Detecting Anomalous Energy Consumption in Buildings using Smart Meter Data

by

Haroon Rashid Lone

under the guidance of

Dr. Pushpendra Singh

Department of Computer Science and Engineering
Indraprastha Institute of Information Technology - Delhi
New Delhi - 110 020, India
December, 2018

Detecting Anomalous Energy Consumption in Buildings using Smart Meter Data

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

Doctor of Philosophy

in

Computer Science and Engineering

by

Haroon Rashid Lone
(Roll No: Phd1307)

under the guidance of

Dr. Pushpendra Singh

Department of Computer Science and Engineering
Indraprastha Institute of Information Technology - Delhi
New Delhi - 110 020, India
December, 2018

Department of Computer Science and Engineering
Indraprastha Institute of Information Technology - Delhi
New Delhi - 110 020, India.

December 15, 2018

Certificate

This is to certify that the work in the thesis entitled ***"Detecting Anomalous Energy Consumption in Buildings Using Smart Meter Data"*** by ***Haroon Rashid Lone*** is a record of an original research work carried out under my supervision and guidance in partial fulfilment of the requirements for the award of the degree of Doctor of Philosopy in Computer Science and Engineering, IIIT Delhi, India.

The results contained in this thesis have not been submitted in part or full to any other university or institute for the award of any degree/diploma.

Dr. Pushpendra Singh
Associate Professor
Department of CSE
IIIT Delhi, India

Dedication

This thesis is dedicated to my parents and teachers.

Acknowledgement

Had my parents not allowed me to go for a Ph.D., I might not have developed the courage to see the hardships and still stay calm. Love, unconditional support, and constant encouragement from parents and siblings help me to grow and become a better person day by day. Making phone calls each day and discussing health, food, and weather allowed me to remain focussed. I owe my life to you.

I am grateful to my advisor, **Prof. Pushpendra Singh** for providing me the opportunity to join his lab. Providing freedom in choosing the research topic and accepting my Ph.D. failures kept motivated me to pursue research. Your continuous support, motivation, and encouragement helped me to reach the stage of writing this thesis. Allowing multiple internships, collaborations, and supporting conference travel is commendable.

I am thankful to my thesis evaluation committee: **Dr. Arunchandar Vasan**, **Prof. David Irwin**, and **Prof. Nirmalya Roy** for the comprehensive feedback. The feedback was valuable in improving the overall quality of the thesis.

My sincere thanks to **Professors Krithi Ramamritham, Vladimir Stankovic, Lina Stankovic**, and **Amarjeet Singh** for hosting me during internships. Krithi's wittiness and curiosity towards research inspire me to work more. The systematic research approach followed by Vladimir and Lina made me realize that doing research is not just prototyping. Amarjeet improved my experimental rigour and technical writing.

Joining an established "energy research group" of **Pandarasamy Arjunan, Nipun Batra, Manoj Gulati**, and **Milan Jain** was indeed a blessing. Pandarasamy helped me to start with the experiments. Nipun's comments improved my paper writing and presentation skills. Discussions with Manoj and Milan over the years helped me to understand the broad aspects of energy research.

Thank you, **Dheryta Jaisinghani, Sonia Soubam, Garvita Bajaj, Deepika Yadav, Anupriya Tuli, Jasmeet Kaur, Ashwini B, Anil Sharma, Parikshit Maini, Alvika Gautam, Sneihil Gopal, Inderpal Singh, Charul, Siddhartha Astana, Vandana Mittal**, and **Akanksha Farswan** for your constant feedback, cooperation, and friendship. It was your company which made me grow up technically each day. A special thanks to Anupriya for adding fun and a different research approach in the Mobile and Ubiquitous Computing group. Your activities made me realize that research can be done while having fun. Informal discussion group, "Brain teasers", with Milan, Anil, Manoj, and Tanya was helpful to get early feedback and understand cross-domain problems.

I am indebted to other friends at the Institute who made my stay memorable.

Thank you, **Venkatesh Vinayakarao, Siddharth Dawar, Lokender Tiwari, Ankita Likhyani**, and **Dhriti Khanna** for all the great time we have shared. In particular, I will miss wisdom-full post-dinner discussions with Venkatesh. Your help in reviewing research proposals will always be remembered.

Many thanks to Tata Consultancy Services (TCS) for supporting my Ph.D. In particular, sincere thanks to TCS research scientists **Dr. Arunchandar Vasan** and **Dr. Venkatesh Sarangan**. Your technical feedback during exams and conferences improved the bar of my work. Annual ITRA project meetings have been hugely beneficial to me. I want to thank **Dr. Shivkumar Kalyanaraman** and **Prof. Archan Mishra** for correcting me during the meetings.

Profound thanks to our former Director, **Prof. Pankaj Jalote** for maintaining the research standard and inspirational speeches. Over the years, I have been hugely benefited from the examination committee members. Thank you, professors **Amarjeet Singh, Srikanta Bedathur, Anghsul Majumdar**, and **Venkata M. Viswanath Gunturi** for the feedback.

Thanks to the administrative staff of IIIT Delhi, who always do their best to sort out the issues timely. In particular, thank you, **Ms. Priti Patel, Ms. Sheetu Ahuja, Ms. Jyoti Singh, Mr. Ashutosh Brahma, Mr. Umesh Singhania, Mr. Bhawani Shah**, and **Mr. Amit Shankdhar** for the support. Also, thanks to our chief librarian **Mr. Rajendra Singh** for arranging research articles from different universities.

My friends back home played an important role during my stay at the institute. Thank you, **Rizwan Akbar, Rouf Rubbani, Kalimullah Lone, Adil Mudasir, Jahangeer Ali, Ubaid-ul-Haq, Saboor Sirwal, Akif Khan**, **Masood** for being there at the moments of despair.

Last but not the least, heartfelt thanks to my dearest wife, **Gazala** for being there for me and the emotional and moral support. Your support pushes me to move forward and to become a better person day by day.

Haroon Rashid Lone

Abstract

Buildings consume 50% of the total available electrical energy. Studies show that up to 20% of the energy gets waste due to several reasons such as appliance misconfigurations, faults, and abnormal user behavior. The instances of abnormal energy consumption by various electrical appliances are known as anomalies, and it is important to detect such anomalies in a timely manner.

Several works use intrusive appliance-level monitors, i.e., submetering, for detecting anomalous behavior of appliances. These approaches detect anomalies, but the intrusive monitoring approach is not scalable as it requires to have a separate monitor for each appliance in a home. A more practical approach is to use the household's aggregate consumption data from a single smart meter. Existing methods using smart meter data suffer from two limitations: (*i*) they result in a high number of false alarms which makes them unreliable, (*ii*) they only detect anomalies and do not identify the anomaly causing appliance, which makes it difficult for building administrator to take prompt action.

Using smart meter data, we handled the problem of anomaly detection both at coarse and fine-granular levels. At coarse-level, we proposed a density-based anomaly detection approach namely, *Monitor*, which takes the power readings of several days as input and outputs anomaly score for each day consumption. At the granular level, we proposed an anomaly detection approach namely, Rimor, and also evaluated the efficacy of Non-Intrusive Load Monitoring (NILM) for identifying anomalous appliance. Results show that proposed approaches reduce the false alarm rate considerably and also identify the anomalous appliances.

We did all our experiments on publicly available datasets namely Dataport, REFIT, AMPds, and ECO. These datasets have both aggregate and appliance-level energy consumption data, but none of them provides information about anomalous instances. We created and publicly released detailed anomaly annotations for REFIT at appliance-level and the remaining three datasets at the aggregate-level. Furthermore, we collected a context-rich four-year energy dataset from our campus and made it publicly available so that it can be leveraged to propose newer anomaly detection approaches with higher accuracy and take the research in a new direction.

Contents

List of Figures

List of Tables

List of Publications

1. **Haroon Rashid**, Pushpendra Singh, Vladimir Stankovic, Lina Stankovic. "Can Non-intrusive Load Monitoring be used for Identifying an Appliance's Anomalous Behaviour?" Applied Energy, Elsevier, 2019.

2. **Haroon Rashid**, Pushpendra Singh, Amarjeet Singh. "I-BLEND, a Campus Scale Commercial and Residential Buildings Electrical Energy Dataset." Scientific Data, Nature, 2019.

3. **Haroon Rashid**, Vladimir Stankovic, Lina Stankovic, Pushpendra Singh. "Evaluation of Non-Intrusive Load Monitoring Algorithms for Appliance-level Anomaly Detection." In IEEE ICASSP 2019.

4. **Haroon Rashid**, Nipun Batra, and Pushpendra Singh. "Rimor: towards identifying anomalous appliances in buildings." In Proceedings of the 5th Conference on Systems for Built Environments (BuildSys '18). ACM, New York, NY, USA, 2018.

5. **Haroon Rashid** and Pushpendra Singh. "Monitor: An Abnormality Detection Approach in Buildings Energy Consumption." In 4th IEEE International Conference on Collaboration and Internet Computing (CIC), Philadelphia, PA, USA, 2018.

6. **Haroon Rashid** and Pushpendra Singh. "Energy disaggregation for identifying anomalous appliance." In Proceedings of the 4th ACM International Conference on Systems for Energy-Efficient Built Environments (BuildSys '17). ACM, New York, NY, USA, 2017. [**Poster**]

7. **Haroon Rashid**, Priyanka Mary Mammen, Siddharth Singh, Krithi Ramamritham, P. Singh, P. Shenoy. "Want to reduce Energy Consumption? Don't depend on the Customers! In 4th ACM International Conference on Systems for Energy-Efficient Built Environments, ACM-BuildSys, Delft, The Netherlands, 2017.

8. **Haroon Rashid**, Pushpendra Singh, Krithi Ramamritham. "Revisiting selection of residential consumers for demand response programs" In 4th ACM International Conference on Systems for Energy-Efficient Built Environments, ACM-BuildSys, Delft, The Netherlands 2017.

9. **Haroon Rashid**, Pandarasamy Arjunan, Pushpendra Singh, and Amarjeet Singh. "Collect, Compare, and Score: A generic data-driven anomaly detection method for buildings."In Proceedings of the Seventh International Con-

ference on Future Energy Systems Poster Sessions (e-Energy '16). ACM, New York, NY, USA [**Poster**]

10. Priyanka Mary Mammen, Hareesh Kumar, Krithi Ramamritham, **Haroon Rashid**. "Want to Reduce Energy Consumption, Whom should we call?" Proceedings of the ninth ACM International Conference on Future Energy Systems, ACM e-Energy, Karlsruhe, Germany, 2018. [**Not included in the thesis**]

Chapter 1

Introduction

Energy is one of the crucial elements in supporting life on earth. Providing access to reliable, affordable, and modern energy is one of the 17 Sustainable Development Goals (SDGs) set by the United Nations in 2015. Ensuring energy sustainability is a prime concern as it interlinks all other SDGs such as eradicating poverty, climatic action, clean water, and sanitation. Still, one billion[1] people around the world do not have access to energy.

Globally, four major sectors consume energy, i.e., commercial, residential, industrial, and transportation. Within these, commercial and residential buildings collectively consume 50% of total electrical energy [1]. Figure 1.1(a) shows the percentage of electric energy consumption across all four sectors. In buildings (commercial and residential), energy is used for lighting, cooling (air conditioners, coolers, refrigerators), heating (boilers, ovens, heaters, stoves, furnaces) and other miscellaneous (washing machine, dishwasher, television, computers) purposes. International Energy Agency (IEA) reports that energy consumption demand of the world is increasing[2] continuously and will go up to 28% between 2015 and 2040 [1]. The reasons for this growing demand include increased access to marketed energy, economic growth, and rapidly growing populations.

However, major energy sources including coal, liquids, natural gas, and nuclear

[1] http://in.one.un.org/page/sustainable-development-goals/sdg-7/

[2] From 575 to 736 quadrillion BTU. one quadrillion is 10^{15} and one BTU = 1055 joules

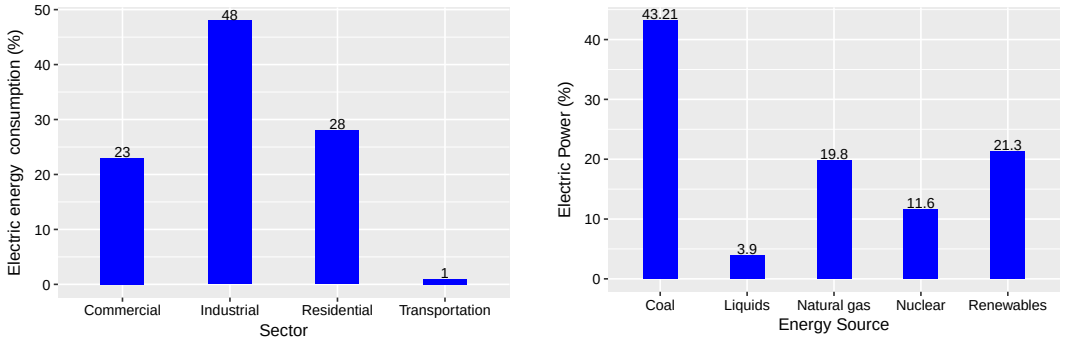


Figure 1.1: (a) Percentage of electric energy consumption across different sectors worldwide, (b) percentage of primary sources used for electric power generation

are limited. Figure 1.1(b) shows the percentage of primary sources used in the electric power generation. Knowing the limited nature[3] of fossil fuels and our technological inability to harness renewable energy sources [2, 3], consumers have to use energy efficiently by reducing energy wastage instances [4].

In buildings, energy wastage happens either due to faulty electrical appliances or consumer behavior. An appliance becomes *faulty* when one of its parts works inappropriately, such as thermistor breakdown of an air conditioner. Table 1.1 lists several faults and the amount of energy wasted due to these faults. While as wastage due to consumer behavior include:

- *Misconfigured appliance* – When a user sets the appliance on inappropriate settings. For example, on a cloudy day, setting air conditioner on sunny day settings.

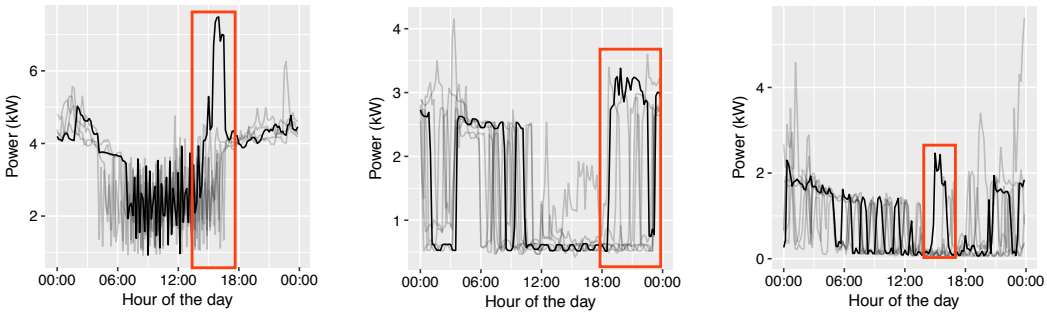
- *Unusual consumer*[4] *behavior* – Either when a user forgets to turn off the appliance after usage or due to a sudden change in weather, working schedule.

Such malfunctioning and misconfigured appliances often result in deviations in energy consumption, generally called as *anomalies*. It is important to detect such anomalies in a timely manner as they waste up to 30% of energy in buildings [5, 6,

[3] https://www.ecotricity.co.uk/our-green-energy/energy-independence/the-end-of-fossil-fuels
[4] In this thesis words such as occupant, user, and consumers are used interchangeably
[5] One quad equals a quadrillion, i.e., 10^{15} BTU

Fault	**%**	**Annual energy consumption (quads)**
Duct leakage	30%	0.300
HVAC left on when space unoccupied	20%	0.200
Lights left on when space unoccupied	18%	0.180
Airflow not balanced	7%	0.070
Improper refrigerant charge	7%	0.070
Dampers not working properly	6%	0.055
Insufficient evaporator airflow	4%	0.035
Improper controls setup/commissioning	2%	0.023
Control component failure or degradation	2%	0.023
Software programming errors	1%	0.012
Improper controls hardware installation	1%	0.010
Air-cooled condenser fouling	1%	0.008
Valve leakage	1%	0.007

Table 1.1: The annual energy impact of faults in buildings



(a) (b) (c)

Figure 1.2: Anomaly types: (a) Point anomaly (b) Sequence anomaly (c) Contextual anomaly. Thin transparent lines show normal consumption on historical days, and thick opaque lines show the consumption on an anomalous day and the red rectangular region marks anomalous portion.

7, 8]. The primary step to reduce the energy wastage is to identify and detect such anomalies automatically.

According to the general classification [9] of anomalies found in time-series data, we differentiate anomalies in energy data as

1. **Point anomaly**: A point anomaly refers to unusual energy consumption for a short duration. Such an anomaly occurs once and does not repeat at regular time intervals. Figure 1.2(a) shows a point anomaly as the aggregate consumption deviates significantly from the historical days. Another example is intermittent spikes observed in refrigerator's power consumption signature.

2. **Sequence anomaly**: A sequence anomaly represents unusual energy consumption for a longer duration or anomalies which occur at regular time intervals. Figure 1.2(b) shows a sequence anomaly in which aggregate consumption deviates significantly for a longer duration. Historical consumption (thin lines) previews that the consumption never remained continuously 3kW for such a long duration.

3. **Contextual anomaly**: A contextual anomaly represents unusual consumption while considering some context such as daytime, occupancy, weather, the day of the week. Figure 1.2(c) represents such an anomaly. The highlighted consumption is marginally different than the remaining consumption of the same day but is significantly different than the historical consumption (thin lines) during the same period (14 – 18 hours). Normally, the consumption used to remain around less than 1 kW, and On this day consumption was above 2 kW for the period.

1.1 Anomaly detection with smart meters

Anomalies in energy consumption can be detected by analyzing smart meter data. Over the last decade, electrical utilities are deploying smart meters aggressively to make electrical grids smart. Until now, half[6] of the U.S. homes are instrumented with smart meters. Unlike traditional billing meters, these meters provide real-time power usage to both the consumer and the power utilities. Such meters allow logging of various electrical parameters such as voltage, current, power factor at a user-defined sampling rate.

Real-time communication and control allow utilities to run various programs such as remote monitoring, energy-saving advices[7], demand response and other energy efficiency checkups. Research shows analysis of smart meter data helps to understand the consumption behavior of consumers and to detect anomalies [10, 11]. There-

[6] https://www.eia.gov/tools/faqs/faq.php?id=108&t=3
[7] http://www.energylens.com/articles/energy-management

fore smart meter data opened the door to various works including *energy wastage monitoring and analysis.*

1.2 Why is anomaly detection important?

The problem of anomaly detection in the buildings energy consumption is important from multiple perspectives. Some of the important ones include:

1. **Detecting faulty appliances:** Monitoring each appliance in large buildings is laborious, and hence it becomes difficult to know when does an appliance gets faulty. Mostly, a faulty appliance works, but the energy consumed is usually high as compared to its normal operation, e.g., faulty fridge compressor. Such appliances are detected either when they stop working or during commissioning, which happens rarely. This means such appliances waste a considerable amount of energy until their detection. These appliances can be identified with anomaly detection approaches as faults manifest in consumption pattern, which differs significantly from the normal one. Furthermore, with anomaly detection approaches, building operators can reduce the failure rate of an appliance by predicting its faulty nature in advance.

2. **Detecting misconfigured appliances:** Electrical appliances provide a range of settings to customize comfort levels. Under different settings, operating characteristics of an appliance vary, hence is the energy consumption. Misconfigurations result in an increase or decrease in energy consumption. An example of misconfiguration is using an AC on a cloudy day with the settings of a hot sunny summer day. Such misconfigurations unnecessarily waste energy. Within the misconfigured scenario, we also include cases where users forgot to switch-off their appliances in odd usage hours, and this also results in energy wastage. These energy wastage instances can be identified in real-time using anomaly detection approaches as against to the end of the month when a consumer receives higher electricity bill.

3. **Computing baseline energy consumption:** Power utilities are often interested in knowing the baseline energy consumption of its consumers. This baseline information is used for various programs, which include: (*i*) Utilities use it to predict the energy consumption in the future (coming days). This prediction gives an estimate to power utilities, how much power they need to generate and hence accordingly manage the resources. (*ii*) This also helps utility companies to run Demand-Response (DR) programs. In DR, utilities select a set of consumers who voluntarily shed their load during peak load hours. This stabilizes the electric grid under peak demand. In return, consumers get some monetary benefits. The selection of consumers in DR is made on their historical usage pattern, i.e., high energy consuming consumers are selected. Both the prediction as well as DR depend heavily on the accurate measurement of baseline energy consumption. If these measurements contain anomalies as well, then both of these programs result in poor performance. Therefore, it is important to separate anomalies before calculating baseline energy consumption.

The importance of the anomaly detection problem is clear. Still, there are various challenges in detecting anomalies accurately.

1.3 Challenges in anomaly detection

Anomaly detection is an unsupervised problem. The energy data are not labeled, and there are no anomaly signature databases. Creating an anomaly signature database is difficult due to the diversity found in the energy consumption behavior of buildings. For examples, Figure 1.3 shows the one-week consumption pattern of a commercial and residential building at IIIT Delhi. The commercial building follows a regular consumption pattern while as the residential building do not follow a pattern across different days. So, anomaly signature in one building may not manifest in the similar form in some other building.

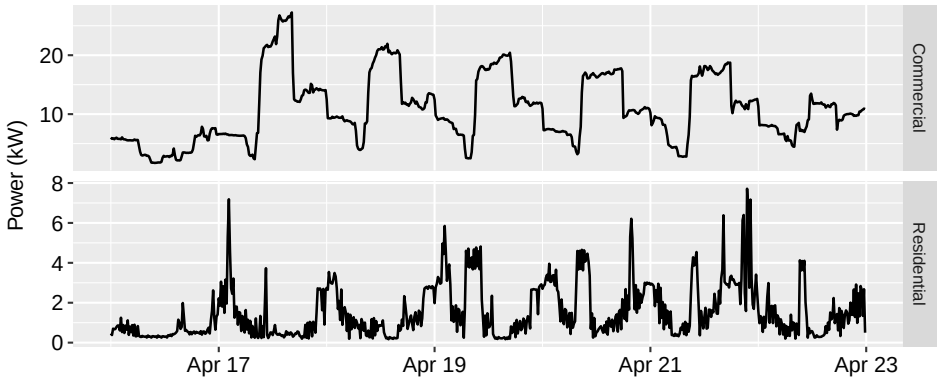


Figure 1.3: Energy consumption pattern of commercial and residential buildings over a complete one week at IIIT Delhi.

Often, an anomaly detection system detects an anomaly, but then the next question arises whether the detected anomaly should be reported to an occupant or not, as all anomalous events do not represent energy wastage. For example, in a home, extra AC usage may happen either due to guests or an occupant might have forgotten to switch off the AC after normal usage. In both of the cases, the extra usage gets flagged as an anomaly. Alarming an occupant in the first case reduces the confidence on the system, and the system is labeled as unreliable, which often happens in most such detection systems.

It is difficult to address these challenges without the availability of additional data in the form of annotations or contextual information. In this thesis, we identified following two major challenges and proposed approaches to address them.

1. **Low anomaly detection accuracy:** Existing approaches are either threshold-based or use historical data for identifying anomalies. Balakrishnan et al. report that in the campus of the University of California, San Diego, threshold-based approach resulted in more than 10,000 alarms per day [12]. The high number of alarms were then ignored by the building administrators thus defeating the purpose of detecting anomalies. Therefore, it is important to develop novel approaches that can adapt to dynamic energy usage and are more accurate, i.e., the rate of false positive alarms is low. Remaining methods detect

an anomaly on a particular day by considering data from previous days. However, in doing so, these methods ignore the consumption patterns and treat all days *equally*. Ignoring patterns existing in consumption affects the accuracy and leads to a high number of false positive alarms.

Every home has a different family size and differs in energy consumption pattern in one or the other way. Some homes have all working members, and a pattern results in energy consumption while some homes have children or elders which consume energy throughout the day, and this results in random energy consumption. This variation in energy consumption behavior across homes makes it difficult for one algorithm to work correctly in all homes and affects large-scale deployments. Further, energy consumption of any home is a function of various parameters such as time of a day, weather (temperature and humidity), occupant energy consciousness, occupant comfort level, etc. To our knowledge, we do not find a single existing algorithm which models all these parameters, so such anomaly detection systems often result in false alarms. Now the challenge is to make a context-aware reliable anomaly detection algorithm. Collecting contextual information is cumbersome and also poses risks to privacy which are the additional challenges.

2. **Inability to identify anomalous appliance:** Smart meters measure buildings' aggregate energy consumption, so an anomaly detected with smart meter data does not show which home appliance resulted in the anomaly. After detecting an anomaly, building manager has to find the anomalous home appliance manually to take necessary action. This manual process of appliance identification slows down the purpose of such automated anomaly detection systems. The manual appliance identification gets difficult with: (i) increase in the number of home appliances, (ii) multiple appliances with same power rating or of same consumption signature type, e.g., refrigerator and freezer have almost same signature and wattage ratings.

Furthermore, lack of publicly available anomaly labeled datasets impedes the re-

search in the domain. Energy community has made many buildings' energy datasets publicly available, and most of them have been collected by academia to report various results. Some of the datasets include: REDD [13], Smart* [14], iAWE [15], UK-DALE [16], DRED [17], ECO [18], AMPds [19], REFIT [20], Dataport [21]. Unfortunately, none of them is labeled with anomalies. Now, post-labeling of such datasets without having the complete contextual information is difficult which makes it impossible to evaluate anomaly detection algorithms on such real-world datasets. In extreme cases, an algorithm can be evaluated on a synthetic anomaly of which anomalous signature is already known as done in [22], but, generally, it is impossible to simulate all anomalous signatures due to the diverse set of appliance models. To our knowledge, not even a single anomaly annotated dataset is publicly available for experimentation.

1.4 Contributions and organization of the thesis

The thesis aims to leverage household aggregate smart meter data in (i) improving the state-of-the-art in the anomaly detection, and (ii) identifying anomalous appliance among various household appliances.

We handled the problem of anomaly detection using smart meters both at coarse and fine-granular levels. At coarse-level, we proposed a density-based anomaly detection approach namely, *Monitor*, which is explained in **chapter 3** of the thesis. The *Monitor* takes the power readings of several days as input and outputs anomaly score for each day consumption [23, 24]. The anomaly score defines how likely the consumption of the day was anomalous. Monitor improves anomaly detection performance considerably by reducing the number of false positive alarms. We have evaluated our approach on 16 weeks smart meter data of our campus buildings. The comparison of this approach with existing baseline approaches shows that our approach improves the accuracy by up to 24% in best scenario and on average by 14%. This improvement in accuracy reduces the rate of false positive alarms significantly and makes it more suitable for real-world deployments.

At the granular level, we proposed an anomaly detection approach namely, RIMOR[8], and also evaluated the efficacy of Non-Intrusive Load Monitoring (NILM) for identifying anomalous appliance using household smart meter data [25]. RIMOR predicts the energy consumption of a home using historical energy data and contextual information. It flags an anomaly when the actual energy consumption deviates significantly from the predicted consumption. Further, it identifies anomalous appliance(s) by using easy-to-collect appliance power ratings. We evaluated it on four real-world energy datasets containing 51 homes and found it to be 15% more accurate in detecting anomalies as compared to four other baseline approaches. RIMOR reports an appliance identification accuracy of 82%. The work comprises **chapter 4** of the thesis.

With the satisfactory progress of NILM, numerous energy companies have deployed NILM based products in various countries including UK, USA, Canada, France, Korea. Primarily, these companies attach NILM fusebox with a household's smart meter and provide appliance-level consumption to consumers. Research shows that giving appliance-level consumption to consumers results in more than 12% energy savings [26]. Also, studies mention that NILM can be used for identifying anomalous appliances in buildings [27, 28]. However, no NILM based work for detecting anomalous appliances is available. So, for the first time, we evaluated the applicability of NILM in identifying anomalous appliances in detail [29, 30, 31]. This work is explained in **chapter 5** of the thesis. We did our experiments using both real and simulated anomalies. Our results show that while NILM could identify anomalies in higher energy consuming appliances reasonably, its performance for lower energy consuming appliances is not satisfactory. More research is needed in the NILM domain to propose anomaly-aware NILM approaches.

We did our experiments on publicly available datasets namely Dataport (USA), REFIT (UK), AMPds (Canada), and ECO (Switzerland). These datasets have both aggregate and appliance-level energy consumption data, but none of them provides

[8]In Latin, RIMOR means investigator, explorer

information about anomalous instances. We created detailed anomaly annotations for REFIT showing both anomalous duration and the anomaly causing appliances. For the remaining three datasets, we noted all anomalous instances found at the aggregate-level and have released it with RIMOR on a publicly accessible link. Availability of anomaly annotated datasets publicly can be leveraged to propose newer anomaly detection approaches with higher accuracy and take the research in a new direction.

Lack of context-rich energy datasets impedes anomaly detection research. So, for the first time, we release 52 months context-rich energy dataset collected from our institute campus [32]. It includes contextual information such as occupancy, weather, and the Institute calendar. The dataset is explained in **chapter 6** of the thesis in detail. To the best of our knowledge, this is the first such large energy dataset from India. Public availability of such fine-granular data will allow users to perform different research tasks such as analyzing the impact of weather or occupancy schedule on energy consumption, detecting anomalies, and developing algorithms for predictive maintenance.

Chapter 2

Literature survey

The problem of anomaly detection in energy consumption is inherently *unsupervised* in nature as the data are not labeled. Further, it is difficult to build a one-time energy consumption model for buildings as energy usage is dynamic due to seasonal variations and external contextual factors such as day type (holiday, working day) or occupancy. Anomalies can be detected in offline or online mode. Generally, offline detection methods are modeled as unsupervised and approach like clustering is used to identify anomalies [22, 33, 34]. While as online detection methods are modeled as *supervised* (classification), where energy usage is forecasted a priori, and then actual usage is compared with the forecast. If the actual usage is found significantly different from the forecast, then an anomaly is flagged. The accuracy of such classification approaches depends on the forecast accuracy and the various external contextual factors such as user behavior.

In buildings, anomalies are detected at two different levels, i.e., appliance level [35, 36, 37, 12, 38, 39, 40, 41] or aggregate household level [33, 42, 22]. At appliance level, anomalies within a specific appliance like refrigerator or air conditioner (AC) are detected by instrumenting the appliance with a separate monitoring kits. Instrumenting each appliance in a home with a separate monitoring kit is a massive overhead due to a combination of installation, maintenance, communications, storage, and data validity checks. While at the aggregate household level, anomalies are detected by using aggregate smart meter data only. In this thesis, we focus on

anomaly detection at the aggregate household meter level only.

For simplicity, we categorize the existing literature on anomaly detection based on the type of approaches used: Statistical, Machine learning, Graph, and Rule. Table 2.1 summarizes important features of existing anomaly detection approaches in table form.

2.1 Statistical test based anomaly detection

To our knowledge, for the first time, John Seem used smart meter data for anomaly detection [43]. He used a pattern recognition algorithm to identify days of a week with similar power consumption while processing features such as daily peak and average consumptions. During this process, anomalous observations were removed using statistical tests such as generalized extreme studentized deviate (GESD) [44] and Wilk's multivariate outlier test [45]. The approach requires two user-defined parameters, initial probability value defining observations as anomalies when no anomaly exists in reality, and an upper bound on the number of anomalies present. A wrong setting of these parameters may decrease anomaly detection accuracy. In 2007, John Seem improved the mentioned approach by using features mean and standard deviation on daily consumption [46]; and by ranking final anomalies by using modified z-score [47]. This score measures the number of standard deviations an anomalous observation is from the computed mean. The higher the difference, the major severe is an anomaly. Similarly, Li et al. [48] used GESD and Q-test on the mean and standard deviation of the observed data to flag anomalous observations. However, Liu et al. [49] proposed a method to detect anomalies in lighting consumption of buildings. Authors first use classification and regression technique (CART) and then GESD to identify anomalous consumption. The difficulty with these approaches is setting the proper thresholds on mentioned parameters.

Liu et al. [50] proposed a *generalized* comprehensive model consisting of a pipeline of different statistical methods (multivariate regression model, variable base degree day regression (VBDD) model [51], and integrated regressive moving average

(ARIMA) model) for anomaly detection, forecasting, finding root cause of energy usage, and energy efficiency of buildings. Specifically, regression models identify the correlation between a building's energy consumption and its characteristics (age, occupancy, and the number of appliances), VBDD model separates building's base load from the weather-dependent usage. Where base load shows consumption due to lighting, cooking, and plug loads alone. And weather-related usage includes base temperature estimates and coefficients of heating/cooling degree days. Finally, ARIMA model is used to capture the effect due to building's tenants and seasonal factors. Anomalies are detected if the energy consumption lies outside of 95% confidence bounds on historical consumption.

Instead of offline anomaly detection methods, Chou et al. [52] proposed an online anomaly detection method. This method works in two stages, i.e., prediction and the anomaly detection stage. Prediction is made on a weekly basis, and the actual usage is considered as anomalous if found outside the predicted usage by two standard deviations. On the other hand, Wrinch et al. [53] showed that the frequency domain representation of metered time-series energy data is more suitable for detecting anomalies in periodic operations. They did their experiments on real-world office buildings data having sampling rates of one and 15 minutes.

2.2 Machine Learning based anomaly detection

Machine learning based approaches are mostly clustering based. For instance, (i) Chen et al. [54] first discretize continuous numeric energy data to symbolic representation [55] by using equal-width binning approach, and later use Suffix trees to find different patterns in energy usage. Finally, clustering on the frequency count of different symbolic sequences is used to identify anomalous instances. (ii) Zhang et al. proposed three different methods, i.e., regression-based, entropy-based and a clustering one [48]. Both regression and entropy-based require some thresholds a priori while as clustering based uses different features such as mean, variance, maximum range, ratio, and does not need to specify any threshold beforehand.

Contrarily, Bellala et al. proposed a density-based anomaly detection approach for commercial buildings [33]. They first create a low dimensional representation of high-dimensional hourly data using Multidimensional Scaling [56] and later use the k nearest neighbor algorithm to compute anomaly score for each day's energy consumption. The higher the score, the higher are the changes of the day's consumption being anomalous.

Few works leveraged existing algorithms for anomaly detection in the energy domain. For instance, (i) Capozzoli et al. evaluated and compared four different existing approaches for anomaly detection [57, 58]. Authors used various statistical approaches and neural networks to detect anomalies in lighting consumption of eight buildings. (ii) Janetzko et al. [59] proposed visualization based anomaly detection techniques for building managers. In the background, two anomaly detection techniques are used: one prediction based proposed by [60] and the second one proposed by [33]. Detected anomalies are visualized through different techniques such as recursive patterns, spirals, traditional line chart. Each technique shows intrinsic anomalous behavior to a building manager in an interpretable way.

2.2.1 Contextual information augmentation

Energy consumption of buildings is a function of different contextual features as explained in the first chapter. So, several machine learning approaches were augmented with contextual information [61, 62, 22]. For example, Arjunan et al. used neighborhood information to detect anomalous instances [22]. Firstly, they compute day wise anomaly score using k-Medoid clustering approach for each home separately. Next, they adjust anomaly scores of each home by comparing home's score with other homes having similar historical energy consumption pattern. The intuition is that homes with similar historical consumption patterns should behave in the same manner. While as, Fontugne et al. proposed Strip, Bind and Search (SBS) – an unsupervised anomaly detection method [37]. SBS explores the relationship among different appliances (AC, fan, lights, computer) usage behavior, *i.e.*, it

groups the devices used at the same time in a particular setting. These intrinsic device relationships are uncovered using Empirical Mode Decomposition [63]. These relationships change with seasonality and hence are purely contextual. A difference observed in any group behavior is flagged as an anomaly. Similarly, Balakrishnan et al. proposed Model, Cluster and Compare (MCC), another unsupervised technique for fault detection [12]. However, MCC explores the relationship between different HVAC zones and build models for similar zones to detect faults.

Several anomaly detection approaches used seasonality as a contextual feature. For example, Ploennigs et al. [64] use buildings' hierarchy of submeters to identify which part of the building results in anomalous consumption using a statistical model known as a generalized additive model (GAM). Apart from historical consumption GAM includes exogenous factors – temperature, time of year, day type and time of the day to identify anomalies, and hence GAM shows whether the anomaly was due to these exogenous factors or not. If an anomaly is detected at building's main meter, then GAM is rerun on the all submeters to identify the anomalous submeter. This process continues recursively until an anomalous meter is identified. This approach can be only used in buildings where meters are installed at different levels. Araya et al. [65] proposed a generic anomaly detection framework. The proposed framework uses a threshold for anomaly detection. Apart from historical power consumption data, it uses generated features (mean, median) from energy data, and contextual features (day of the year, season, month, the day of the week and hour of the day) for anomaly detection.

However, Hayes et al. [66] proposed a generalized contextual based anomaly detection algorithm and evaluated it on the energy data. It works in two steps: (1) Content anomaly detection – In this, it first builds a statistical training model on historical data, and then it compares new value with the built model. A significant deviation observed is flagged as an anomaly. In the next step, it verifies the anomalous nature by using contextual information. (2) Context anomaly detection – In this, firstly, it creates sensor profiles with multivariate context based clustering

algorithm and predicts some value. Next, it compares the predicted value with the actual value observed. If the new value too deviated from the predicted value by a defined threshold, then usage is flagged as anomalous. The applicability of the approach is limited as the authors did not explain how to set thresholds in both of the steps optimally.

2.3 Graph based anomaly detection

Cheng et al. used random graph traversal over kernel matrix to determine anomalies [67]. A kernel matrix is constructed from the time series energy data, and less traversed nodes are flagged out as anomalies. Further, authors find a relationship between different time-series by aligning kernel matrices; and these relationships can be used as contextual information. However, Qiu et al. used Granger graphical models to identify anomalies [68]. They first built causal graph time series data by learning temporal causality, next compute anomaly score of the graph with Kullback-Leibler divergence and finally flag anomalies at a particular threshold.

2.4 Rule based anomaly detection

Wijayasekara et al. [69] proposed a fuzzy logic rule-based anomaly detection technique. During training, clustering is used to group the consumption data and later fuzzy logic rule extraction is used to create rules for identifying normal consumption. During testing, energy consumption is compared using the already built fuzzy logic rule-based model. A deviation from the model is flagged as an anomaly. Further, the proposed approach presents an anomaly status to a building administrator in the form of linguistic descriptions. Providing linguistic descriptions to building administrators have been found more useful than anomaly scores. Similarly, Pena et al. [70] proposed another rule-based anomaly detection system. Besides electricity data, indoor and outdoor sensor data (occupancy, weather) were used to create a comprehensive set of rules for identifying anomalous consumption.

Paper	Dataset Type	Sampling Rate	Sensing Device	Approach
John [43, 46]	Commercial	60 minutes	Smart Meter	Robust statistical measures on historical data are used to detect anomalies
Chou et al. [52]	Commercial	1 minute	Smart Meter	Two standard deviations rule is used to separate anomalous usage from the normal usage
Chen et al. [54]	Residential		Sensors	Abnormal patterns (anomalies) are identified from the normal patterns by using using suffix tree representation
Li et al. [71]	Commercial	30 minutes	Smart Meter	Usage is considered as anomalous, if not following the trained classification model
Zhang et al. [48]	Residential	60 minutes	Smart Meter & Sensors	three approaches were used to identify anomalies, which include regression, entropy and clustering
Capozzoli et al. [57, 58]	Commercial	15 minutes	Smart Meter	Statistical techniques along with neural networks were used to identify anomalies
Liu et al. [49]	Commercial	60 minutes	Smart Meter	Statistical generalized extreme studentized deviate is used to separate anomalous lighting consumption in office buildings
Arjunan et al. [22]	Commercial & Residential	60 minutes	Smart Meter	First clustering is used to find anomalous score for day consumption and later neighbourhood information is used to adjust score
Bellala et al. [33]	Commercial	60 minutes	Smart Meter	Density based approach is used to identify anomalous days
Fontugne et al. [72, 37]	Commercial		Sensors	Anomalies were identified in frequency domain analysis by monitoring device relationships
Balakrishnan et al. [12]	Commercial		Sensors	Faults in the HVAC VAV values are identified by monitoring entities working at same time
Ploennings et al. [64]	Commercial	60 minutes	Smart Meter at different levels	Finds anomalies at two levels, i.e., if using main meter anomaly is detected then, on all sub meters, anomaly detection approach is run to find the exact device causing anomaly

Table 2.1: Categorization of existing literature on anomaly detection in buildings energy consumption.

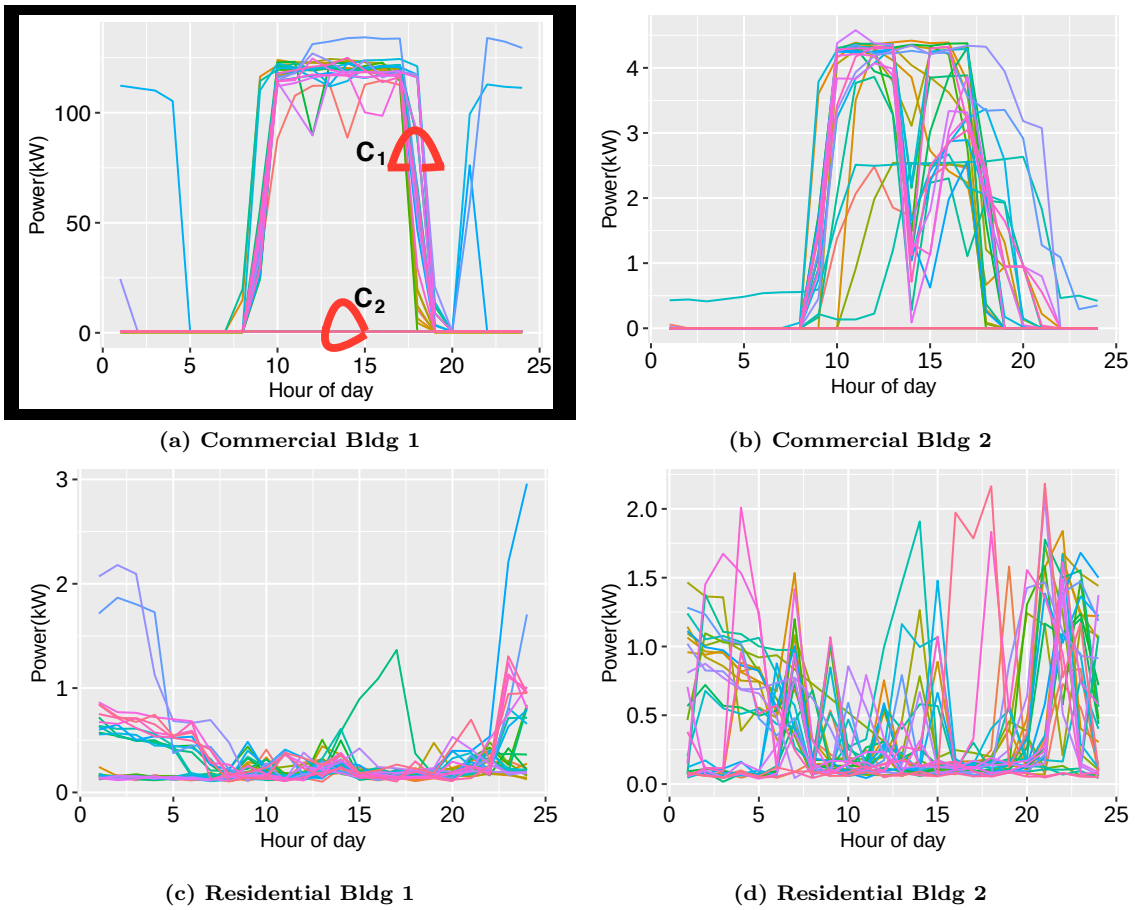
Chapter 3

Offline anomaly detection at the household level

3.1 Introduction

The identification of anomalies is essential for reducing energy wastage. Timely identification can be used to prompt customers for corrective actions. As the scale of data grows, automated methods are needed to analyze and provide actionable feedback to customers as well as organizations. A study shows that giving feedback to customers on daily or weekly basis results in up to 8.4% energy savings [26]. However, for effective feedback, it is important to accurately identify anomaly instances to avoid cases of false positive alarms.

A simple approach to detect anomalies is to use threshold-based monitoring. In this approach, an anomaly detection system generates an alarm when the energy usage is found higher than the defined threshold. But, this approach does not cope well with dynamic energy usage behavior of buildings. Balakrishnan et al. report that in the campus of the University of California, San Diego, this approach resulted in more than 10,000 alarms per day [12]. The high number of alarms were then ignored by the building administrators thus defeating the purpose of identifying anomalies. Therefore it is important to develop novel approaches that can adapt to



(a) Commercial Bldg 1

(b) Commercial Bldg 2

(c) Residential Bldg 1

(d) Residential Bldg 2

Figure 3.1: Hourly average energy usage of different commercial and residential buildings for a duration of one month. Each wiggly line represents energy usage of a different day of month. The thick dark bands represent different dominant usage patterns, where each band consist of similar usage days.

dynamic energy usage and are more accurate, i.e., the rate of false positive alarms is low. Existing methods detect an anomaly on a particular day by considering data from all previous days [46, 33, 22]. However, in doing so, these methods ignore the consumption patterns and treat all days *equally*. We show in Section 3.4 that ignoring patterns existing in consumption affects the accuracy and leads to high false positive alarms.

This chapter presents *Monitor*, which is a density based anomaly detection approach. We presented the preliminary version of *Monitor* in the poster session of ACM e-Energy 2016 [23] with a name of Collect, Compare, and Score (CCS), later the final version was presented in the IEEE-CIC [24]. In our approach, we first identify prominent patterns present in past consumption; thereafter we cluster days in different groups as per their pattern; at the end, we classify an instance

as anomalous if that instance differs significantly from the prominent pattern of the cluster. Our approach of first identifying patterns resembles closely with the real-world usage, e.g., in commercial settings often working weekdays have a very different and distinct pattern than non-working weekends or holidays and even in residential apartments we can see distinct patterns when the occupants are in house whole day or while they are out for regular work (say office hours). For clustering days as per their patterns, we use Local Outlier Factor (LOF) [73]. The LOF first clusters all past days into several clusters having different energy consumption patterns. Next, it computes anomaly score in the range of 0 to 1 for each day within a specific cluster by using other days of the same cluster only. Therefore, days in other clusters do not affect anomaly score calculation. For Example, Figure 3.1(a) shows energy consumption on different days of a commercial building. Days of a different type (working, non-working) result in different clusters where days of one type usually go into one cluster. For example, Cluster C_1 represent energy consumption of working days and C_2 that of non-working days only. Both of these clusters differ due to inherent day type context. Existing methods would detect an anomaly on a particular day with respect to all past days which results in a large number of false alarms [33, 22]. Our proposed approach handles this limitation by first dividing days into relevant clusters. Afterward, the anomaly score is calculated locally and days in the remaining clusters do not affect anomaly score calculation.

For evaluation, we use a dataset comprising of different *loads* at IIIT-Delhi Institute campus that show different usage patterns, thus allowing us the opportunity to test our approach on diverse energy consumption patterns. The dataset is explained in Section 3.3.1 in detail. For generalizability, we refer loads to different energy consuming sources such as residential apartments or HVAC chiller. The energy usage pattern of commercial and residential buildings (loads) is different as shown in Figure 3.1. The commercial building shows 1–3 distinct patterns while among the residential buildings only one of them show a pattern. Further, we compare the performance of *Monitor* with two existing Anomaly Detection Methods

ALGORITHM 1: Steps of *Monitor*

Input: $X[M]$: Time series power consumption data of M days.
Output: $A[M]$: Anomaly score for each day in the range of 0 to 1.

1 Transform input time series sequence X into Matrix $Y[M, T]$ where M represents number of days and T represents number of power readings per day.
2 Calculate DFT for each row of Y separately as, $Y^{'}[i, \ ] = DFT(Y[i, \ ]), \forall i \in \{1, \cdots, M\}$.
3 Calculate dissimilarity matrix $\Delta[M, M]$ for all pairs of M days with Euclidean distance measure, i.e., $\Delta[i, j] = [\sum_{k=1}^{T}(Y^{'}[i, k] - Y^{'}[j, k])^2]^{1/2}$ where $i, j \in \{1, \cdots, M\}$
4 Reduce dimensionality of Δ from M to 2 using MDS technique, i.e., $\widehat{\Delta}[M, 2] = MDS(\Delta[M, M])$
5 Calculate LOF $L[M, K]$ using $\widehat{\Delta}[M, 2]$ for various K, where K refers to the number of neighbors
6 Calculate final LOF score for each day M as, $\widehat{L}[i] = max(L[i, k]), k \in \{1, \cdots, K\}, \forall i \in \{1, \cdots, M\}$
7 Standardize $\widehat{L}[M]$ in the range of 0 to 1 to calculate anomaly score for each day as $A[M] = Standardise(\widehat{L}[M])$

(ADM), namely, ADM-I [33] and ADM-II [22]. Our approach, *Monitor* improves the accuracy in detecting anomaly up to 24% in best scenario and on average by 14%. The improved accuracy of *Monitor* results in lesser false alarms as compared to the existing approaches. The contributions of this chapter are:

- We propose a novel approach *Monitor* to detect anomalous instances of energy usage using smart meter data only.
- We evaluate *Monitor* using both commercial and residential buildings energy consumption data. Previous works have been extensively tested either under commercial [71, 33, 42, 52, 22, 58, 74, 43] or residential settings [54, 48, 22] only.

3.2 Methodology

Monitor takes time series power readings from a smart meter as input, and then for each day, anomaly score is computed separately. The anomaly score varies between 0 and 1. The higher the score, the higher are the chances of the day being

anomalous. *Monitor* consists of three steps. Algorithm 1 shows stepwise execution of *Monitor*.

Step-I: In this step, firstly, it transforms input power consumption data X of M days into a Matrix Y such that each row of Y contains power consumption of a consecutive day consisting of T readings. Next, it computes frequency density vector corresponding to each row of Y separately as Matrix $Y^{'}$, since the data is periodic and it is found that frequency representation is more sensitive to anomalies [75, 76]. Frequency representation is obtained using Discrete Fourier Transform (DFT) function [77]. It considers only magnitude part of DFT for further calculations. This contains all the necessary information of power consumption pattern.

Step-II: This step compares power consumption across all the input days of Step-I (Lines 3-4). Firstly, it computes the dissimilarity matrix $\Delta[M, M]$ by calculating the Euclidean distance between each pair of M days of $Y^{'}$. Euclidean distance reflects the underlying similarity in energy usage data. Another distance metric, Dynamic Time Warping (DTW) can also be used to measure the distance between two time-series energy signals. DTW first allows aligning according to the shape of the input series and then computes the distance whereas Euclidean metric computes the distance between corresponding points of input series directly. Since the occurrence of an anomaly in energy data is time-sensitive, so we use the Euclidean metric. Next, step-II reduces the dimensionality of $\Delta[M, M]$ to $\widehat{\Delta}[M, 2]$ by using Multidimensional Scaling (MDS) algorithm. MDS essentially provides a two-dimensional representation of higher (T) dimensional data. Two-dimensional representation makes it easy to compare the consumption among different days. Figure 3.2 shows corresponding MDS plots of different loads shown in Figure 3.1. Each • on the plot represents power consumption of a different day and a number below each • represents the respective day of the month. Each plot shows a different distribution of •, meaning each load follows a different energy consumption pattern. The distribution does not reflect the lower or higher power consumption days but shows the variation in power usage on various days. Days within a cluster consume energy in the similar pattern

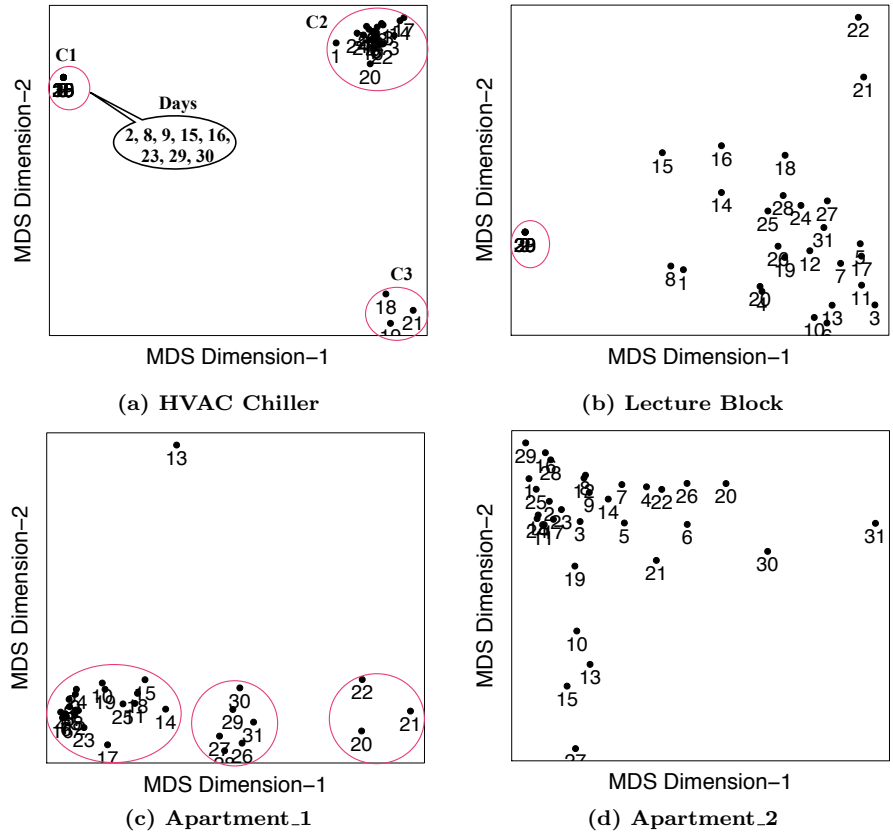


Figure 3.2: Multidimensional Scaling (MDS) plots of selected commercial and residential loads. Each • represents a different day and the distribution of • shows the variation in the average power consumption on different days of a month. Circles represent clusters, i.e., days with similar power consumption pattern.

and different from the days in remaining clusters.

Step-III: This step calculates the anomaly score for each day power consumption (Lines 5-7). It uses LOF to calculate the anomaly score, which has been extensively used in various domains such as network intrusion detection [78]. LOF takes $\widehat{\Delta}[M, 2]$ as input and represents each row with a single observation thus $\widehat{\Delta}[M, 2]$ reduces to M observations. Next, it computes density for each observation with its k neighbors. Finally, it computes anomaly score for each observation by comparing its density with neighboring observations. Observations with significantly lower density as compared to neighbors get higher anomaly score.

LOF scores do not have a fixed range, so to interpret the scores *Monitor* standardizes output LOF scores using the approach in [79]. This approach first performs regularization and then normalization to output scores between 0 and 1.

To compute anomaly score, k value is specified (Line 5) in LOF. A single specific value of k cannot be used for all the loads as each load follows different energy usage pattern and hence result in different data distribution. Therefore, range of k values are used to calculate LOF scores, similar to the approach in [80]. The final LOF score considered is the maximum of LOF scores calculated with different k's as suggested in [73]. Please refer section 3.5 for detailed information about selection of k and the final LOF score.

3.3 Evaluation

3.3.1 Dataset

We use a real-world dataset to evaluate the performance of the *Monitor*. It contains power consumption readings of both the residential and commercial loads at the IIIT-D campus.

The residential dataset contains energy consumption data of faculty apartments on the campus. Each apartment consists of three bedrooms, a hall, and a kitchen. These apartments are equipped with common home appliances, such as lighting systems, refrigerator, heaters, and air-conditioners. A smart meter is installed for *each apartment* separately. The residential apartments remain operative throughout the week (24 x 7). The apartments for our experiments were selected on the basis of: *i*) apartments with no missing readings, *ii*) apartments showing *diversity*. Diversity (Figure 3.1) refers to different power consumption patterns and we focus on this to check the competency of *Monitor*.

For the commercial dataset, we chose HVAC chiller, Lecture block due to two reasons: *i*) these loads mostly follow regular working hours, and *ii*) these are high energy consuming loads. The Lecture block comprises of 12 classrooms spread over the three floors. The Lecture block and HVAC chiller run usually from 0800 - 1700 hours on weekdays.

The sampling rate of smart meters is 30 seconds to get the fine-grained power

consumption details. The data from all the smart meters on campus are stored in a centralized meter data aggregation system using an open source system, sMAP [81]. This dataset consists of continuous sixteen weeks of data starting from I^{st} Aug. till 29^{th} Nov. 2015. We chose this period of the year primarily for two reasons: (*i*) This is a period of the academic semester, so everyone is on campus and every electric consuming equipment remains operational. (*ii*) This period of the year shows seasonality effect, where the months of August and September are hot and are most energy consuming while the remaining two months do not normally require AC and are low energy consuming months. We downsampled the data to hourly average readings as the power utilities generally collect data at hourly scale [82]. Therefore, the motivation is to check the performance of the *Monitor* on any diverse dataset collected by various power utilities. We have released our dataset[1] for the public use.

3.3.2 Baseline methods

We correlate the performance of *Monitor* with two baseline methods, i.e., ADM-I [33] and ADM-II [22]. We chose these well-known methods because both of these (*i*) use aggregate hourly meter readings, and (*ii*) compute anomaly score corresponding to each day. Both ADM-I and ADM-II use energy readings of n consecutive days to identify anomalous instances. For each day energy consumption, ADM-I uses k-NN (nearest neighbor) to compute the density and finally assigns an anomaly score with a global day having highest density. ADM-II is a clustering based anomaly detection approach. It uses k-medoid clustering algorithm and the number of clusters in data are computed using Partitioning Around Medoids algorithm.

3.3.3 Physical interpretation of k

In *Monitor* and the baseline methods, we need k-nearest neighbors to calculate the anomaly score for each observation that corresponds to entire day energy usage. k

[1] `www.bit.ly/2xX0Vsb`

nearest neighbors of an observation represent k other observations which deviate less from the considered observation as compared to remaining observations. Fig. 3.2(c) shows that energy usage on day 21 differs less from days 20 and 22 as compared to remaining days on the plot. So, for day 21, days 20 and 22 represent two nearest neighbors.

For *Monitor*, k value is set between 4 - 7, as suggested to use a range of k values in [73]. Please refer section 3.5 for sensitive analysis of k. For ADM-I, k is set to 6 as computed with the formula in the paper [33]. In ADM-II, an optimal k value is determined automatically using Partitioning Around Medoids algorithm.

3.3.4 Evaluation metrics

Anomaly score: The *Monitor*, ADM-I, and ADM-II calculate anomaly score for the energy usage of each day. The score varies between zero and one. The higher the score, the higher are the chances that the energy consumption of the day is anomalous.

The anomaly score can be presented through two different means [83, 9]: *i*) Top-s: In this method, a ranked list of anomaly score is sorted in the decreasing order, and then the building administrator chooses upper s days as being anomalous, *ii*) Threshold-based: In this method, building administrator selects a threshold on anomaly score, and then the days with score $\geq$ threshold are declared as anomalous. The range of the threshold varies between zero and one, where a value of zero means normal and one means anomalous with 100% confidence.

We use the threshold-based method as it gives finer control to an administrator, i.e., an administrator can check anomalies of any severity level which top-s does not allow. Further, in top-s, correct information is required to extract only anomalous instances [80]. We use a threshold value of 0.75 in all the three approaches to maintain consistency.

Area Under Receiver Operating Characteristic Curve: We used a standard metric known as Area under Receiver Operating Characteristic (ROC) curve to

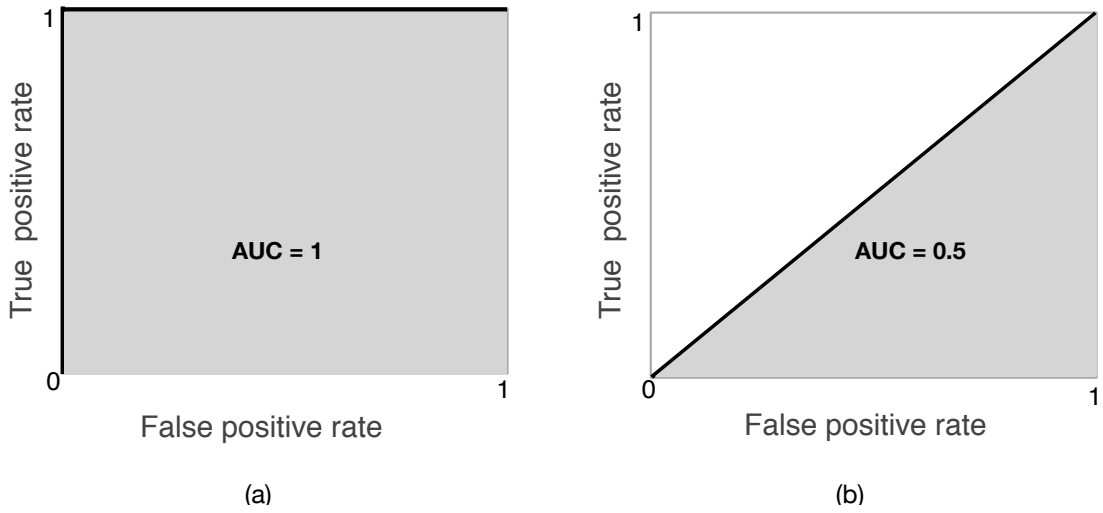


Figure 3.3: Receiver operating characteristic curves

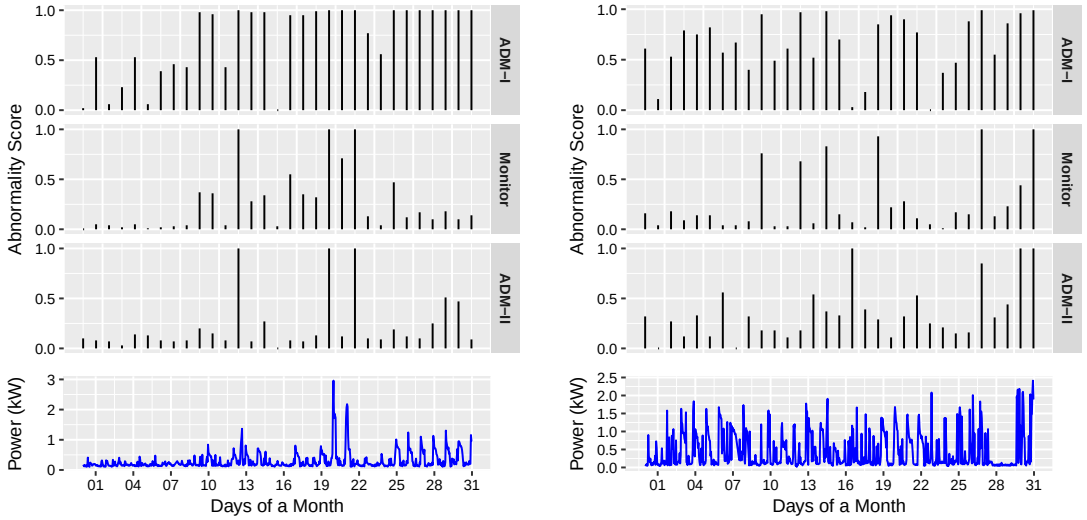
measure the performance of all the three methods at *different thresholds*[2]. The ROC curve is plotted between the false positive rate (FPR) and the true positive rate (TPR) of any binary classifier at different thresholds as shown in Figure 3.3. Where the TPR gives information about the correctly identified instances, and the FPR gives information about misclassified instances. The ideal ROC curve has a TPR of 1 and an FPR of 0 as shown in Figure 3.3(a). Figure 3.3(b) shows ROC curve of a classifier classifying instances randomly.

ROC curves are often summarized with with a single value called as **A**rea **U**nder ROC **C**urve (AUC). The AUC value ranges between 0 and 1. The higher the AUC value, the better is the performance of the method.

3.3.5 Anomaly verification

Collecting ground truth information in the energy domain is an open issue. There is no publicly available anomaly annotated dataset, so all existing works consult building administrator (BA) to mark the ground truth [33, 22]. Accordingly, we obtained ground truth information from our BA. He analyzed the raw power readings manually (visual inspection) and marked all the anomalous days using historical usage pattern and the service log book entries. We will discuss the reasons for

[2] `https://en.wikipedia.org/wiki/Receiver_operating_characteristic`



(a) Apartment_1 **(b) Apartment_2**

Figure 3.4: The raw power consumption of two residential apartments for a month and anomaly scores computed using ADM-I, ADM-II and *Monitor*

anomalous behavior as mentioned by the BA on a case by case basis.

3.3.6 Experimental setup

For all loads, anomaly score for each day was computed from August 1^{st} 2015 to November 29^{th} 2015 using all the three methods. All methods were run on a rolling basis, and the window size was set to one month. We set it to one month empirically, such that a window has enough number of weekend and weekday days. The `R` implementation of ADM-II is publicly[3] available, *Monitor* and ADM-I were implemented in `R` too. In *Monitor*, `Rlof` package was used for LOF, and for LOF score standardization `HighDimOut R` package was used [84, 85]. We have created a publicly-available implementation of *Monitor* on GitHub[4].

3.4 Results

This section initially presents the analysis of *Monitor* on residential and commercial loads, and later explains the sensitivity analysis of k. At the end, performance

[3] `https://github.com/pandarasamy/anomaly_detection`
[4] `https://github.com/loneharoon/LoF_Anomaly`

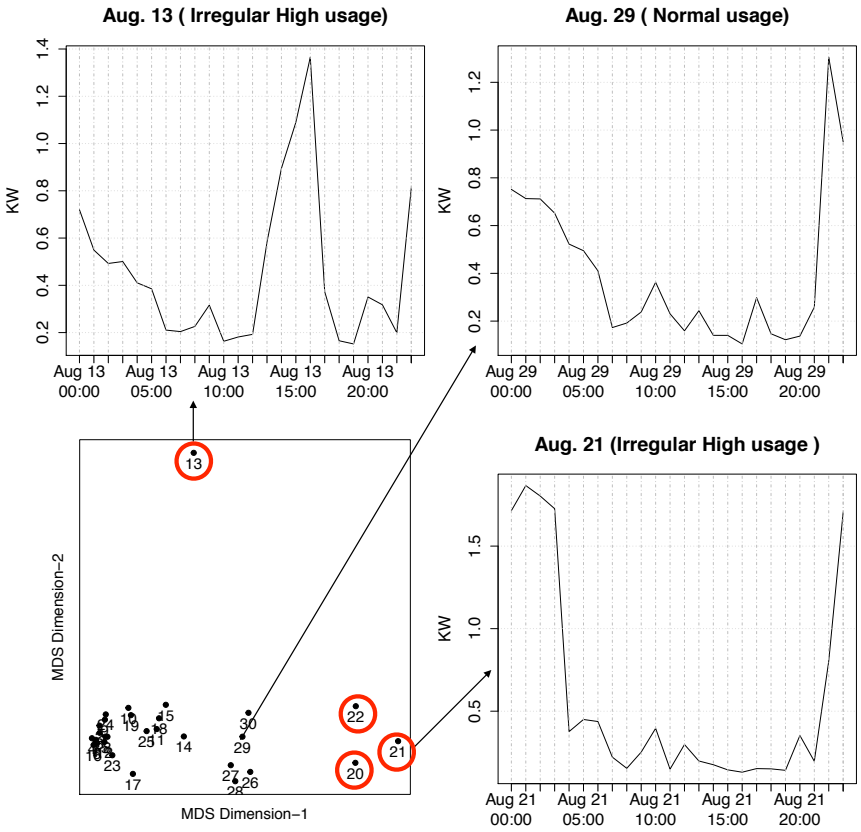


Figure 3.5: MDS plot along with power consumption pattern of selected days of Apartment_1. Encircled days represent anomalous instances.

comparison of *Monitor* with baseline methods ADM-I and ADM-II using standard Area Under Curve (AUC) [86] metric is presented.

3.4.1 Analysis of residential buildings' data

As we observe in Figure 3.4(a), hourly average raw power readings of apartment_1 shows two distinct patterns: (*i*) power consumption is below 1000 watts from Aug. 1 to Aug. 25 excluding days 13, 20, 21, and 22, (*ii*) power consumption is above 1000 watts from Aug. 26 until Aug. 30. It indicates that days of the considered month should result in two distinct clusters corresponding to two usage patterns found. This observation is confirmed by the MDS plot shown in Figure 3.5.

From raw power analysis, day 13 is anomalous due to unusual high power consumption between 1430 - 1620 hours as shown in a subplot of Figure 3.5. Similarly, days 20, 21, and 22 are anomalous due to high power usage during night hours.

Metric	Method	Apt_1	Apt_2	Lecture Block	Chiller
False Positives	ADM-I	15	9	7	20
	ADM-II	0	1	2	2
	Monitor	0	2	0	0
False Negatives	ADM-I	0	0	2	0
	ADM-II	1	1	2	2
	Monitor	1	1	3	1

Table 3.1: False Positives and False Negatives found at a threshold of 0.75 anomaly score with different methods. Short form, Apt in the column names refer to apartment.

Discussions with the apartment owner revealed all these instances were due to AC usage during unusual day hours.

Contrarily, the raw power usage of apartment_2 in Figure 3.4(b) does not show any distinct pattern across days; as a result, the MDS plot in Figure 3.2(d) does not lead to any separate cluster. The absence of any pattern in power consumption makes it difficult to identify anomalous days accurately. But, in Figure 3.2(d) we find days 10, 13, 15, 27, and 31 to be different from the rest, as these days are far away from the remaining days in the plot. On manual analysis of raw power readings and discussions with the apartment owner, we found that on these days AC was running for a longer duration (day 31) continuously and at irregular intervals of the day (days 10, 13, 15, and 27).

Table 3.1 shows false positives and false negatives for both residential loads using *Monitor*, ADM-I and ADM-II. In both Apartments, *Monitor* resulted in a single false negative at a threshold of 0.75. These false negatives are day 21 of Apartment_1 and day 13 of Apartment_2. *Monitor* has assigned an anomaly score of 0.74 to both of these days as shown in Figure 3.4(a) and 3.4(b), but as the threshold is set to 0.75 both of them resulted as false negatives. Total number of anomalous days in apartment_1 and apartment_2 are 4 (days 13, 20, 21, 22) and 4 (days 15, 27, 30, 31) respectively. We discuss the cause of high false positives in the ADM-I in section 3.6.

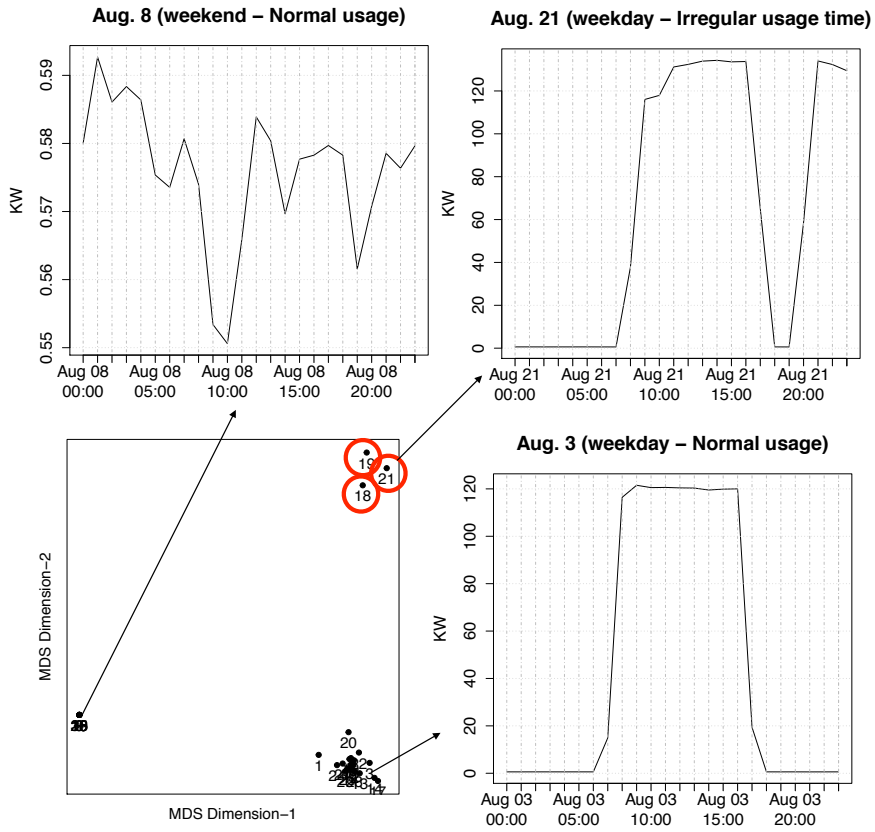
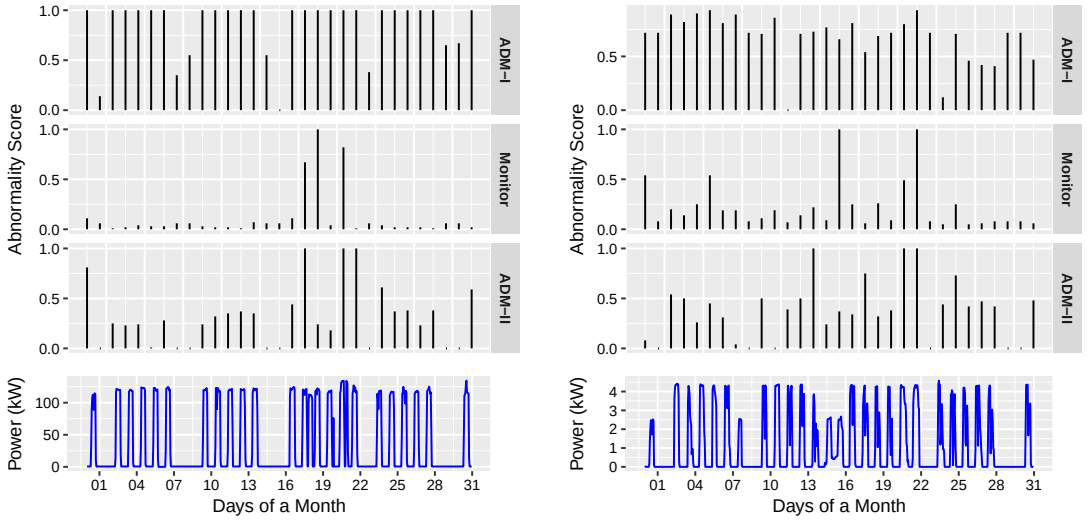


Figure 3.6: MDS plot along with power consumption pattern of selected days of Chiller. Encircled days represent anomalous instances.

3.4.2 Analysis of commercial buildings' data

Figure 3.6 shows the MDS plot of an HVAC chiller. It depicts two clusters, which correspond to energy usage of weekends and weekdays. It also shows that days 18, 19, and 21 are far from both the clusters centers. On examining the raw energy usage of 21, we found that Chiller was working through late night hours as depicted in the upper right plot of Figure 3.6. During conversations with the building management system (BMS) operator, we were informed that in IIIT-Delhi campus two HVAC chillers are installed, which run in alternate time periods. But, on days 18, 19 and 21 both chillers were running simultaneously for a duration of 2 hours due to high demand, and this resulted in anomalies on respective days.

In comparison to Chiller consumption, we find one tiny, dense cluster in the power consumption of Lecture block as shown in Figure 3.2(b). This cluster represents the weekend days as the power consumption is 0 for all these days. As the lecture block



(a) Chiller **(b) Lecture Block**

Figure 3.7: The raw power consumption of two commercial loads for a month and anomaly scores computed using ADM-I, ADM-II and the *Monitor*

consists of 12 classrooms, it shows more variation in the power consumption due to the difference in the timetable for each day and due to dynamic usage patterns (either all fans/lights are on or a few). Furthermore, we found that on some days either lights/fan were switched off completely during lunch hours or hack-nights were organized. This results in dynamic usage behavior on the different days of a month.

Figure 3.2(b) shows that days 21 and 22 are far from the remaining days, and Figure 3.7(b) indicates that the *Monitor* assigns high anomaly scores to days 16 and 22. On analyzing the raw power traces of these days we found: *i*) on day 16, there was a continuous usage of 500 watts (see raw power in Figure 3.7(b)) from 0000 - 0800 hours, which is unusual and hence high anomaly score is assigned. During discussions with the BA, we found that on some floor lights and fans were not switched off completely. *ii*) on day 22, usual power usage was extended from 1700 - 2100 hours. From the institute calendar, we found day 22 as Institute's technical festival day and hence resulted in extra usage. This analysis shows that usage on day 16 during morning hours was energy wastage, and the *Monitor* detected it.

Table 3.1 shows false positives and false negatives for both loads with *Monitor*, ADM-I and ADM-II. For these loads, we do not find a single false positive with

Monitor while ADM-I contains false positives in higher proportion. Total number of anomalous days in chiller and lecture block are 4 (days 18, 19, 20, 21) and 5 (days 14, 15, 16, 21, 22) respectively.

We differentiate the anomalies detected by various methods as *actionable* and *non-actionable*. Actionable anomalies represent energy wastage instances that can be eliminated upon timely detection. While as non-actionable represent significantly high energy usage instances that result due to some genuine cause. Although non-actionable anomalies cannot be eliminated but their detection helps in understanding the cause of high electricity bills and thus provide the knobs to consumers for optimal appliance settings. Our results show both actionable (Day 16, Lecture block) and non-actionable anomalies (Apartments 1 & 2, Chiller), but without discriminating the nature of anomaly detected we proved that *Monitor* is capable to detect both type of anomalies.

3.5 Sensitivity analysis

Optimal value of k: The performance of the *Monitor* depends upon k value, where k refers to the numbers of neighbors of an observation. We remove the sensitivity of *Monitor* on k by using *ensemble* approach as suggested in [80, 73]. In this approach, anomaly score is computed for a range of k values instead on a single value of k. This approach is important as it ensures that not a specific k value affects the anomaly score of an object. We used k values in the range of 4 to 7. To find the lower limit, we varied k in the range of 2 to 10 for all the loads. On analyzing the results, we found that anomaly scores approximately remain same for the k range of 2 to 4, so to reduce the computing cycles we set lower limit = 4. And we set higher limit = 7 owing to the reason that in most cases energy usage follows a weekly pattern due to seasonality or occupancy behavior. This means that on average the energy usage should follow similar usage pattern for 7 days.

Figure 3.8 explains the effect of diversity on anomaly score with different *k's*. We observe two patterns in the Figure: *i*) low variability score with different k, e.g.,

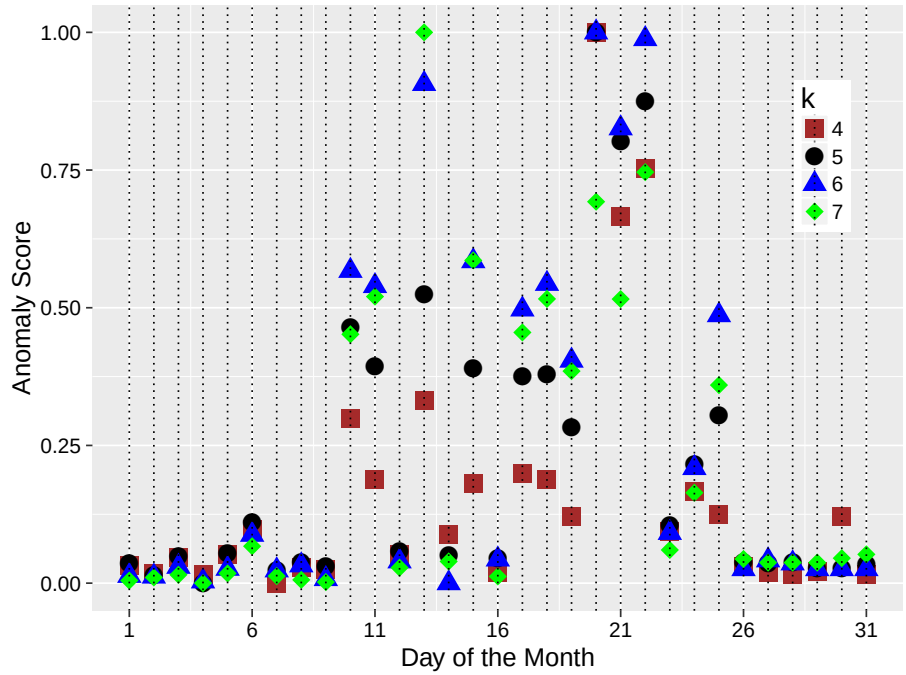


Figure 3.8: Comparison of anomaly scores within each day with different values of k (4 to 7) for the Apartment_1. Days 10 – 25 show distinct variability in score due to variation in calculated density with different k neighbors. **[Best viewed in color]**

days 1 - 9, and 26 - 31, *ii*) high variability in score with different k, e.g., mostly days between 10 - 25. We illustrate the effect of k on anomaly score as: assume anomaly threshold is set to 0.75, then days with anomaly score $\geq$ 0.75 are considered as anomalous. With the assumed threshold and the k value of 4 or 5, day 13 is non-anomalous while as day 20 is anomalous; on the other hand with the k value of 7, day 13 is anomalous while as day 20 is normal. But, we know both of these days are anomalous from the ground truth information (Section 3.3). This ambiguity shows that we can not rely on a particular k value, hence we should use ensemble approach as suggested by [73, 80] to find the outlier nature of an object.

Final anomaly score: We find different methods to combine the anomaly scores corresponding to different k, which include: *i*) Breadth First Scheme (BFS) [87], *ii*) Cumulative sum [87], *iii*) Mean [73], and *iv*) Maximum value [73]. In both the BFS and the Cumulative sum, anomaly scores on different k are sorted and ranked according to decreasing order of their magnitudes. We empirically computed anomaly scores with all the four, and we chose Maximum value method over the rest

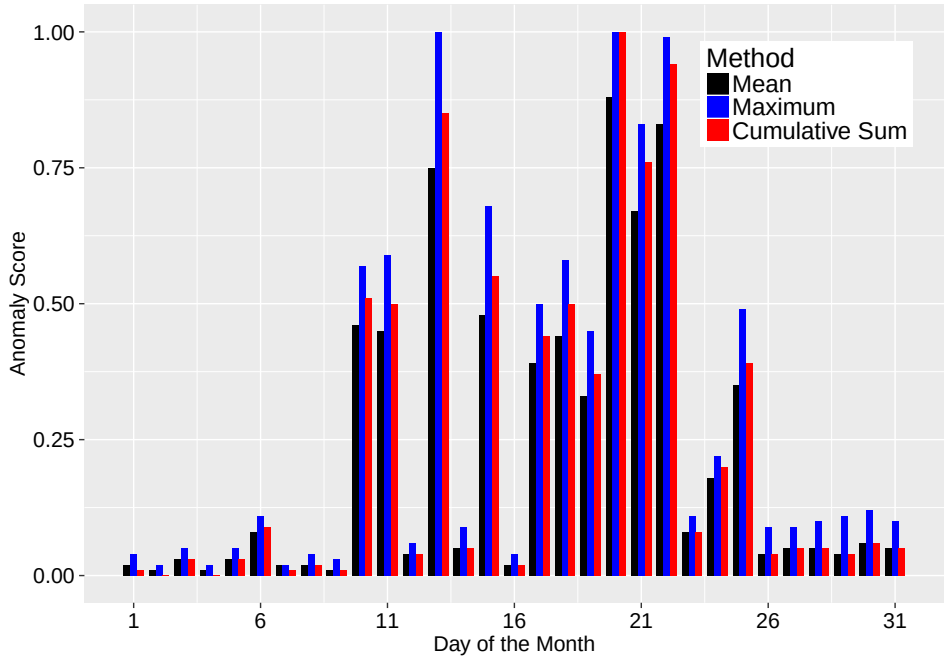
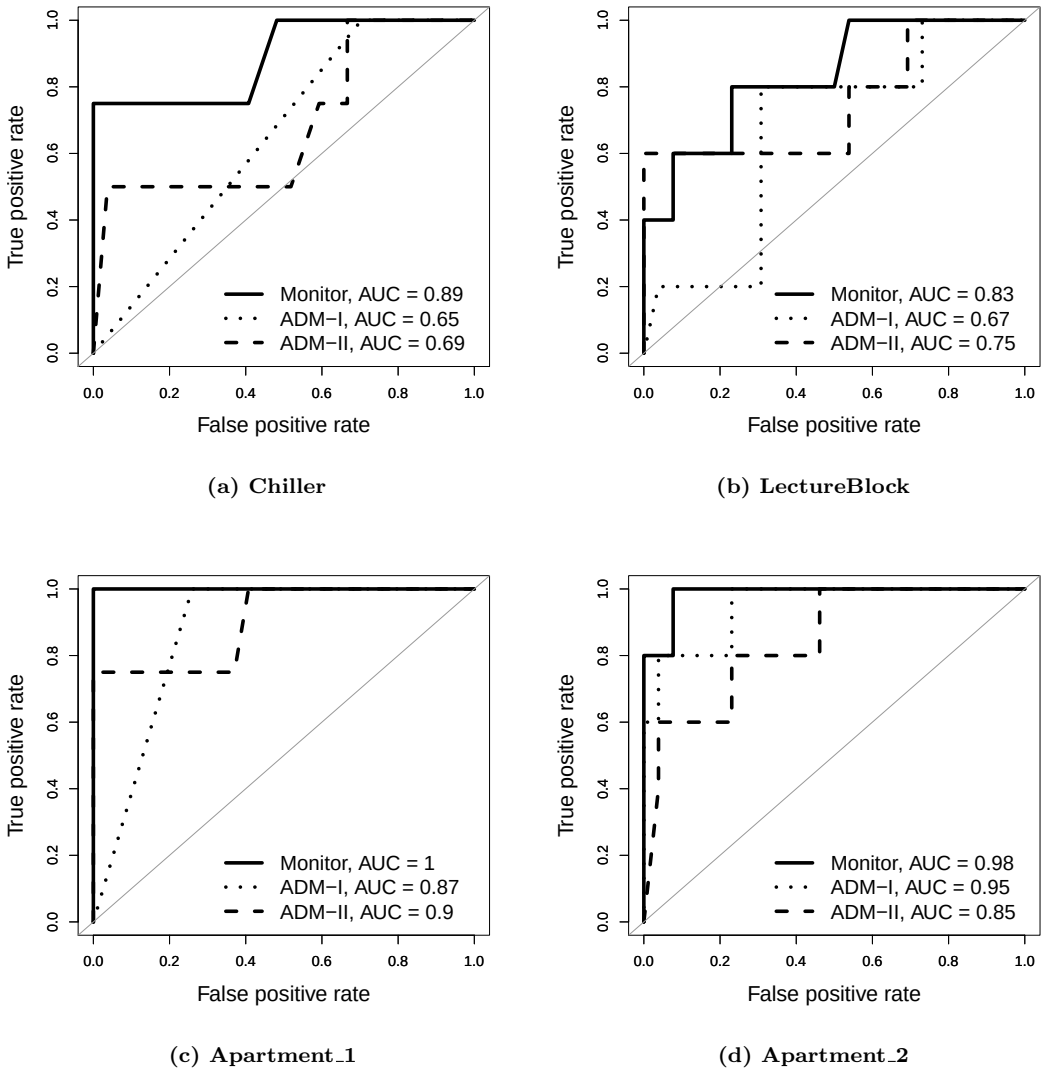


Figure 3.9: Anomaly scores computed using different aggregation methods (Mean, Maximum, and Cumulative) for Apartment_1 dataset. Each method aggregates the anomaly scores calculated using different k values in the range of 4 to 7. [**Best viewed in color**]

as it always highlights the outlier instance, if an object behaves as an anomalous with any k. Also, the maximum value method is least sensitive to the cut-off value considered for the objects to be anomalous. For days 13, 21 and 22, Figure 3.9 shows the effect of maximum value method on anomaly score in comparison with other methods. The mean and cumulative sum dilutes the outlier nature of an object as shown in Figure 3.9. While as, in some datasets we found BFS is sensitive to the order of anomaly detection method, and this is highlighted in [87].

3.6 Performance comparison

We use Area Under Curve (AUC) to correlate the performance of *Monitor* with the baseline methods. Figure 3.10 shows receiver operating characteristic (ROC) curves for all loads with different methods. The true positive rate (TPR) in ROC curve shows the number of correctly labeled anomalies while the false positive rate (FPR) depicts number of normal data points wrongly labeled as anomalous. The



(a) Chiller **(b) LectureBlock**

(c) Apartment_1 **(d) Apartment_2**

Figure 3.10: Comparison of ROC curves computed using baseline ADM-I, ADM-II and the proposed *Monitor* methods for different loads. It shows that the *Monitor* outperforms ADM-I and ADM-II as the AUC is high for all the loads.

ideal ROC have TPR of 1 and FPR of 0. The performance is measured from ROC in terms of AUC value, which ranges between 0 and 1. The higher the value, the better is the performance [86].

Table 3.2 shows AUC values for *Monitor*, ADM-I and ADM-II. For instance, AUC value for *Monitor*, ADM-I and ADM-II in chiller is 0.89, 0.65, and 0.69 respectively. It is observed from the AUC values that the *Monitor* performs better than both ADM-I and ADM-II. Higher AUC of *Monitor* is attributed to its lower FPR as compared to the baseline methods. Note that ROC curve is not drawn at a particular threshold. Instead, it uses different thresholds automatically [86].

Method	Chiller	Lecture Block	Apartment_1	Apartment_2
Monitor	0.89	0.83	1.00	0.98
ADM-I	0.65	0.67	0.87	0.95
ADM-II	0.69	0.75	0.90	0.85

Table 3.2: Area under curve (AUC) values for different loads using *Monitor*, ADM-I, and ADM-II

Chiller	Lecture Block	Apartment_1	Apartment_2	Average
24	14	10	8	**14**

Table 3.3: AUC percentage Improvement in *Monitor* over ADM-I & ADM-II

Table 3.3 summarizes percentage improvement in the AUC values of *Monitor* over the baseline methods.

1. Insights: Figures 3.4 and 3.7 show calculated scores by *Monitor*, ADM-I and ADM-II and Figure 3.2 shows the corresponding MDS plots. While looking at the MDS plots and the corresponding assigned anomaly scores by the ADM-I we infer that *"if there are clusters, i.e., dense clusters (e.g., Chiller, Apartment_1) found in power consumption, then the ADM-I results in high false positive rate"*. While as with the *Monitor* we do not find any impact of presence or absence of clusters on the score calculation. This inference shows that *Monitor* can be used in any similar type of energy consumption scenario as shown in Figure 3.2, ensuring minimal FPR.

2. High False Positive Rate in ADM-I: The anomaly score, a probability value is calculated by using a density estimation algorithm, namely k-NN. For each day's power consumption (observation), firstly density is calculated using k-NN, and then probability value (anomaly score) is calculated with respect to the observation having highest density. This essentially means that a *single* observation with highest density influences the anomaly scores of all the remaining observations in a window.

We illustrate this with an example: In Chiller (Figure 3.2(a)), we found that the day 2 has highest local density of 13.7 and the average density of days (2, 8, 9, 15, 16, 23, 29, and 30) clustered in the same cluster as of day 2 is 9.17. Also, the average density of remaining 22 days of the month is 0.00020. With this statistics,

it is clear that on average, each week/working day is given an anomaly score of 0.99 (= 1 - 0.00020/13.7) and each non-working day is given 0.33 (= 1- 9.17/13.7). This example illustrates that ADM-I should not be used in scenarios where we have $1-2$ clear dense clusters, as in the case of Chiller. In such scenarios, an observation belonging to the densest cluster always affect the anomaly score calculation badly, and hence results in high FPR.

3. High Accuracy of *Monitor*: The clusters in data essentially depict that the points within a cluster behave similarly and different from the points in other clusters. The denser a cluster is, higher is the similarity between data points. Therefore, density is the most important metric to find similar data points.

Monitor takes advantage of this observation and uses well-known density-based LOF technique to identify anomalous days [73]. LOF uses only nearby points to calculate the anomaly score of any data point, *i.e.*, only points within a cluster are used to calculate score. We elucidate this further with an example as - consider Figure 3.2(a) in which we find three clusters $C1$ (represents weekend days of Aug. 2015) and $C2$ (working days of Aug. 2015) and $C3$ (anomalous days). There is no relation between $C1$ and $C2$ since both of these represent energy consumption of two different type of days. Hence, we should not use data points of one cluster to calculate anomaly score of some other data point in a different cluster. LOF uses the same concept and hence results in less FPR.

3.7 Summary

Automated anomaly detection approaches detect energy wastage timely. Unfortunately, building administrators often ignore the alarms due to their high false positive alarm rate. In this process, genuine (true positives) alarms may also get ignored leading to energy wastage. This unreliability makes the existing anomaly detection approaches useless. This chapter presented *Monitor* – a reliable anomaly detection approach which reduces false alarms to a greater extent. The lesser rate of false alarms makes *Monitor* applicable for real-world building management systems.

Monitor computes anomaly score for each day after having observed the energy consumption trend for the day. This approach has two limitations: *i*) if a day gets high anomaly score, building administrator has to manually go through the entire day (24 hours) energy consumption log to find the exact anomalous usage time interval and to find the exact cause; *ii*) if anomaly occurs in the early hours of day, it results in energy wastage for the entire day. In the next chapter, we present another anomaly detection approach which handles both of these limitations.

Chapter 4

Real-time contextual anomaly detection at the household level

4.1 Introduction

The current anomaly detection techniques using smart meter data only detect anomalies in *offline mode* and do not *identify* the appliance causing the anomaly [71, 54, 42, 33, 22]. Identification of such appliances in *real-time* mode will allow consumers to take prompt action and result in energy savings. Furthermore, identifying an anomalous appliance among n appliances of a home using a single smart meter is more economical than using n separate monitors for each appliance. Current offline techniques report the anomaly status at the end of the day [71, 54, 42]. These techniques use power data throughout an entire day to detect whether the day has anomalous usage or not [33, 22]. In related areas, such as energy prediction [88, 89, 90], contextual information (such as day of week or external weather conditions) have been used, but none of the existing anomaly detection techniques in the energy domain uses such information. We believe that adding contextual information should increase the anomaly detection accuracy because the energy consumption of buildings is a function of these contextual factors.

In this chapter, we propose a novel technique called RIMOR[1] - that can *iden-*

[1]In Latin, RIMOR means investigator, explorer

tify the appliance causing the anomaly by using aggregate smart meter data and contextual information in real-time mode. The basic intuition behind RIMOR is that homes should have similar temporal energy patterns if we account for variations in context, such as weather. RIMOR works in three steps. Firstly, it predicts the energy consumption of a target day by using historical energy data and contextual information. Secondly, it flags the energy consumption to be anomalous if there is a significant difference between the predicted energy consumption and the actual energy consumption. Lastly, it *identifies* the anomalous appliance by comparing the difference between the predicted and the actual energy consumption to the rated power consumption of appliances present in the household. The rated power consumption of different home appliances is publicly available and can be easily collected [91].

We evaluate RIMOR on 51 homes from four different real-world publicly available datasets namely Dataport [21], AMPds [92], ECO [93] and REFIT [20]. All these datasets contain appliance level consumption data in addition to aggregate home consumption collected using smart meters. None of these datasets are annotated with anomalies. Moreover, no anomaly-annotated dataset is publicly available in the energy domain. Previous works [71, 33, 42, 22, 34, 70] have manually annotated datasets using domain experts, but they have not been released. For the first time, we annotate these datasets manually and make them available to the research community. We compare RIMOR with four other baselines: Multi-User Anomaly Detection (MUAD) [22], Monitor [34, 24], Twitter Anomaly Detection (TAD) [94, 44] and Real-time Anomaly Detection (RAD) [52]. We find that RIMOR reports an improvement of **15%** in F-score in the detection of anomalies as compared to the baseline approaches. The anomalous appliance identification accuracy is found as **82%**. Our analysis shows that contextual information accounts for a 16% improvement in energy prediction. We release RIMOR as an open-source web application[2] that can potentially *identify* anomalous appliance in any home that has a smart

[2] `https://github.com/loneharoon/AnomAppliance`

meter installed. Smart meters are now becoming ubiquitous IoT devices to manage countries' energy demands[3].

The contributions of this chapter are:

- We propose a near real-time anomaly detection technique, RIMOR, which identifies the anomalous appliance by using easily available appliance's power ratings. Further, we show that adding contextual information improves anomaly detection accuracy.
- We evaluate RIMOR with four other existing techniques. All these techniques can only detect the anomaly but do not identify the anomalous appliance. The evaluation shows that our technique improves detection accuracy by 15%.
- We make the anomaly-annotated dataset public[4], along with a web application of RIMOR.

4.2 Methodology

The overall goal of RIMOR is to detect anomalies and *identify* the anomaly-causing appliances in real-time mode. To achieve this goal, RIMOR uses contextual information for a more accurate detection of anomalies. For example, homes can have different energy patterns on weekdays and weekends, and accounting for those differences can lead to better anomaly detection. RIMOR uses two types of contextual information: real-valued (such as temperature and humidity) and binary (such as: is it the weekend?) denoted by K_{real} and K_{binary} respectively. Algorithm 2 provides an outline of steps in RIMOR.

Since RIMOR aims to detect anomalies in real-time mode, we divide a day into W equally spaced chunks (for example $W = 24$ means 24 chunks of an hour each). Now, our problem reduces to detecting each of these $w_i \in W$ (where $i \in \{1, \ldots, |W|\}$) chunks as anomalous or not, and for anomalous chunks identifying the anomaly

[3] `https://goo.gl/5GuaUJ`
[4] `https://goo.gl/SjozdF`

ALGORITHM 2: Steps in proposed anomaly detection approach, RIMOR

Input: Historical power consumption Y_{train} of N train days; Contextual variables K_{real}, K_{binary}, and actual energy consumption Y_{test} of a test day; and power ratings a_l^u of n appliances present in home

Output: Anomalous appliance

1 Predict energy consumption $\widehat{Y_{test}}$ and prediction band $\widehat{Y_{test,error}}$ for the test day using Y_{train}, K_{real}, and K_{binary}

2 Compare Y_{test} with $\widehat{Y_{test}}$ and $\widehat{Y_{test,error}}$ to flag anomalous time instances of the test day

3 Identify anomalous appliance if any anomalous time instances found with

$$\arg\min_{a_l} \ (abs(\widehat{Y_{test}} - Y_{test}) - a_l^u), \forall l \in \{1, \cdots, n\}$$

return a_l `/* i.e., the anomalous appliance,if any */`

causing appliance. An important consideration in RIMOR is that the number of samples (T) collected from the smart meter should be greater than the chunk duration. Higher T will provide more points in a chunk and would likely be better for deciding whether the chunk has anomalous usage or not. RIMOR uses following three steps to detect anomalies and identify anomalous appliance:

I. Energy prediction: This step takes energy consumption data of the previous N days ($Y_{train}^d, \forall d \in \{1, \ldots, N\}$), the K_{real} values, and the K_{binary} contextual data points of a test[5] day as input variables to predict the energy consumption ($\widehat{Y_{test}}$) of test day.

The relation between the input variables and the energy consumption of test day can be either linear or non-linear. So two different regression approaches are used, i.e., linear regression and neural networks. Linear regression models the linear relationship between input variables ($Y_{train}^d, K_{real}, K_{binary}$) whereas neural networks model non-linear relationship, if exists [95].

The proposed regression model is summarized as follows:

$$\widehat{Y_{test}^t} = C + \sum_{d=1}^{N} \alpha^d Y_{train}^{d,t} + \sum_{j=1}^{K_{real}} \beta^j Z_{real}^{j,t} + \sum_{j=1}^{K_{binary}} \gamma^j Z_{binary}^{j,t} + \epsilon_{test}^t, \forall t \in \{1, \ldots, T\} \quad (4.1)$$

[5]Test day refers to a day for which prediction is to be done, and train days refer to historical days

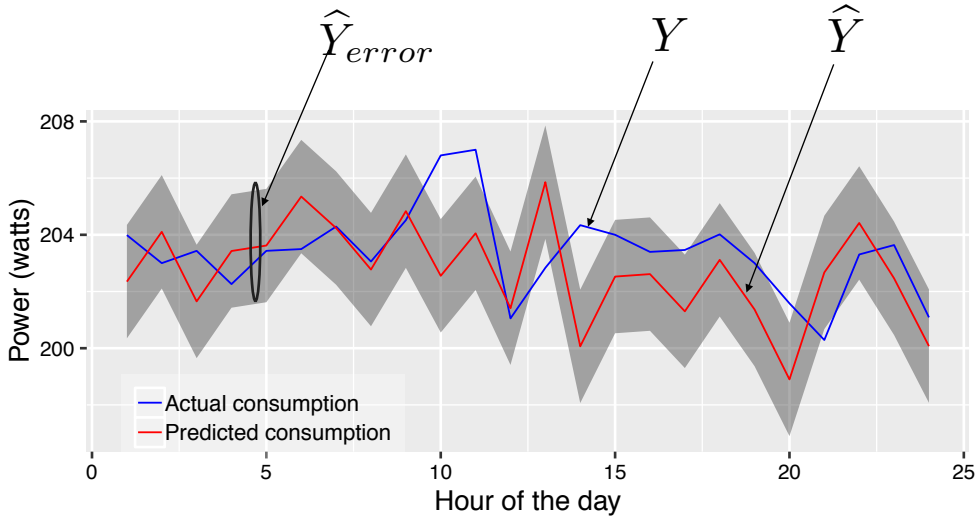


Figure 4.1: Visual representation of actual consumption (Y), predicted consumption ($\widehat{Y}$), and the prediction band ($\widehat{Y}_{error}$).

where $\widehat{Y^t_{test}}$ represents predicted energy consumption at the t^{th} time instant of a *test* day, C represents the intercept term. α^d is the coefficient corresponding to energy consumption (Y^d_{train}) of d^{th} historical day; $Z^{j,t}_{real}$ and $Z^{j,t}_{binary}$ represent the value of the j^{th} real and binary external context variables at the t^{th} time instant of the day. β^j and γ^j correspond to the coefficients for the j^{th} real and binary contextual variables, and ϵ^t_{test} represents modelling error.

In addition to $\widehat{Y^t_{test}}$, the prediction error band is computed as:

$$\widehat{Y^t_{test,error}} = \pm\delta * \sigma^t, \forall t \in \{1, \ldots, T\} \tag{4.2}$$

where σ^t represents standard deviation at the t^{th} time instant of a day, and $\delta \in \{1, \ldots, 3\}$ represents number of standard deviations. Figure 4.1 shows a visual representation of the predicted consumption ($\widehat{Y}$) and the prediction error band ($\widehat{Y}_{error}$). Note that $\widehat{Y}$ and $\widehat{Y}_{error}$ are generalized representations of $\widehat{Y^t_{test}}$ and $\widehat{Y^t_{test,error}}$ respectively.

Neural networks take same variables ($Y^d_{train}, K_{real}, K_{binary}$) as input and output $\widehat{Y^t_{test}}$ and $\widehat{Y^t_{test,error}}$. The exact relationship between input variables keeps changing so we cannot represent any specific form of the equation here. The set parameters of neural networks are defined in Section 4.3.5 (Experimental Setup) in detail.

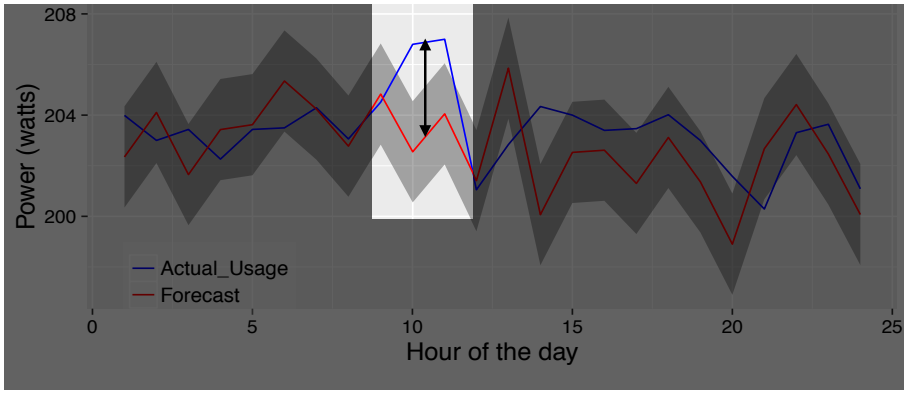
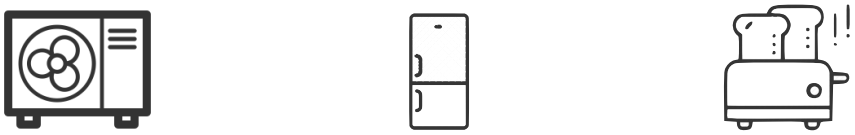


Figure 4.2: Figure highlighting an anomalous chunk (w_i).



(a) AC (b) Refrigerator (c) Toaster

Figure 4.3: Home appliances with different power ratings.

Mostly, the energy consumption throughout a day does not follow complete *stationarity*[6], so we created four separate models corresponding to every six consecutive hours of a day in both regression and neural network approaches. In this way, daytimes at which complete stationarity is violated do not affect the prediction accuracy of the remaining times of the day.

II. Anomaly Detection: This step detects whether the actual energy consumption Y_{test} on a test day of any chunk w_i ($w_i \in W$) is anomalous or not. It uses a single rule - if Y_{test} deviates significantly from $\widehat{Y_{test}}$ for a duration S ($S <= \texttt{timeduration}(w_i)$) of chunk w_i, then w_i is considered as anomalous. Algorithm 3 mention the steps in detail.

Note that the first `if` statement in the Algorithm 3 considers only the upper prediction band because appliances mostly consume higher energy in an anomalous state than in their normal state. Figure 4.2 highlights an anomalous chunk when the actual energy consumption is found outside the prediction band.

[6]Mostly, at morning and evening times, energy consumption is higher and different than remaining times of a day

ALGORITHM 3: Check whether a chunk is anomalous

Input: chunk w_i, Y_{test}, $\widehat{Y_{test}}$, $\widehat{Y_{test,error}}$, S
Output: binary output, whether w_i is anomalous

```
count = 0
for t = 1 to length(w_i) do
    /* t is a time-series index, used to identify required
       observations of w_i in Y_test, Ŷ_test, Ŷ_test,error          */
    if Y^t_test > (Ŷ^t_test + Ŷ^t_test,error) then
        count = count + 1
if count ≥ S then
    print (chunk w_i is anomalous)
    return 1
else
    return 0
```

III. Anomalous Appliance Identification: Once an anomalous chunk is detected, the next step is to *identify* the anomalous appliance a_l which resulted in an anomaly from the set of n appliances $A = \{a_1, \ldots, a_n\}$ present in household. This step uses all n appliance's power ratings (in Watts) to identify the anomalous appliance.

Typically, each home appliance has a different power consumption wattage. For example, AC consumes 1.5 kW, a refrigerator has a wattage of 150 watts, and a toaster has a wattage of 800 watts as mentioned in Figure 4.3. While knowing this, we assume that an appliance with higher wattage will result in an anomaly signature that is different from an anomaly caused due to an appliance of lower wattage, i.e., an anomaly by AC will be of higher magnitude than that of a refrigerator.

Let a_l^u denotes the power rating of appliance a_l where $a_l \in \{A\}$. Among n appliances, the anomalous appliance is identified with the following equation

$$\arg\min_{a_l} (abs(\widehat{Y_{test}} - Y_{test}) - a_l^u), \forall l \in \{1, \ldots, n\} \tag{4.3}$$

i.e., the appliance a_l which *minimizes* the difference between $abs(Y_{test} - \widehat{Y_{test}})$ and its power rating (a_l^u) is flagged as anomalous.

Dataset	Homes used (#)	Sampling rate (s)	Appliances per home (#)	Country
Dataport	24	60	09	USA
AMPds	01	60	20	Canada
ECO	06	01	08	Switzerland
REFIT	20	08	09	UK

Table 4.1: Dataset characteristics

4.3 Evaluation

4.3.1 Dataset

Energy data: We use four different publicly available datasets spanning four countries, for the evaluation of RIMOR. The datasets include Dataport [21], AMPds [92], ECO [93] and REFIT [20]. Table 4.1 shows the details of each dataset. We used consecutive three months of energy data with minimal missing values from each home for anomaly detection. The selected months were: June to August 2014 from Dataport and REFIT, January to March 2014 from AMPds, and August to October 2012 from ECO. While the Dataport dataset has data from more than 500 homes, we chose 24 homes having a consistent set of appliances. These datasets logged power readings at rates varying from 1 to 60 readings a minute. However, most electrical utilities log data once every 10 to 15 minutes [96]. We downsampled the data from all the datasets to 10 minutes.

Weather data: Weather data, i.e., temperature and humidity required for predictions is obtained from a publicly available weather service, WeatherUnderground[7]. The data is available at the sampling rate of thirty minutes to one hour. We upsampled weather data to the same rate of 10 minutes as energy data via interpolation.

Appliance ratings: Power ratings of any appliance can be easily obtained by knowing the make and model of the appliance. However, in our datasets, only Dataport contained appliance make and model, and that too for a limited number of homes and appliances. Thus, we obtained the appliance power ratings using the

[7] `https://www.wunderground.com/`

appliance power traces made available in these datasets. Most appliances can be modeled as finite state machines (FSMs), where each state corresponds to a mode of operation (e.g. off, on, intermediate) and has an associated power draw (e.g. 0 Watts when off, 200 Watts when on) [97]. We learned the appliance ratings, i.e. the amount of power draw in different states of operations using standard clustering procedures [98].

4.3.2 Ground truth collection

Collecting ground truth information about anomalies in the building energy domain is considered a challenging and tedious task [33]. Very few studies [38] had access to the necessary infrastructure to automate the process of collecting ground truth partially. This ground truth labeling task is inherently manual and requires a domain expert to analyze (through visualizations) the total building energy data, appliance energy, weather parameters, and service logs. Most prior works [71, 33, 42, 22, 34, 70] use domain expertise to label anomalies. We followed the same approach to label the ground truth in these four datasets by consulting with a domain expert. We explain the labelling process separately at `https://github.com/loneharoon/PowerViz` in detail. Here, we briefly mention the steps followed.

1. Plot power consumption for each month in a subplot pattern, where each subplot shows aggregate and appliances consumption for separate days.

2. Plot weather variables temperature and humidity of the same duration in subplot style.

3. Analyze all these plots at once and identify the deviations, if a significant deviation was observed in aggregate consumption and weather variables do not explain the deviation then we labeled that an anomaly.

Datasets used in prior works are not publicly available, but we release our anomaly labeled dataset for public reuse[8].

[8] `https://goo.gl/SjozdF`

Dataset	Min # per day	Max # per day	Mean # per day	Total # of anomalies
Dataport	0	3	0.35	480
AMPds	0	1	0.25	18
ECO	0	2	0.08	30
REFIT	0	2	0.21	280
Total	0	3	0.22	808

Table 4.2: Statistics of anomalies present in datasets of three months duration.

Dataset	Top 3 anomalous appliances (% contribution)
Dataport	Dryer (35%), Air conditioner (9%), Dishwasher (7%)
AMPds	Dryer (58%), Heat pump (29%), Oven (5%)
ECO	Fridge (54%), Dryer (27%), Washing machine (18%)
REFIT	Dishwashwer (34%), Dryer (25%), Washing machine (7%)

Table 4.3: Statistics on the top-3 anomaly causing appliances and the percentage of anomalies contributed by them.

In total, 808 anomalies were observed across the four datasets as shown in Table 4.2. Dataport and ECO had the highest and the lowest mean number of anomalies per day, respectively. Table 4.3 shows the top-3 anomaly causing appliances across four datasets. Appliances, such as dryer, dishwasher, and washing machine cause the maximum number of anomalies. All these three appliances generally contribute significantly to the overall energy consumption in a home[9].

4.3.3 Baseline techniques

We compare the performance of Rimor with four other techniques:

I. Multi-User Anomaly Detection (MUAD)[22]: It breaks down the daily power consumption in W chunks as we did in our technique. For each chunk, this technique has two steps. First, it computes anomaly score for energy consumption by using *k-medoid* clustering algorithm for each home separately. Next, it compares anomaly scores of all homes in the same geographical locality to adjust the final anomaly score.

[9] `http://www.eia.gov/consumption/residential/data/2009/`

II. Monitor [34, 24]: This technique takes as an input the same feature vector as MUAD. However, instead of *k-medoid* clustering, it uses Local Outlier Factor (LOF), a density-based approach to compute the anomaly score for the energy consumption in each user-defined time interval. It assigns an outlier score, i.e., the degree of outlierness, to each object based on the neighboring objects only.

III. Twitter Anomaly Detection (TAD) [94, 44]: In 2015, Twitter released an open-source anomaly detection package. This uses seasonal hybrid Extreme Studentized Deviate (ESD) algorithm, which builds upon Generalized ESD test.

IV. Real-time Anomaly Detection (RAD) [52]: This technique firstly uses an auto-regressive neural network-based approach for prediction, and later observation which lies outside two standard deviations from the predicted consumption are flagged as anomalies. The two main limitations of this technique are: (i) predictions are made a week ahead, which means predictions do not include the effect of recent historical days; (ii) detected anomalous observations are not removed from training data for future predictions. Therefore, anomalous observations affect predictions in this technique.

4.3.4 Evaluation metrics

The metrics used to measure the accuracy of RIMOR are defined as:

F-score: It is interpreted as a harmonic average of precision and recall as

$$\text{F score} = 2 * \frac{precision * recall}{precision + recall} \tag{4.4}$$

Precision measures the percentage of true anomalies against the total number of reported anomalies. Recall measures the percentage of true reported anomalies to the total number of anomalies present. The value of the F-score ranges between [0, 1]. The higher the value, the better is the performance.

Appliance identification accuracy: Anomaly detection results in true positives, true negatives, false positives, and false negatives. To compute the true appliance identification accuracy, we consider only true positive cases by using the following

formula

$$\textit{Identification Accuracy} = \frac{\textit{Total \# of correct identified appliances}}{\textit{Total \# of true positive anomalies}} \tag{4.5}$$

Symmetric mean absolute percentage error: The anomaly detection performance of RIMOR depends on the accuracy of the prediction (Step 2: Anomaly detection). We use a standard performance metric, symmetric mean absolute percentage error (SMAPE) [99], to measure the prediction accuracy of regression and neural network models. SMAPE is defined as

$$SMAPE = \sum_{t=1}^{T} \frac{|\widehat{Y^t_{test}} - Y^t_{test}|}{|\widehat{Y^t_{test}}| + |Y^t_{test}|} \tag{4.6}$$

Where, $\widehat{Y^t_{test}}$ and Y^t_{test} are predicted and actual values of power consumption data at time instant t. The lesser the value of SMAPE, the better is the prediction accuracy.

4.3.5 Experimental setup

For Rimor technique: We use leave-one-out cross-validation to find the optimal number of historical train days N using Bayesian Information Criteria (BIC). N was varied between 1 to 21 and was finally set at $N = 4$ as it resulted in the lowest BIC value. Anomaly detection rate, $W = 24$, is done at an hourly rate, and the S parameter is set to 30 minutes. We will present the sensitivity analysis of all these parameters in Section 4.5. Note that there is no division of training and testing dataset, RIMOR works in a rolling window manner, where the current day for which prediction (or anomaly detection) is done is considered as a test day and historical days are considered as train days. It must be noted that an outlier or anomaly in the train days can lead to learning incorrect models. Thus, we check and remove such outliers in train days before training the regression model or the neural networks

using Rlof `R` package[10].

Existing implementation of neural networks in the `Caret` package of `R` was used to get prediction results. With cross-validation, the optimal neural parameters were found as: number of hidden nodes = 10, weights = 0.05, and the number of repeats as 500. For anomaly detection in Step 2 of RIMOR, two standard deviation rule was used to flag the anomalies, i.e. $\delta = 2$. All parameters were set empirically, due to space constraints we will show the analysis of important ones only. The implementation code of RIMOR is publicly available in the form of `R` shiny application at Github[11].

For baseline techniques: The `R` code of MUAD and Monitor were provided by their authors. Implementation of TAD is publicly available as an `AnomalyDetection`[12] `R` package. RAD results were obtained using the `R Forecast` package as used by the authors of the paper. For all the four baseline techniques, we used the parameter values as suggested by their authors for optimal performance.

4.4 Results

4.4.1 Anomaly detection performance

Our main result in Figure 4.4 shows that our approach, RIMOR, reports an average 15% better anomaly detection accuracy compared to the four baselines. For both our approach and the baselines, we found that the false positive rate (FPR) was generally higher than the false negative rate (FNR). For our approach, most of these false positives result due to the underprediction of actual energy consumption, which, in turn, results from irregularity in pattern across several consecutive days. In contrast, the high FPR in the baseline approaches is because they do not account for contextual variables, and they do not remove the outliers in the historical data.

While, across all the datasets, our approach outperforms the four baselines, the

[10] `https://cran.r-project.org/web/packages/Rlof/index.html`
[11] `https://github.com/loneharoon/AnomAppliance`
[12] https://github.com/twitter/AnomalyDetection

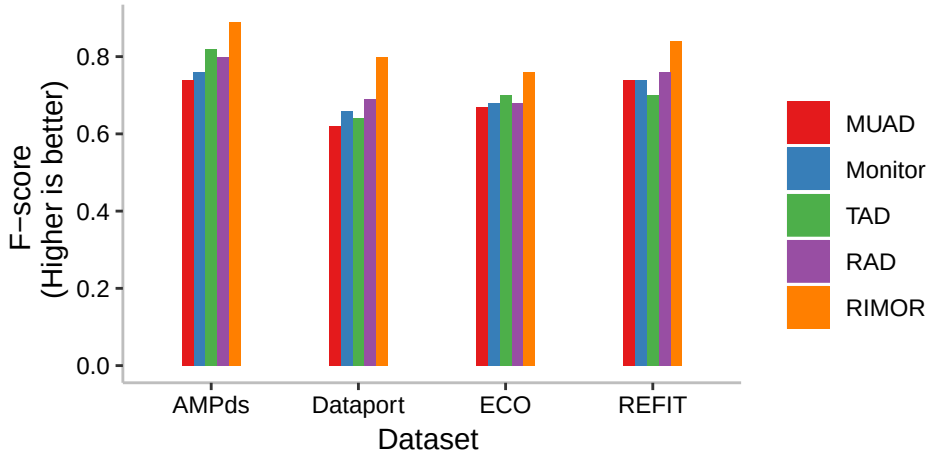


Figure 4.4: Our approach (RIMOR) gives 15% better accuracy in detecting anomalies. [**Best viewed in color**]

	SMAPE (Lower is better)	
Dataset	**Regression**	**Neural Networks**
Dataport	0.60	**0.33**
AMPds	0.42	**0.29**
ECO	0.62	**0.39**
REFIT	0.54	**0.31**

Table 4.4: Prediction accuracy using SMAPE metric.

performance for AMPds and REFIT is better than the other two datasets. This can be attributed to the better energy prediction accuracy on AMPds and REFIT as shown in Table 4.4. AMPds and REFIT show better accuracy since the power patterns are more regular when compared to the other two datasets. We believe that there can be additional contextual parameters to improve the energy consumption prediction for Dataport and ECO.

From Table 4.4, we can also see that neural networks modeling the non-linear relationship between the input features and the output power consumption outperform the linear regression model. This indicates the presence of non-linear relationships among the considered data variables. Since neural networks perform better than the regression model, we use only neural network prediction results for anomaly detection and appliance identification.

Dataport	AMPds	ECO	REFIT
0.85	0.88	0.76	0.81

Table 4.5: Anomalous appliance identification accuracy

Home #	Appliances	Dataset	Approx. ratings (W)
1	Clothes Washer & Microwave	REFIT	0450
2	Cooktop and AC	Dataport	1200
3	Water heater and AC	Dataport	1100
4	Heat pump and wall oven	AMPds	1800
5	Furance and kitchen plugs	Dataport	0450

Table 4.6: Appliances found in few homes with approximately similar power ratings. Due to similar power ratings, such appliances typically lower appliance identification accuracy.

4.4.2 Appliance identification performance

The appliance identification accuracy is shown in Table 4.5. On average, RIMOR reports 82% appliance identification accuracy. We observe that the ECO dataset shows the lowest appliance identification accuracy. This is because the dryer and the washing machine, which collectively account for 45% of the anomalies in the dataset, have very similar appliance power ratings. Since our identification approach currently leverages only appliance power ratings, it will show low accuracy in the presence of appliances having similar power ratings. Table 4.6 shows a few pairs of appliances having a similar power rating across the different datasets.

4.5 Sensitivity analysis

4.5.1 Number of historical days

The accuracy of predictions in prediction models depends upon the number of historical days (N). Figure 4.5 shows the effect of the change in the number of historical days on BIC value across the different datasets. The lower the BIC value, the better the model is. As shown, prediction models achieve lower BIC values when the days are between 4 and 5. As we increase history from 1 to 4 days, the BIC de-

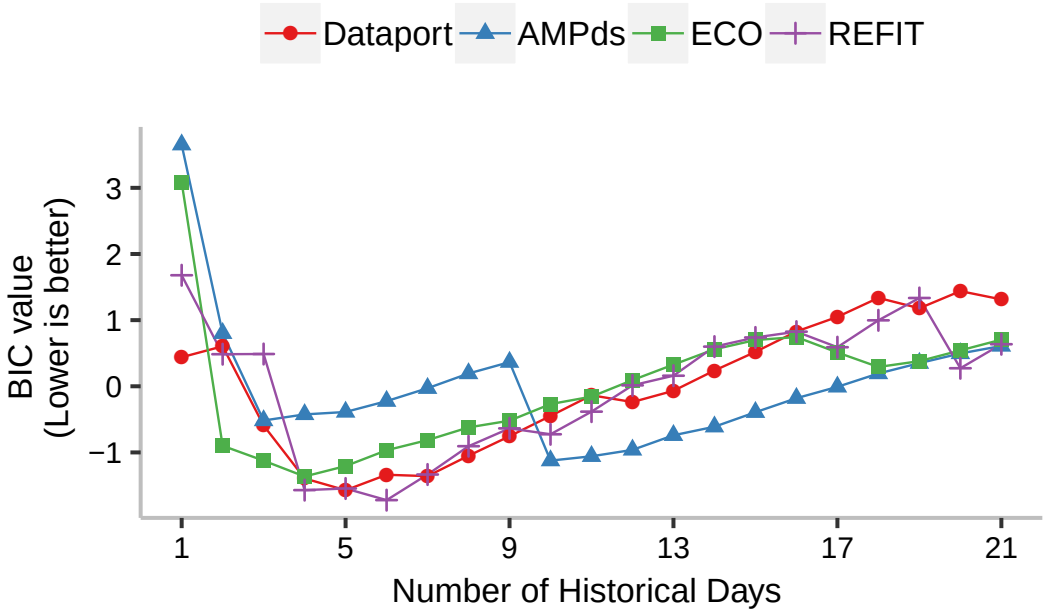


Figure 4.5: Effect of number of historical days N on energy consumption prediction. Using 4-5 historical days gives the best model accuracy. [**Best viewed in color**]

creases, signifying that the regression models work better with more historical data. However, beyond 6 or 7 days, adding more historical information increases BIC, signifying that the additional historical data is generally less representative of the recent energy consumption trends.

4.5.2 W and S parameter selection

The W parameter controls the chunk size; for example, $W = 24$ means each chunk consists of one-hour consumption data ($= \frac{\text{\# of hours in a day}}{W}$). The S parameter defines the threshold on the number of observations within a chunk behaving abnormally. First, we discuss the effect of S on various metrics such as precision, recall, and F score while keeping W constant, and later, we discuss the effect of varying W on the mentioned metrics.

For $W = 24$, Figures 4.6(a) and (b) show as S increases, both precision and recall increases for all the four homes. This means that with increasing S, the number of false positives and false negatives decreases. Our data is at a 10-minute sampling rate, so $W = 24$ means that a chunk of hour size consists of 6 observations. For this

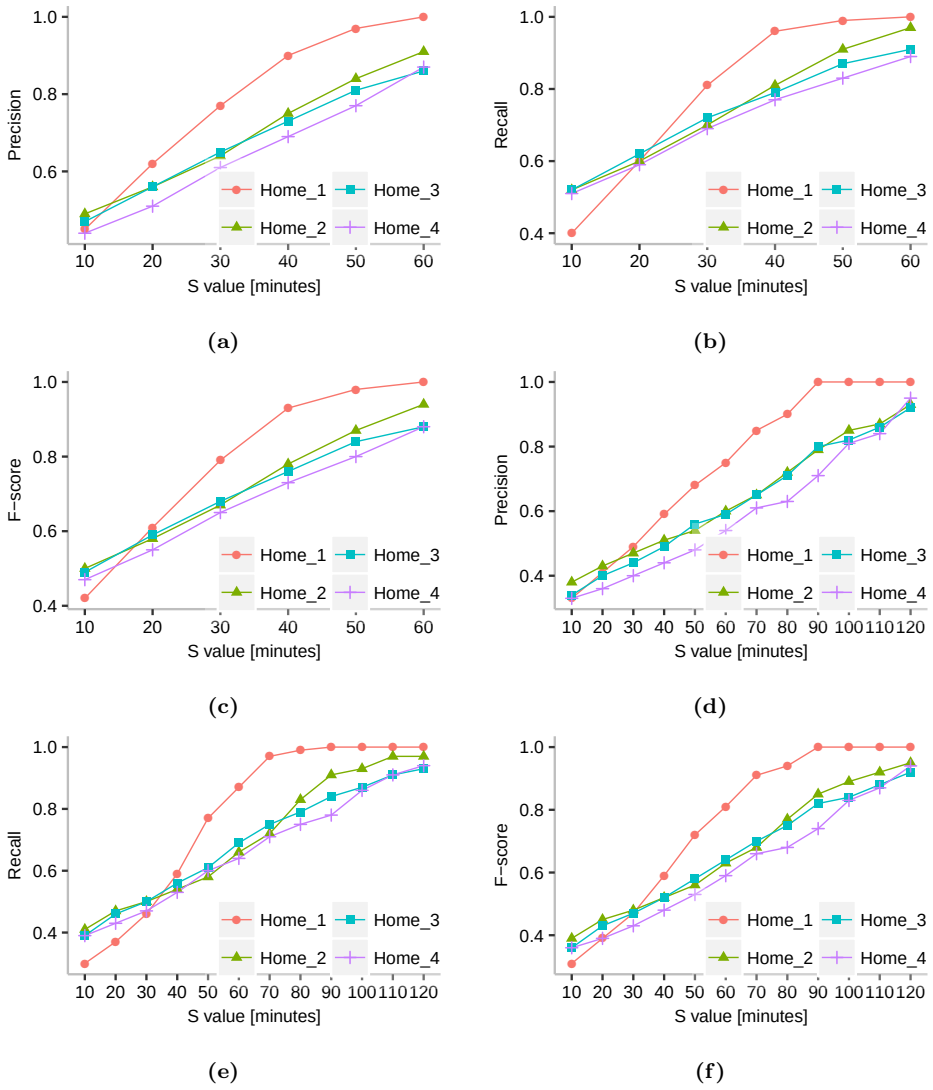


Figure 4.6: At W = one hour : (a) Precision, (b) Recall, (c) F-score. At W = two hours : (d) Precision, (e) Recall, (f) F-score. Figures show with increasing S both false positives and false negatives decrease. And with increasing chunk size, F-score decreases. [**Best viewed in color**]

W, the chance of flagging a chunk as erroneously anomalous is higher at $S = 10$ as compared to $S = 50$, because $S = 50$ defines the behavior of five observations as compared to a single observation ($S = 10$). This means a smaller S value results in more false positives as compared to a larger S value within the same chunk size. Figure 4.6(c) shows the combined effect of precision and recall in terms of F score. The higher the F score, the better the performance of the technique.

For $W = 12$, i.e., two hourly chunk size, Figures 4.6(d), (e), and (f) show precision, recall, and F-score, respectively. Figures 4.6(c) and (f) show that F-scores decreases as chunk size increases. F-score of 0.8 is achieved at 50 and 90 minutes

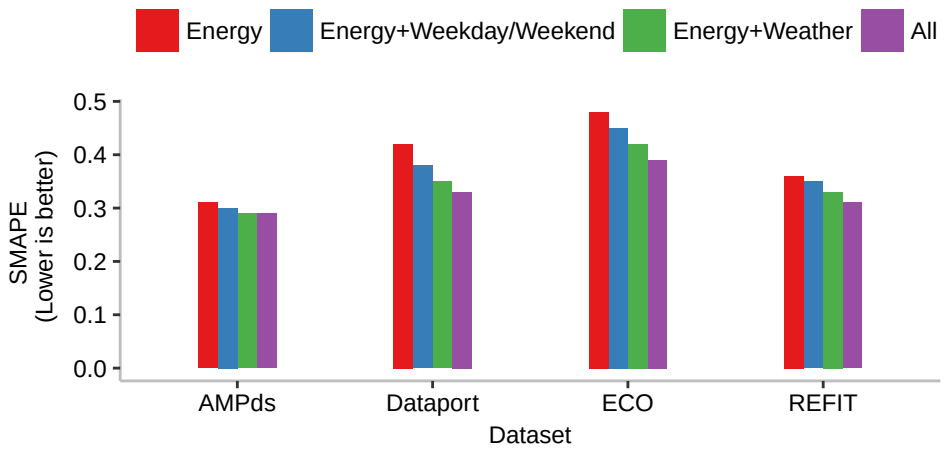


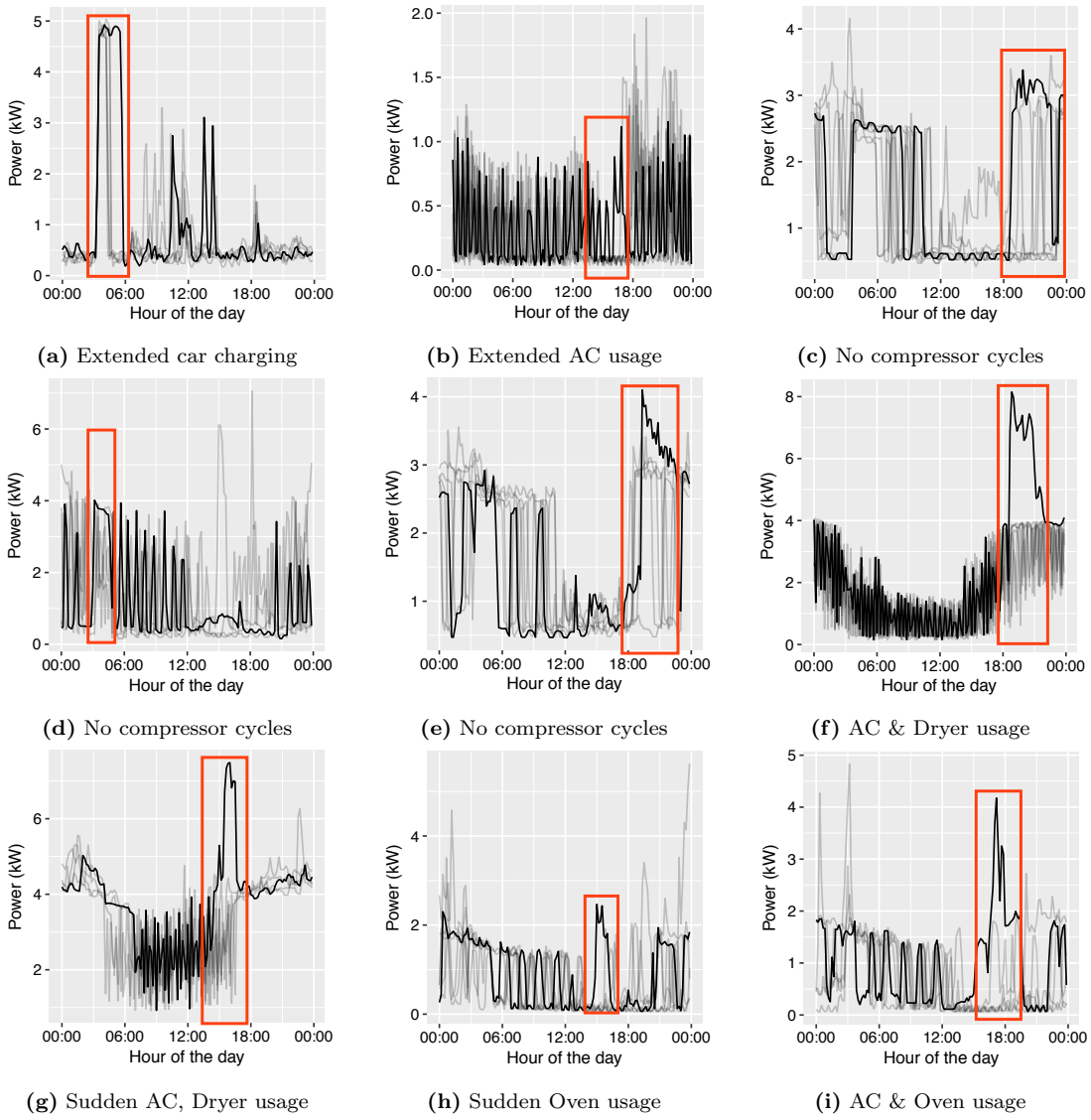
Figure 4.7: Adding more contextual features decreases SMAPE error and, hence, improves energy prediction. [**Best viewed in color**]

in Figures 4.6(c) and (f), respectively. A chunk size of one hour contains six observations while a chunk size of two hours contains 12 observations. The chances of behaving any three observations as erroneously anomalous are higher in 2-hour chunks as compared to one-hour chunks. As a result, the same value of S results in higher F-score in one hour chunk as compared to a two-hour chunk.

From the analysis of W and S, we conclude that for a chunk size of C minutes, the S value needs to be $> 0.5 * C$ minutes to achieve a better F-score. A domain expert can define both of these parameters according to the nature of energy consumption. If anomaly detection is critical, then W can be set as low as possible, say hourly.

4.5.3 Impact of contextual information

We hypothesized that adding contextual information would improve the anomaly detection accuracy as it would lead to a more accurate energy prediction. Figure 4.7 shows that adding the weekday/weekend context and weather information reduces the energy prediction errors compared to using energy data, by 6% and 11%, respectively. Adding the two contexts reduces the error by 16%, thus validating our hypothesis.



(a) Extended car charging **(b)** Extended AC usage **(c)** No compressor cycles

(d) No compressor cycles **(e)** No compressor cycles **(f)** AC & Dryer usage

(g) Sudden AC, Dryer usage **(h)** Sudden Oven usage **(i)** AC & Oven usage

Figure 4.8: Aggregate power consumption of homes over several consecutive days. Transparent lines represent power consumption of non-anomalous days, and the opaque line that of an anomalous day. Red rectangular regions shows anomalous time intervals. The separate caption of each figure defines the cause of the anomaly.

4.6 Insights and potential impact

Having discussed RIMOR and its evaluation, we now revisit the original motivation: can we help reduce the energy consumption by identifying anomalous usage? To answer this question, we now discuss the classification of flagged anomalies into actionable and non-actionable anomalies, followed by a discussion on the potential energy and monetary savings if these anomalies are timely rectified.

4.6.1 Actionable anomalies

These anomalies represent energy wastage instances where timely action can lead to energy savings. The marked regions in Figure 4.8(c), (d) and, (e) show detected actionable anomalies. On these particular days, the air conditioner took a longer ON compressor cycle during the marked time duration. As a result, the aggregate usage deviated from the historical days, and this resulted in an anomaly. We observed similar external weather conditions during the previous non-anomalous days and these particular anomalous days. Thus, we believe that these anomalies can be attributed either to the air conditioner misconfiguration [100] or some fault, like air conditioner gas leakage. However, since the fault does not persist over time, we believe that the most probable cause is setpoint misconfiguration.

4.6.2 Non-actionable anomalies

Such anomalies arise when users change their energy consumption behavior as per their needs. For instance, if occupants have guests at home, their energy consumption may increase. This raises an important question on the subjectivity of the definition of an anomaly. If we have additional contextual information, such as the presence of guests, this usage will not be treated as anomalous. Figure 4.8(a) shows a non-actionable anomaly from the dataset. We found these anomalies were flagged out due to the extended usage of the car charger. Normally, the charging used to happen for one hour over several days, but on this particular day, it took two

Dataset	Min. units (kWh)	Max. units (kWh)	Avg. units (kWh)
Dataport	04	101	48.8
AMPds	93	93	93.0
ECO	53	74	64.0
REFIT	09	96	48.4
	04	101	**63.5**

Table 4.7: Statistics of anomalous energy units consumed over the duration of two months.

hours. Figure 4.8(b) reports anomalies between 1300 - 1600 hours because the air conditioner was running continuously, which is unusual compared to historical days. Similarly, Figure 4.8(g) shows anomalies that occurred when the air conditioner and the dryer were operated simultaneously. Note that these time intervals are flagged as anomalous because these devices were never used simultaneously in the same time interval during historical days. Figure 4.8(h) shows anomalies that result from the sporadic untimely usage of the oven.

These anomalies have limited the scope of energy saving, but their detection is still important because utilities may want to target users with more consistent usage for programs such as Demand Response (DR) [101, 102]. Having information about abnormal usage may improve the efficiency of DR programs.

4.6.3 Potential energy savings

Table 4.7 shows the statistics on energy consumed due to anomalies in our dataset. From each dataset, we report only homes with minimum and maximum amount of anomalous energy consumed. We also report the average amount of anomalous energy over all the homes of a dataset. In summary, this table shows that Rimor has the potential to save up to 63.5 units of anomalous energy. In relative terms, this would amount to an energy saving of around 8% per home. This shows that Rimor can save energy by using existing smart meter data and publicly available weather information. Moreover, it does not require any new hardware to be installed in the home.

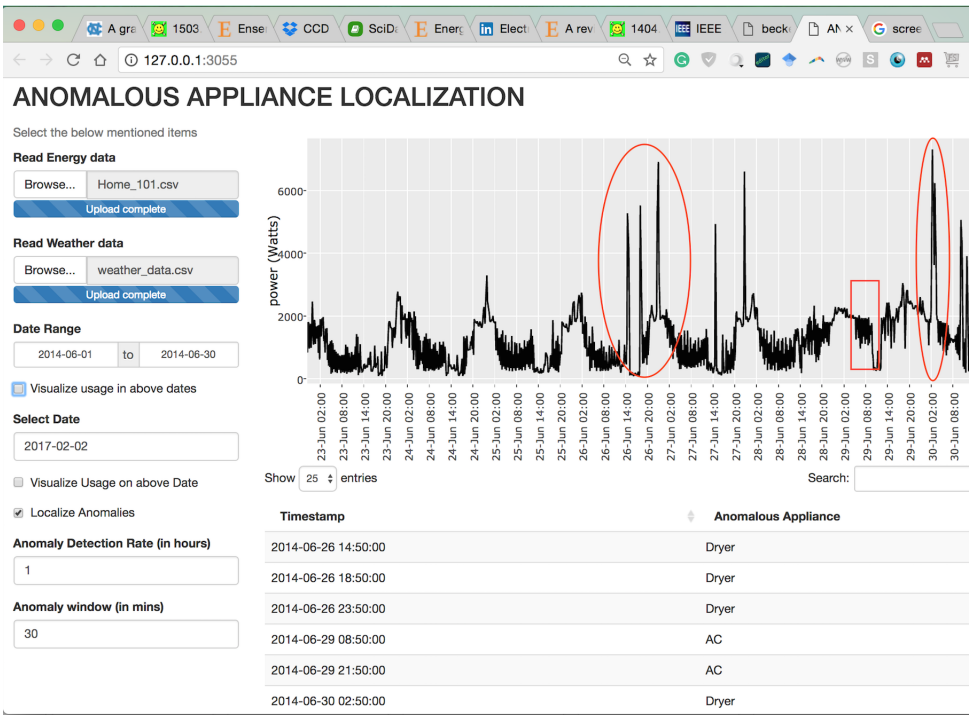


Figure 4.9: Screenshot of prototype application

4.6.4 Application prototype of Rimor

We show the utility of RIMOR by developing a web application[13], which requires energy consumption and weather data. Figure 4.9 shows a screenshot of our prototype application. Energy consumption data can be readily obtained from installed home-level smart meters, and weather data can be obtained from the same home if available or from any publicly available weather services, such as Weatherunderground[14] or openweathermap[15]. Also, the two user-defined parameters W and S are provided to tune the anomaly detection rate (chunk size) and anomaly duration, respectively. Before deploying RIMOR, we plan to:

- Develop a dashboard which will show energy consumption in real-time manner to a user and provide potential anomaly alerts while running RIMOR in the background. This dashboard will also include an appliance registry for a one-time registration of appliances present in a home. A homeowner needs to add only the make and model of the appliances.

[13] `https://github.com/loneharoon/AnomAppliance`
[14] `https://www.wunderground.com`
[15] `http://openweathermap.org`

- Obtain consent from the homeowners and the power utility to interface RIMOR with their smart meter data.

4.7 Extensions of Rimor

The current implementation of our approach has following limitations which provide further opportunities to work on the problem.

1. Our approach currently localizes appliances just using appliance power ratings. However, as we saw in our analysis, many appliances can have similar appliance ratings. For solving such ambiguities, appliance time-series power signal can be used, in addition to the rated power. Further, if multiple instances of the same appliance are present in the home, our approach can not distinguish between them. Adding location context [17] can potentially resolve this ambiguity.

2. Currently, our approach cannot differentiate between actionable and non-actionable anomalies. Most non-actionable anomalies are felt as false alarms by consumers, as a result consumers find these type of approaches unreliable. In future, active learning approaches [103] can be explored to learn whether an anomaly should be flagged as actionable or not.

4.8 Summary

In this chapter, we presented RIMOR to *identify* anomalous appliances in real-time mode. We evaluated RIMOR on four publicly available datasets from different geographical locations and found it to be 15% better in detecting anomalies. Our results showed that adding contextual information helped to improve anomaly detection by up to 16%. Given that the data required to run our approach – smart meter data and external weather data – is readily available, we believe our application can be scaled to a large number of homes.

Chapter 5

Anomaly detection with Non-Intrusive Load Monitoring

5.1 Introduction

In previous chapters, we used aggregate smart meter data, collected at the household level for anomaly detection, and we found that it is difficult to identify an anomalous appliance several scenarios. In this chapter, we will explore an alternative approach, namely, Non-Intrusive Appliance Load Monitoring (NILM), for identifying an anomalous appliance. The idea is to predict/extract appliance-level consumption data from aggregate smart meter data first, and then run an anomaly detection algorithm on extracted appliance level data. So, the accuracy of this alternative approach will depend upon: (i) how accurately appliance-level data is predicted from smart meter data, and (ii) accuracy of the anomaly detection approach.

NILM [98], a software-based disaggregation approach, has been used to infer individual appliance's consumption using smart metered high-frequency energy data. Over the years, the NILM research community has shown improvement in the appliance classification accuracy demonstrating that NILM is suitable for numerous applications, ranging from demand response and energy feedback to activity recognition, and appliance retrofit decisions [98, 28, 104, 105, 106].

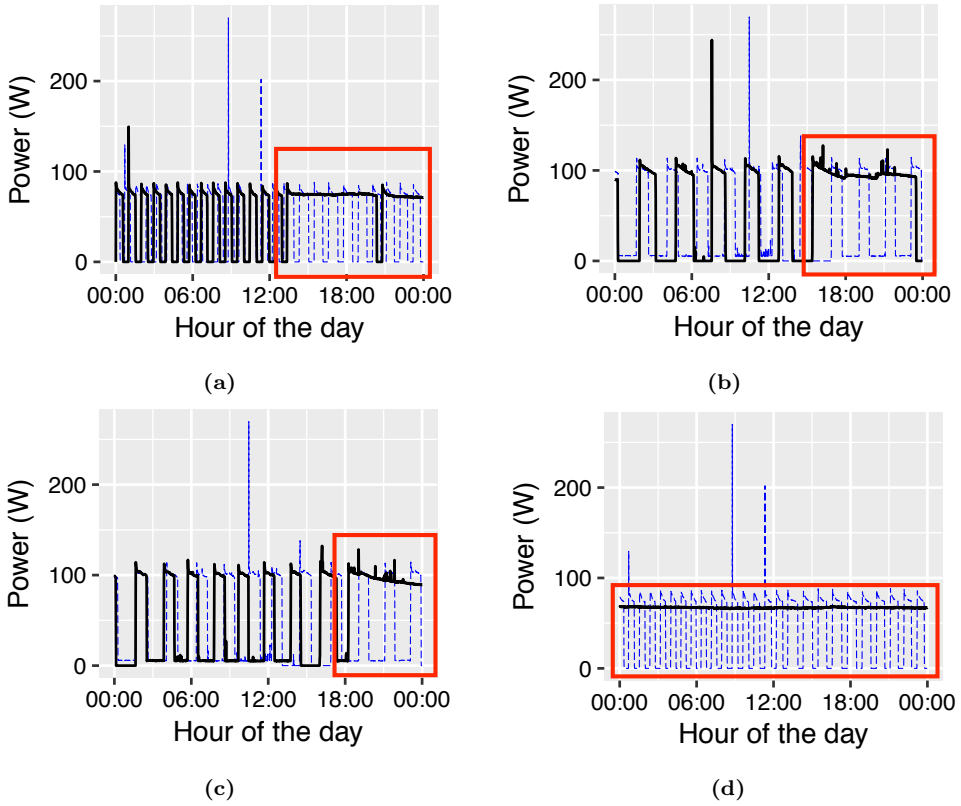


Figure 5.1: Power consumption signatures of freezer. In each plot, thin dashed blue line shows appliance's normal working pattern and thick solid black line shows consumption on an anomalous day. On the anomalous day, either appliance took longer ON cycles (Figures a, b, c) or remained ON throughout the day (Figure d) as highlighted with rectangles.

Building on the large success of NILM over the past few years [104], this chapter explores the usability of state-of-the-art NILM methods for appliance's anomaly detection. Towards this, we propose an anomaly detection approach, namely, Appliance Energy Monitor (**AEM**) for detecting anomalies at an appliance level. It is a supervised approach, which first builds a model of an appliance load by training on normal operation electrical measurements, and then monitors energy consumption of appliance for anomalies using a built-in training model. We test AEM's usability on UK's publicly available dataset, REFIT [20]. REFIT contains aggregate smart meter and submetered appliance's continuous electrical load measurements of 20 residential homes for a duration of about two years. Visual analysis (Figure 5.1) of the dataset shows that electric heater, freezer and fridge have periods of anomalous behaviour, so we examined thoroughly these appliances for anomaly detection.

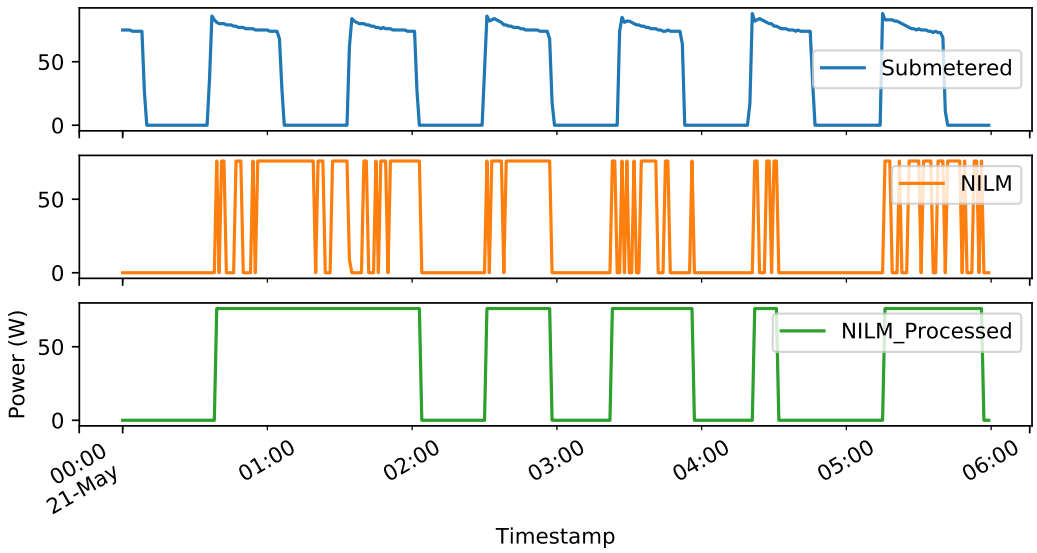


Figure 5.2: Submetered (top), NILM (middle) and NILM post processed (bottom) data.

For evaluation of the suitability of NILM data with our proposed anomaly detection algorithm, we perform disaggregation using existing state-of-the-art NILM methods, both state-based and event-based, all publicly available [104]. During experiments, we observed that sometimes NILM methods introduce redundant ON-OFF cycles as shown in the middle panel of Figure 5.2, so we proposed an algorithm for post-processing of NILM data. Apart from NILM data we also run **AEM** on available submetered data to find its baseline performance.

Our contributions in this chapter include:

1. We test the usability of NILM data, obtained with five different NILM methods, for the identification of anomalous appliances. The usability is checked by proposing and testing an anomaly detection approach on both NILM and submetered data. Testing on submetered data reports its baseline performance whereas testing on NILM data shows the usability of NILM for identifying anomalous appliances.

2. We propose a post-processing algorithm for improving anomaly detection capability of traditional NILM.

3. We release publicly annotated anomalies of the REFIT dataset. Presently, no

such detailed annotations are publicly available for any of the energy datasets.

5.2 Related work

NILM techniques provide the breakdown of smart meter readings into individual appliances consumption. Broadly, NILM approaches are categorized into two categories (i) *supervised* – approaches that require submetered appliance data prior to actual disaggregation process [107, 13, 108, 109, 110, 111, 112]. (ii) *unsupervised* – approaches that do not require labeled training data for the disaggregation purpose [113, 114, 115, 116]. It has been found that supervised approaches outperform unsupervised ones as the former ones are trained on specific homes' appliance data [107]. NILM has become a very active research topic. Indeed, from 2010 onwards, the number of NILM-related papers has increased linearly to 160 per year [117]. Over the years, NILM's accuracy improved significantly, and currently, worldwide, more than 30 companies are providing NILM-based solutions including appliance and activity feedback [118]. According to [27], NILM outputs can be used for anomaly detection, but empirical evaluation has not been preformed yet on building's/household's aggregate meter data.

NILM's applicability in detecting anomalies has been evaluated only in some special cases and for specific applications. For example, in [119], NILM is used for anomaly detection by disaggregating vacuum-assisted waste-disposal system consumption to estimate consumption of vacuum pump and two other pumps. Similarly, authors [120, 121] use NILM data to detect faults in a refrigeration system (or AC) by monitoring changes in the individual parameters of a compressor, a condenser, and an evaporator. All such parameters were obtained from the power consumption signal of a refrigerator (or AC), without considering aggregate building's power consumption.

To the best of our knowledge, this work is the first to evaluate the suitability of building's smart metered NILM data for anomaly detection in detail.

5.3 Methodology

This section first explains the proposed anomaly detection approach, then gives a brief overview of used NILM approaches, and finally, explains post-processing algorithm for NILM data.

5.3.1 Anomaly detection approach

Due to availability of data, we focus on appliances that periodically pass through two cycles during their operation, i.e., appliances that have ON cycle followed by an OFF cycle such as refrigerator, freezer, electric heater etc.

The first step in our methodology is to define the anomalous behaviour of an appliance. Generally, anomaly is any significant departure from usual consumption patterns. The analysis of cyclical appliances shows that anomalies are reflected in the appliance signature in two ways: (i) the appliance takes longer in the ON cycle as compared to normal ones. The reasons for this include appliance malfunctioning, fridge/freezer door left open, cracked door, or mis-configured settings [35]; (ii) the appliance takes a significantly higher number of cycles as compared to normal operation [122]. This is due to appliance malfunctioning, due to age or a fault.

In the first case, the energy consumed in an anomalous cycle should be significantly higher than in the normal cycles and in the second case the number of cycles in a specific time duration will be significantly higher. Thus resulting in higher wear and tear of the mechanical equipment. With these two scenarios, we create a rule-based anomaly detection algorithm, namely **AEM**, for cyclical appliances.

AEM works in two phases - training and testing. In the training phase, it learns an appliance's power consumption signature during normal operation and computes average energy consumed by an appliance's ON cycle and the number of cycles by the appliance in a specific time. In the testing phase, it firstly computes the mentioned parameters of the power consumption of the test day and then compares these with the statistics computed during the training phase. A deviation observed in any value is flagged as anomaly. Next, we formally describe each of these two

phases.

Training Phase: First, power consumption readings of an appliance for D normal days are collected. For each day, D_i, count the number of cycles taken by an appliance as c_i, and compute energy consumption of different cycles as vector $\mathbf{e}_i$, where size of $\mathbf{e}_i$ equals to c_i. Next, using the computed statistics of D days, average number of cycles C_{train} and energy per cycle E_{train} are computed as:

$$C_{train} = mean(c_i), i \in \{1, \ldots, D\} \tag{5.1}$$

$$\sigma^C_{train} = std(c_i), i \in \{1, \ldots, D\} \tag{5.2}$$

$$E_{train} = mean(\mathbf{e}_i), i \in \{1, \ldots, D\} \tag{5.3}$$

$$\sigma^E_{train} = std(\mathbf{e}_i), i \in \{1, \ldots, D\}. \tag{5.4}$$

where σ^C_{train} and σ^E_{train} represent standard deviation of number of cycles and energy consumption per cycle respectively.

Testing Phase: In this phase, **AEM** takes power consumption signature of the appliance during the target test day and computes statistics C_{test} and E_{test} as above. Next, it uses the following set of rules to flag anomalies:

Rule # 1: If the average energy consumption of the test day cycles is significantly greater than the train day cycles, i.e.,

$$E_{test} > \alpha * (E_{train} + n * \sigma^E_{train}) \tag{5.5}$$

where n is the number of standard deviations and α represents the number of times energy consumption on a test day deviate from the normal.

Rule # 2: If the number of cycles taken by appliance on the test day are significantly greater than the train day cycles

$$C_{test} > C_{train} + n * \sigma^C_{train} \tag{5.6}$$

5.3.2 NILM algorithms

NILM algorithms take smart meter data as input and estimate power consumption of each appliance in the household. We used five different well-known NILM algorithms to obtain the appliance level NILM data from the aggregate smart meter data. These include Combinatorial Optimization (CO) [98], Factorial Hidden Markov Model (FHMM) [123], Latent Bayesian Modeling (LBM) [108], Super-state Hidden Markov Model (SSHMM) [107], and Graph-based Signal Processing (GSP) [106]. All of these are publicly available and are considered state-of-the-art.

CO: CO defines the energy disaggregation problem as

$$Y_t = \sum_{i=1}^{N} y_t^i + e_t \tag{5.7}$$

where Y_t represents aggregate power consumption of a home at time t, y_t^i represents i^{th} appliance's consumption at time t, N the number of appliances and e represents residual. Alternatively, above equation can be written as

$$e_t = \arg\min_{e_t} |Y_t - \sum_{i=1}^{N} y_t^i| \tag{5.8}$$

At every t, CO finds the optimal combination of the set of ON appliances and their power consumption that result in minimum e_t.

FHMM: FHMM is an extension of basic Hidden Markov Model (HMM) in which each appliance is represented with an independent Markov chain, and the output is represented as an addition function of all hidden states.

LBM: LBM is an extension of FHMM. It adds extra features in the form of appliance's summary statistics including total energy consumption, duration of use, and usage frequency.

SSHMM: SSHMM is an extension of HMM where each state is computed as the cartesian product of different possible states of household appliances. It uses only recent last and current meter readings to get disaggregated outputs for household

ALGORITHM 4: Steps in post-processing NILM data

Input: Appliance's submetered power consumption $Y_{metered}$ of D normal days and NILM data of the appliance Y_{nilm}

Output: Post-processed NILM data, $Y_{processed}$

Compute duration d_i of each OFF cycle in $Y_{metered}$, where $i \in \{1, \cdots, H\}$; H is the number of cycles in D consumption days

Compute mean and standard deviation over all d_i as

$$d_{metered} = mean(d_i), i \in \{1, \ldots, H\} \tag{5.9}$$

$$\sigma^d_{metered} = std(d_i), i \in \{1, \ldots, H\} \tag{5.10}$$

for $j \leftarrow 1$ **to** *number of OFF cycles in* Y_{nilm} **do**
    Compute duration d^j_{nilm} of j^{th} OFF cycle
    if $d^j_{nilm} < (d_{meterd} - 2 * \sigma^d_{metered})$ **then**
        Find first and last readings of d^j_{nilm} as o_f and o_l
        $d^{intpol}_{nilm} \leftarrow$ Interpolate linearly all readings between o_f & o_l
        Update Y_{nilm} with d^{intpol}_{nilm}
$Y_{processed} \leftarrow Y_{nilm}$
return $Y_{processed}$

appliances using the proposed sparse Viterbi algorithm.

GSP: GSP is an unsupervised, graph-based disaggregation algorithm that exploits adaptive thresholding and clustering to build a graph and then pattern matching to differentiate ON and OFF states. The paper authors represent a dataset with a set of nodes and a weighted adjacency matrix. A node corresponds to an element in the dataset and the adjacency matrix defines all edge weights in the graph. Graph weights represent the correlation between the corresponding nodes.

5.3.3 NILM data post-processing

The top two panels of Figure 5.2 show submetered and NILM data obtained with SSHMM. The analysis of various other plots like Figure 5.2 shows that NILM detects events but often gets confused with the ON-OFF cycle frequency. Proposed anomaly detection approach **AEM** detects anomalies by monitoring the average energy consumption of each cycle. So, using NILM data as such will result in wrong anomaly results (discussed in Results section). So, we post process NILM data first

and then **AEM** uses post processed data for anomaly detection.

Post-processing consists of two steps: (1) Take the submetered data $Y_{metered}$ of target appliance and compute the average duration $d_{metered}$ of all OFF cycles. Also, compute the standard deviation $\sigma^{d}_{metered}$ of all OFF cycles as explained in Algorithm 4. (2) Take NILM data Y_{nilm} and compute OFF duration of each j^{th} cycle as d^{j}_{nilm}. If the OFF duration is found significantly lower than $d_{metered}$ as computed in step 1, then "first" and "last" readings of the OFF duration are taken and then in-between readings are linearly interpolated as explained in the algorithm. In this way, high frequent cycles introduced due to disaggregation are handled and the data becomes ready for anomaly detection. The bottom panel of Figure 5.2 shows post NILM processed data.

5.4 Evaluation

This section explains the dataset, used metrics, and the experimental setup.

5.4.1 Dataset and anomaly annotation

All experiments were conducted on a publicly open, residential energy dataset, REFIT [20]. It contains both household aggregate and appliance's submetered power consumption data of 20 UK homes for about two years. The default sampling rate of the data is eight seconds and we downsampled it uniformly to one minute rate. The data was downsampled as many electrical utilities log data at minutes intervals.

Out of 20 homes, we selected five homes (namely House 1, 10, 16, 18 and 20), since all these selected five homes contain a higher number of anomalies compared to the remaining homes. Further, we selected four months of data from each of these homes in such way that one month did not contain any anomaly and remaining three contain anomalies. We ensured that the power consumption pattern did not change significantly in these months.

For each of the mentioned homes, we selected top six highest energy consuming

Home number	1	10	16	18	20
Noise percentage	68.42	58.97	63.17	45.28	55.92

Table 5.1: Noise percentage in different homes.

Home (#)	1	2	3	4	5	6
1	DW	WM	Fridge	Freezer 1	**Electric Heater**	Freezer 2
10	DW	TV	Toaster	WM	**Chest Freezer**	Blender
16	DW	Computer	TV	Freezer 2	**Fridge Freezer 1**	Dehumidifier
18	DW	TV	Freezer	Computer	**Fridge freezer**	Fridge
20	DW	TV	kettle	Fridge	**Freezer**	Dryer

Table 5.2: List of appliances used for disaggregation in different homes. Anomaly detection is done on Appliance # 5 only. DW and WM corresponds to dishwasher and washing machine.

appliances for disaggregation as is done commonly in NILM community [124, 107, 13] and the aggregate smart meter data is retained as such. Therefore, the aggregate data is not equal to the summation of these six energy consuming appliances, and contains other 'unknown' appliances, commonly referred to as noise [107, 125]. The percentage of noise in each home is calculated as [107]

$$Noise\ percentage = \frac{\sum_{t=1}^{T} |Y_t - \sum_{i=1}^{N} y_t^i|}{\sum_{t=1}^{T} Y_t} * 100, \tag{5.11}$$

where Y_t represents aggregate power consumption at time t, y_t^i represents consumption of appliance i, N the number of appliances to be disaggregated. Table 5.1 reports the noise percentage present in the aggregate meter reading of each of the mentioned five homes. Table 5.2 lists top six appliances for each of these homes. This paper targets only appliance # 5 (heater or freezer) of the table for anomaly detection and we refer to such appliances as target appliances hereafter.

The current release of the REFIT dataset does not have anomaly annotations publicly available. For the first time, we visualized the entire dataset for continuous one month and marked all appliance's anomalies in a separate CSV file. Rules defined in Table 5.3 were used to mark anomalies. Figure 5.3 shows a screen-shot of the created anomaly annotated file highlighting different anomaly parameters

House_No	Appliance	Time_Duration	Status	Explanation	Comments
1	Freezer_1	"2014-06-18 23:00:00 ; 2014-06-19 04:00:00"	S	Continuous ON state	
1	Freezer_1	"2014-08-01 18:00:00 ; 2014-08-02 13:00:00"	NS	Sensor seems stuck	
1	Freezer_2	"2014-08-01 18:00:00 ; 2014-08-02 13:00:00"	NS	Sensor seems stuck	
1	Freezer_2	"2014-09-16 15:30:00 ; 2014-09-16 19:20:00"	S	Continuous ON state	

Figure 5.3: A screenshot of the anomaly annotated file.

• **Case 1:** Appliance's consumption found significantly different from its historical normal consumption. **Action 1:** Flagged as anomalous and marked as S (sure) **Action 2:** Noted time-duration of anomaly
• **Case 2:** Appliance's consumption found significantly different from its historical normal consumption, but anomalous duration seems either due to measuring meter malfunctioning (when a meter gets stuck at some higher value) or the values were interpolated by using 'previous known value filling approach' **Action 1:** Flagged as anomalous and marked as NS (not-sure) **Action 2:** Noted time-duration of anomaly
• **Case 3:** Appliance's ON cycle duration found significantly longer than its historical consumption and the predecessor OFF cycle also found longer. **Action 1:** Not marked as anomaly as it is normal in considered appliances **Action 2:** Nothing

Table 5.3: Rules for marking anomalies in REFIT dataset.

noted for each anomaly. Anomalies shown in Figure 5.1 are from the same REFIT dataset. Currently, no anomaly annotated energy dataset is publicly available. With this paper, we release the anomaly annotation file for the energy research community publicly at mirror links (link1[1], link2[2]).

5.4.2 Evaluation metrics

We chose Root Mean Squared Error (RMSE) and Pearson Correlation Coefficient to show the disaggregation performance of NILM algorithms after a detailed analysis of existing disaggregation metrics such as RMSE, Mean absolute error (MAE), Fscore, fraction of total energy assigned correctly (FECA) and normalised error in assigned energy (ANE) [124]. Fscore is a classification metric which only reflects the appliance's ON and OFF accuracy detection and does not reflect the energy

[1] `https://bit.ly/2Oc9Mww`
[2] `https://doi.org/10.15129/9729a2a0-11ce-4cce-b0d0-144c483fcb33`

consumed. Quantifying MAE, ANE and FECA is difficult since there is no uniform scale for comparing results, but quantifying RMSE is easy as explained below.

Root Mean Squared Error: For an appliance a, RMSE is

$$RMSE^a = \sqrt{\frac{\sum_{t=1}^{T}(y_t^a - \hat{y}_t^a)^2}{T}} \tag{5.12}$$

where T represents number of sample readings, y_t^a represents submetered power readings of appliance a at time t and $\hat{y}_t^a$ represents predicted power readings with a NILM algorithm. Ideally, the RMSE for any appliance should be zero. For an appliance of 100 watts, if RMSE is 20 watts then that means on average the predicted NILM values deviate by 20 watts from the actual average consumption. For a 40 watt appliance, RMSE of 20 is significantly higher as compared for an appliance of 400 watts.

Pearson Correlation Coefficient, ρ: In our work, Pearson coefficient measures the correlation between appliance's submetered (s) and predicted (p) NILM readings.

$$\rho_{s,p} = \frac{cov(s,p)}{\sigma_s \sigma_p} \tag{5.13}$$

where cov is covariance, σ_s and σ_p are the standard deviation of s and p. The value of $\rho_{s,p}$ varies in the range [-1, 1], where -1 means either s readings increase and p readings decrease or vice-versa. Value 1 means both s and p readings either increase or decrease together.

RMSE shows by how much predicted readings were off from the appliance wattage and ρ shows ON-OFF correlation between the NILM and submetered readings.

Fscore: The anomaly detection performance of **AEM** is measured in terms of true positive (TP), false positive (FP) and false negative (FN), where a TP means an anomaly detected by **AEM** is actually present in the appliance, FP means anomaly detected by **AEM** is not in the appliance, and FN means an anomaly in appliance was missed by **AEM**. Number of TP, FP and FN for an appliance are often rep-

resented in terms of three accuracy metrics, Precision, Recall and Fscore. These accuracy metrics vary in the range of [0 - 1]. The higher the value of metric, the better is the performance of the approach.

$$\text{Fscore} = 2 * \frac{precision * recall}{precision + recall} \tag{5.14}$$

where $precision = \frac{TP}{TP + FP}$, and $recall = \frac{TP}{TP + FN}$

Matthews Correlation Coefficient (MCC): Unlike F-score, MCC also includes TN while calculating the final value. MCC is defined as

$$MCC = \frac{(TP * TN) - (FP * FN)}{\sqrt{(TP + FP) * (TP + FN) * (TN + FP) * (TN + FN)}} \tag{5.15}$$

Like correlation coefficient, MCC value varies between -1 and +1. A value of +1 represents a perfect positive correlation, -1 represents total disagreement, 0 represents a random prediction between actual and predicted class.

5.4.3 Experimental setup

Firstly, we explain the settings of various NILM algorithms used and later we discuss the **AEM** parameter values.

NILM algorithms: For supervised NILM algorithms- CO, FHMM, LBM and SSHMM, one month of data was used for training and the remaining months for testing as reported in Table 5.4. Testing was done in a sliding-window manner with a window size of one day. Therefore, one month of training data was appropriate for one day of test data. GSP is unsupervised, no training was required and testing was done on testing duration data of Table 5.4. Publicly available implementations of CO and FHMM from NILMTK[3] toolkit, LBM[4], SSHMM[5], and GSP[6] were used

[3] https://github.com/nilmtk/nilmtk
[4] https://github.com/MingjunZhong/LatentBayesianMelding
[5] https://github.com/smakonin/SparseNILM
[6] https://github.com/loneharoon/GSP_energy_disaggregator

Home	Training duration	Testing duration
1	Dec. 2014	Jan. 1, 2015 - March 31, 2015
10	April 2014	May 1, 2014 - June 30, 2014
16	March 2014	April 1, 2014 - June 30, 2014
18	July 2014	Aug. 1, 2014 - Oct. 31, 2014
20	May 2014	June 1, 2014 - Aug. 31, 2014

Table 5.4: Division of data for testing and training purpose.

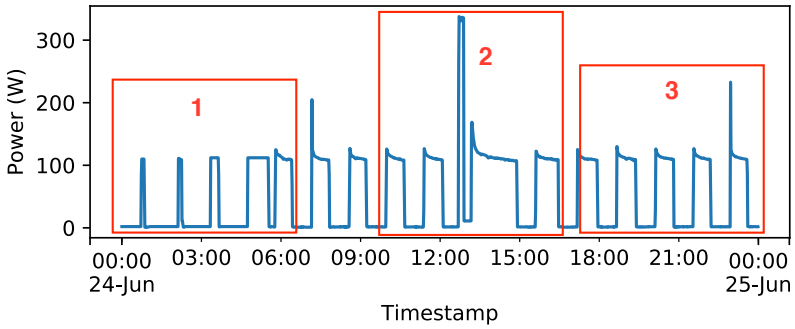


Figure 5.4: One day power consumption signature of a freezer. Regions marked with different rectangles have different consumption signature from one another. [Best visible in color]

to get disaggregation results.

To avoid evaluation bias, all unsupervised NILM algorithms were run with default parameter settings as mentioned by their respective authors. For GSP, the appliance threshold was set to 40 watts obtained empirically since the wattage was above 40 watts for all appliances in the considered dataset. GSP was run in sliding window manner where each window was of size one month. Window size of more than one month takes more time to compute intermediate matrices and optimization step as mentioned in the GSP paper too.

AEM: It uses training data to learn the statistical model of an appliance. Table 5.4 shows the training data duration used to train the models of appliances of mentioned homes.

Visual inspection of appliance's power consumption shows that some appliances consume energy differently at different times of a day. For example, Figure 5.4 shows one day power consumption signature of a freezer. Red colored rectangles

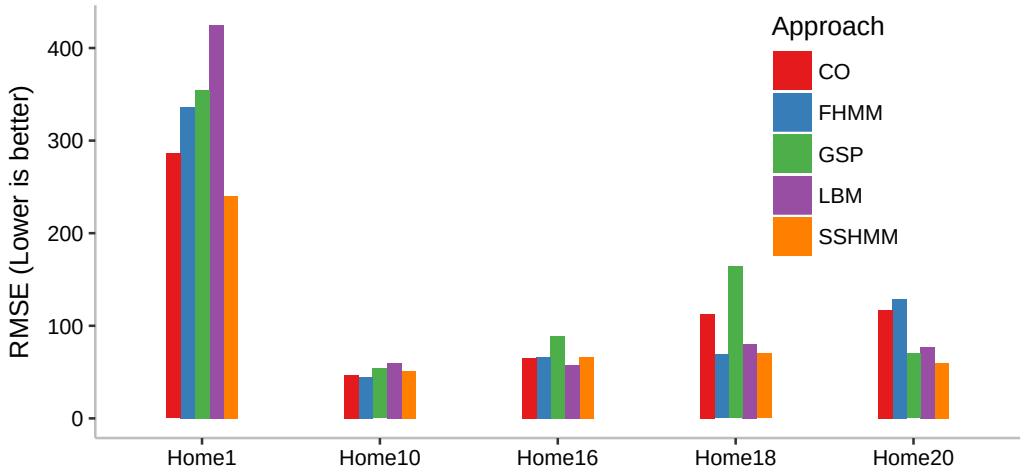


Figure 5.5: RMSE for NILM approaches on heater (Home1) and Freezer (Homes 10, 16, 18 and 20). [Best viewed in color]

mark three different portions of the day showing different consumption signatures where rectangle # 1 has smaller ON cycle duration and rectangle # 2 has larger ON cycle duration as compared to rectangle # 3 ON cycle duration. This shows that the appliance consumes energy differently at different day hours. So, we run **AEM** in sliding window manner of window size six hours. Note that a particular window for which **AEM** runs always uses built training model of the same window. Remaining parameters of **AEM** in Equation 5.5 were set as: $\alpha = 2$ and $n = 2$. We will present the analysis of all these three parameters in a separate section.

5.5 Results

This section first explains the disaggregation performance of NILM algorithms and later the anomaly detection accuracy.

Disaggregation performance: Figure 5.5 shows RMSE of different NILM approaches on target (Heater in home 1 and Freezer in remaining homes) appliances of five homes. Electric heater in Home1 has wattage $>$ 1 KW so RMSE values of it are higher than freezer of remaining homes. The appliance of home 10 has lowest RMSE as compared to appliances of homes 16, 18 and 20 because home 10 has only one freezer as compared to two freezers found in remaining homes. Having

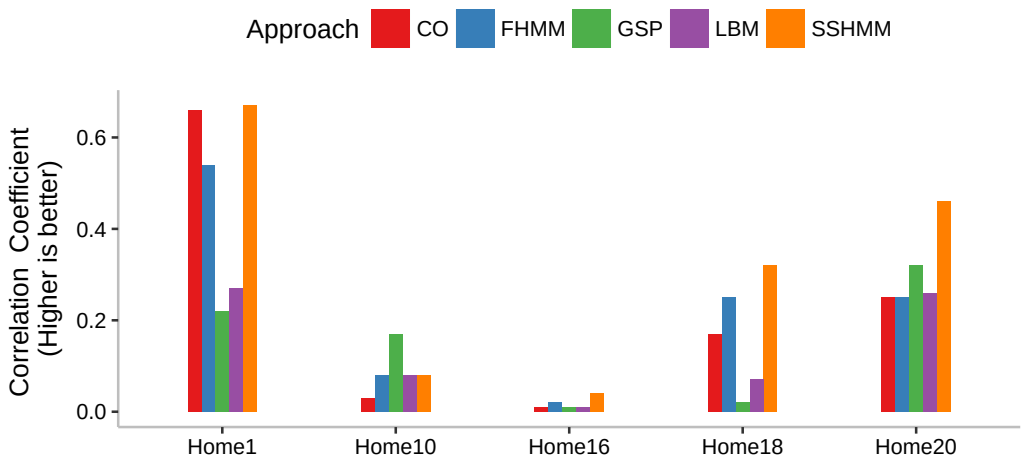


Figure 5.6: Pearson Correlation coefficient for NILM approaches on heater (Home1) and Freezer (Homes 10, 16, 18 and 20). [Best viewed in color]

appliances of distinct wattage decreases RMSE. Figure 5.6 shows the Pearson correlation Coefficient of target appliances of five homes with different disaggregation approaches. Electric heater of home 1 has highest correlation coefficient due to its distinct higher wattage as compared to appliances of other homes.

Overall, Figures 5.5 and 5.6 show SSHMM performs better in terms of lower RMSE and higher correlation coefficient than other competing approaches because it considers all possible combinations of all appliances states while handling the sparsity which other HMM based methods were not able to do. Note that GSP is an unsupervised approach and still its performance is in par to other supervised approaches. Knowing that SSHMM performs better, we expect that its disaggregated data should detect anomalies more accurately as compared to remaining ones.

	Home 1						**Home 10**						**Home 16**						**Home 18**						**Home 20**					
	Submetered	CO	FHMM	LBM	SSHMM	GSP	Submetered	CO	FHMM	LBM	SSHMM	GSP	Submetered	CO	FHMM	LBM	SSHMM	GSP	Submetered	CO	FHMM	LBM	SSHMM	GSP	Submetered	CO	FHMM	LBM	SSHMM	GSP
Precis.	0.7	0.1	0.1	0.01	0.1	0.14	1	0	0.5	0	0.5	0.5	0.9	0	0.25	0.2	0.3	0.2	1	0	0.04	0.07	0.04	0.04	1	0.5	0.3	0.14	0.11	0.13
Recall	1.0	1.0	1.0	1.00	1.0	1	1	0	0.9	0	1	0.6	0.9	0	1	0.5	1	0.6	0.7	0	1	0.67	1	1	0.6	0.1	0.5	0.5	1	0.6
Fscore	0.8	0.2	0.2	0.15	0.2	0.25	1	0	0.7	0	0.7	0.5	0.9	0	0.4	0.3	0.4	0.3	0.8	0	0.08	0.13	0.08	0.08	0.7	0.17	0.3	0.22	0.2	0.21
MCC	1	0	0	0	0	0	1	0	1	0	1	0	1	0	0	0	0	0	1	-1	0	0	0	0	1	0	0	0	0	0

Table 5.5: Precision, Recall, and Fscore of **AEM** with post-processed NILM data.

	Home 1						**Home 10**						**Home 16**						**Home 18**						**Home 20**					
	Submetered	CO	FHMM	LBM	SSHMM	GSP	Submetered	CO	FHMM	LBM	SSHMM	GSP	Submetered	CO	FHMM	LBM	SSHMM	GSP	Submetered	CO	FHMM	LBM	SSHMM	GSP	Submetered	CO	FHMM	LBM	SSHMM	GSP
Precision	0.7	0.1	0.1	0.08	0.1	0.14	1	0	0.6	0	0	0	0.9	0	0	0.23	0	0.24	1	0	0	0.06	0	0.04	1	0	0.33	0.13	0	0.33
Recall	1.0	1.0	1.0	1.00	1.0	1	1	0	0.27	0	0	0	0.9	0	0	0.43	0	0.48	0.7	0	0	0.33	0	1	0.6	0	0.3	0.4	0	0.1
Fscore	0.8	0.2	0.2	0.15	0.2	0.25	1	0	0.37	0	0	0	0.9	0	0	0.3	0	0.32	0.8	0	0	0.1	0	0.08	0.7	0	0.31	0.2	0	0.15
MCC	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	-1

Table 5.6: Precision, Recall, and Fscore of **AEM** with non post-processed NILM data

Anomaly detection performance with NILM: For each target appliance (heater in home 1 and freezer in remaining homes), we use post-processed NILM data of all approaches and submetered data to compute and compare the anomaly detection accuracies. Accuracy results obtained on submetered data are considered as baseline results. The NILM approach providing closest match to submetered data results is considered as the best NILM approach for appliance level anomaly detection.

Table 5.5 reports precision, recall and F-score of **AEM** with both submetered and five different post-processed NILM data of target appliances. We find following inferences from the table:

1. Recall is found to be better than precision for every home, meaning there are less false negatives as compared to false positives. A false positive results whenever NILM signature deviates significantly from the actual energy consumption and the deviation found matches to an anomaly signature.

2. Fscore on SSHMM data is marginally better as compared to remaining approaches but still is low compared to Fscore obtained with submetered data. Better performance of SSHMM as mentioned already explains this marginal improvement of F-score as compared to remaining approaches.

3. **AEM** results in an acceptable Fscore (≥ 0.8) on submetered data, and insignificantly lower F-score on using NILM data obtained with different approaches. Lower Precision results in the drop of Fscore and low Fscore on NILM data as compared to submetered data means **AEM** does not perform well on NILM data. This shows that NILM data of Freezer and heater obtained with existing state-of-the-art NILM approaches cannot be used for anomaly detection.

We computed similar accuracy metrics on unprocessed NILM data too to show the improvement in F-score after post-processing of NILM data. Table 5.6 shows precision, recall and F-score of different approaches on using non post-processed NILM data of target appliances. Note that in Table 5.6 submetered column is copied from former Table 5.5 for the comparison purpose with the remaining columns of the

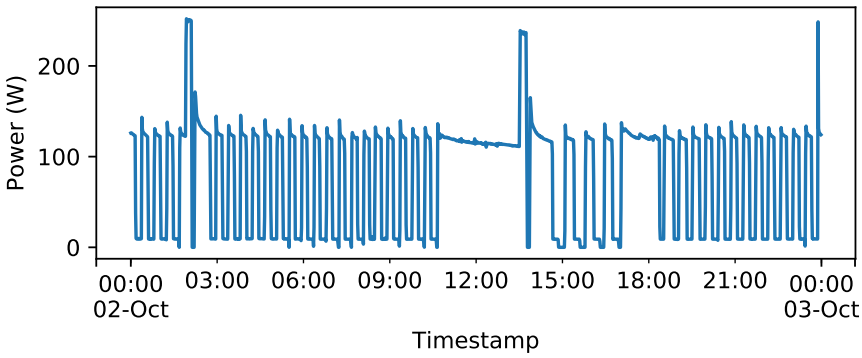


Figure 5.7: Signature of home 18 Fridge Freezer

table. Comparing Fscore of Table 5.5 and 5.6 we find post-processing has improved results significantly, particularly in FHMM and SSHMM approaches. With post-processing, **AEM** is able to flag all genuine anomalies.

5.6 Discussion

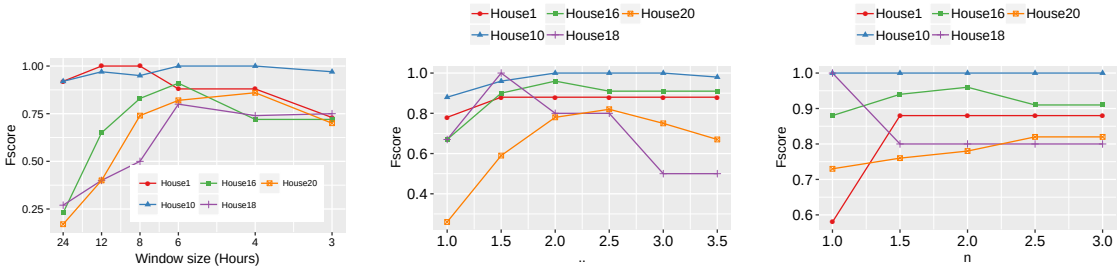
In this section, we answer the following questions.

(i) Why **AEM** *does not result in Recall equal to one with submetered data?*

Recall not being equal to one means there are few false negatives. Figure 5.7 shows one such anomaly where anomalous duration is from 1045 to 1450 hours, i.e., four hours of unusual consumption. Normally, the appliance has 15 minute of ON cycle followed by 15 minutes of OFF cycle. The anomaly is not detected by **AEM** at a rolling window size of 3 or 4 hours. At bigger window sizes such as 6 or 8 hours it gets detected. At window size of 3 hours, the anomalous region comes in one window, and it also contains eight normal consumption cycles. So, getting normal and anomalous cycles in one window smoothens the anomaly effect. while as, at a window size of 4 hours, the anomaly gets partitioned into two windows and the anomalous signature is not detected. This explains that false negatives result due to inappropriate window size. This can not be fixed as different anomalies have different duration and can occur at any time of a day.

(ii) Does sampling rate affect the performance of mentioned NILM algorithms?

On evaluating NILM algorithms on high-frequency data (eight seconds) we did not



(a) (b) (c)

Figure 5.8: Effect of (a) Window size, (b) α, and (c) n on Fscore in **AEM**. [**Best viewed in colour**]

find considerable improvement in any of the used metrics compared to using 1-minute down sampled data.

(iii) If home 18 appliance has better disaggregation performance than Home16 appliance as shown in Figures 5.5 and 5.6 then why **AEM** *results in lower Fscore in home 18 appliance as reported in Table 5.5?*

Home 18 has lesser number of anomalies as compared to home 16. Therefore a small change in home 18 gets much amplified as compared to home 16. So we see significant changes in Fscore of home 18 compared to home 16.

5.7 Analysis of AEM parameters

AEM has three parameters including window size, α, and n. In this section, we analyze their effect on Fscore.

1. Effect of window size: Figure 5.8(a) shows the effect of changing window size on Fscore. Window size refers to the hours of data taken once as input by **AEM** for anomaly detection. Generally, Fscore is low for all homes at larger window sizes, and it improves initially and later remains almost constant while decreasing the window size.

At window size = 24 hours, **AEM** takes entire day's power consumption at once for anomaly detection. Since **AEM** detects anomalies by monitoring energy consumption of each cycle within a window and in cases where anomalous cycles are few as compared to normal cases then such anomalous cycles get smoothened out

due to large number of normal cycles within a window. Due to this smoothening effect, **AEM** misses some anomalies and results in False Negatives, which ultimately drop Fscore. As we decrease the window size, the smoothening effect decreases and anomalies get detected easily. The lower Fscore value at larger window sizes in the figure is due to smoothening effect.

2. Effect of α: Figure 5.8(b) shows the effect of changing α value in Equation (5.5) on Fscore. The value of α signifies by "how many times" the average energy consumption of a window (testing period) is more than normal historical consumption as defined in training model. This means **AEM** flags out anomaly at particular value of α only if that α value satisfies Equation (5.5). The figure shows that Fscore increases generally until $\alpha \leq 2.0$ and then it either remains the same or decreases for $\alpha > 2.0$. Fscore being low at $\alpha = 1$ as compared to remaining α values means few anomalies were missed initially and were caught later as α value increased. In Homes 18 and 20, Fscore again decreases after $\alpha > 2.5$ meaning some anomalies missed again because the criteria defined in Equation (5.5) for anomaly does not match.

3. Effect of n: Figure 5.8(c) shows the effect of changing n in Equations (5.5) and (5.6) on Fscore. n signifies by how many standard deviations the average energy consumption of a window (testing period consumption) deviates from the historical consumption. We found empirically that changing the value of n alone is not enough to capture anomalies but combining α with it results in the flagging of anomalies successfully. Due to its less influence on anomaly detection, the figure shows Fscore does not vary significantly while varying n.

To generalize the settings of mentioned parameters for large-scale experiments, we suggest setting window size = 6 hours, because human activities within a home often change after 6 hours and hence can be set for all the homes. While knowing the significance of α, its value can be set according to the magnitude of anomalies. For example, if an appliance consumes 20 kWh normally and a user wants to flag anomalies only if the consumption exceeds twice of 20 kWh then α should be set to

2. n does not affect Fscore significantly so setting it to 2 works well.

5.8 Summary

This chapter reports detailed experiments to assess whether current state-of-the-art NILM algorithms outputs can be used for appliance level anomaly detection. Our experiments are focused on two major energy-consuming appliances (Freezer and Heater) of a real-world UK residential energy dataset. We conclude that appliance level anomalies cannot be detected using NILM data directly because the appliance-level NILM signatures do not resemble submetered signatures since NILM algorithms are trained on normal appliance operation. While the obvious solution would be to train NILM algorithms with anomalous signatures, this is challenging because each appliance results in a different anomalous signature depending on the cause of the anomaly and knowing all anomalies signatures a priori is not always realistic. We also analyze the output of the NILM algorithms to understand the cause of the poor anomaly detection performance, and while the anomalies are detected, there is some confusion in detecting the events accurately with current metrics because of the frequency of the ON-OFF cycles. We propose a post-processing algorithm of NILM outputs to improve anomaly detection accuracy and show improved accuracy.

Chapter 6

Context-rich energy dataset

6.1 Introduction

Data collection from smart meters is a cumbersome process due to the unreliable nature of data collection hardware, communication issues, and the continuous monitoring of the data collection setup. Furthermore, collecting extra *contextual information* such as occupancy and weather data makes the process more difficult. Due to these challenges, it is often difficult to make large contextualized datasets ready for public release, especially from developing regions where technology is still in its infancy. Some of the major publicly available energy datasets are shown in Table 6.1. These datasets span from days to years with data collected from a few to 100s of buildings. From India, only one dataset spanning a period of 73 days from a single household is available [15].

For the first time, we release a long duration contextualized dataset from India, namely Indian BuiLdings Energy coNsumption Dataset (*I-BLEND*)[1][32]. It is a 52-month energy dataset from seven buildings of our institute campus. This dataset contains five different parameters – voltage, current, power, frequency, and power factor – at a sampling frequency of one minute. With *I-BLEND*, we release other contextual data including occupancy, institute calendar, building architecture details, and four months of local weather (temperature, humidity). The weather

[1] `https://doi.org/10.6084/m9.figshare.c.3893581.v1`

Dataset	Parameters	Duration	Sampling rate (sec.)	Houses (#)	Data type
AMPds [19]	V, I, f, PF, P, Q, S	2 years	60	1	Agg. & App.
Dataport [21]	P	since 2011	60	> 500	Agg. & App.
DRED [17]	P	5 months	1	1	Agg. & App.
ECO [18]	V, I, ϕ	8 months	1	6	Agg. & App.
GREEND [127]	P	1 year	1	9	Agg. & App.
REDD [13]	V, P	19 days	15kHz (Agg.), 3 (Cir.)	6	Agg. & Cir.
Smart* [14]	P, S	3 months	1	3	Agg. & Cir.
UK-DALE [16]	V, I, P	4 years	16 kHz (V, I), 6 (P)	5	Agg. & App.
REFIT [20]	P	2 years	8	20	Agg. & App.

Acronyms: V→ Voltage, I→ Current, f→ Frequency, PF→ Power factor, P→ True power, Q→ Reactive power, S→ Apparent power, $\phi \rightarrow$ Phase angle, E→ Energy, Agg.→ Aggregate, App.→ Appliances, Cir.→ Circuits

Table 6.1: Details of publicly available electrical energy datasets.

parameters (temperature, humidity, wind speed, and wind direction) of a nearby weather station are already publicly available at 30 minute intervals from a free weather service, Weather Underground (WU) [126]. The occupancy dataset is at 10-minute resolution for all seven buildings.

I-BLEND is unique from existing energy datasets in several ways including: (*i*) different environmental factors – voltage fluctuations due to poor infrastructure and supply deficit, power failure, and poor network connectivity [15], (*ii*) a mix of commercial and residential buildings on an academic campus (both types of buildings vary in terms of operational hours and energy consumption patterns, and hence offer diverse research opportunities), and (*iii*) the long duration of the dataset makes it a good fit for deep learning and neural network applications [128]. The dataset also fills the gap from emerging economies.

Availability of the contextual data makes *I-BLEND* a good fit for anomaly detection research. Furthermore, the dataset can be used for other research topics including

- **Understand the impact of weather and occupancy on consumption:**

Contextual variables such as weather and occupancy define buildings' energy demand. Weather variables (e.g., temperature and humidity) determine the heating and cooling demand of a building. These context variables keep changing across days and years, so the question to answer is: which of these variables affects the consumption the most, and which of these can be tuned to ensure optimal energy consumption.

- **Evaluate the potential of various applications, such as energy prediction, demand response [129, 102]:** A demand response (DR) program is run by electric utilities to curtail a portion of the peak demand by shifting the energy usage. The success of this program rests on the accuracy of energy prediction. *I-BLEND* can be used to understand the viability of DR in the campus buildings.

- **Understand the impact of various energy parameters on one another:** Each energy parameter (current, voltage, power, frequency, and power factor) defines a distinct feature of the energy data and offers different insights. For example, paper [130] explains a scenario of how power factor was used to identify anomalous air conditioner instead of normally-used energy parameters (current and voltage). This shows releasing all five parameters in *I-BLEND* provides numerous research opportunities to understand the relationship between these parameters.

- **Building's energy modelling:** Building's energy modelling provides a different research direction. Various software packages (EnergyPlus, OpenStudio) take building details (contextual and energy data) as input and answer multiple questions such as building performance rating, building stock analysis, architectural design. So, *I-BLEND* can be used to study the mentioned metrics of the Institute buildings.

Building	**Floors (#)**	**Total area (ft^2)**	**HVAC area (ft^2)**
Academics	5	61308	22785
Lecture	3	17548	12101
Library	4	26401	18354
Dining	4	50191	11840
Facilities	3	09769	NA
Boys Dormitory	6	72745	30120
Girls Dormitory	4	38126	12818

Table 6.2: Building details. Facilities building do not have a centralized HVAC.

6.2 Buildings' details

Indraprastha Institute of Information Technology Delhi (IIIT-D), spread over 25 acres, is an autonomous research institute in Delhi, India. The campus of the institute started in 2012 with newly constructed buildings. It has seven buildings, namely, Academic, Lecture, Library, Facilities, Dining, Boys dormitory, and Girls dormitory. The energy demands of all buildings are met by electricity received from the state electricity board; diesel generators are used in the events of power cuts. Table 6.2 provides an overview of these buildings. The average minimum temperature in Delhi during winters is around 8° Celsius, and average maximum temperature during summers reaches to around 42° Celsius. All buildings on campus, except Facilities building, are connected to a centralized Heating, Ventilation, and air Conditioning (HVAC) system. Therefore, the total energy consumed by these buildings (excluding facilities) includes the energy consumed by HVAC components such as air handler units (AHU). Almost every building is used for a different purpose, so electrical loads used and operational timings vary for each of them. The following description provides an overview of the characteristics of each building.

Academic Building: This building consists of faculty offices, server room (hosting most of the computer servers on campus), and research labs used by research scholars. The electrical loads in this building include computers, fans, lights, AHUs, and two lifts. Most faculty offices remain open from 0800 till 1800 hours, whereas research scholars do not follow a particular schedule, and hence, the building remains occupied with limited numbers during night hours and on Saturdays and Sundays

as well.

Lecture Building: This building consists of nine classrooms. Typically, classes finish on weekdays by 1730 hours, but on some special events (such as "hack nights"), with an average frequency of one per month, a few classes remain occupied till midnight. Electrical loads in this building include lights, fans, and fan coil units of HVAC.

Facilities Building: This building consists of five administrative offices and an electrical panel room. Major electrical loads in this building include lights, fans, and seven window ACs.

Library Building: This building consists of an open area, library, and computer labs. The open area remains open 24x7 for reading; the library remains open from 0830 until midnight. Electrical loads in this building include lights, fans, two lifts, AHUs, and three computer labs, consisting of around 150 computers in total, for courses.

Dining Building: This building consists of a cafeteria which remains open 24x7, a dining floor, computer labs, and a floor with a gym and table tennis rooms. Electrical loads in this building include lights, fans, a lift, AHUs, computers, and gym equipment (two treadmills).

Dormitory Buildings: There are separate dormitories for boys and girls. Each dormitory has single and shared (double, triple) rooms and four lifts. Electrical loads in these buildings include lights, fans, AHUs, and high wall units. The HVAC unit remains operational in these buildings during night hours only. Dormitories have both mains and UPS power supply. Single rooms have one tube light, one fan (on UPS), two plug points (one on mains and another on UPS), one study light (on UPS), and an AC high wall unit (on mains). In the case of the shared rooms, all loads or plug points are multiplied by 2 (or 3) according to sharing type. Students are allowed to use electricity for laptops, desktops, and mobile phones only.

Year	Jan. - April		May - July		Aug. - Nov.	
	Boys	Girls	Boys	Girls	Boys	Girls
2015	394	182	90	56	389	183
2016	428	192	247	144	456	195
2017	438	190	254	129	450	205

Table 6.3: Occupancy count of boys' and girls' dormitories during three sessions for three different years.

Table 6.3 shows the number of students staying in boys' and girls' dormitories during three different sessions for three years. Months January to April and August to November represent two academic semesters and months December and May to July represent winter and summer vacations respectively. During vacations, only graduate students remain on campus. During academic semesters, the number of boys staying in dormitories are approximately double as compared to the number of girls. HVAC remains operational during day hours (9 AM - 5 PM) in Academic, Lecture, and Library; and during night hours (9 PM - 3 AM) in dormitories.

6.3 Energy data collection

Firstly, we explain the metering part, and then the data-collection, storage, and retrieval part.

6.3.1 Metering

We use standard panel meters (Schneider EM6400 [131]) designed for industrial and commercial installations to measure the different electrical parameters. Each unit costs approximately $150. They can measure, display, and communicate approximately 25 different electrical parameters using Modbus protocol. They are three-phase meters and receive input voltage signals via three potential transformers (PTs) and current signals via three current transformers (CTs). Using these voltage and current signals, other electrical parameters are computed internally using various formulae mentioned in the User Manual [131]. They store these pa-

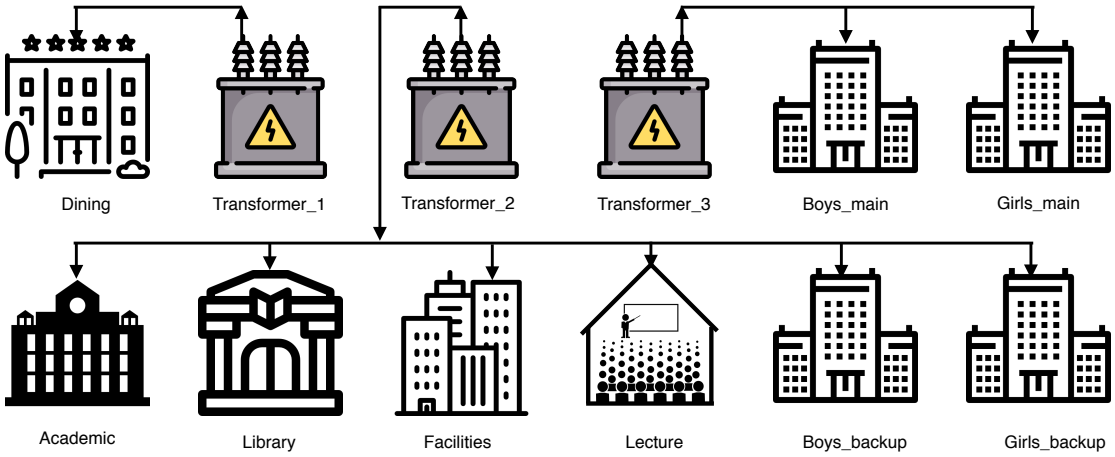


Figure 6.1: Flow diagram of transformers and building meters. Transformer_1 and Transformer_2 have several auxiliary loads that are not released in the paper.

rameters temporarily in registers, and these registers are updated every few seconds (we experimentally observed the update interval to be two seconds). In accordance with IEC 62053-21 standards, they have an accuracy class of 1.0, meaning that measurement error can be up to 1 percent. They are precalibrated meters, and the manufacturer does not recommend any subsequent recalibration. Over a duration of 4.5 years, we did not find any case of meter breakdown.

Each building on campus has a separate EM6400 meter installed on building mains. The institute has three transformers, which supply energy to different buildings on campus, and each transformer has a meter installed. We label these transformers as Transformer_1, Transformer_2, and Transformer_3. The mains and UPS consumption in dormitories are measured with separate meters, and accordingly, meter names are suffixed with the words "mains" or "backup" (e.g., Boys_main, Boys_backup). Figure 6.1 shows a flow diagram between the transformers and the different buildings on campus.

6.3.2 Data collection

We use an open source Simple Measuring and Actuation Profile (sMAP) platform to collect, store, and retrieve the data [132]. Figure 6.2 provides an overview of the entire setup. The EM6400 meters connected to the building's main supply

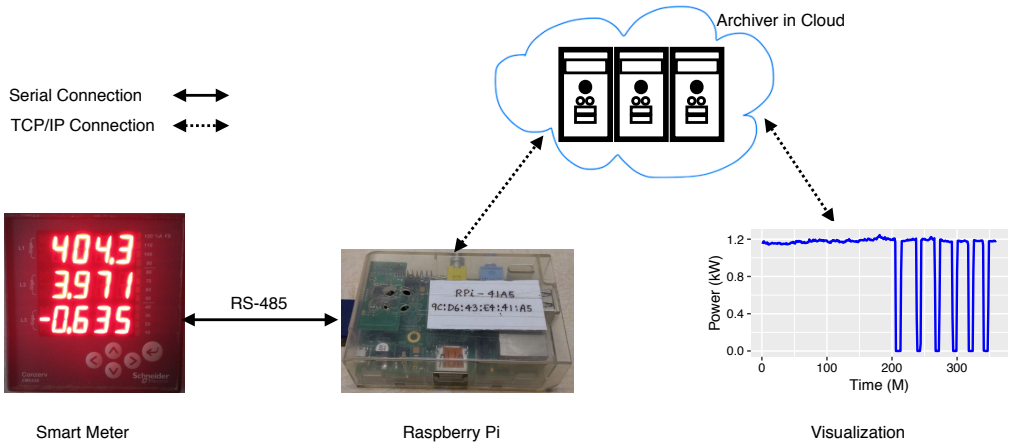


Figure 6.2: Data collection and visualization setup

measure different parameters such as voltage, current, power, frequency, and power factor. These meters are then connected to Raspberry Pi modules via RS-485 serial transmission standard, and the measured parameters are collected from meters to Raspberry Pi through a Modbus communication protocol. A self-designed USB-RS485 converter is used to read data from meters into the sMAP application. The sMAP application, running in Raspberry Pi, polls these meters at a frequency of 30 seconds and temporarily stores the response data (parameter values) on the memory card of the Raspberry Pi. Raspberry Pi's are connected to the reliable Internet via an Ethernet interface, so all Pi's remain time synchronized. In real-time, sMAP components on the same Raspberry Pi format each received parameter values from a smart meter by assigning a unique identifier and attaching the required metadata. The sMAP modules on Pi publish these time-series parameters to another sMAP module in Cloud, called as *Archiver*, where this data is stored in a *readingdb* database [133]. The sMAP Archiver provides API for both visualizing and fetching the data.

6.4 Contextual data collection

Apart from buildings' details, other contextual data includes building level occupancy, weather data, and the institute calendar.

ALGORITHM 5: Steps to compute occupancy count of each building

Input: X - SNMP traps log for a time period T.
Output: Time series occupancy count B_i^o for each of seven buildings for time period T, where $i \in \{1, \ldots, 7\}$

1 Separate X into different buildings B_i where $i \in \{1,\ldots, 7\}$ and represents building number. Each trap in X contains a building identifier.
2 **for** $i \leftarrow 1$ **to** 7 **do**
3 $C \leftarrow$ Extract only `ciscoLwappDot11ClientMovedToRunState` traps from B_i.
4 **foreach** *trap c in C* **do** Create a time series ON sequence with start and stop timings of c
5 Bind all ON sequences along the time series index to create a dataframe DF
6 Perform row wise addition of DF to create final time series occupancy count B_i^o of the building B_i
7 return B_i^o

6.4.1 Occupancy data

With the ubiquity of wireless devices (laptops, smartphones, etc.), Wi-Fi infrastructure is increasingly being used for indoor localization, buildings occupancy monitoring, and footfall measurement [134, 135, 136, 137, 138, 139, 140]; and it has been found comparable in accuracy to other occupancy monitoring techniques. IIIT-Delhi, being a newly constructed academic campus, has a rich deployment of wireless Cisco Access Points (series 1100, 1600, 1800, 2800, 3700, and 1500) across the campus and has seamless wireless Internet access. Each access point covers a radius of around 15 meters. On top of the Wi-Fi infrastructure, we have built a separate system [138, 139], which collects SNMP (Simple Network Management Protocol) traps of all the buildings on campus. It has been operational since February 16, 2014. A trap is a data packet generated by a wireless Access Point whenever a client (laptop, mobile phone, etc) connects or disconnects with the access point. Each trap distinguishes itself from the remaining traps as each trap is associated with unique information consisting of a client_id, an access point_id, the trap type, a timestamp, etc. The access point_id is created such that it acts as a building identifier, e.g., ACB3FAP2 refers to the second access point that is installed on the third floor of the academic building. All traps are forwarded to a central server,

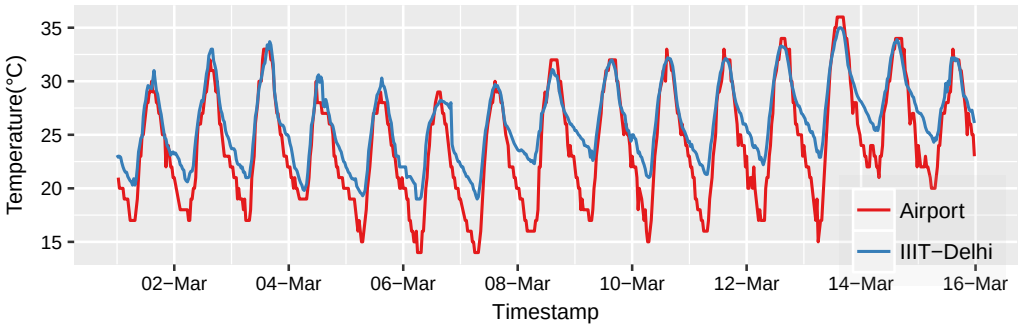


Figure 6.3: Half-hourly temperature at IGIA Airport and IIIT-Delhi campus from March 1, 2018 until March 15, 2018. **[Best viewed in color]**

which creates a log of all received traps. Analysis of these logs provides various details, such as how long a client was connected to a specific access point or how many clients are connected to an access point. We use these Wi-Fi logs to extract the occupancy details for each building on the campus. We first separate traps into different groups using a building identifier and later compute the occupancy count for each building separately using Algorithm 5. The occupancy system is explained in detail in [138, 139].

By default, the occupancy dataset is at a one-minute resolution, and for this paper, we downsampled it to a 10-minute resolution by taking the maximum value of each 10-minute consecutive window. Downsampling does not reduce the occupancy accuracy significantly as the average difference found between a maximum and a minimum values of a window is eight, i.e., downsampling at max reduces the occupancy accuracy by eight occupants. The supplied occupancy data is from February 16, 2014, to November 3, 2017 (almost four years).

6.4.2 Weather data

Weather data is publicly available from a free weather service, namely, Weather Underground (WU) [126]. The weather station is deployed and maintained by the Indian Metrological department, and the data is available through WU API. This weather station records about 13 weather parameters at a nearby location, IGIA Airport, New Delhi, which has the coordinates: 28.5667° latitude, 77.1167° longi-

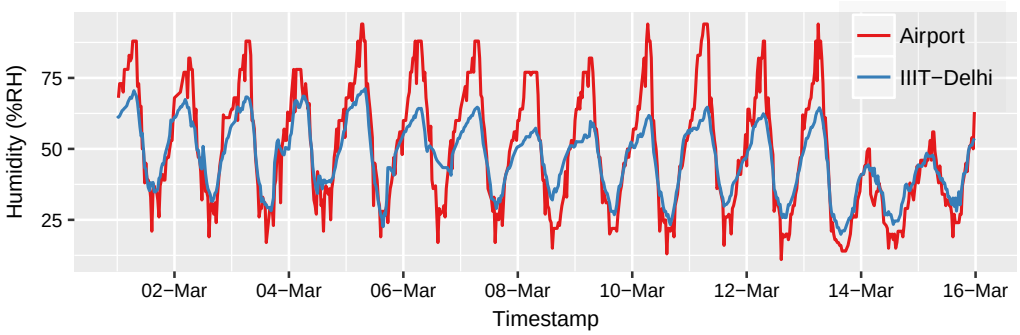


Figure 6.4: Half-hourly humidity at IGIA Airport and IIIT-Delhi campus from March 1, 2018 until March 15, 2018. **[Best viewed in color]**

tude, and 237 meters elevation. The coordinates of IIIT-Delhi campus buildings are 28.5463° latitude, 77.2732° longitude, and 226 meters elevation. The distance between the airport weather station and IIIT-Delhi campus buildings is approximately 14.8 Km. The half-hourly weather data of the station was downloaded using WU API. We have downloaded temperature, humidity, wind speed, and wind direction data for the same duration as the power consumption data with our script at GitHub [141]. The exact position and the make and model of the measuring instruments at the IGIA Airport was not revealed to us by the Indian Meteorological department.

To understand the difference in weather at IIIT-Delhi compared to the airport, we measured the outdoor temperature and humidity at IIIT-Delhi campus with an off-the-shelf Elitech RC-4HC [142] temperature and humidity data logger, costing around $120, continuously for four months. The data logger was fixed on an outside wall facing south but was shielded from direct sunlight. It has a temperature accuracy and resolution of ± 0.5°C and 0.1°C respectively, and humidity accuracy of ±3%RH. The Pearson Correlation coefficient of the temperatures at the two sites (IIIT-Delhi and IGIA Airport) is 0.96, and that of humidity is 0.90. This suggests that temperatures and humidity at the two sites vary in the same pattern.

Figure 6.3 shows the variation in temperature at both sites for two continuous weeks, and Table 6.4 shows mean temperature and standard deviation at both sites during daytime (8 AM - 6 PM) and nighttime hours separately. When comparing

Time	Mean temperature		Standard deviation	
	Airport	IIITD	Airport	IIITD
Day	34.53	34.01	5.64	4.2
Night	28.63	30.44	5.61	4.2

Table 6.4: Day and night temperature features – mean and standard deviation at IGIA airport and IIIT-Delhi campus over a duration of four months from March to June 2018.

these sites, daytime temperatures did not differ, but during night hours, a mean difference of 1.81 Celsius was found. The airport area is a non-residential open area, while IIIT-Delhi (Okhla, Phase III) is a residential-cum-industrial area. As a result, nighttime temperatures are relatively lower at the airport station when compared to the IIIT-Delhi station. Similarly, Figure 6.4 shows the variation in humidity at both sites. A mean difference of 4.82 and 1.58 is found in humidity among the two sites during the day and night hours respectively. The insignificant difference in weather between the two sites suggests that IGIA weather data can be used to study the impact of weather variables on IIIT-Delhi buildings' energy consumption.

6.4.3 Calendar

The Institute has a calendar which shows working, non-working, and the semester days. We encode all this information in three column CSV file where the first column contains dates, the second column shows whether the day was working or not, and the third column shows whether the day was classified as high or low activity period. High activity period corresponds to days when the academic semester was going on, and students were in campus dormitories. Semester breaks, summer and winter breaks, and festival breaks of several consecutive days during the semesters were considered as low activity period. This information would be helpful for energy forecasting applications.

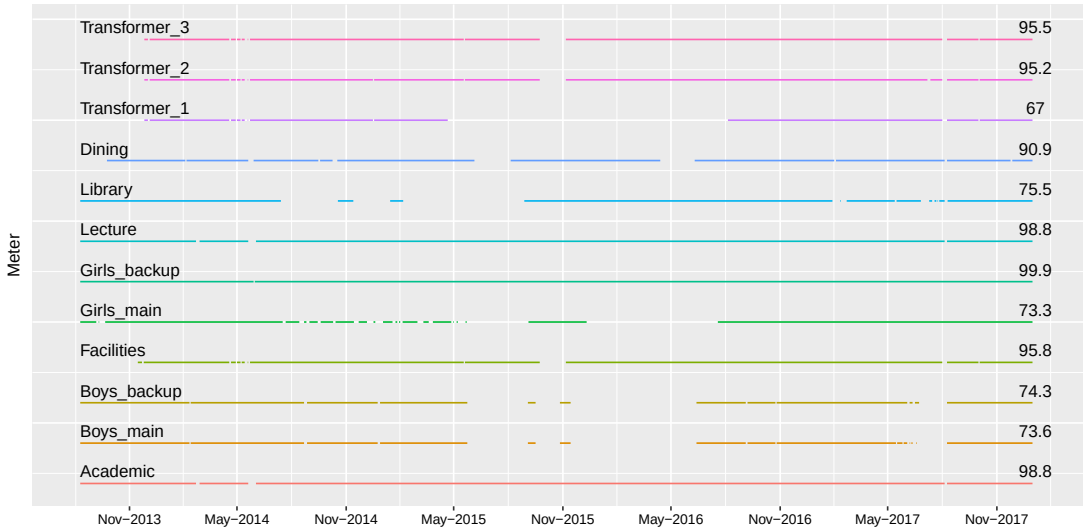


Figure 6.5: Gaps in the lines represent days on which more than a quarter of the meter readings are missing. Uptime represents the percentage of days for which more than a quarter of the meter readings are present. [**Best viewed in color**]

6.5 Data cleaning

The smart meters installed at the campus log data every 30 seconds. We down-sampled data to a uniform one minute rate because of the following reasons: (i) readings from different meters were not time synchronized with one another, so for better comparison we aligned them at uniform one-minute durations, and (ii) the dataset is not targeted for energy disaggregation research, so the sampling frequency of 1 minute is good enough for remaining applications, such as forecasting and benchmarking.

Data logging for the academic building, the lecture building, dormitories, and the library building started from August 10, 2013. And for the facilities building and transformers, it started from November 15 and 26, 2013 respectively. For this paper, we have considered data until December 31, 2017. Gaps in Figure 6.5 show the different days in which more than a quarter of the readings ($> \frac{1440}{4}$) from a particular meter were missing. The secondary y-axis of the plot shows the number of days in percentage when the meter has more than a quarter of readings. We denote this with the uptime metric. This plot shows that meters, namely, Academic, Girls_backup, and Lecture, have the least missing data. Meters showing similar gap patterns are

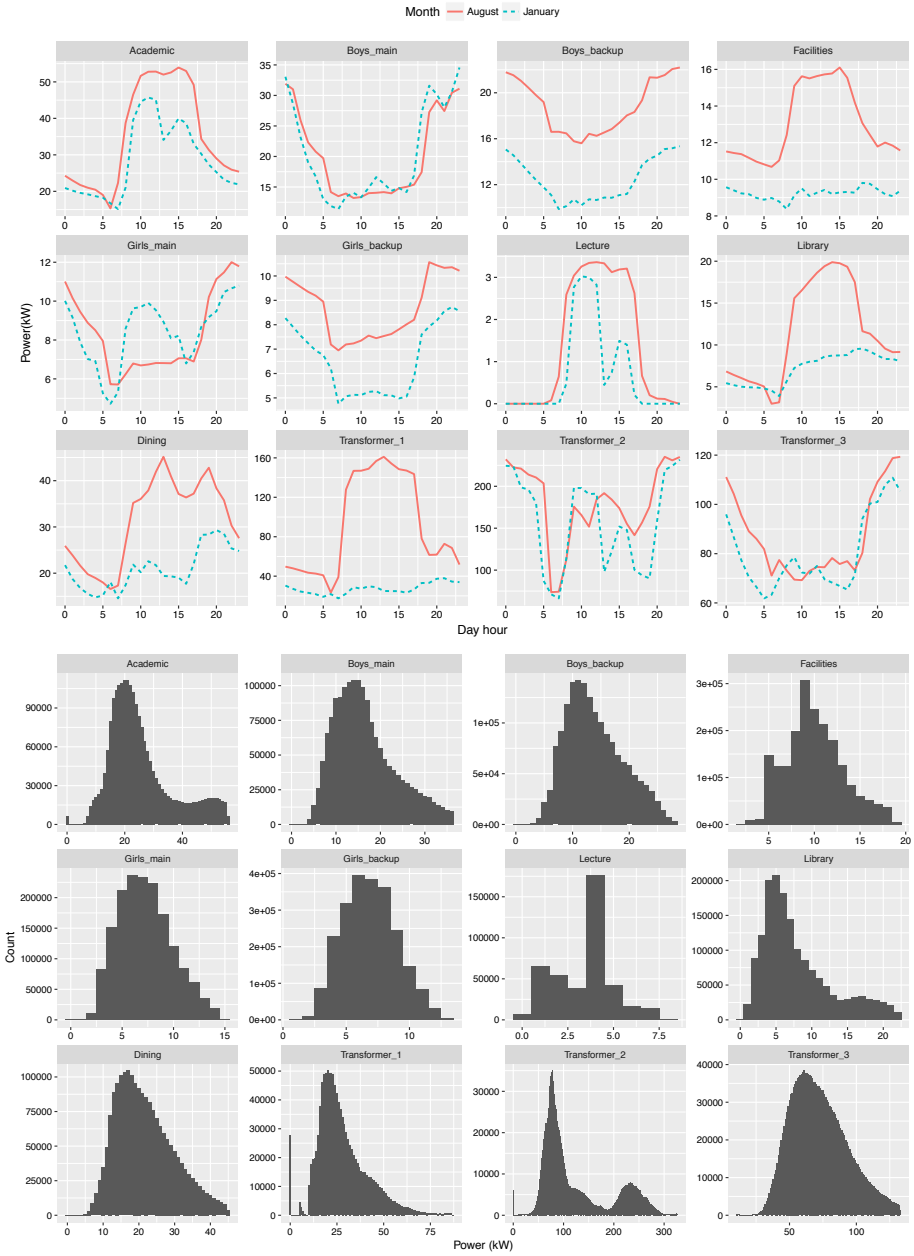


Figure 6.6: **Top:** Average power consumption across the same hours on different days of Aug. 2016 & Jan. 2017 (Months with different seasonality pattern). **Bottom:** Histograms of the power consumption of different meters from August 2013 until December 2017 when the histogram bin width is 1 kW. [**Best viewed in color**]

controlled by the same Raspberry Pi. Reasons for missing data include faulty power supply or corrupted memory card of Raspberry Pi. Running Little's test [143] for Missing Completely at Random (MCAR), we got a p value of 0.51, suggesting that all missing instances are MCAR, and hence, no hidden factor controls the pattern of such missing observations.

6.6 Data validation

In this section, first, we will show the technical validation of the energy data and later of the occupancy data.

6.6.1 Energy data

Figure 6.6(top) shows the average power consumption at different daytime hours in August 2016 and January 2017. These months differ in terms of academic semesters, and seasonality and hence represent summer and winter consumption patterns. The academic building remains occupied mostly between 0800 and 1800 hours, so the consumption is highest in this time range as compared to the remaining daytime hours. The lecture building follows a strict daily operation schedule because most classes finish by 1730 hours with the exception of when some extra classes happen. Dormitories consume more power during the nighttime than during the daytime since all students remain inside their rooms. Consumption in facilities building remains constant during the night and day in January 2017 as it is isolated from the centralized HVAC system. While the higher consumption during daytime hours in August 2016 is due to window ACs. The Dining building has some fixed usage due to the gym and labs; apart from this, its consumption increases at lunch and dinner time.

Figure 6.6(bottom) shows histograms of the power consumption of all buildings and transformers from August 2013 to December 2017. These plots indicate that academic building follows a bimodal distribution having 20 and 50 kW frequently

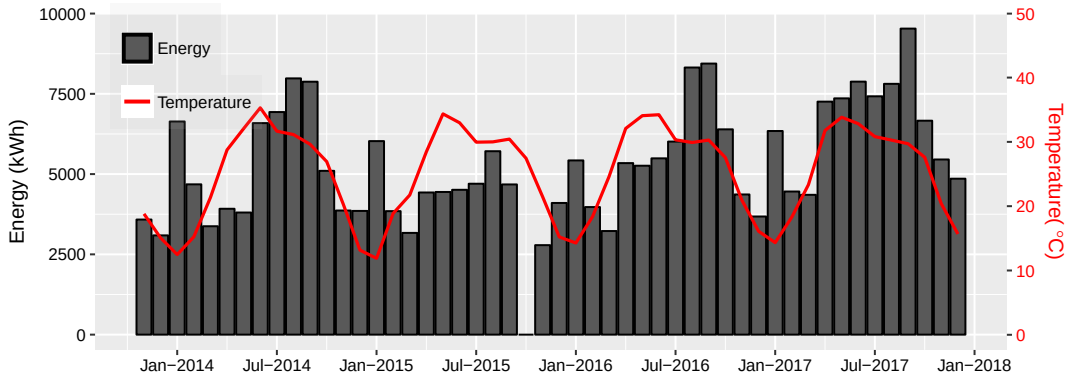


Figure 6.7: Monthly average energy consumption and temperature from November 26, 2013, until December 2017.

occurring loads. 50 kW represents working hours consumption, and 20 kW represents night hours consumption. Boys' and girls' dormitories have 31.5 kW (17.5 of main and 14 of backup) and 13.8 kW (7 of main and 6.8 of backup) frequently occurring loads, respectively. Facilities, Lecture, Library, and Dining have frequently occurring loads of around 10, 2, 10 and 20 kW, respectively.

Figure 6.7 shows the monthly average energy consumption of the campus (sum of transformers 1, 2, and 3) and temperature from 2013 until 2017. Note in Figure 6.5, that the meter connected to Transformer_1 did not log data from May 2015 until August 2016, and meters of the remaining two transformers did not log data for October 2015. As a result, in Figure 6.7, the increase in energy consumption during summer 2015 is not clear as found in the remaining years of the figure. The higher consumption during summers and January is due to cooling and heating loads, respectively.

6.6.2 Occupancy data

Figure 6.8 shows the occupancy pattern of all seven buildings on campus for a week. The occupancy count hardly ever reaches zero as the campus is residential and students work during days as well as nights. Lecture rooms remain closed during night hours, so occupancy reaches to zero in the lecture building at night. Figure 6.9 shows the power consumption and occupancy count of the academic building for

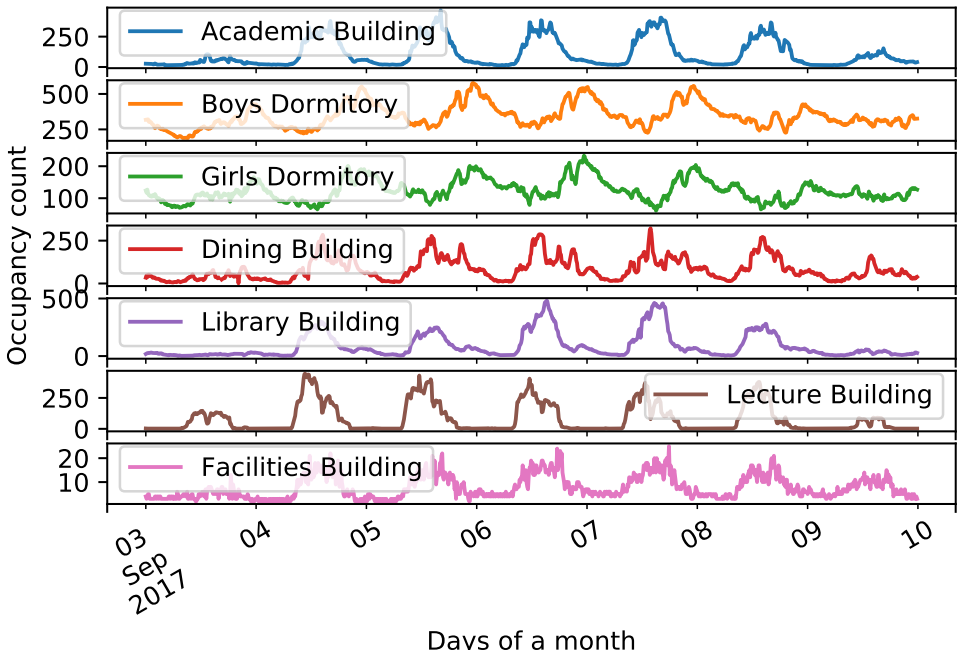


Figure 6.8: Occupancy of the campus buildings for one week of September 2017. [**Best viewed in color**]

five consecutive days at half-hourly intervals. April 1 was a Saturday, so only non-teaching staff and some Ph.D. students were present in the building as compared to April 3-5, which were working days. The Pearson correlation coefficient between power consumption and the occupancy count of the plotted data is 0.89. For all the buildings on campus, we computed the correlation coefficient between the power consumption and the occupancy count using one week (June 11 to June 16, 2017) of data. We chose this duration because there were no missing values in any of

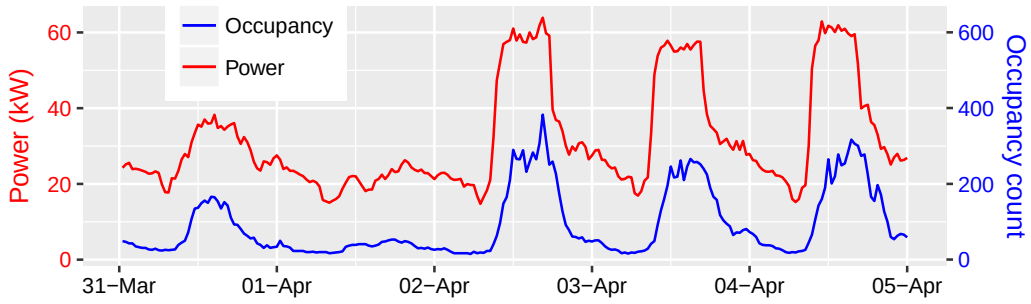


Figure 6.9: Half-hourly power consumption and occupancy of academic building from April 1, 2017, until April 5, 2017. [**Best viewed in color**]

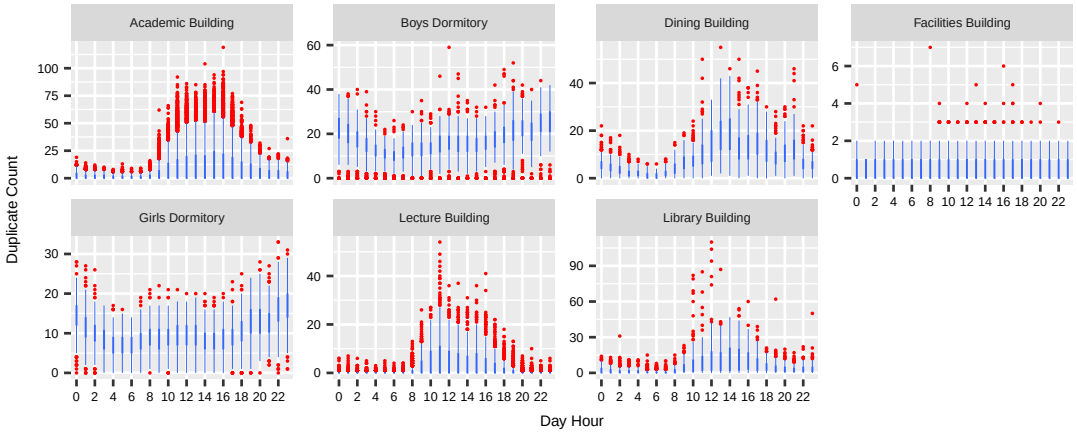


Figure 6.10: Box plots showing the distribution of occupants (duplicate count) using multiple devices at the same time at different day hours over two months (Aug. and Sept. 2017). [**Best viewed in color**]

the buildings' data. The correlation coefficient for the academic building, the boys' dormitory, the dining building, the library, the lecture, and facilities buildings is 0.87, 0.75, 0.75, 0.83, 0.80, 0.71, and 0.87, respectively. A significant correlation ($\geq$ 0.70) between the power consumption and the occupancy suggests that occupancy data can be used as a parameter for predicting the energy consumption of a building.

Occupancy estimated via SNMP traps have two error sources: (i) when an occupant has more than one device connected to the Wi-Fi simultaneously, and (ii) when an occupant does not connect to Wi-Fi. The former case results in occupancy over-estimation, while the latter case results in occupancy under-estimation. We calculated the occupancy over-estimation, but it is difficult to find the under-estimation due to the unavailability of the required data. However, under or over-estimation may vary across the sessions (mentioned in Table 6.3), but it remains constant across the days of the same session as the number of students staying on campus remains constant throughout a session. Occupancy over or under-estimation should not limit the study of occupancy data on energy consumption since we found a significant impact of the change in occupancy on the energy consumption duration during our analysis.

The IT department of the institute maintains a registration directory which lists each current user and the set of devices registered by the user. While knowing MAC

addresses from this directory, we counted the number of active devices from SNMP logs pointing to the same user at a specific timestamp as both the registry and SNMP logs contain device MAC addresses. In this way, we calculate the number of duplicate connections for all buildings on the campus. Box plots in Figure 6.10 show the distribution of such duplicate connections hour-wise for August and September 2017. The figure shows that the number of duplicates increases during day hours in the academic, the dining, the lecture, and the library buildings, while in the dormitories, it increases during night hours, and in the facilities building, no significant change is found. Box plots show that in the academic building, which witnesses high movement, occupancy count from SNMP overestimates by 50 and in remaining buildings, it overestimates up to 20. We could not estimate the duplicate connections for all the months of the dataset due to the unavailability of the required data as the registration directory is updated in July every year with the new admissions data. During the update, new MAC addresses are added, and older ones are removed.

6.7 Summary

Unavailability of context-rich energy datasets makes it difficult to verify the underlying nature of detected anomalies. Now, with the availability of our campus dataset researchers can study different topics such as analyzing the impact of weather or occupancy schedule on energy consumption, detecting anomalies, and developing algorithms for predictive maintenance.

Chapter 7

Conclusion and future work

7.1 Conclusion

Buildings waste a considerable portion of the total available electrical energy. Studies show a significant percentage of it can be reduced by identifying the wastage instances timely. So, to propose economical solutions for such a compelling problem, we explored the use of smart meters in identifying such instances in a timely manner. Smart meters are installed at buildings' mains line by electrical utilities for smart billing and other energy efficiency programs. Leveraging smart meter data for identifying energy wastage (anomalous) instances does not require extra instrumentation and such meters allow logging of different energy parameters reliably. Governments across the countries are making it mandatory to instrument buildings with smart meters. To use smart meter data for identifying anomalies, we identified two challenges. Firstly, we found that existing detection techniques result in a high number of false alarms. As a result, building administrators find them unreliable and ignore the alarms. Secondly, we found that existing techniques only detect anomalies and do not identify the anomaly causing appliance. Manual identification of the anomaly causing appliance is a difficult process and results in energy wastage for a longer duration. Furthermore, we found a lack of publicly available anomaly annotated and contextual rich datasets.

Towards the first challenge, we proposed an anomaly detection approach, *Mon-*

itor. The comparison of this approach with existing approaches shows that it improves the anomaly detection accuracy by up to 24% in best scenario and on average by 14%. This improvement in accuracy reduces the false positive anomaly detection rate significantly and makes it suitable for real-world deployments.

Towards the second challenge, we proposed an anomaly detection approach, RIMOR. It detects anomalies at user-defined time intervals and identifies the anomaly causing appliance. Results show that it is 15% more accurate in detecting anomalies as compared to four other baseline approaches and it reports an appliance identification accuracy of 82%. Further, we performed detailed experiments to understand the applicability of NILM in identifying anomalous appliances. Results show that current NILM techniques can identify anomalies caused due to high energy consuming appliances only. More work is required in the NILM domain for proposing anomaly aware disaggregation techniques for all appliances.

While proposing *Monitor* and RIMOR we annotated various[1,2,3,4,5] publicly available datasets thus showing anomalous instances. Furthermore, we also collected a context rich dataset spanning almost four years from our campus. All these datasets are made publicly available and can be leveraged to move the state-of-the-art forward in the domain.

7.2 Future work

While working on the mentioned challenges, we identified various other issues which need to be addressed properly. The issues and the possible approaches to handle them include:

1. **Automate anomaly annotation:** Current energy data collecting systems only collect consumption data and do not collect additional information such

[1]https://tinyurl.com/y4ydtwek
[2]https://tinyurl.com/y3xlk69m
[3]https://tinyurl.com/y35x9l83
[4]https://tinyurl.com/y68h6bxy
[5]https://tinyurl.com/y5evvp7x

as unusual events. Such datasets can be used for several research purposes, but as such, they cannot be used to propose and validate anomaly detection algorithms, until and unless incidence logs have been maintained. So, it is vital to collect additional information which can then be used for anomaly detection to reduce energy wastage. It can be done by running an existing anomaly detection algorithm, at the time of data collection (for example [144]) in the background on the smart meter data, and whenever any unusual event is detected, ask the consumers for this unusual behavior and log the response. Also, consumers can be requested to voluntarily log events whenever any unusual activity is found.

2. **Maintain anomaly signatures database:** Almost every anomaly shows a different consumption signature and knowing appliances' anomalous signatures helps to understand appliances' behaviors during different faults or misconfigurations. Like ODDS Library [145], energy community needs to maintain a database of anomalous appliance signatures. Maintaining such a database would help in two ways (i) Researchers may understand anomalies in a better manner and propose improved anomaly detection algorithms. (ii) Appliance manufacturers may use the database for further testing of the appliances.

3. **Develop anomaly aware energy disaggregation techniques:** Over past several years, energy disaggregation community has shown significant improvement in disaggregating appliances consumption from the aggregate smart meter data [13, 107]. However, this work has been done on the data of appliances behaving normally. Our detailed experiments show that when state-of-the-art NILM algorithms [13, 98] are used on data from appliances showing anomaly, the predicted output is not sufficiently accurate. It is found that disaggregation process does not recover the energy consumption signature of an appliance correctly and it deviates significantly from appliance's true signature. This shows that energy disaggregation research needs to be extended to include appliances showing anomalies. There are two directions for future work:

(i) developing novel anomaly detection rules that are suitable for NILM-based detection; (ii) designing anomaly aware NILM algorithms, without relying on learning normal operation load signatures.

4. **Develop context-aware anomaly detection algorithms:** An anomaly can be of any type and the reason responsible for the anomaly either can be appliance malfunctioning or due to change in contextual factors. So, flagging an anomaly solely on detecting an unusual change in smart meter data is not a complete and smart solution to the energy wastage problem [33, 46, 22]. Instead, maximum contextual information should be fused in new algorithms which eventually will reduce false alarm rate and thus improve occupants reliability on such automated systems. We believe that instrumenting buildings with hard sensors extensively is not an elegant solution. Instead, we need to develop new inference algorithms which will extract the required information from already installed sensors [146].

Availability of smart meter data has made it possible to propose economical, effective, and scalable solutions for identifying energy wastage instances. In this thesis, we have worked towards developing anomaly detection techniques using smart meter data only to provide a scalable approach for anomaly detection.

Bibliography

[1] EIA. International Energy Outlook. Technical report, Energy Information Administration (EIA), 2017.

[2] David Lindley. Smart grids: The energy storage problem. *Nature News*, 463(7277):18–20, 2010.

[3] Omar Ellabban, Haitham Abu-Rub, and Frede Blaabjerg. Renewable energy resources: Current status, future prospects and their enabling technology. *Renewable and Sustainable Energy Reviews*, 39:748–764, 2014.

[4] Science—Business. Resouce Innovation - New Ideas for Managing Scarce Resources and Energy. Technical report, Science—Business Symposium, Report at: http://goo.gl/HVFRtK, 2012.

[5] EnergyCentral. US Now Leads in Energy Waste. `https://www.energycentral.com/c/ec/us-now-leads-energy-waste`, 2013.

[6] W. Sisson, C. Van-Aerschot, C. Kornevall, R. Cowe, D. Bridoux, T. B. Bonnaire, and J. Fritz. Energy efficiency in buildings: Transforming the market. *Switzerland: World Business Council for Sustainable Development (WBCSD)*, 2009.

[7] K. Srinivas and M. Brambley. Methods for Fault Detection, Diagnostics, and Prognostics. *HVAC and R Research*, 11:3–25, 2005.

[8] K. W. Roth, D. Westphalen, M. Y. Feng, P. Llana, and L. Quartararo. Energy impact of commercial building controls and performance diagnostics: market

characterization, energy impact of building faults and energy savings potential. *Prepared by TAIX LLC for the US Department of Energy. November. 412pp (Table 2–1)*, 2005.

[9] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41(3):15, 2009.

[10] Soma Shekara Sreenadh Reddy Depuru, Lingfeng Wang, and Vijay Devabhaktuni. Smart meters for power grid: Challenges, issues, advantages and status. *Renewable and sustainable energy reviews*, 15(6):2736–2742, 2011.

[11] Andrés Molina-Markham, Prashant Shenoy, Kevin Fu, Emmanuel Cecchet, and David Irwin. Private memoirs of a smart meter. In *Proceedings of the 2nd ACM workshop on embedded sensing systems for energy-efficiency in building*, pages 61–66. ACM, 2010.

[12] Balakrishnan Narayanaswamy, Bharathan Balaji, Rajesh Gupta, and Yuvraj Agarwal. Data driven investigation of faults in HVAC systems with model, cluster and compare (MCC). In *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, pages 50–59. ACM, 2014.

[13] J Zico Kolter and Matthew J Johnson. REDD: a public data set for energy disaggregation research. In *Workshop on Data Mining Applications in Sustainability (SIGKDD), San Diego, CA*, volume 25, pages 59–62, 2011.

[14] Sean Barker, Aditya Mishra, David Irwin, Emmanuel Cecchet, Prashant Shenoy, and Jeannie Albrecht. Smart*: An open data set and tools for enabling research in sustainable homes. *SustKDD, August*, 111:112, 2012.

[15] Nipun Batra, Manoj Gulati, Amarjeet Singh, and Mani B Srivastava. It's different: Insights into home energy consumption in india. In *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings*, pages 1–8. ACM, 2013.

[16] Jack Kelly and William Knottenbelt. The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes. *Scientific data*, 2:150007, 2015.

[17] Akshay SN Uttama Nambi, Antonio Reyes Lua, and Venkatesha R Prasad. LocED: location-aware energy disaggregation framework. In *Proceedings of the 2Nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, pages 45–54. ACM, 2015.

[18] Wilhelm Kleiminger, Christian Beckel, and Silvia Santini. Household occupancy monitoring using electricity meters. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 975–986. ACM, 2015.

[19] Stephen Makonin, Bradley Ellert, Ivan V Bajić, and Fred Popowich. Electricity, water, and natural gas consumption of a residential house in canada from 2012 to 2014. *Scientific data*, 3, 2016.

[20] David Murray, Lina Stankovic, and Vladimir Stankovic. An electrical load measurements dataset of united kingdom households from a two-year longitudinal study. *Scientific Data*, 4:160122, 2017.

[21] Dataport. https://dataport.pecanstreet.org/, 6 2018.

[22] Pandarasamy Arjunan, Harshad D Khadilkar, Tanuja Ganu, Zainul M Charbiwala, Amarjeet Singh, and Pushpendra Singh. Multi-user energy consumption monitoring and anomaly detection with partial context information. In *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, pages 35–44. ACM, 2015.

[23] Haroon Rashid, Pandarasamy Arjunan, Pushpendra Singh, and Amarjeet Singh. Collect, compare, and score: a generic data-driven anomaly detection method for buildings. In *Proceedings of the Seventh International Conference on Future Energy Systems Poster Sessions*, page 12. ACM, 2016.

[24] Haroon Rashid and Pushpendra Singh. Monitor: An abnormality detection approach in buildings energy consumption. In *2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC)*, pages 16–25.

[25] Haroon Rashid, Nipun Batra, and Pushpendra Singh. Rimor: Towards identifying anomalous appliances in buildings. In *Proceedings of the 5th ACM Conference on Systems for Built Environments*, pages 33–42. ACM, 2018.

[26] K. C. Armel, A. Gupta, G. Shrimali, and A. Albert. Is disaggregation the holy grail of energy efficiency? the case of electricity. *Energy Policy*, 52:213–234, 2013.

[27] Huan Yang. Residential energy data analysis using green button data. Technical report, Leigh University, Bethlehem, PA(United States), 2014.

[28] Jack Kelly and William Knottenbelt. Neural NILM: Deep neural networks applied to energy disaggregation. In *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, pages 55–64. ACM, 2015.

[29] H. Rashid and P. Singh. Energy disaggregation for identifying anomalous appliance. In *Proceedings of the 4th ACM International Conference on Systems for Energy-Efficient Built Environments*, page 41. ACM, 2017.

[30] Haroon Rashid, Pushpendra Singh, Vladimir Stankovic, and Lina Stankovic. Can non-intrusive load monitoring be used for identifying an appliance's anomalous behaviour? *Applied Energy*, 238:796–805, 2019.

[31] Haroon Rashid, Vladimir Stankovic, Lina Stankovic, and Pushpendra Singh. Evaluation of non-intrusive load monitoring algorithms for appliance-level anomaly detection. In *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019.

[32] Haroon Rashid, Pushpendra Singh, and Amarjeet Singh. I-BLEND, a campus scale commercial and residential buildings electrical energy dataset. *Scientific Data*, 2019.

[33] Gowtham Bellala, Manish Marwah, Martin Arlitt, Geoff Lyon, and Cullen E Bash. Towards an understanding of campus-scale power consumption. In *Proceedings of the Third ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings*, pages 73–78. ACM, 2011.

[34] H. Rashid, P. Arjunan, P. Singh, and A. Singh. Collect, compare, and score: A generic data-driven anomaly detection method for buildings. In *Proceedings of the Seventh International Conference on Future Energy Systems Poster Sessions*, e-Energy '16, pages 12:1–12:2, Waterloo, Ontario, Canada, 2016. ACM.

[35] Tanuja Ganu, Dwi Rahayu, Deva P Seetharam, Rajesh Kunnath, Ashok Pon Kumar, Vijay Arya, Saiful A Husain, and Shivkumar Kalyanaraman. Socketwatch: an autonomous appliance monitoring system. In *Pervasive Computing and Communications (PerCom), 2014 IEEE International Conference on*, pages 38–43. IEEE, 2014.

[36] Welma Pereira, Alois Ferscha, and Klemens Weigl. Unsupervised detection of unusual behaviors from smart home energy data. In *International Conference on Artificial Intelligence and Soft Computing*, pages 523–534. Springer, 2016.

[37] Romain Fontugne, Jorge Ortiz, Nicolas Tremblay, Pierre Borgnat, Patrick Flandrin, Kensuke Fukuda, David Culler, and Hiroshi Esaki. Strip, bind, and search: a method for identifying abnormal energy consumption in buildings. In *Proceedings of the 12th international conference on Information processing in sensor networks*, pages 129–140. ACM, 2013.

[38] Shravan Srinivasan, Arunchandar Vasan, Venkatesh Sarangan, and Anand Sivasubramaniam. Bugs in the freezer: Detecting faults in supermarket refrigeration systems using energy signals. In *Proceedings of the 2015 ACM Sixth*

International Conference on Future Energy Systems, pages 101–110. ACM, 2015.

[39] Y. Zhao, S. Wang, and F. Xiao. Pattern recognition-based chillers fault detection method using support vector data description (SVDD). *Applied Energy*, 112:1041–1048, 2013.

[40] Z. Du, X. Jin, and Y. Yang. Fault diagnosis for temperature, flow rate and pressure sensors in VAV systems using wavelet neural network. *Applied energy*, 86(9):1624–1631, 2009.

[41] H. Han, Z. Cao, B. Gu, and N. Ren. PCA-SVM-based automated fault detection and diagnosis (AFDD) for vapor-compression refrigeration systems. *HVAC and R Research*, 16(3):295–313, 2010.

[42] J. Ploennigs, B. Chen, A. Schumann, and N. Brady. Exploiting Generalized Additive Models for Diagnosing Abnormal Energy use in Buildings. In *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings*, pages 1–8. ACM, 2013.

[43] John E. Seem. Pattern Recognition Algorithm for Determining Days of the Week with Similar Energy Consumption Profiles. *Energy and Buildings*, 37(2):127–139, 2005.

[44] Bernard Rosner. Percentage points for a generalized ESD many-outlier procedure. *Technometrics*, 25(2):165–172, 1983.

[45] Chrys Caroni and Philip Prescott. Sequential application of wilks's multivariate outlier test. *Applied Statistics*, pages 355–364, 1992.

[46] John E Seem. Using intelligent data analysis to detect abnormal energy consumption in buildings. *Energy and Buildings*, 39(1):52–58, 2007.

[47] Boris Iglewicz and David C Hoaglin. *How to detect and handle outliers*, volume 16. ASQC Quality Press Milwaukee (Wisconsin), 1993.

[48] Y. Zhang, W. Chen, and J. Black. Anomaly Detection in Premise Energy Consumption Data. In *Power and Energy Society General Meeting, 2011 IEEE*, pages 1–8. IEEE, 2011.

[49] D. Liu, Q. Chen, K. Mori, and Y. Kida. A method for detecting abnormal electricity energy consumption in buildings. *Journal of Computational Information Systems*, 6(14):4887–4895, 2010.

[50] Fei Liu, Huijing Jiang, Young M Lee, Jane Snowdon, and Michael Bobker. Statistical modeling for anomaly detection, forecasting and root cause analysis of energy consumption for a portfolio of buildings. In *12th International Conference of the International Building Performance Simulation Association*, 2011.

[51] John Kelly Kissock, Jeff S Haberl, and David E Claridge. Inverse modeling toolkit: numerical algorithms. *ASHRAE transactions*, 109:425, 2003.

[52] J.-S. Chou and A. S. Telaga. Real-time Detection of Anomalous Power Consumption. *Renewable and Sustainable Energy Reviews*, 33:400–411, 2014.

[53] Michael Wrinch, Tarek HM El-Fouly, and Steven Wong. Anomaly detection of building systems using energy demand frequency domain analysis. In *Power and Energy Society General Meeting, 2012 IEEE*, pages 1–6. IEEE, 2012.

[54] Chao Chen and Diane J Cook. Energy outlier detection in smart environments. *Artificial Intelligence and Smarter Living*, 11:07, 2011.

[55] Jessica Lin, Eamonn Keogh, Stefano Lonardi, and Bill Chiu. A symbolic representation of time series, with implications for streaming algorithms. In *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, pages 2–11. ACM, 2003.

[56] Richard O Duda, Peter E Hart, and David G Stork. *Pattern classification*. John Wiley & Sons, 2012.

[57] I. Khan, A. Capozzoli, S. P. Corgnati, and T. Cerquitelli. Fault detection analysis of building energy consumption using data mining techniques. *Energy Procedia*, 42:557–566, 2013.

[58] A. Capozzoli, F. Lauro, and I. Khan. Fault Detection Analysis using Data Mining Techniques for a Cluster of Smart Office Buildings. *Expert Systems with Applications*, 42(9):4324–4338, 2015.

[59] Halldor Janetzko, Florian Stoffel, Sebastian Mittelstädt, and Daniel A. Keim. Anomaly Detection for Visual Analytics of Power Consumption Data. *Computer and Graphics*, 38:27–37, 2014.

[60] Ming C. Hao, Halldor Janetzko, Sebastian Mittelstädt, Walter Hill, Umeshwar Dayal, Daniel A. Keim, Manish Marwah, and Ratnesh K. Sharma. A Visual Analytics Approach for Peak-Preserving Prediction of Large Seasonal Time Series. *Computer Graphics Forum*, 30(3):691–700, 2011.

[61] V. M. Catterson, S. D. J. McArthur, and G. Moss. Online Conditional Anomaly Detection in Multivariate Data for Transformer Monitoring. *IEEE Transactions on Power Delivery*, 25(4):2556–2564, 2010.

[62] M. Gupta, A. B. Sharma, H. Chen, and G. Jiang. Context-aware Time Series Anomaly Detection for Complex Systems. In *WORKSHOP NOTES*, page 14, 2013.

[63] Norden E Huang, Zheng Shen, Steven R Long, Manli C Wu, Hsing H Shih, Quanan Zheng, Nai-Chyuan Yen, Chi Chao Tung, and Henry H Liu. The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, number 1971, pages 903–995. The Royal Society, 1998.

[64] J. Ploennigs, B. Chen, A. Schumann, and N. Brady. Exploiting Generalized Additive Models for Diagnosing Abnormal Energy use in Buildings. In *Pro-*

ceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings, pages 1–8. ACM, 2013.

[65] Daniel B Araya, Katarina Grolinger, Hany F ElYamany, Miriam AM Capretz, and G Bitsuamlak. Collective contextual anomaly detection framework for smart buildings. In *Neural Networks (IJCNN), 2016 International Joint Conference on*, pages 511–518. IEEE, 2016.

[66] Michael A Hayes and Miriam AM Capretz. Contextual anomaly detection framework for big sensor data. *Journal of Big Data*, 2(1):2, 2015.

[67] H. Cheng, P.-N. Tan, C. Potter, and S. A. Klooster. Detection and Characterization of Anomalies in Multivariate Time Series. In *SDM*, volume 9, pages 413–424. SIAM, 2009.

[68] H. Qiu, Y. Liu, N. A. Subrahmanya, and W. Li. Granger Causality for Time-Series Anomaly Detection. In *12th International Conference on Data Mining (ICDM)*, pages 1074–1079. IEEE, 2012.

[69] Dumidu Wijayasekara, Ondrej Linda, Milos Manic, and Craig G Rieger. Mining building energy management system data using fuzzy anomaly detection and linguistic descriptions. *IEEE Trans. Industrial Informatics*, 10(3):1829–1840, 2014.

[70] Manuel Peña, Félix Biscarri, Juan Ignacio Guerrero, Iñigo Monedero, and Carlos León. Rule-based system to detect energy efficiency anomalies in smart buildings, a data mining approach. *Expert Systems with Applications*, 56:242–255, 2016.

[71] X. Li, C. P. Bowers, and T. Schnier. Classification of Energy Consumption in Buildings with Outlier Detection. *IEEE Transactions on Industrial Electronics*, 57(11):3639–3644, 2010.

[72] R. Fontugne, N. Tremblay, P. Borgnat, P. Flandrin, and H. Esaki. Mining anomalous electricity consumption using ensemble empirical mode decompo-

sition. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5238–5242. IEEE, 2013.

[73] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander. LOF: Identifying Density-based Local Outliers. In *ACM SIGMOD conference proceedings*, pages 93–104. ACM, 2000.

[74] D. B. Araya, K. Grolinger, H. F. ElYamany, M. A. M. Capretz, and G. Bitsuamlak. Collective Contextual Anomaly Detection Framework for Smart Buildings. In *IEEE International Joint conference on Neural Networks*, 2016.

[75] R. Agrawal, C. Faloutsos, and A. N. Swami. Efficient similarity search in sequence databases. In *Proceedings of the 4th International Conference on Foundations of Data Organization and Algorithms*, FODO '93, pages 69–84, London, UK, UK, 1993. Springer-Verlag.

[76] X. Wu, R. Kotagiri, and K. B. Korb. *Research and Development in Knowledge Discovery and Data Mining: Second Pacific-Asia Conference, PAKDD'98, Melbourne, Australia, April 15-17, 1998, Proceedings*, volume 1394. Springer, 2006.

[77] John G Proakis and Dimitris G. Manolakis. *Digital signal processing: principles, algorithms, and applications*. Pearson Prentice Hall, 2007.

[78] A. Lazarevic, L. Ertöz, V. Kumar, A. Ozgur, and J. Srivastava. A comparative study of anomaly detection schemes in network intrusion detection. In *SDM*, pages 25–36. SIAM, 2003.

[79] H.-P. Kriegel, P. Kröger, E. Schubert, and A. Zimek. Interpreting and Unifying Outlier Scores. In *SDM*, pages 13–24. SIAM, 2011.

[80] A. Zimek, R. J. G. B. Campello, and J. Sander. Ensembles for Unsupervised Outlier Detection: Challenges and Research Questions. *ACM SIGKDD Explorations Newsletter*, 15(1):11–22, 2014.

[81] S. D.-Haggerty, X. Jiang, G. Tolle, J. Ortiz, and D. Culler. sMAP: A Simple Measurement and Actuation Profile for Physical Information. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, pages 197–210. ACM, 2010.

[82] EIA. "`https://www.eia.gov/tools/faqs/faq.php?id=108&t=1`", 6 2017.

[83] Charu C Aggarwal. *Outlier analysis*. Springer Science & Business Media, 2013.

[84] Y. Hu, W. Murray, Y. Shan, and Australia. *Rlof: R Parallel Implementation of Local Outlier Factor(LOF)*, 2015. R package version 1.1.1.

[85] C. Fan. *HighDimOut: Outlier Detection Algorithms for High-Dimensional Data*, 2015. R package version 1.0.0.

[86] T. Fawcett. An Introduction to ROC Analysis. *Pattern recognition letters*, 27(8):861–874, 2006.

[87] A. Lazarevic and V. Kumar. Feature Bagging for Outlier Detection. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 157–166. ACM, 2005.

[88] Melissa Hart and Richard de Dear. Weather sensitivity in household appliance energy end-use. *Energy and Buildings*, 36(2):161–174, 2004.

[89] Marcelo Espinoza, Caroline Joye, Ronnie Belmans, and Bart De Moor. Short-term load forecasting, profile identification, and customer segmentation: a methodology based on periodic time series. *IEEE Transactions on Power Systems*, 20(3):1622–1630, 2005.

[90] Liping Wang, Paul Mathew, and Xiufeng Pang. Uncertainties in energy consumption introduced by building operations and weather for a medium-size office building. *Energy and Buildings*, 53:152–158, 2012.

[91] Hung-Chia Yang, Sally M Donovan, Scott J Young, Jeffery B Greenblatt, and Louis-Benoit Desroches. Assessment of household appliance surveys collected with amazon mechanical turk. *Energy Efficiency*, 8(6):1063–1075, 2015.

[92] Stephen Makonin, Bradley Ellert, Ivan V Bajić, and Fred Popowich. Electricity, water, and natural gas consumption of a residential house in Canada from 2012 to 2014. *Scientific data*, 3, 2016.

[93] Christian Beckel, Wilhelm Kleiminger, Romano Cicchetti, Thorsten Staake, and Silvia Santini. The ECO data set and the performance of non-intrusive load monitoring algorithms. In *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, pages 80–89. ACM, 2014.

[94] Owen Vallis, Jordan Hochenbaum, and Arun Kejariwal. A novel technique for long-term anomaly detection in the cloud. In *HotCloud*, 2014.

[95] Max Kuhn and Kjell Johnson. *Applied predictive modeling*, volume 26. Springer, 2013.

[96] Leidos Inc. An assessment of interval data and their potential application to residential electricity end- use modeling. Technical report, U. S. Energy Information Administration, February 2015.

[97] Sean Barker, Sandeep Kalra, David Irwin, and Prashant Shenoy. Empirical characterization and modeling of electrical loads in smart homes. In *Green Computing Conference (IGCC), 2013 International*, pages 1–10. IEEE, 2013.

[98] George William Hart. Nonintrusive appliance load monitoring. *Proceedings of the IEEE*, 80(12):1870–1891, 1992.

[99] SMAPE. http://goo.gl/ihYBj5, 12 2017.

[100] Nipun Batra, Amarjeet Singh, and Kamin Whitehouse. If you measure it, can you improve it? exploring the value of disaggregation. In *Proceedings of the*

2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments. ACM, 2015.

[101] Peter Palensky and Dietmar Dietrich. Demand side management: Demand response, intelligent energy systems, and smart loads. *IEEE transactions on industrial informatics*, 7(3):381–388, 2011.

[102] H. Rashid, P. Singh, and K. Ramamritham. Revisiting selection of residential consumers for demand response programs. In *Proceedings of the 4th ACM International Conference on Systems for Energy-Efficient Built Environments*, page 30. ACM, 2017.

[103] Dezhi Hong, Hongning Wang, and Kamin Whitehouse. Clustering-based active learning on sensor type classification in buildings. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 363–372. ACM, 2015.

[104] Anthony Faustine, Nerey Henry Mvungi, Shubi Kaijage, and Kisangiri Michael. A survey on non-intrusive load monitoring methodies and techniques for energy disaggregation problem. *arXiv preprint arXiv:1703.00785*, 2017.

[105] Ahmed Zoha, Alexander Gluhak, Muhammad Ali Imran, and Sutharshan Rajasegarar. Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey. *Sensors*, 12(12):16838–16866, 2012.

[106] Bochao Zhao, Lina Stankovic, and Vladimir Stankovic. On a training-less solution for non-intrusive appliance load monitoring using graph signal processing. *IEEE ACCESS*, 4:1784–1799, 4 2016.

[107] Stephen Makonin, Fred Popowich, Ivan V Bajić, Bob Gill, and Lyn Bartram. Exploiting HMM sparsity to perform online real-time nonintrusive load monitoring. *IEEE Transactions on Smart Grid*, 7(6):2575–2585, 2016.

[108] Mingjun Zhong, Nigel Goddard, and Charles Sutton. Latent bayesian melding for integrating individual and population models. In *Advances in Neural Information Processing Systems*, pages 3618–3626, 2015.

[109] Chao Liu, Adedotun Akintayo, Zhanhong Jiang, Gregor P Henze, and Soumik Sarkar. Multivariate exploration of non-intrusive load monitoring via spatiotemporal pattern network. *Applied Energy*, 211:1106–1122, 2018.

[110] Younghun Kim, Thomas Schmid, Zainul M Charbiwala, and Mani B Srivastava. Viridiscope: design and implementation of a fine grained power monitoring system for homes. In *Proceedings of the 11th international conference on Ubiquitous computing*, pages 245–254, 2009.

[111] Shwetak N Patel, Thomas Robertson, Julie A Kientz, Matthew S Reynolds, and Gregory D Abowd. At the flick of a switch: Detecting and classifying unique electrical events on the residential power line. In *International Conference on Ubiquitous Computing*, pages 271–288. Springer, 2007.

[112] Sidhant Gupta, Matthew S Reynolds, and Shwetak N Patel. Electrisense: single-point sensing using emi for electrical event detection and classification in the home. In *Proceedings of the 12th ACM international conference on Ubiquitous computing*, pages 139–148. ACM, 2010.

[113] Oliver Parson, Siddhartha Ghosh, Mark J Weal, and Alex Rogers. Non-intrusive load monitoring using prior models of general appliance types. In *AAAI*, 2012.

[114] Karim Said Barsim, Roman Streubel, and Bin Yang. An approach for unsupervised non-intrusive load monitoring of residential appliances. In *Proceedings of the 2nd International Workshop on Non-Intrusive Load Monitoring*, 2014.

[115] Men-Shen Tsai and Yu-Hsiu Lin. Modern development of an adaptive non-intrusive appliance load monitoring system in electricity energy conservation. *Applied Energy*, 96:55–73, 2012.

[116] Bo-Jhang Ho, Hsin-Liu Cindy Kao, Nan-Chen Chen, Chuang-Wen You, Hao-Hua Chu, and Ming-Syan Chen. Heatprobe: a thermal-based power meter for accounting disaggregated electricity usage. In *Proceedings of the 13th international conference on Ubiquitous computing*, pages 55–64. ACM, 2011.

[117] Oliver Parson. Overview of the NILM field. `http://blog.oliverparson.co.uk/2015/03/overview-of-nilm-field.html`, 2015.

[118] NILM Community. Companies offering nilm products and services. `http://wiki.nilm.eu/companies.html`, 2017.

[119] Robert Williams Cox. *Minimally intrusive strategies for fault detection and energy monitoring*. PhD thesis, Massachusetts Institute of Technology, 2006.

[120] Peter R Armstrong, Christopher R Laughman, Steven B Leeb, and Leslie K Norford. Detection of rooftop cooling unit faults based on electrical measurements. *HVAC and R Research*, 12(1):151–175, 2006.

[121] Michael R Brambley. A novel, low-cost, reduced-sensor approach for providing smart remote monitoring and diagnostics for packaged air conditioners and heat pumps. Technical report, Pacific Northwest National Lab.(PNNL), Richland, WA (United States), 2009.

[122] Berkeleyheating. http://berkeleyheating.com/blog/why-does-my-air-conditioner-compressor-turn-on-and-off, 05 2017.

[123] J Zico Kolter and Tommi Jaakkola. Approximate inference in additive factorial HMMs with application to energy disaggregation. In *Artificial Intelligence and Statistics*, pages 1472–1482, 2012.

[124] Nipun Batra, Jack Kelly, Oliver Parson, Haimonti Dutta, William Knottenbelt, Alex Rogers, Amarjeet Singh, and Mani Srivastava. NILMTK: an open source toolkit for non-intrusive load monitoring. In *Proceedings of the 5th international conference on Future energy systems*, pages 265–276. ACM, 2014.

[125] Stephen Makonin and Fred Popowich. Nonintrusive load monitoring (NILM) performance evaluation. *Energy Efficiency*, 8(4):809–814, 2015.

[126] Weather Underground. Weather underground API. `https://www.wunderground.com/weather/api/d/docs`, 2016.

[127] Andrea Monacchi, Dominik Egarter, Wilfried Elmenreich, Salvatore D'Alessandro, and Andrea M Tonello. GREEND: an energy consumption dataset of households in italy and austria. In *Smart Grid Communications (SmartGridComm), 2014 IEEE International Conference on*, pages 511–516. IEEE, 2014.

[128] Michael A Nielsen. Neural networks and deep learning, 2015.

[129] Pierluigi Siano. Demand response and smart grids—a survey. *Renewable and Sustainable Energy Reviews*, 30:461–478, 2014.

[130] Kartik Palani, Nabeel Nasir, Vivek Chil Prakash, Amandeep Chugh, Rohit Gupta, and Krithi Ramamritham. Putting smart meters to work: Beyond the usual. In *Proceedings of the 5th international conference on Future energy systems*, pages 237–238. ACM, 2014.

[131] Schneider Electric. EM6400 Series Meter. `https://bit.ly/2NI0A2v`, 2015.

[132] Berkeley.edu. sMAP: the simple measurement and actuation profile. `https://people.eecs.berkeley.edu/~stevedh/smap2/index.html`, 2013.

[133] S. Dawson-Haggerty. Database readingdb. `https://github.com/stevedh/readingdb`, 2016.

[134] Ryan Melfi, Ben Rosenblum, Bruce Nordman, and Ken Christensen. Measuring building occupancy using existing network infrastructure. In *Green Computing Conference and Workshops (IGCC), 2011 International*, pages 1–8. IEEE, 2011.

[135] Bharathan Balaji, Jian Xu, Anthony Nwokafor, Rajesh Gupta, and Yuvraj Agarwal. Sentinel: occupancy based HVAC actuation using existing wifi infrastructure within commercial buildings. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, page 17. ACM, 2013.

[136] Andrea Vaccari, Stephen Samouhos, Leon Glicksman, and Carlo Ratti. MIT Enernet: correlating wifi activity to human occupancy. In *Proceedings of Healthy Buildings*, 2009.

[137] Mohamed M Ouf, Mohamed H Issa, Afaf Azzouz, and Abdul-Manan Sadick. Effectiveness of using wifi technologies to detect and predict building occupancy. *Sustainable Buildings*, 2:7, 2017.

[138] Digvijay Singh and Pushpendra Singh. Enabling ubiquitous applications using existing infrastructure. Master's thesis, IIIT Delhi, 2015.

[139] Joy Aneja and Pushpendra Singh. Analysing space utilization using indoor localization. Master's thesis, IIIT Delhi, 2016.

[140] Garvita Bajaj and Pushpendra Singh. Sensing human activity for assessing participation in evacuation drills. In *Adjunct Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers*, pages 1423–1432. ACM, 2015.

[141] H. Rashid. Weather underground data pull. `https://git.io/fN42n`, 2018.

[142] Elitech. Elitech RC-4HC Temperature and Humidity Data Logger. `https://bit.ly/2mKa2ax`, 2017.

[143] A. Alexander Beaujean. Bayloredpsych: R package for baylor university educational psychology quantitative courses. r package version 0.5, 2012.

[144] Li Wei, Nitin Kumar, Venkata Lolla, Eamonn J. Keogh, Stefano Lonardi, and Chotirat Ratanamahatana. Assumption-free anomaly detection in time

series. In *Proceedings of the 17th International Conference on Scientific and Statistical Database Management*, SSDBM'2005, pages 237–240, 2005.

[145] Shebuti Rayana. Odds library, `http://odds.cs.stonybrook.edu`, 2016.

[146] Anshul Agarwal, Vitobha Munigala, and Krithi Ramamritham. Observability: A principled approach to provisioning sensors in buildings. In *Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments*, BuildSys '16, pages 197–206, 2016.