University of London
Imperial College of Science, Technology and Medicine
Department of Computing

Disaggregation of Domestic Smart Meter Energy Data

Jack (Daniel) Kelly

Submitted in part fulfilment of the requirements for the degree of
Doctor of Philosophy in Computing of the University of London and
the Diploma of Imperial College, April 2017

Abstract

Many countries are rolling out smart electricity meters. A smart meter measures the *aggregate* energy consumption of an entire building. However, appliance-by-appliance energy consumption information may be more valuable than aggregate data for a variety of uses including reducing energy demand and improving load forecasting for the electricity grid. Electricity disaggregation algorithms – the focus of this thesis – estimate appliance-by-appliance electricity demand from aggregate electricity demand.

This thesis has three main goals: 1) to *critically evaluate* the benefits of energy disaggregation; 2) to develop tools to *enable* rigorous disaggregation research; 3) to *advance* the state of the art in disaggregation algorithms.

The first part of this thesis explores whether disaggregated energy feedback helps domestic users to reduce energy consumption; and discusses threats to the NILM. Evidence is collected, summarised and aggregated by means of a critical, systematic review of the literature. Multiple uses for disaggregated data are discussed. Our review finds no robust evidence to support the hypothesis that current forms of disaggregated energy feedback are more effective than aggregate energy feedback at reducing energy consumption in the general population. But the absence of evidence does not necessarily imply the absence of any beneficial effect of disaggregated feedback. The review ends with a discussion of ways in which the effectiveness of disaggregated feedback may be increased and a discussion of opportunities for new research into the effectiveness of disaggregated feedback. We conclude that more social science research into the effects of disaggregated energy feedback is required. This motivates the remainder of the thesis: to enable cost-effective research into the effects of disaggregated feedback, we work towards developing robust NILM algorithms and software.

The second part of this thesis describes three tools and one dataset developed to enable disaggregation research. The first of these tools is a novel, low-cost data collection system, which records appliance-by-appliance electricity demand every six seconds and records the whole-home voltage and current at 16 kHz. This system enabled us to collect the UK's first and only high-frequency (kHz) electricity dataset, the UK Disaggregated Appliance-Level Electricity dataset (UK-DALE). Next, to help the disaggregation community to conduct open, rigorous, repeatable research, we collaborated with other researchers to build the first open-source disaggregation framework, NILMTK. NILMTK has gained significant traction in the community, both in terms of contributed code and in terms of users. The third tool described in this thesis is a metadata schema for disaggregated energy data. This schema was developed to make it easier for researchers to describe their own datasets and to reduce the effort required to import datasets.

The third part of this thesis describes our effort to advance the state of the art in disaggregation algorithms. Three disaggregation approaches based on deep learning are discussed: 1) a form of recurrent neural network called 'long short-term memory' (LSTM); 2) denoising autoencoders; and 3) a neural network which regresses the start time, end time and average power demand of each appliance activation. The disaggregation performance was measured using seven metrics and compared to two 'benchmark' algorithms from NILMTK: combinatorial optimisation and factorial hidden Markov models. To explore how well the algorithms generalise to unseen houses, the performance of the algorithms was measured in two separate scenarios: one using test data from a house not seen during training and a second scenario using test data from houses which were seen during training. All three neural nets achieve better F1 scores (averaged over all five appliances) than either benchmark algorithm. The neural net algorithms also generalise well to unseen houses.

Copyright

Acknowledgements

I would like to thank the following people:

- My supervisor, Professor William Knottenbelt, who has been the perfect supervisor for me. Will gave me the freedom and the trust to explore new and interesting research directions, while providing frequent encouragement and guidance. Will also led by example: he is a shining light of honesty, warmth and sharing; both in the context of research and in the context of teaching. I have been very lucky to have had such a wonderful supervisor.

- My second supervisor, Professor Andrew Davison, for first introducing me to the joys of machine learning when I took his Robotics course during my Computing MSc, and for his insight and encouragement during my PhD.

- My collaborators on NILMTK and other projects, Dr Oliver Parson (who was at the University of Southampton and is now at Centrica Connected Home) and Nipun Batra (at IIIT Delhi). I *thoroughly* enjoyed our collaboration, learnt a lot, and met two of the most friendly and ambitious researchers in the community!

- Professor Mario Bergés (CMU) for his encouragement, generosity and insight.

- Professor Murray Shanahan for allowing me to co-supervise some highly inspiring and exciting MSc projects on the intersection between neuroscience and machine learning.

- Pete Swabey for providing many fascinating (and sceptical) links on smart homes.

- Dr Gareth Jones for his feedback on my draft thesis.

- Dr Carrie Armel for engaging with me in email discussions about one of her papers, Armel et al. 2013.

- Becky Sokoloski and Professor Wesley Schultz for email discussions about Becky's masters thesis, Sokoloski 2015.

- The Computing MSc students who worked on group projects and individual projects that I proposed. Co-supervising MSc projects was one of the most enjoyable and rewarding activities during my PhD and I was very lucky to work with such motivated and smart people; many of whom have gone on to do great things.

- Members of the community who have contributed code and comments to NILMTK and NILM Metadata.
- The EPSRC for funding my PhD DTA.
- Intel for providing a generous Fellowship throughout my PhD.
- The Department of Computing's Support Group (CSG) for their rapid support and unwavering patience. I would especially like to thank Dr Lloyd Kamara, Mr Geoff Bruce and Mr Duncan White.
- Dr Amani El-Kholy, Mrs Ann Halford, Ms Teresa Ng and Mr Hassan Patel for their constant and patient support.
- My two PhD examiners, Professor Duncan Gillies and Dr. Nigel Goddard, for their attention to detail.
- Last but not least, my wonderful wife, Ginnie Kelly, for her love, support, patience and encouragement. I really do love you with all my heart, Ginnie. Thank you.

Dedication

To my two children, Max and Olive.

“One concern I’ve always had is that most people have *no idea* where their energy goes, so any attempt to conserve is like optimizing a program without a profiler.”

Bret Victor 2015

Contents

Abstract i

Copyright iii

Acknowledgements v

1 Introduction 1

1.1 Brief overview of energy disaggregation . . . 1

1.1.1 The many names of 'energy disaggregation' . . . 2

1.1.2 The NILM research community . . . 3

Companies offering NILM products . . . 4

1.2 Objectives . . . 10

1.3 Motivation: proposed benefits of NILM . . . 12

1.3.1 A large reduction in CO_2 emissions is required to maintain a stable climate 12

1.3.2 Demand reduction is the most cost-effective way to reduce CO_2 emissions 14

1.3.3 Smart meters are rolling out in many countries . . . 14

1.3.4 Energy feedback can reduce energy consumption . . . 15

1.3.5 Use-cases for NILM . . . 16

Generating itemised energy bills to help people to save energy . . . 16

Energy literacy . . . 18

Detailed logs of appliance usage . . . 18

"Did I leave something on?" . . . 18

Personalised energy saving recommendations . . . 19

Identify malfunctioning or inefficient appliances . . . 19

Commercial and industrial consumers . . . 19

Help architects to design energy-efficient buildings . . . 20

Allow grid operators to improve predictions of energy demand . . . 21

Enable higher accuracy whole-system energy models 21
Allow utility companies to better segment their users 22
Track changes in energy demand in response to interventions 22
Targeted demand side response 22
Appliance recognition 25
Enable large-scale surveys into energy usage behaviour 25
Apportion energy consumption to individuals or activities 25
Occupancy monitoring 25
Drive consumers to engage with utilities online 26
1.4 Contributions and thesis outline 27
1.4.1 Critical evaluation of the benefits of energy disaggregation 27
1.4.2 Tools to enable rigorous disaggregation research 27
1.4.3 Advancing the state of the art in disaggregation algorithms 28
1.5 Statement of originality 28
1.6 Publications 28
1.6.1 Formatting conventions in this thesis 30

I Critically evaluate the effectiveness of disaggregation 31

2 Does disaggregated electricity feedback reduce domestic electricity consumption? 32

2.1 An introduction to systematic reviews 33
2.2 Methodology 34
2.3 Can disaggregated electricity feedback enable ‘energy enthusiasts’ to save energy? 34
2.4 How much energy would the *whole* population save if given disaggregated data? 37
2.5 Is ‘fine-grained’ disaggregation necessary? 39
2.6 Does aggregate or disaggregated feedback enable greater savings for the whole population? 40
2.7 Suggestions for future research 41
2.8 Conclusions 45

3 Threats and challenges to NILM **47**

3.1 Will ‘smart appliances’ & ‘smart plugs’ make NILM obsolete? 47

3.1.1 Smart appliances 49

Are manufacturers losing interest in energy? 49

3.1.2 Low-cost appliance power monitors (‘smart plugs’) 53

3.1.3 Networking for the smart home 55

Wi-Fi is emerging as dominant for *mains powered* smart appliances. . . . 57

A common API for energy 57

Too much diversity? 59

3.1.4 Conclusions regarding smart homes 60

Concerns over smart meters 62

3.2 Conclusions 63

II Tools to enable disaggregation research **65**

4 The UK domestic appliance-level electricity dataset (UK-DALE) **66**

4.1 Introduction 66

4.2 Methods 68

4.2.1 Individual appliance monitoring 68

4.2.2 Measuring whole-house power demand 71

Calibration 75

Open source implementation 75

4.2.3 Complete metering setup 76

4.2.4 Selecting houses to record 77

4.3 Details of the recorded data 78

4.4 Exploration of the recorded data 81

4.4.1 Correlations with weather 88

4.5 Reception 90

4.6 Acknowledgments for UK-DALE 91

5 An open-source tool kit for non-intrusive load monitoring (NILMTK) 92

5.1 Why develop an open-source NILM toolkit? 92

5.2 Overview of NILMTK 94

5.3 The NILMTK developers 95

5.4 Other open-source tool kits 96

5.5 NILMTK's design and features 96

5.5.1 NILMTK can handle an arbitrary amount of data 97

5.5.2 Classes 98

The `DataStore` class 99

The `TimeFrame` class and the `TimeFrameGroup` class 100

The `Electric` superclass, `ElecMeter` class and `MeterGroup` class 100

The `Appliance` class 101

The `Building` class and the `DataSet` class 101

The `Disaggregator` class 102

5.5.3 Data format 103

5.5.4 Disaggregation 103

5.5.5 Automated unit testing 104

5.6 NILMTK and the community 104

5.7 Conclusion 105

5.7.1 Future work: simplifying NILMTK 105

6 A metadata schema for disaggregated energy data 107

6.1 Introduction 108

6.2 Related work 109

6.3 Design 111

6.3.1 Dataset 111

6.3.2 Building 113

6.3.3 Meters are distinct from appliances 113

6.3.4 ElecMeters and MeterDevices 114

6.3.5 Mains wiring 114

6.3.6 Appliance and ApplianceType 115

6.3.7 Inheritance for ApplianceTypes 115

6.3.8 Appliance categorisation 116

6.3.9 Appliances can contain other appliances 117

6.3.10 Prior knowledge . . . 118
6.3.11 Learnt models of appliances . . . 119
6.4 Limitations and future work . . . 120
6.4.1 Simplification . . . 120
6.5 Usage within the community . . . 121
6.6 Conclusions . . . 121

III Disaggregation algorithms 122

7 Review of NILM algorithms 123
7.1 Disaggregation framed as an optimisation problem . . . 123
7.1.1 Optimisation is computationally intractable . . . 124
7.2 Blind source separation . . . 125
7.3 Extracting steady states from a smart meter signal . . . 126
7.4 Hart's non-intrusive load monitoring algorithm . . . 128
7.4.1 Performance of Hart's NILM algorithm . . . 132
7.4.2 Limitations of Hart's NILM algorithm . . . 134
7.4.3 Extending Hart's NILM algorithm . . . 134
Transients . . . 134
Using additional features . . . 135
7.5 Sparse coding . . . 136
7.6 Hidden Markov models . . . 137
7.6.1 Conditional factorial hidden semi-Markov models . . . 137
7.6.2 Generic models and house-specific models . . . 139
7.6.3 Computationally efficient methods for inference in FHMMs . . . 140
Tuplets of overlapping appliances . . . 140
7.6.4 Approximate inference in additive factorial HMMs . . . 141
7.6.5 Disadvantages of using HMMs to model multi-state appliances . . . 142
7.7 Other approaches . . . 143

8 Manual feature extraction experiments 144
8.1 Spectrogram . . . 146
8.2 Spike histogram . . . 146
8.2.1 Tests on real smart meter data . . . 150
8.3 Discussion . . . 150

9 Deep neural networks for energy disaggregation **152**

9.1 Motivation . . . 153
9.2 Introduction to artificial neural nets . . . 155
9.2.1 Forwards pass . . . 156
9.2.2 Backwards pass . . . 156
9.2.3 Convolutional neural nets . . . 157
9.3 Training data . . . 157
9.3.1 Choice of appliances . . . 159
9.3.2 Extract activations . . . 160
9.3.3 Select windows of real aggregate data . . . 161
9.3.4 Synthetic aggregate data . . . 161
9.3.5 Implementation of data processing . . . 162
9.3.6 Standardisation . . . 162
9.4 Neural network architectures . . . 163
9.4.1 Recurrent neural networks . . . 165
9.4.2 Denoising autoencoders . . . 166
9.4.3 Regress start time, end time & power . . . 168
9.4.4 Neural net implementation . . . 169
9.5 Disaggregation . . . 170
9.6 Results . . . 170
9.7 Conclusions & future work . . . 176
9.7.1 Train on more data . . . 177
9.7.2 Unsupervised pre-training . . . 178
9.7.3 Additional ideas for future work . . . 178

10 Conclusion **180**

10.1 Summary of thesis achievements . . . 180
10.2 Future work . . . 181
10.2.1 A NILM algorithm competition . . . 181
10.2.2 Build a NILM web service . . . 182
10.2.3 Run a large randomised controlled trial . . . 183

A UK-DALE **184**
A.1 Radio frequency (RF) details 184
A.2 Disadvantages of using a CT clamp connected to a wireless transmitter 185
A.3 Calculation of measurement resolution 186
A.4 Known issues 187
A.5 Usage notes 187

B NILMTK **189**
B.1 Dependencies 189

C NILM Metadata **191**
C.1 Implementation 191
C.2 File organisation 191
C.3 Example 192

D Reactive power versus real power **194**

E Copyright permissions **195**

Bibliography **197**

Index **222**

Glossary **223**

List of Tables

1.1 Companies who can disaggregate 'standard' smart meter data. 'Date' column is when the company's NILM product was announced. Table adapted and extended from Parson 2012–2016, with permission. 7

1.2 NILM companies who require their own metering hardware to be installed. 'Date' column is when the company's NILM product was announced. Adapted and extended from Parson 2012–2016, with permission. 8

1.3 Responses to the question *"Do you think accurate information of how to reduce the electricity consumption of appliances will help you reduce your household's electricity bills?"* from 654 subjects. Adapted from Mansouri et al. 1996, p261, with permission 17

2.1 Studies on the effectiveness of disaggregated energy feedback. 38

3.1 Example individual appliance meters (also called 'smart plugs'). US Dollars and Euros have been converted to GB Pounds at conversions rates of \$1:£0.70 and €1:£0.78. Prices are per appliance meter, excluding the wireless base station. . . 54

4.1 Summary statistics for each house. 82

5.1 Summary of dataset results calculated by the diagnostic and statistical functions in NILMTK. Each cell represents the range of values across all households per data set. The three numbers per cell are the minimum, median and maximum values. AMPds, Smart* and iAWE each contain just a single house, hence these rows have a single number per cell. 102

9.1 Number of training activations per house. 159

9.2 Number of testing activations per house. 159

9.3 Houses used for training and testing. 160

9.4 Arguments passed to `get_activations()`. 160

E.1 Copyright permissions table 195

List of Figures

1.1 Number of NILM publications per year. Source: Parson 2015 (with permission). 2

1.2 Screenshot of Smappee's itemised energy bill (from St. John 2015b). Note the large number of disaggregated appliances. The Smappee app claims that there are two irons, one of which is the single biggest energy user. I am sceptical that a single clothes iron would use 62 kWh in one month. The power demand of an iron ranges from 800–2000 kW. But irons do not keep their heaters on constantly. So even at the top end of this range, the iron would need to be used for about two hours *every* day in the month. Which is *possible* but unlikely. 6

1.3 Fossil-fuel emissions estimated to be compatible with a global temperature rise of 2°C above pre-industrial temperatures. Source: Gasser et al. 2015 (with permission). 12

1.4 Past and future changes in global mean sea level. Source: Clark et al. 2016 (with permission). Projections for the next 10 000 years are for four emissions scenarios (1 280; 2 560; 8 840 and 5 120 PgC). Vertical grey bars show the range of long-term sea level rise for each scenario. The images show reconstructions of ice sheets on Greenland (top) and Antarctica (bottom) for today (left) and for the 5 120 PgC emission scenario (right). 13

1.5 GB electricity demand profiles for two typical days: one in summer and one in winter. Source: Gavin 2014. 23

2.1 The desktop PC used to display disaggregated energy data in the trial performed by Dobson & Griffin 1992. 36

2.2 An example of the 'coarse-grained' disaggregation performed by HEA. Source: http://corp.hea.com/how-it-works (with permission). 40

3.1 A touchscreen 'Home Manager' made by Unity Systems and installed in 1990. Unity Systems made Home Managers from 1985 until 1999. It controls sockets, switches, the sprinkler, security, temperature and more. Source: http://imgur.com/a/Jb6jW. 48

3.2 General Electric's Nucleus energy manager. This represents GE's vision of the smart home back in 2010. Source: Dahl 2011. 51

3.3 General Electric's Nucleus iPhone application showing energy usage for a single smart appliance. Source: General Electric 2012. 52

3.4 Smart homes can make life *more* complex. Source: Cate 2016 (with permission). 60

4.1 Two EDF Transmitter Plugs installed in our kitchen. 69

4.2 System diagram for the data collection system. The system has three major components: 1) the data logging PC; 2) the sound card power meter and 3) the 'RFM EDF Ecomanager' which uses a Nanode to communicate over the air with a set of individual appliance monitors (IAMs) and current transformer (CT) sensors. On the left is the circuit diagram for interfacing a sound card to a CT clamp and AC-AC adaptor to measure mains current and voltage, respectively. The circuit was adapted from Robert Wall's work (Wall 2012). Each diode is a 1N5282 (1.3 V forward voltage bias). 70

4.3 An EDF EcoManager Transmitter Plug (left) and a Nanode (right). 71

4.4 The data logging PC is the black case on the bottom. The Nanode is inside the box on the top. The four wires running in the plane of the ground are the RF ground plane. Each wire is 1/4-wavelength. The striped wire running upwards is the 1/4-wavelength antenna. 77

4.5 Power demand for a typical day (Sunday 2014-12-07) in House 1. The thin grey line shows the mains (whole-house) active power demand recorded using our sound card power meter. The stacked and filled coloured blocks show the power demand for the top five appliances (by energy consumption) and the dark blue block shows all the other submeters summed together. The thin white gap between the top of the coloured blocks and the mains plot line represents the power demand not captured by any submeter. 84

4.6 Time periods when meters were recording. The five houses in the dataset are represented by the five panels in this plot. The height of each panel is proportional to the number of meters installed in each house. Each thin row (marked by each y-axis tick mark) represents a meter. Blue areas indicate time periods when a meter was recording. White gaps indicate gaps in the dataset. The data for House 1 continues into 2015 and 2016 but that data has been left off this plot in order to provide enough space to see the detail in the plots for Houses 2-5. . . 85

4.7 Mains electricity data. **a,** 16 kHz sampling of mains voltage and current using our sound card power meter from House 1 on 2014-09-03 21:00:00+01:00. The green line shows the current and the blue line shows the voltage. Panel **b** shows histograms of mains power demand for each house. The five subplots represent the five houses in the dataset. There is some density above 500 watts but this has been cropped from this plot to allow us to see detail in the range between 0 and 500 watts. 85

4.8 Electrical appliance usage in House 1. a, Histograms of daily appliance usage patterns. Panel **b** shows average daily energy consumption of the top-five appliances in House 1. All appliances were ranked by the amount of energy they consumed and the top-five are shown here. All lights were grouped together. The 'remainder' block at the bottom represents the difference between the total mains energy consumption and the sum of the energy consumption of the top five appliances. As such, the top edge of the bar shows the average daily total energy consumption for House 1. 86

4.9 Histograms of appliance power demand from House 1. 87

4.10 Maximum relative measurement error for power measurements across a range of loads. A 'Watts up? PRO' meter was used to record the ground-truth. 88

4.11 Seasonal variation in boiler usage. In June (bottom panel), the boiler usage histogram is dominated by the hot water program which runs twice a day. In February (top panel), considerably more energy is used on space heating than on hot water heating. 89

4.12 Linear regressions showing correlation between appliance activity and weather. R^2 denotes the coefficient of determination, m is the gradient of the regression line and n is the number of data-points used in each regression. Each data-point represents one day. Historical daily averages from Heathrow weather station (20 miles due west of the premises under consideration) were obtained from the UK Met Office under their Educational program. Days for which the appliance usage was zero were ignored because we assume that the house was unoccupied on these days. 90

5.1 NILM papers published per year. Source: Parson 2015 (with permission). 93

5.2 The NILMTK data processing pipeline (Figure created in collaboration with Oliver Parson and Nipun Batra). 95

5.3 NILMTK v0.2 can process an arbitrary quantity of data by loading data from disk in chunks. This figure illustrates the loading of a chunk of aggregate data from disk (top) and then pushing this chunk through a processing pipeline which ends in saving appliance estimates to disk chunk-by-chunk. 97

5.4 UML diagram for NILMTK v0.2. 99

5.5 Histograms of power consumption. The filled grey plots show histograms of normalised power. The thin, grey, semi-transparent lines drawn over the filled plots show histograms of un-normalised power. 100

5.6 Top five appliances in terms of the proportion of the total energy used in a single house (house 1) in each of REDD (USA), iAWE (India) and UK-DALE. 101

5.7 Lost samples per hour from a representative subset of channels in REDD house 1.102

5.8 Predicted power estimates generated by the CO and FHMM algorithms and, for comparison, the ground truth for air conditioner 2 in the iAWE data set. Figure created by Nipun Batra. 103

6.1 UML Class Diagram showing the relationships between classes. A dark black diamond indicates a 'composition' relationship whilst a hollow diamond indicates an 'aggregation' relationship. For example, the relationship between `Dataset` and `Building` is read as '*each `Dataset` contains any number of `Building`s and each `Building` belongs to exactly one `Dataset`*'. We use hollow diamonds to mean that objects of one class *refer* to objects in another class. For example, each `Appliance` object refers to exactly one `ApplianceType`. Instances of the classes in the shaded area on the left are intended to be shipped with each dataset whilst objects of the classes on the right are common to all datasets and are stored within the NILM Metadata project. Some `ApplianceType`s contain `Appliance`s, hence the box representing the `Appliance` class slightly protrudes into the 'common metadata' area on the right. 112

6.2 The illustration on the left shows a cartoon mains wiring diagram for a domestic building. Black lines indicate mains wires. This home has a split-phase mains supply (common in North America, for example). The washing machine draws power across both splits. All other appliances draw power from a single split. The text on the right shows a minimalistic description (using the NILM Metadata schema) of the wiring diagram on the left. 112

7.1 Six washing machine waveforms. *a* - *e* were produced by the same washing machine. 126

7.2 Hart's "steady-state" detection. Taken from G. W. Hart 1992. 127

7.3 Tumble drier signature sampled at 1 Hz 128

7.4 George Hart's 'signature taxonomy'. Source: G. W. Hart 1992 (with permission). 129

7.5 Distinguishing between a heater and a fridge by comparing real and reactive power consumption. The heater is a purely resistive load and hence pulls no reactive power. The refrigerator mostly pulls real power but also pulls some reactive power. These two variables allow us to discriminate between most devices. This diagram is taken from US patent 4858141, filed by George Hart and colleagues from MIT in 1989 (G. W. Hart et al. 1989). 130

7.6 A heat pump compressor waveform. The "edge" is the initial spike at about 70 seconds. The "slope" is the increase from 70 seconds onwards. Taken from Cole & Albicki 1998b. 136

8.1 Analysis of a bread maker (panels *a* - *d*) and a CRT TV (panels *e* - *h*). The forward difference of the signature (denoted by $\Delta_{signature}$) at each time t = *signature(t+1) - signature(t)*. 145

8.2 Simple example illustrating the spectrogram (top panel) which results from a "chirp" (bottom panel) which is a sinusoidal signal whose frequency increases over time. The diagonal line in the spectrogram indicates that the chirp increases in frequency over time. The two panels share the same x-axis (time). 147

8.3 Synthetic test data (bottom panel) analysed using a spectrogram (top panel) and my "spike histogram" algorithm (middle panel). All three panels share the same x-axis (which represents time). During the first half of the synthetic data the amplitude of each rectangular wave remains the same but the frequency increases. During the second half of the synthetic data the frequency remains constant but the amplitude decreases. The pixel colour in the spectrogram represents the amount of energy attributed to the frequency indicated on the y-axis at the time indicated on the x-axis. 148

8.4 Real smart meter data (bottom panel) analysed using a spectrogram (top panel) and my "spike histogram" algorithm (middle panel). All three panels share the same x-axis (which represents time). The middle panel has been manually annotated to illustrate that the spike histogram feature detector can discriminate between the time-domain patterns emitted by the breadmaker, dishwasher, washing machine and TV. 149

9.1 Example washing machine power demand (from UK-DALE House 1). 153

9.2 Example outputs produced by all three neural network architectures for three appliances. Each column shows data for a different appliance. The rows are in three groups (the tall grey rectangles on the far left). The top group shows measured data from House 1. The top row shows the measured aggregate power data from House 1 (the input to the neural nets). The Y-axis scale for the aggregate data is standardised such that its mean is 0 and its standard deviation is 1 across the data set. The Y-axis range for all other subplots is $[0, 1]$. The second row shows the single-appliance power demand (i.e. what the neural nets are trying to estimate). The middle group of rows shows the raw output from each neural network (one pass through each network). The bottom rows show the result of sliding the network over the aggregate data with `STRIDE=16` and overlapping the output. For the 'rectangles' net, the the rectangle height should be the *mean* power demand over the duration of the identified activation. 173

9.3 Disaggregation performance on a house not seen during training. 174

9.4 Disaggregation performance on houses seen during training (the time window used for testing is different to that used for training). 175

A.1 Sniffing the wireless configuration parameters from the Current Cost base station. The image on the left shows a Bus Pirate (the small red circuit board on the far right of the image) connected to a Current Cost base station, ready to 'sniff' configuration data from the serial bus on the Current Cost base station. The image on the right shows a close up of the wireless module on the Current Cost base station. 185

Chapter 1

Introduction

1.1 Brief overview of energy disaggregation

Energy disaggregation (also called non-intrusive load monitoring or NILM) is a computational technique for estimating the power demand of individual appliances from a single meter which measures the combined demand of multiple appliances. One use-case is the production of itemised electricity bills from a single, whole-home smart meter. The ultimate aim might be to help users reduce their energy consumption; or to help operators to manage the grid; or to identify faulty appliances; or to survey appliance usage behaviour.

Research on NILM started with the seminal work of George Hart in the mid-1980s (G. W. Hart 1985; G. W. Hart 1992). Today, in 2016, there is a lot of excitement about energy disaggregation. Since 2010 there has been a dramatic increase in the number of papers published on energy disaggregation algorithms (see Figure 1.1) and since 2013 there have been over 100 papers published each year (Parson 2015). Some authors have gone as far as to suggest that energy disaggregation is the 'Holy Grail' of energy efficiency Armel et al. 2013. Disaggregation is big business: in November 2015 company Bidgely raised $16.6 million USD (Richardson 2015). There are now at least 30 companies who offer disaggregation products and services (see section 1.1.2).

For recent reviews on energy disaggregation, see Jiang et al. 2011; Zeifman & Roth 2011; Zoha et al. 2012; Christensen et al. 2012; Armel et al. 2013; Bonfigli et al. 2015.

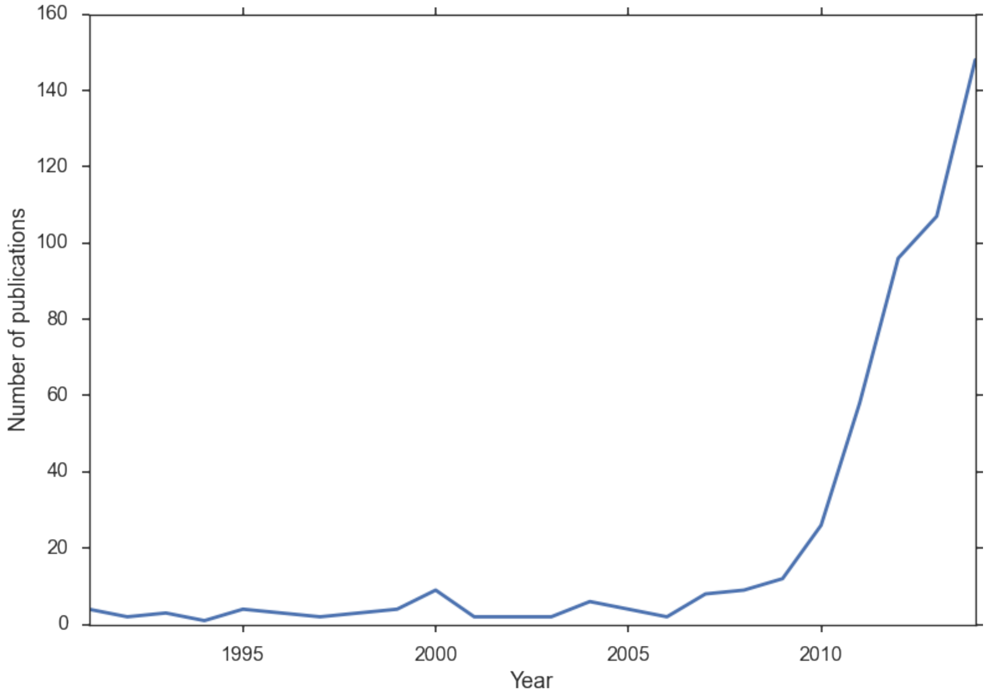


Figure 1.1: Number of NILM publications per year. Source: Parson 2015 (with permission).

1.1.1 The many names of 'energy disaggregation'

Authors use many different names to refer to 'energy disaggregation'. All the names that we have come across in the literature are listed below. The number of papers where the acronym is used in the title or abstract are indicated in brackets (the sample is my literature database[1] of 1,039 papers).

NILM Non-Intrusive Load Monitoring (118)

NIALM Non-Intrusive Appliance Load Monitoring (20)

NALM Nonintrusive Appliance Load Monitoring (4) (this is the acronym used by George Hart in his seminal review paper, G. W. Hart 1992.)

NIALMS Non-Intrusive Appliance Load Monitoring System (3)

ALM Appliance Load Monitoring (1)

C-NILMS Commercial Non-Intrusive Load Monitoring System (1)

In this thesis (and in my papers) I use 'energy disaggregation' and 'NILM' interchangeably.

[1] https://github.com/JackKelly/reference-library

1.1.2 The NILM research community

There have been a total of three International NILM Workshops[2]:

- 2012, May 7th at Carnegie Mellon University, Pittsburgh, PA, USA[3] with about 40 people, organised by Mario Bergés[4] and Zico Kolter[5].
- 2014, June 3rd at University of Texas, Austin, TX, USA[6]; with about 80 people; organised by Mario Bergés and Zico Kolter.
- 2016, May 14-15th at Simon Fraser University, Vancouver, BC, Canada[7]; with about 90 people; organised by Stephen Makonin[8].

There have been two European NILM Workshops[9] (and a third is scheduled):

- 2014, 3rd-4th September at Imperial College London[10]; about 30 people.
- 2015, 8th July at Imperial College London[11]; about 80 people.
- 2016, 18th October in London[12]; organised in association with the Electric Power Research Institute[13] (EPRI) in the US.

I helped to organise the 2014 and 2015 European NILM Workshops.

There is an Energy Disaggregation mailing list[14] which I started in January 2015.

The number of NILM papers published per year is shown in Figure 1.1.

[2] http://nilmworkshop.org
[3] http://www.ices.cmu.edu/psii/nilm/
[4] http://www.marioberges.com
[5] http://www.zicokolter.com
[6] http://nilmworkshop.org/2014
[7] http://nilmworkshop.org/2016
[8] http://makonin.com
[9] http://www.nilm.eu
[10] http://www.oliverparson.co.uk/nilm-2014-london
[11] http://www.nilm.eu/nilm-workshop-2015
[12] http://www.nilm.eu
[13] http://www.epri.com
[14] https://groups.google.com/forum/#!forum/energy-disaggregation

There has been substantial growth in the community over the last few years. This is shown both in the increase in the numbers of people attending NILM workshops (which increased from 40 people to 90 people between the 2012 and 2016 International NILM Workshops), and in the numbers of papers published per year, which is now above 100 papers per year.

Companies offering NILM products

Tables 1.1 and 1.2 list a total of 32 companies who offer NILM products. Table 1.1 shows 13 companies who can disaggregation 'standard' smart meter data. Table 1.2 shows 19 companies who require users to install specialised metering hardware. I will briefly discuss a few companies to give a feel for the world of commercial NILM products as of August 2016.

Bidgely[15] (a word which means "electricity" in Hindi) is based in California and was founded in 2011. Bidgely offers a range of energy analytics - including disaggregation - for smart meter data. A few years ago they sold their services directly to consumers but more recently their business model has shifted towards partnering with utility companies (Bidgely are "utility-facing", to use the industry's term). They state on their website[16] that there are "*5.6 billion people with access to electricity across the world today, and this will only continue to grow. We want to be the engine for every household everywhere to understand their energy usage and make informed decisions that are good for their pocketbooks and good for the planet. We won't stop until we get there.*"

In November 2015, Bidgely raised $16.6 million USD to scale up their 'HomeBeat' platform in USA and Europe (Tweed 2015). Bidgely has about 20 large utility clients (*ibid.*) including RWE, who have customers in the UK, (Tweed 2016) and TXU Energy in Texas (Tweed 2014).

Bidgely have been involved in several academic papers on the effectiveness of disaggregated energy data to help users to save energy so we will revisit Bidgely in chapter 2 when we consider the literature on whether disaggregated energy data helps users to save energy.

Another American disaggregation company, PlotWatt[17], was founded in 2008 and is based in North Carolina. Similarly to Bidgely, PlotWatt began by selling disaggregation services directly

[15] http://www.bidgely.com
[16] https://www.bidgely.com/company
[17] http://plotwatt.com

to domestic users using aggregated data recorded by standard home energy monitors. Over the last few years, PlotWatt have focused on offering disaggregation services and other energy analytics to chain restaurants, using PlotWatt's own meter hardware. Chain restaurants often have similar appliances installed across the chain and so it is possible to train a disaggregation algorithm on one restaurant in the chain and it should be able to disaggregate appliances from the other restaurants in the same chain. PlotWatt's system is installed in some KFC and Dunkin' Donuts restaurants. PlotWatt also offer disaggregation solutions for utilities.

An alternative, user-facing business strategy is exemplified by companies such as Smappee[18], based in Belgium; and Neurio[19], based in Canada. Both require users to install dedicated measurement hardware which costs $239.99 (USD) from Neurio and $249 (USD) from Smappee. Smappee's meter samples at kHz which allows them to, apparently, identify many home appliances (see Figure 1.2 for an example of Smappee's itemised energy bill). A review of Smappee's disaggregation product is provided by Tong 2014.

All the companies discussed above attempt to do *fine-grained* disaggregation by recognising patterns in relatively rapidly-sampled aggregated data. An alternative is to use hourly or monthly meter data to do *coarse-grained* disaggregation. There are several approaches but the general idea is to use prior knowledge of the average itemised energy consumption from *similar* homes; and to find correlations between energy use and weather; and household characteristics. Opower[20] began offering a form of 'coarse' disaggregation in 2014 to utility customers (Spradlin et al. 2014). And Home Energy Analytics[21] (HEA) combine hourly smart meter data with a short user survey to provide a breakdown into five categories including 'heating load' (energy usage which correlates with cold weather) and 'cooling load' (energy usage which correlated with warm weather).

Companies who have experimented with NILM but do not currently offer a NILM product include Intel, Belkin and Wattics.

[18] http://smappee.com
[19] http://neur.io
[20] http://opower.com
[21] http://corp.hea.com

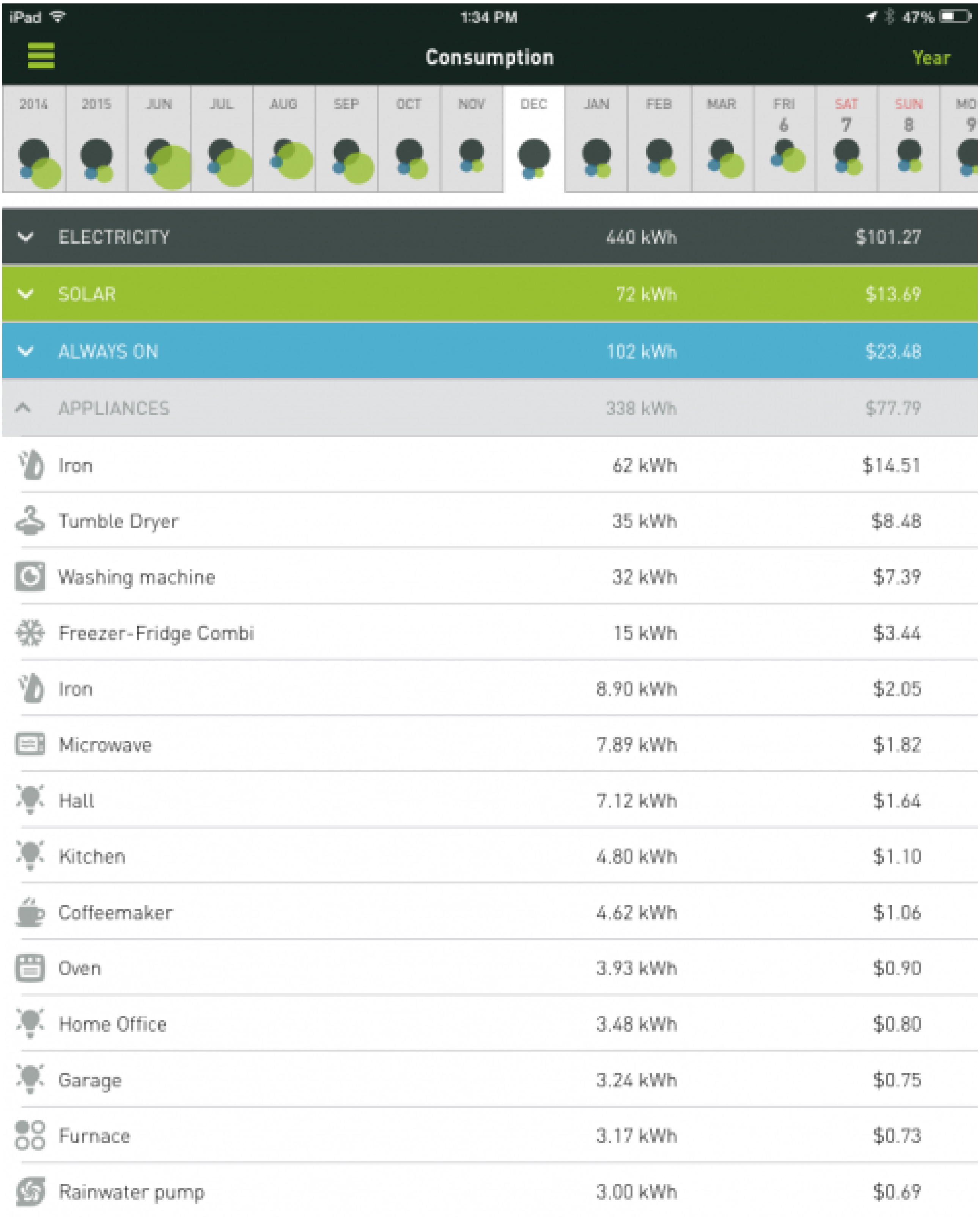


Figure 1.2: Screenshot of Smappee's itemised energy bill (from St. John 2015b). Note the large number of disaggregated appliances. The Smappee app claims that there are two irons, one of which is the single biggest energy user. I am sceptical that a single clothes iron would use 62 kWh in one month. The power demand of an iron ranges from 800–2000 kW. But irons do not keep their heaters on constantly. So even at the top end of this range, the iron would need to be used for about two hours *every* day in the month. Which is *possible* but unlikely.

Table 1.1: Companies who can disaggregate 'standard' smart meter data. 'Date' column is when the company's NILM product was announced. Table adapted and extended from Parson 2012–2016, with permission.

Company	HQ	Date	Inputs	Description
Bidgely	California, USA	2011	Meter-agnostic (medium- & low-freq?)	API for sending data *to* Bidgely.
Ecotagious	Vancouver, Canada	2010?	Hourly smart meter data	Residential and utility-facing
EEme	Pittsburgh, USA	2012	15-minute smart meter data	Ex-CMU students. Suman Giri (who did his PhD with Mario Bergés) was a data scientist here.
Fludia	Paris, France	$\leq$ 2013 ?	1 minute data & household survey	Retrofit 'dumb' meters with 'Fludiameter' which gives 1 minute data. Their disaggregation product is 'Beluso'.
Grid4C	Austin, Texas, USA	2015?	Smart meter data	Provides NILM for Centrica's 'Direct Energy' brand in US/Canada using data from 15M smart meters.
Home Energy Analytics	California, USA	2008	Hourly smart meter data. Short user survey.	Have over 3,500 users. Saved an average of 12.8%.
HOMEpulse	Aix en Provence, France	2013?	Sample period 1-10 seconds?	Formerly WattGo. Elec & gas disag in near real time (minutes). They have a whitepaper: Hochedez et al. 2015.
Onzo	London, UK	2012?	Meter-agnostic (medium- & low-freq?)	Can also infer household occupancy schedules and appliance diagnostics. Utility-facing.
Opower	Virginia, USA	2014	Smart meter data: monthly or faster	Enables utilities to deliver disag data to customers e.g. as a pie chart on their bill. Not doing "disag" as such. Instead use building models which do well on average but possibly not so well for individuals.
PlotWatt	North Carolina, USA	2008	Sample period 1-100 secs. Also have their own meters.	Products for chain restaurants and for utilities. Detects appliance health issues. API for pushing data *to* PlotWatt.
Powerly	Michigan, USA	2014	The "Powerley Energy Bridge (PEB)" - a smart meter (ZigBee SEP) to WiFi bridge	Offers utility-branded energy management platform. Demand response. Appliance health monitoring. Real-time. PEB talks ZigBee, Z-Wave, BT & Thread

Continued on next page

Table 1.1 – continued from previous page

Company	HQ	Date	Inputs	Description
Silver Spring	California, USA	2015	Smart meter data	Silver Spring acquired Detectent in Jan 2015 and inherited their NILM technology. Utility-facing.
Watt Is	Portugal	2012?	Smart meter data	Lots of focus on actionable recommendations

Table 1.2: NILM companies who require their own metering hardware to be installed. 'Date' column is when the company's NILM product was announced. Adapted and extended from Parson 2012–2016, with permission.

Company	HQ	Date	Inputs	Description
AlertMe	Cambridge, UK	2012?	CT at 1 Hz	British Gas acquired AlertMe in 2015.
Ecoisme	Krakow, Poland	2015	CT clamp, V&I, high freq	Personalised tips. IoT integration. Detect faulty appliances. Funded on IndieGoGo. Open API coming.
Enetics SPEED	New York, USA	1996	2 CT clamps or Form 2S socket. High freq V&I	Enetics' disag product is called "SPEED". Enetics have been doing research into NILM since 1993.
Green Running	London, UK	2006?	kHz V&I	Have their own hardware and are rapidly expanding.
Informetis	Japan & Cambridge, UK	2013	High freq?	"Near real time" disag. Sony spinoff. Uses techniques developed for AIBO dog. Zoubin Ghahramani is their tech advisor. Tokyo Elec. Power Co. trialled the system in 300 homes in March 2015.
Ipsum Energy	Netherlands	2011?	Their "Coded Power" system in the meter box	Very few details on their website
LoadIQ	Nevada, USA	2011	Their 'EI.X' meter measures V, I, PF etc. High freq?	Focussed on disag for commercial & industrial customers.
Navetas	Oxford, UK	?	8 kHz	When I checked in August 2016, their website no longer mentions NILM, although they used to provide NILM products.

Continued on next page

Table 1.2 – continued from previous page

Company	HQ	Date	Inputs	Description
Neurio	Vancouver, Canada	2013	2 CT clamps. Measures V, I, active power & PF at 1 Hz	Formerly 'Energy Aware'. Kickstarter funded. RESTful API for appliance switch events. Real-time notifications. Detection limited to appliances >400 W. Disag product is $250 USD. Normative comparisons.
Powersavvy	Castlebar, Ireland	2009?	Their own meter	Products for businesses and households.
Sense	MA, USA	2015	2 CT clamps. Measures V & I at >1 Hz	Appliance health early warning. Consumer-facing.
Smappee	Kortrijk, Belgium	2014	CT clamp. Khz.	Real-time. Disag system costs $250. Optional "Comfort Plugs" to switch appliances. IoT integration using IFTTT. Disaggregates many appliances (if users interacts).
SMART Impulse	Paris, France	2008?	Their own meter. V&I. High freq?	For commercial buildings.
smartB	Berlin, Germany	?	4 kHz V&I and 50 Hz PQ data	SmartB are the energy disaggregation arm of Yetu. Personalised recommendations
Verdigris	California, USA	2012	8 kHz circuit-level metering using CT clamps	Monitor large buildings. Real-time fault detection.
verlitics	Oregon, USA	2012?	Their own meter	Products for businesses and households. Formerly 'Emme'.
Wattseeker	Nice, France	?	Their own meter with CT clamps. High freq?	Their NILM product is 'LYNX'. Claims that each CT clamp can disag up to 12 appliances with an accuracy of ±2%. For businesses.
Watty	Stockholm, Sweden	?	?	Young startup. Few details on website.
You Know Watt?	Brussels, Belgium	2013	Requires electrician. V&I at 1.6 kHz	"The Virtual Submetering Company". Personalised tips. Normative comparison.

1.2 Objectives

My ultimate aim is to help to reduce energy demand globally by providing a disaggregation system. The 'disaggregation scenario' that I have had in mind throughout my research is:

- Whole-house aggregate power demand data would be collected from a smart meter via the home area network.
- Users may have to buy a consumer access device (CAD) to transfer the 10-second data from the smart meter to the internet for disaggregation in the cloud.
- Users should not have to fill out a survey in order to start using the disaggregation service.
- Users should not have to train the disaggregation system. Instead the disaggregation system should be trained by the developer(s) across enough training examples to allow the disaggregation system to generalise to unseen instances of learnt appliance types.
- Whole-house aggregate power demand will be displayed on an in-home display (IHD).
- Users can view disaggregated data on a website or smart phone application. Summaries of disaggregated energy consumption will be provided on printed energy bills and/or regular but infrequent emails.
- Disaggregated data will update as quickly as possible (ideally as soon as new aggregate data arrives: i.e. once every 10 seconds). But 'real-time' disaggregation is hard because the algorithm has to recognise *incomplete* appliance signatures. Hence it may not be possible to provide users will 'real-time' disaggregation. Users can also view energy usage aggregated over various time intervals (e.g. daily, yearly, the current billing cycle).

There are two sets of users that we need to distinguish:

The general public: In order to achieve large energy savings, the majority of homeowners in the country need to reduce their energy consumption when given disaggregated data. They are unlikely to want to spend additional money. As such, the majority of users will not have dedicated displays for disaggregated data (because these would cost too much).

'Energy enthusiasts': These users may be a tiny proportion of the general population but may achieve large energy savings. These users may purchase dedicated displays for disaggregated data. Especially enthusiastic users may have one dedicated display per large appliance so they can conveniently see appliance energy consumption at the point-of-use.

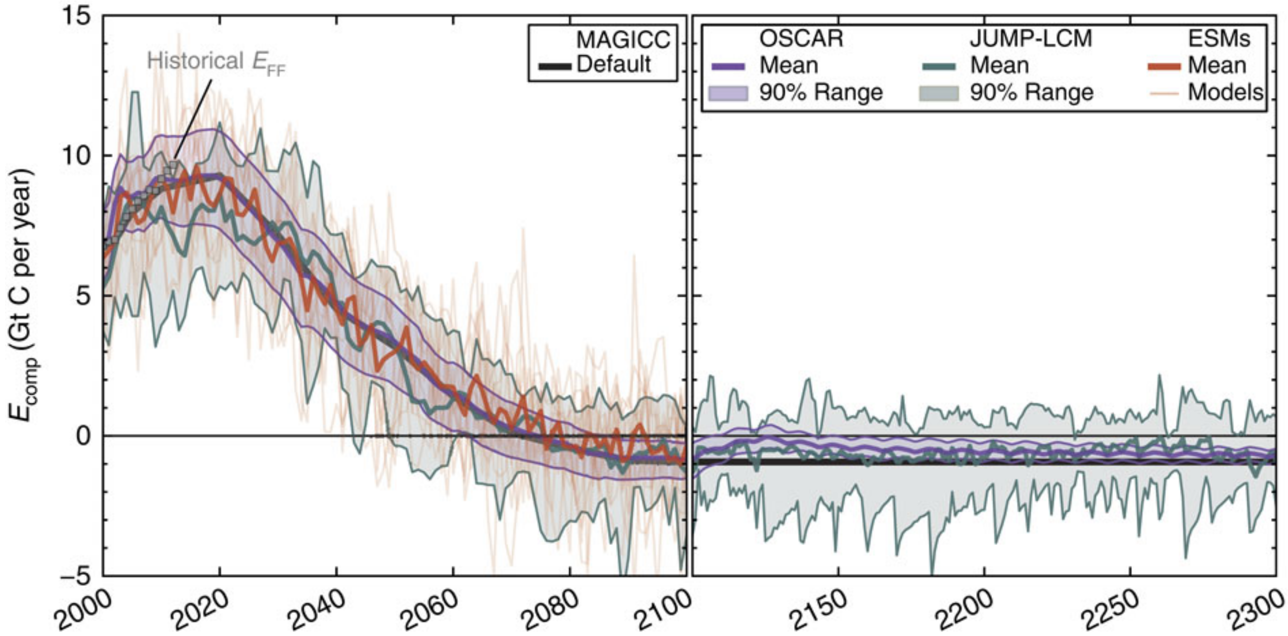


Figure 1.3: Fossil-fuel emissions estimated to be compatible with a global temperature rise of 2°C above pre-industrial temperatures. Source: Gasser et al. 2015 (with permission).

1.3 Motivation: proposed benefits of NILM

In this section, we will step through the arguments for working on NILM.

1.3.1 A large reduction in CO_2 emissions is required to maintain a stable climate

At COP 21 in Paris[22] in December 2015, the countries attending the United Nations Framework Convention on Climate Change reached an agreement to tackle climate change. A key result of this agreement was to set a goal of limiting the warming of the earth's surface to no more than 2°C above pre-industrial levels by the year 2100. The countries also agreed to "pursue efforts to" limit global warming to 1.5°C.

Limiting global warming to 1.5°C by 2100 is a highly ambitious task: Wagner et al. 2016 show that we need to reduce global CO_2 emissions by 50% by 2020 if we are to avoid locking the climate in to a 1.5°C temperature rise. A 2°C limit is still extremely challenging. Figure 1.3 shows the emissions trajectories compatible with a 2°C limit.

[22] http://unfccc.int/paris_agreement/items/9485.php

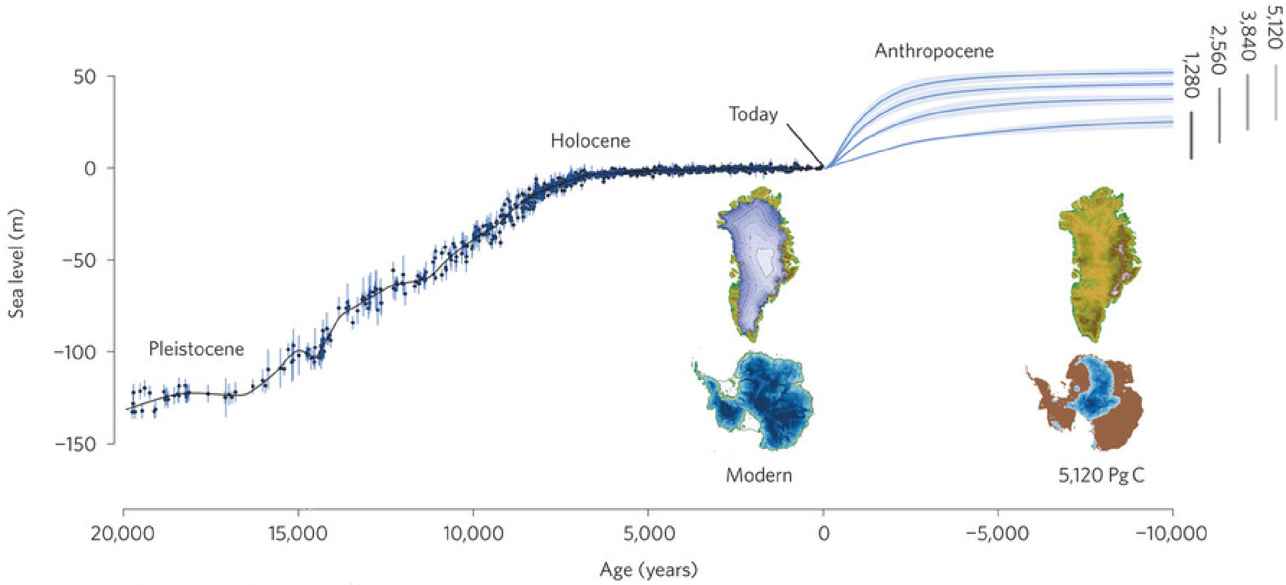


Figure 1.4: Past and future changes in global mean sea level. Source: Clark et al. 2016 (with permission). Projections for the next 10 000 years are for four emissions scenarios (1 280; 2 560; 8 840 and 5 120 PgC). Vertical grey bars show the range of long-term sea level rise for each scenario. The images show reconstructions of ice sheets on Greenland (top) and Antarctica (bottom) for today (left) and for the 5 120 PgC emission scenario (right).

Much of discourse about climate change only considers consequences up until the year 2100. But climate change, driven by decisions that we make in the next years and decades, is likely to persist for tens of thousands of years (Clark et al. 2016). Figure 1.4 shows the global mean sea level from 20 000 years ago to 10 000 years into the future. There are four projections into the future. These projections are for different emissions scenarios. The worst-case scenario in the paper projects that sea level will rise to almost 50 meters above current levels within a few thousand years. The worst-case scenario in this figure is not even as bad as it *could* get because the scenario considers the release of 5 120 petagrams of carbon (PgC) but current attainable carbon reserves are estimated to be between ~9 500 and ~15 700 PgC.

There is also a strong mandate to pursue ambitious climate change mitigation strategies. A 2016 YouGov poll (Dahlgreen 2016) showed that climate change is considered by the world's population to be the third most serious issue threatening the world.

1.3.2 Demand reduction is the most cost-effective way to reduce CO_2 emissions

There are two broad ways to reduce CO_2 emissions: 1) use *renewable* energy sources such as wind and solar which produce little or zero CO_2 and 2) reduce demand for energy. Both of these approaches are required to mitigate climate change. Building new energy generation capacity is expensive, time-consuming and can be politically fraught. Reducing energy demand, on the other hand, is seen by many as more cost-effective, faster and a politically favourable way to reduce CO_2 emissions. Reducing energy demand has the added benefits of reducing energy bills and increasing energy security as soon as the efficiency measures are implemented.

The International Energy Agency (IEA) is enthusiastic about energy efficiency. Fatih Birol, the IEA's Executive Director, says:

> "*Mobilising energy efficiency is an urgent priority. To transition to the sustainable energy system of the future, we need to decouple economic growth from greenhouse gas (GHG) emissions. Energy efficiency is the most important "arrow in the quiver" to achieve this. For its part, the International Energy Agency (IEA) is pursuing a number of strategies to improve energy efficiency both among its member governments and with partner countries.*" - International Energy Agency 2015, page 3

In the UK in 2015, electricity generation was responsible for the emission of 136 million tonnes of CO_2-equivalent ($MtCO_2e$), which is 34% of the UK's total CO_2 emissions (DECC 2016, page 9). This supplied a total of 336 terawatt hours (TWh). Hence the average CO_2 intensity of the UK's electricity supply system in 2015 was 405 grams of CO_2 per kWh.

1.3.3 Smart meters are rolling out in many countries

By the end of 2020, there should be about 830 million electric smart meters installed worldwide (Telefónica 2014, page 9). Of these, 438 million will be in China, 132 million will be in the United States and 27 million will be in the UK.

The UK's smart meters will confirm to the second version of the government's Smart Meter Equipment Technology Specification (SMETS2) (DECC 2014). SMETS2 meters will have two communications interfaces: a ZigBee home area network (HAN) interface which will deliver readings of whole-home active power once every ten seconds. This will be displayed on an in home display (IHD). Users can use a consumer access device (CAD) to collect their smart meter data and send it to the cloud. SMETS2 meters will also have a wide area network (WAN) for sending data to utility companies once every half an hour.

Research conducted in the UK for Smart Energy GB [23] by independent research company Populous [24] surveyed people in the UK with a smart meter in 2014 and found that 84% of people with a smart meter would recommend one to others and that 79% of people with a smart meter had taken steps to reduce their energy consumption (Populus 2015).

Recent reviews of smart meter deployments include Bhatt et al. 2014; Xenias et al. 2015; Sharma & Saini 2015; Colak et al. 2015; Sataøen et al. 2015; Tricoire 2015; Bertoldo et al. 2015 and Naus et al. 2015.

1.3.4 Energy feedback can reduce energy consumption

There is good reason to believe that *aggregate* energy feedback (such as that provided by an in home display) can reduce energy consumption by about 3% (Davis et al. 2013). Hence smart meters have a role to play in mitigating against climate change.

We will present a critical, systematic review on the question of whether *disaggregated* energy bills can reduce energy consumption in chapter 2.

[23] www.smartenergygb.org
[24] www.populus.co.uk

1.3.5 Use-cases for NILM

In this section, we will consider some of the main use-cases for energy disaggregation.

Generating itemised energy bills to help people to save energy

Perhaps the most commonly cited use-case for NILM is to produce itemised, appliance-by-appliance energy bills from a single electricity meter.

In this section, we will briefly describe the *theoretical* support for the idea that disaggregated energy feedback might help people to use less energy. We will discuss the *empirical* evidence for whether disaggregated energy feedback *actually* helps people to save energy in chapter 2.

At least since 1978, researchers have hypothesised that disaggregated energy data could help people to save energy. Socolow 1977–1978, page 212 states:

> *"[Energy] savings were anticipated by our psychologists, who look on energy conservation as a problem in learning new skills... The analog of the future meter is the sportscar's dashboard, giving consumption (in money units?) separately for the major appliances, with buttons to reset some meters to zero. The future bill makes comparisons with one's own past performance and with the current performance of one's peers."*

Furthermore, people, in general, *want* itemised energy bills. Multiple studies report that users express a preference for (free) disaggregated energy data (Mansouri et al. 1996; Wilhite et al. 1999; Darby 2006; W. Anderson & White 2009; Fitzpatrick & Smith 2009; Ståhlberg 2010; Karjalainen 2011; Vassileva et al. 2012; Snow et al. 2013; Krishnamurti et al. 2013; Rettie et al. 2014).

Let us illustrate this trend with a specific example from the literature: Wilhite et al. 1999 estimated disaggregated usage using aggregate data and a short questionnaire. The authors sent a survey and disaggregated electricity use presented as a pie chart to 2 000 households in Norway. 95% of respondents reported that they would be interested in receiving disaggregated information on their electricity bill in the future; only 7% of respondents reported that the pie

Table 1.3: Responses to the question *"Do you think accurate information of how to reduce the electricity consumption of appliances will help you reduce your household's electricity bills?"* from 654 subjects. Adapted from Mansouri et al. 1996, p261, with permission

'A great deal'	'Moderately'	'Not very much'	'Not at all'
35.0%	47.9%	14.7%	2.4%

chart was difficult to understand; 84% reported that the pie chart gave them a better understanding of their home's energy use; and 84% reported that the pie chart provides information not available through other sources. However, only 20% of respondents reported that they would like to receive disaggregated energy information via the internet (instead of on paper bills). But remember that this study was done in 1999 when the world-wide-web was less than ten years old!

When asked *"Do you think accurate information of how to reduce the electricity consumption of appliances will help you reduce your household's electricity bills?"* 82.9% of people responded either 'a great deal' or 'moderately' (Mansouri et al. 1996 - also see Table 1.3). Mansouri et al. 1996 also found that 91% of people 'regularly' or 'sometimes' make a conscious effort to save electricity.

But expressing a *preference* does not automatically imply that users will *benefit* from disaggregated data.

There is also evidence that many people do not know how much energy each appliance uses: that people have an "information deficit".

Studies on residential energy users show that the vast majority are poor at estimating either the consumption of individual devices or total aggregate consumption. Residents often underestimate the energy used by heating and overestimate the consumption of perceptually salient devices like lights and televisions (Kempton & Montgomery 1982). Residents' failure to correctly estimate energy consumption leads to higher total consumption.

How significant is occupant behaviour in determining total energy usage? Energy use can differ by two or three times among identical homes with similar appliances occupied by people from similar demographics (Socolow 1977–1978; Winett & Neale 1979; Seryak & Kissock 2003).

These large differences in energy consumption are attributed to differences in *behaviour*. If the home provided better feedback about which devices used the most energy then users could tweak their behaviour to make more efficient use of appliances.

The implication of the “information deficit” model is that if we can “fix” this information deficit then people will be empowered to save energy. For example, Wilhite & Ling 1995 write:

> We can say with certainty that the following relationship holds true: Increased feedback on consumption $\rightarrow$ Decrease in consumption.

This “information deficit” theory is now viewed with scepticism by some social scientists. See, for example Hargreaves et al. 2010; Strengers 2013; Hargreaves et al. 2013.

Energy literacy

Saving energy is often highlighted as the main purpose of providing people with disaggregate energy bills. But perhaps increasing *energy literacy*, in and of itself, is a worthy outcome of providing disaggregated energy feedback.

For example, T. Schwartz et al. 2013 conducted a field trial and demonstrated that the participant’s energy literacy was increased when they were given a home energy management system which recorded appliance level energy consumption using smart plugs.

Detailed logs of appliance usage

If users can be provided with logs of exactly when appliances are used then these logs could be used to check for faulty automatic controls.

“Did I leave something on?”

Imagine a phone application which detects when you are about to leave the house and then checks to make sure the clothes iron is off before you leave. This requires real-time disaggregation of small appliances which is hard to compute. Nevertheless, Bidgely claim to offer a service which can tell users which appliances are switched on, in real time (Bidgely 2015).

Personalised energy saving recommendations

Disaggregated data might enable more accurate, personalised recommendations for saving energy. These recommendations might come from human advisers or from automated recommender systems. For example, Fischer et al. 2013 provided automated advice for changing tariffs; and Makriyiannis et al. 2014; Makriyiannis et al. 2016 provided automatic recommendations for changing energy suppliers.

Identify malfunctioning or inefficient appliances

If disaggregated energy data is sufficiently accurate then it may be possible to identify broken appliances. For example, identifying a fridge which defrosts more frequently than normal might suggest that the fridge seal is damaged and should be replaced. However, Batra et al. 2016b found that the NILM algorithms they tested had insufficient accuracy to detect faulty fridges.

Additionally, appliances such as digital video recorders, printers etc. which fail to fall asleep appropriately could also be identified so their settings can be changed. Martin & Poll 2014 found that a 39% reduction in energy use could be possible on the multi-function devices they studied just by changing the time-to-sleep setting. And H.-H. Chang et al. 2015 worked on identifying ageing appliances using NILM.

Commercial and industrial consumers

Applying NILM to commercial and industrial buildings is an attractive proposition: commercial organisations tend to be more economically rational than domestic users and so might be more likely to want to reduce expenditure on energy consumption.

However, commercial NILM is hard. There might be many instances of the same type of appliance (for example, Imperial's Department of Computing must have hundreds if not thousands of computers!) There are also likely to be many more simultaneous switching events in a commercial building (Holmegaard & Kjærgaard 2015). A further problem is a lack of public data from commercial buildings (there is only one public dataset from a commercial building that we are aware of: COMBED[25] (Batra et al. 2014b)).

[25] http://combed.github.io

Despite these challenges, companies such as Verdigris[26] claim to be able to provide NILM for commercial buildings (M. Chang 2015).

For a recent review of NILM applied to commercial settings, see Liu & Chen 2014.

Help architects to design energy-efficient buildings

Whole building energy simulation (BES) software attempts to predict the energy use for an occupied building. These predictions are typically categorised by end-use (cooling, lighting, heating, etc.). An example is the free, open-source EnergyPlus[27] software developed by the US Department of Energy. Whole building energy simulation can be used in at least four contexts:

Building design: Estimate the energy performance of a proposed building to confirm whether it will comply with energy-related building regulations and to predict energy costs.

Explore alternative energy efficiency measures: Either during the design phase of a proposed building, or during the planning phase for an energy retrofit for an existing building, designers can compare the cost-effectiveness of alternative energy conservation measures.

Identify inefficiencies in a commissioned building: If a building uses more energy than it was predicted to use then comparing the actual, disaggregated usage with the predicted usage may identify specific actions to reduce actual energy consumption. For example, Yan et al. 2012 demonstrate a simplified building energy performance assessment method using energy disaggregation and energy performance analysis.

Model Predictive Control (MPC): Build a forward model of the building and use this, combined with real-time measurements from the building, to inform a controller. This controller also has some optimisation objective (e.g. to minimise energy consumption). The controller aims to find the optimal set of commands for systems within the building in order to satisfy its optimisation objective. For example, Hu & Karava 2014 developed a linear time-variant state-space model predictive controller which found sequences of binary (open/close) commands for the motorised windows. Mahendra et al. 2015 developed a 'reactive' energy management system which responds to unplanned events in order to

[26] http://verdigris.co

[27] https://energyplus.net

maintain the occupant's comfort and energy requirements. Mehendra et al. did not use energy disaggregation but disaggregated data would surely help to inform their controller.

Whole building energy simulation models for commercial buildings are routinely calibrated with disaggregated energy data available from the building management system present in many commercial buildings (Reddy et al. 2007; Bertagnolio 2012; Coakley et al. 2012). For a recent review of building model calibration see Royapoor & Roskilly 2015.

A well-documented issue with BES models is there is often a large gap between the *predicted* energy performance and the *actual* performance. Energy disaggregation could help to pinpoint the appliances or usage behaviours which contribute to the 'performance gap' and hence can help building owners to improve energy efficiency of existing buildings and can help architects to build better predictive models for future design projects.

If a building already has a building management system (BMS) then *some* disaggregated energy data is probably already available on the BMS (measured by sub-meters connected to the BMS). Even so, in a review of methods to match building energy simulation models to measured data, Coakley et al. 2014 cite energy disaggregation as a way to help align model with measurement even in commercial buildings with a BMS. Furthermore, *domestic* buildings rarely have a BMS. Instead, domestic buildings only have a single whole-house meter. Hence disaggregation could be used to identify the energy consumed by individual appliances in order to help to identify why a domestic building is not performing as predicted and to help in the future to inform the modelling of proposed domestic buildings.

Allow grid operators to improve predictions of energy demand

In order to manage the electricity grid, the operator needs to be able to predict demand because large generators take time to react. Disaggregation may help the electricity grid to better predict demand. For a recent review of modelling residential electricity demand, see Torriti 2014.

Enable higher accuracy whole-system energy models

Models of the entire energy system of a country or sub-national area are used for a variety of tasks including planning new electricity infrastructure. Bottom-up physical models of buildings currently tend to use aggregated 'typical home' consumption patterns. This aggregation may provide sufficiently accurate models at *national* level but may be inaccurate at sub-national level (Cheng & Steemers 2011; Natarajan et al. 2011). As such, disaggregated energy data might help to inform building physics models. The International Energy Agency recently wrote:

> "The Energy Information Administration (EIA) is investigating the potential benefits of incorporating interval electricity data into its residential energy end use models. This includes interval smart meter and submeter data from utility assets and systems. It is expected that these data will play a significant role in informing residential energy efficiency policies in the future. Therefore, a long-term strategy for improving the RECS end-use models will not be complete without an investigation of the current state of affairs of submeter data, including their potential for use in the context of residential building energy modeling." - U.S. Energy Information Administration (EIA) 2015

See Kavgic et al. 2010 for a review of bottom-up physics modelling of housing stock.

Allow utility companies to better segment their users

Utility companies segment their users in a variety of ways. Disaggregated energy data may allow for more fine-grained segmentation.

Track changes in energy demand in response to interventions

When users make a change to their energy system, such as swapping incandescent lights for LED lights, it may be useful to be able to measure and attribute a specific energy saving to that intervention. NILM could help to track changes and attribute changes in total energy demand to specific interventions.

Targeted demand side response

Demand side response (DSR) involves electrical loads (such as fridges or aluminium smelters) modifying their consumption behaviour in response to a signal from the electricity grid (Strbac

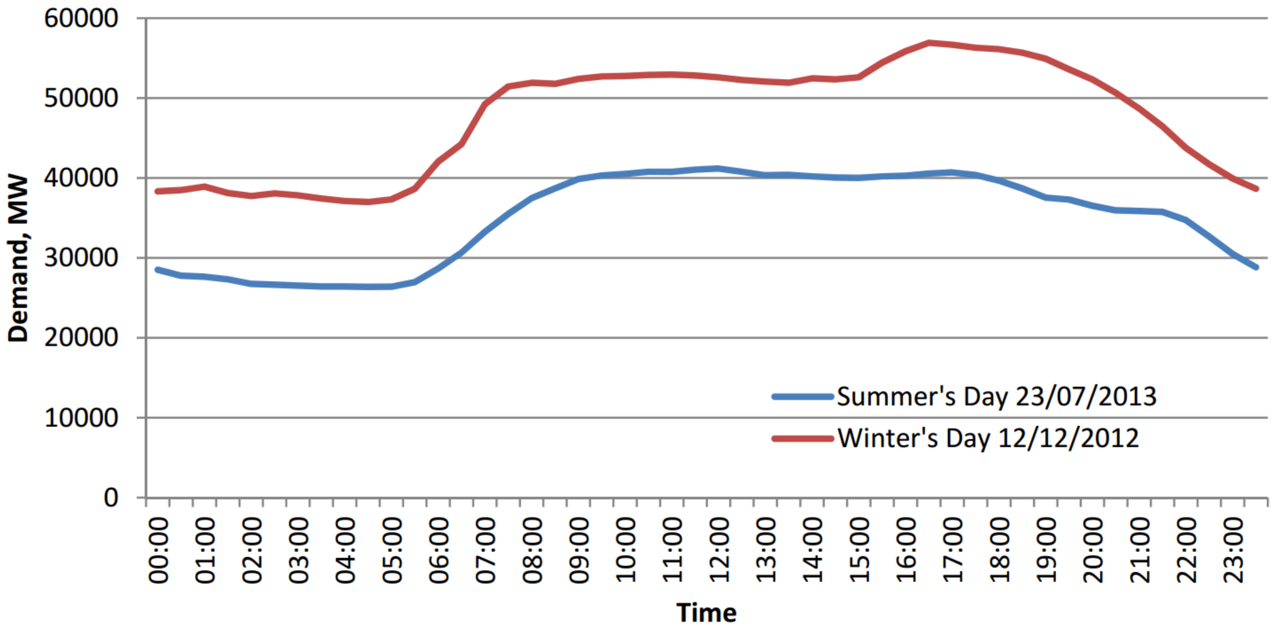


Figure 1.5: GB electricity demand profiles for two typical days: one in summer and one in winter. Source: Gavin 2014.

2008). Devices such as fridges and aluminium smelters have significant "thermal mass" and so they maintain a temperature adequate for their continued operation for a short while after they turn off. Hence these loads can safely turn off for short times without affecting their performance. There are two reasons why demand side response might be useful.

Firstly, each country's electricity demand changes over the course of the day (see Figure 1.5). In winter, Britain's electricity demand peaks at around 55 GW at 5pm and drops to under 40 GW overnight. The country needs to pay for and build sufficient generation and transmission capacity to satisfy the *peak* demand; even though the *average* demand is significantly lower than the peak. This results in an energy system which is more expensive to run than if demand was flat during the day. Demand side response can help to shift demand away from the peak, hence reducing the total cost of the energy system, even though the average daily consumption remains the same. A recent review of the costs and benefits of DSR in the UK found that "the economic care for [demand response] in UK markets in positive" (Bradley et al. 2013).

The second benefit of DSR is that it can help to reduce the grid's *carbon intensity* (i.e. reduce amount of CO_2 emitted per kWh of electricity generated). Renewable generation sources such as wind, solar and tidal do not generate electricity continuously; and electricity storage is expensive and of very limited capacity. If we cannot control the *supply* then we must control the *demand* to match the supply, which is what DSR enables us to do. For example, Harris et al.

2015 performed a modelling study and found that greenhouse gas emissions could be reduced by 21-35% through a “best-case” application of their “Locational Marginal Price Emissions Estimation Method (LEEM)” to shift demand.

How could NILM help with DSR? One answer is that NILM could be used to validate “behavioural DSR” where utility companies send messages to the human *users* of appliances to ask them to turn down their appliances (e.g. “please turn off your air conditioning unit for half an hour to help smooth out the current peak in demand”).

Even when utility companies have nominally automatic control over appliances, that control can fail. So, in this scenario, it can be beneficial to use NILM to validate that appliances have turned off (Witherden et al. 2013).

Furthermore, Yilmaz et al. 2015 built a simulator of appliance usage. This simulator was based on information such as switch-on times taken from real power demand data from appliances. The authors built a model to estimate the demand response potential. This sort of simulation could be informed by NILM, rather than having to go to the expense of installing individual meters on each appliance. For a recent review of simulations of household appliances with DSR enabled, see Ozoh & Apperley 2015.

There is evidence to suggest that users respond favourably to automated energy management (Buchanan et al. 2016).

Energy disaggregation can also help to identify which loads are flexible enough for demand side response Su et al. 2015.

The European Commission ran a project in 2009 called “Smart Domestic Appliances in Sustainable Energy Systems (Smart-A)” (Intelligent Energy Europe 2009) which looked at ways to match domestic electricity demand to intermittent renewable supply by modifying the run times of domestic appliances. The report was based on a literature survey to compile data on the usage patterns of domestic appliances. In the future, this appliance usage data could be gathered from a much wider population by using NILM.

Appliance recognition

A task which is related to NILM but distinct from it is appliance recognition. Here the task is that we have data from a meter which measures a single appliance but we do not know the identity of that appliance. An appliance recognition system will attempt to identify the appliance just from its power demand profile. For examples of this type of work, see Antonio Ridi et al. 2013 and S. Barker et al. 2014.

Enable large-scale surveys into energy usage behaviour

Large-scale surveys into domestic energy behaviour are expensive to conduct if every home must have multiple sub-meters installed. For example, collecting the HES dataset (Zimmermann et al. 2012) cost UK taxpayers £850 000. Hence disaggregation could help to substantially reduce the financial cost of such studies (Cambridge Architectural Research Ltd. & Loughborough University 2013).

Apportion energy consumption to individuals or activities

"Conventional NILM" attempts to apportion energy consumption to appliances. But people often use appliances as part of a larger task. For example, cooking dinner might require the kettle, hob, oven and microwave. The aim of this research is to provide users with a measure of total energy consumed by activities or by specific people in a shared dwelling.

For example, Bedwell et al. 2014 reviewed the issues involved in apportioning energy consumption to individuals in the workplace. And Stankovic et al. 2015 studied domestic appliance usage through their association with common activities.

Occupancy monitoring

Given disaggregated energy data, it may be possible to infer the occupancy state of the building; and possibly to infer whether the occupant is moving around. This could be used to remotely monitor the health of, for example, elderly people (Belley et al. 2013; Kalogridis & Dave 2014;

Alcalá et al. 2015). Or it could be used to inform a heating controller (Spiegel & Albayrak 2014; Spiegel 2015).

This technology also has security and privacy implications. We are aware of one NILM company who publicly advertised their ability to detect the occupancy schedules of individuals so that utility companies could schedule marketing cold-calling campaigns.

Drive consumers to engage with utilities online

Call centres are expensive. So utility companies can reduce costs if they can increase the proportion of their customers who engage with the company via their website rather than via call centres. Adding disaggregated energy information to the utility company's website might help to encourage more users to interact via the web. For example, Chakravarty & Gupta 2013 showed high levels of consumer engagement with disaggregated energy data.

1.4 Contributions and thesis outline

This thesis makes a number of contributions, which fall under three broad categories: 1) to *critically evaluate* the benefits of energy disaggregation; 2) to develop tools to *enable* rigorous disaggregation research; and 3) to *advance* the state of the art in disaggregation algorithms.

1.4.1 Critical evaluation of the benefits of energy disaggregation

In chapter 2 we describe what is – to our best knowledge – the only systematic review of studies on the effectiveness of disaggregated energy feedback to help people to reduce their energy consumption.

In chapter 3 we discuss some of the most prominent threats and challenges to NILM.

We conclude that more social science research into the effects of disaggregated energy feedback is required. This motivates the remainder of the thesis: to enable cost-effective research into the effects of disaggregated feedback, we work towards developing robust NILM algorithms and software.

1.4.2 Tools to enable rigorous disaggregation research

We present three tools:

Energy disaggregation researchers require data recorded in the field in order to train and validate disaggregation algorithms. We designed a novel, cost effective data collection system and used this to collect the only UK dataset with kilohertz temporal resolution. This dataset is called the UK Disaggregated Appliance-Level Electricity dataset (UK-DALE) and is described in chapter 4.

The second tool was designed to help the disaggregation community to conduct open, rigorous, repeatable research. We collaborated with other researchers to build the first open-source disaggregation framework, NILMTK (the non-intrusive load monitoring toolkit). NILMTK has gained significant traction in the community, both in terms of contributed code and in terms of users. NILMTK is described in chapter 5.

The third tool described in this thesis is a metadata schema for disaggregated energy data. This schema was developed to make it easier for researchers to describe their own datasets and to reduce the effort required to import datasets. The NILM Metadata schema is described in chapter 6.

1.4.3 Advancing the state of the art in disaggregation algorithms

We describe three disaggregation approaches that we developed using deep learning. One advantage of deep learning is that it can automatically learn features from the data, rather than requiring manual feature engineering. Our Neural NILM experiments are presented in chapter 9. The disaggregation performance was measured using seven metrics and compared to two 'benchmark' algorithms from NILMTK: combinatorial optimisation or factorial hidden Markov models. To explore how well the algorithms generalise to unseen houses, the performance of the algorithms was measured in two separate scenarios: one using test data from a house not seen during training and a second scenario using test data from houses which were seen during training. All three neural nets achieve better F1 scores (averaged over all five appliances) than either benchmark algorithm. The neural net algorithms also generalise well to unseen houses.

Finally, in chapter 10 we conclude the thesis.

1.5 Statement of originality

I declare that this thesis was composed by myself, and that the work that it presents is my own except where otherwise stated.

1.6 Publications

Where a publication forms the basis of a chapter in this thesis, that chapter is stated in bold type after the publication reference. My list of publications is also available on Google Scholar[28].

[28]https://scholar.google.co.uk/citations?user=Z9L0TgsAAAAJ&hl=en

1. Jack Kelly & William Knottenbelt (2012a). 'Disaggregating Multi-State Appliances From Smart Meter Data'. In: *Imperial College Energy and Performance Colloquium*. London, UK

2. Jack Kelly & William Knottenbelt (2012b). 'Disaggregating Multi-State Appliances From Smart Meter Data'. In: *SIGMETRICS*. ACM. London, UK - poster

3. Nipun Batra; Jack Kelly; Oliver Parson; Haimonti Dutta; William Knottenbelt; Alex Rogers; Amarjeet Singh & Mani Srivastava (2014a). 'NILMTK: An Open Source Toolkit for Non-intrusive Load Monitoring'. In: *5th International Conference on Future Energy Systems*. e-Energy. ACM. Cambridge, UK. DOI: 10.1145/2602044.2602051 (**Chapter 5**)

4. Jack Kelly; Nipun Batra; Oliver Parson; Haimonti Dutta; William Knottenbelt; Alex Rogers; Amarjeet Singh & Mani Srivastava (2014). 'NILMTK v0.2: A Non-intrusive Load Monitoring Toolkit for Large Scale Data Sets'. In: *1st International Conference on Embedded Systems For Energy-Efficient Buildings*. BuildSys. ACM. Memphis, TN, USA. DOI: 10.1145/2674061.2675024. arXiv: 1409.5908 (**Chapter 5**)

5. Jack Kelly & William Knottenbelt (2014). 'Metadata for Energy Disaggregation'. In: *Computer Software and Applications Conference Workshop at the 2nd International Workshop on Consumer Devices and Systems*. CDS. IEEE. Västerås, Sweden. DOI: 10.1109/COMPSACW.2014.97. arXiv: 1403.5946 (**Chapter 6**)

6. Menelaos Makriyiannis; Tudor Lung; Robert Craven; Francesca Toni & Jack Kelly (2014). 'Smarter Electricity through Argumentation'. In: *4th International Workshop on Combinations of Intelligent Methods and Applications in conjunction with the IEEE International Conference on Tools with AI*. IEEE

7. Jack Kelly & William Knottenbelt (2015b). 'The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes'. In: *Scientific Data* 2.150007. DOI: 10.1038/sdata.2015.7 (**Chapter 4**)

8. Jack Kelly & William Knottenbelt (2015a). 'Neural NILM: Deep Neural Networks Applied to Energy Disaggregation'. In: *2nd Workshop On Embedded Systems For Energy-Efficient Buildings*. BuildSys. ACM. Seoul, South Korea, pages 55–64. DOI: 10.1145/2821650.2821672. arXiv: 1507.06594 (**Chapter 9**)

9. Oliver Parson; Grant Fisher; April Hersey; Nipun Batra; Jack Kelly; Amarjeet Singh; William Knottenbelt & Alex Rogers (2015). 'Dataport and NILMTK: A building data set designed for non-intrusive load monitoring'. In: *1st International Symposium on Signal Processing Applications in Smart Buildings at 3rd Global Conference on Signal & Information Processing*. GlobalSIP. IEEE. Orlando, Florida, USA (**Chapter 5**)

10. Menelaos Makriyiannis; Tudor Lung; Robert Craven; Francesca Toni & Jack Kelly (2016). 'Smarter Electricity and Argumentation Theory'. In: *Combinations of Intelligent Methods and Applications: Proceedings of the 4th International Workshop, CIMA 2014, Limassol, Cyprus, November 2014 (at ICTAI 2014)*. Edited by Ioannis Hatzilygeroudis; Vasile Palade & Jim Prentzas. Volume 46. Springer, pages 79–95. DOI: `10.1007/978-3-319-26860-6_5`

11. Jack Kelly & William Knottenbelt (2016). 'Does disaggregated electricity feedback reduce domestic electricity consumption? A systematic review of the literature'. In: *3rd International NILM Workshop*. Vancouver, Canada. arXiv: `1605.00962` - also presented at *the First Energy Feedback Symposium*[29]*: "Feedback in energy demand reduction: Examining evidence and exploring opportunities"*, Edinburgh, 4-5th July 2016. (**Chapter 2**)

1.6.1 Formatting conventions in this thesis

Blue text denotes a hyperlink. The link may be to an external web page or to a label in the PDF file. Whenever possible, URLs to external web pages will also be printed as footnotes.

Academic papers and articles on websites are cited as per normal in an academic document. Hyperlinks to websites are used when talking about specific websites (as apposed to articles within the website) or to products.

Code inline with the main text is shown with `typewriter font`.

[29]`https://teddinet.org/activities/energy-feedback-symposium/`

Part I

Critically evaluate the effectiveness of disaggregation

Chapter 2

Does disaggregated electricity feedback reduce domestic electricity consumption?

In this chapter, which is based on Kelly & Knottenbelt 2016, we present a systematic review of twelve studies on the efficacy of disaggregated energy feedback. In section 1.3.5 we discussed the *theories* which suggest that disaggregated feedback should enable users to save energy. In the following section, we discuss the *empirical* evidence as to whether people *actually* save energy when given disaggregated data. We discuss four main questions:

1. Can disaggregated energy data help an already-motivated sub-group of the general population ('energy enthusiasts') to save energy?

2. How much energy would the *general population* save if given disaggregated data?

3. Is *fine-grained* disaggregation required?

4. For the general population, does disaggregated energy feedback enable greater savings than *aggregate* data?

2.1 An introduction to systematic reviews

Systematic reviews are common in fields such as medicine and the social sciences. Systematic reviews aim to find results which are robust across *multiple* studies as well as opportunities for future research. The process starts with a search, using predefined criteria, for existing papers. Results and possible biases are extracted from each paper, collated and combined. See Garg et al. 2008 for a discussion of systematic reviews.

Our work is, to the best of our knowledge, the first systematic review on the effectiveness of domestic, disaggregated electricity feedback.

There is a distinction between *narrative* reviews and *systematic* reviews. Most review articles are narrative reviews. These are written by domain experts and contain a discussion of existing papers. Narrative reviews are often very valuable. But they are rarely explicit about how papers were selected and rarely attempt a quantitative synthesis of the results.

Systematic reviews aim to cover *all* papers which match defined criteria relevant to a specific research question. Systematic reviews are explicit about how papers were selected and present a quantitative summary of each paper and a quantitative synthesis of the results. Systematic reviews may contain a 'meta-analysis' where results from each study are combined into a single statistical analysis which provides greater statistical power than any individual study can deliver.

Systematic reviews are not perfect, of course. Bias can still creep in via the selection process; and different statistical analyses may present different results.

Why bother with systematic reviews? *Replication* is essential to the scientific process. Peer review is necessary *but not sufficient* to ensure that individual studies present an accurate estimate of the 'true' state of the world. Reviewers rarely, if ever, attempt to replicate results; possibly because there is insufficient reward to motivate reviewers to spend time on replication. Instead it is left to the community to attempt to replicate results. Recent large-scale replication projects suggest that replicable results may be the *exception* rather than the rule. An attempt to replicate results from 98 psychology papers could only replicate 39% of the results (Baker 2015; Open Science Collaboration 2015). A similar study in cancer biology found that only 6 of the results in 53 high-profile papers could be replicated (Baker 2015; Begley & Ellis 2012).

Hence it is advisable to exercise appropriate scientific scepticism when reading any single study; and it is beneficial to collect all papers on a specific question to identify results which are robust across studies.

2.2 Methodology

Broadly, this chapter discusses whether deployment of disaggregation across the entire population is likely to reduce energy consumption. We assume that disaggregated data for a population-wide deployment would be delivered via websites, smart-phone applications and/or paper bills.

We found twelve groups of studies on the question of whether disaggregated energy data helps users to reduce their energy demand. These studies are summarised in Table 2.1.

We aimed to do an exhaustive search of the literature (although we acknowledge the possibility that we missed studies). We used three search engines: Google Scholar, the ACM Digital Library and IEEE Xplore. The search terms we used were 'disaggregated [energy|electricity] feedback' and 'N[I|A|IA]LM feedback'. These searches produced a huge number of results, many of which were not relevant to our research question. We manually selected papers which test the effectiveness of disaggregated electricity feedback. We accepted experiments conducted either in a laboratory environment or in a field test. We also searched the bibliography sections of papers to find more papers. For example, a review article by Ehrhardt-Martinez et al. 2010 contained references to five relevant studies on disaggregated energy feedback.

2.3 Can disaggregated electricity feedback enable 'energy enthusiasts' to save energy?

The mean reduction in electricity consumption across the twelve studies (weighted by the number of participants in each study) is 4.5%. However, as we will discuss below, this figure is likely to be positively-biased and has a substantial (although unquantifiable) amount of uncertainty associated with it.

Aggregating the results by taking the *mean* of the energy savings across the twelve studies is a crude approach. It would have been preferable to do a full meta-analysis where biases are identified and compensated for (Garg et al. 2008). Davis et al. 2013 did such a meta-analysis for studies on *aggregate* energy feedback. But the studies on disaggregated feedback appear to us to be too varied and, perhaps most fundamentally, six of the twelve studies only provided a point estimate of the effect size. At the very least, a meta-analysis requires that each study provides a point estimate *and* a measure of the *spread* of the results.

We must also be explicit about the likely biases in each study. Please note that this is *not* an attack on the papers in question. We appreciate that it is not possible to conduct a 'perfect' study. The real world is messy and researchers cannot control for everything. Being explicit about the biases allows us to assess how much trust we should put into the assertion that disaggregated energy feedback reduces consumption by 4.5%.

There are several sources of positive bias present in the papers. All twelve studies are prone to 'opt-in' bias, where subjects self-selected to some extent and so are likely to be disproportionately interested in energy.

Eight studies did not control for the Hawthorne effect. This strange effect is where participants reduce their energy consumption simply because they know they are in an energy study. For example, D. Schwartz et al. 2013 conducted a controlled study on 6 350 participants, split equally between control and treatment groups. Subjects in the treatment group received a weekly postcard saying: *'You have been selected to be part of a one-month study of how much electricity you use in your home... No action is needed on your part. We will send you a weekly reminder postcard about the study...'* Participants who received these postcards reduced their consumption by 2.7%. Hence studies on disaggregated energy feedback which do not control for the Hawthorne effect are likely to over-estimate energy savings attributable to the disaggregated energy feedback.

Six studies used feedback displays which were probably more attention-grabbing than the feedback mediums that would be used in a population-wide roll-out of disaggregated energy feedback. For example, Dobson & Griffin 1992 installed dedicated desktop PCs in participants' kitchens (see Figure 2.1). Some studies gave home-visits to some participants to enable additional reductions (e.g. T. Schwartz et al. 2015; Brown 2014; HEA 2015). All but two studies were too short to observe whether energy reductions persist long-term. Perhaps some authors

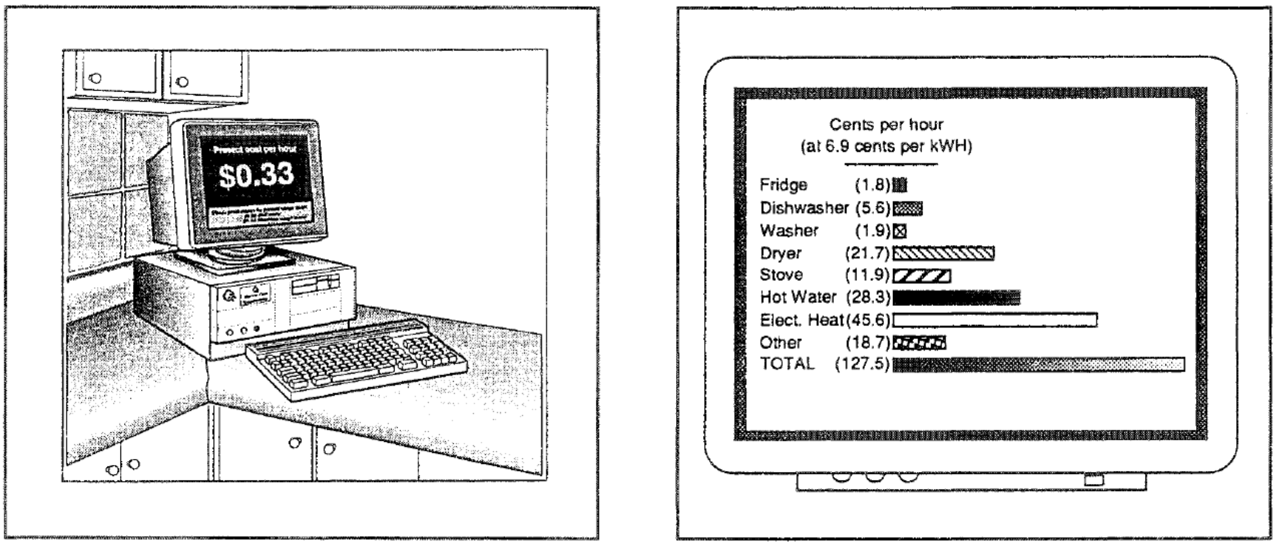


Figure 2.1: The desktop PC used to display disaggregated energy data in the trial performed by Dobson & Griffin 1992.

experimented with multiple statistical techniques until one delivered a significant result. And, finally, Churchwell et al. 2014 found that some users do not trust the output from disaggregation algorithms so it is significant that eight studies used disaggregated data collected by installing *meters* on individual appliances instead of using a disaggregation algorithm; hence avoiding any mistrust of disaggregated estimates.

As well as being explicit about biases in each study, we must acknowledge that the *literature* as a whole may be prone to publication bias. How many *negative* results exist unpublished? Perhaps academics fear that reviewers would reject a null result? Might companies fear that customers or shareholders would be driven away? A study on publication bias in the social sciences found that positive results are 60% more likely to be written up than null results and 40% more likely to be published (Franco et al. 2014). Franco et al. 2014 propose that science would benefit from mechanisms to reduce the effect of publication bias, such as pre-registering experiments.

Despite these sources of bias, there is evidence that energy disaggregation *can* enable energy savings for 'energy enthusiasts'. Two large studies illustrate this assertion:

One group of studies analysed the disaggregation service provided by Home Energy Analytics (HEA) (Schmidt 2012; HEA 2012; HEA 2013; Brown 2014; HEA 2015). All participants *opted into* HEA's system and hence could loosely be considered 'energy enthusiasts'. In total the HEA papers examine 1 623 users. 1 239 used the system for up to 44 months; the rest used the system for one year. The average reduction in electricity consumption across all 1 623 'energy enthusiasts' was 6.1%. The top-quartile (310 'super-enthusiasts') reduced their electricity consumption by 14.5%. But note that none of the HEA studies had a control group; and some participants received home visits to help them to reduce their energy consumption.

Another large study was performed in 2014 over three months on 1 685 PG&E users (Churchwell et al. 2014; Bidgely 2015). Half received an in-home display (IHD) and half received access to Bidgely's website (which includes disaggregation). No statistically significant reduction in consumption was found across all 1 685 users, despite positive biases (e.g. users could *choose* between the IHD or Bidgely). However, a sub-group of users on a time-of-use (TOU) tariff ('energy enthusiasts') saved 7.7%; this group consisted of 142 IHD users and 136 Bidgely users.

2.4 How much energy would the *whole* population save if given disaggregated data?

All twelve studies suffer from opt-in bias to some extent. Seven studies have a *high* risk of opt-in bias because participants sought out the intervention. As such, the study participants are unlikely to be representative of the general population. No 'perfect' correction for opt-in bias exists.

What will the *average* energy saving be across the population if, say, the majority of the population completely ignores disaggregated energy feedback but a small sub-population of 'energy enthusiasts' save 4.5%?

How can we estimate the proportion of 'energy enthusiasts' in the population? Three studies reported the number of people *approached* to participate versus the number who *agreed* to participate (Wood & Newborough 2003; HEA 2015; Sokoloski 2015). This 'opt-in rate' is a crude estimate on the *lower bound* of the proportion of the population who are 'energy enthusiasts' (because, in order to agree to participate in an energy study, people probably need to be energy enthusiasts *and* also have time to participate in the study *and* be willing to let experimenters into their homes etc.). The average opt-in rate is 16%. This is consistent with Murtagh et al. 2014 who estimate that 20% of the population are '[energy] monitor enthusiasts'. If 16% of the population reduced their energy consumption by 4.5% then the mean reduction

Table 2.1: Studies on the effectiveness of disaggregated energy feedback.

Study	Feedback presentation	Num. houses in disag. group	Num. houses in study	Num. disaggregation categories	Duration (months of disag)	Reduction in electricity use[U] (%)	Reduction is for whole house?	Sample period of meter	Feedback delay	Timing: Historic or Concurrent?	Time frames for historic[T]	Recommendations given?[R]	Control group?	Controlled for Hawthorne?	Volunteer bias?[V]	Controlled for weather?
"RECS" Dobson & Griffin 1992	dedicated computer	25	100	∼ 8	2	12.9	✓	0.6 sec	0	H&C	HDM	✗	✓	✓	L[a]	✓
McCalley & Midden 2002	Virtual washing machine [b]	25	100	1	-	0.0	✗	-	0	H&C	-	G	✓	✓	L	-
Wood & Newborough 03; Mansouri & Newborough 99	LCD by cooker	10	44	1	≥ 2	12.2	✗	15 sec	0	C	-	✗[c]	✓	✓	L	✓
"ECOIS-I" Ueno et al. 2006b; Ueno et al. 2006c	Dedicated laptop	8	8[d]	16	2	9	✓	30 min	next day	H	D, 10D	P	✗	✗	H#	✓[e]
"ECOIS-II" Ueno et al. 2005; Ueno et al. 2006a; Ueno et al. 2006c	Dedicated laptop	10	19	16	3	18	✓	30 min	next day	H	D, 10D	P	✓	✓	H#	✓
"EnergyLife" trial 1 Jacucci et al. 2009; Spagnolli et al. 2011; Gamberini et al. 2011	iPhone	13	13	7	3	5	✓	?	1-2 min	H&C	D	P	✗#	✗#	H#	✗#
"EnergyLife" trial 2 Gamberini et al. 2012	iPhone	4	4	7	4	38	✗	?	1-2 min	H&C	D	P	✗	✗#	H#	✗#
Home Energy Analytics HEA 2012; HEA 2013; Brown 2014; HEA 2015	Web & email & home visits	1623	1623	5	≤ 44	6.1	✓	hourly	0	H	Y	P	✗	✗	L	✓
Bidgely Chakravarty & Gupta 2013; Gupta & Chakravarty 2014	Web, mobile, email	163	328	≥ 3?	-	6	✓	30 sec & 1 hr	0[f]	H&C[f]	DBY	P	✓	✗	H	✓
PG&E Pilot Churchwell et al. 2014; Bidgely 2015	Web, mobile, email	844	1685	≥ 3?	3	2.1	✓	30 sec	0[f]	H&C[f]	DBY	P	✓	✗	H	✓
T. Schwartz et al. 2015	Web, mob, TV	6	6	∼ 10	18	7.8	✓	?	0?	H&C	?	?	✗	✗	H	✗
Sokoloski 2015	Web, mob, email	12	70	≥ 3?	0.75	3	✓	30 sec	0[f]	H&C[f]	DBY	P	✓	✗	L	✓

A dash '-' in a cell means 'not applicable (NA)' and '?' means 'not specified in paper'.

[U] Absolute reductions minus reductions for the no-contact control (or the most similar group to a no-contact control available).

[R] Recommendations can be 'P' for 'personalised' or 'G' for 'general' or '✗' for none given.

[V] Volunteer bias can be 'H' for 'high' (subjects sought out the intervention) or 'L' for 'low' (subjects were approached by the experimenters but only a fraction agreed to participate).

[T] H=hourly, D=daily, M=monthly, Y=yearly, B=current billing cycle.

[#] Paper is silent on this question. Assume the worst.

[a] Dobson & Griffin 1992 do not state exactly how households were recruited. They write "*100 all-electric households were qualified from a random sample drawn from a population of approximately 8800 such houses*". I assume households were not *forced* to participate so they must have self-selected to some extent.

[b] A washing machine control was simulated on a computer. The reported energy reduction is *only* for the simulated washer. The no-feedback-no-goal condition and the feedback-no-goal conditions achieved the same reduction (11%), hence the difference in energy savings between those two conditions is 0%.

[c] One group received both real-time energy feedback for the cooker and a printed information pack of general recommendations but this group achieved lower energy savings (8.9%) than the group which only received energy feedback.

[d] ECOIS-I started with 9 houses but one house was excluded because it had solar PV installed.

[e] Ueno et al. 2006b report that the "*average ambient temperatures before and after installation were 6.4 and 6.8 ℃, respectively. Generally, the power consumption of the whole household increases with the fall in ambient temperature in winter; hence, it is thought that the true effect was more than this 9% value.*"

[f] Aggregate data was displayed real-time. Disaggregated data was not real-time.

would be 0.7%.

This may seem rather pessimistic. We assumed that 84% of the population (the 'disinterested') would save *no* energy. Perhaps this is a little unrealistic: we might hope that some proportion of the 'disinterested' group would save a little energy. Furthermore, we used a crude method to determine a lower bound on the proportion of 'energy enthusiasts' in the population. But remember that we have multiple reasons for believing that a 4.5% saving across the 'energy enthusiast' population is an *over-estimate*. We assume that these negative and positive biases cancel out, although we cannot be sure.

Also note that we simply have *no* good evidence for how the *general population* would react to disaggregated energy data. However, related studies have found that effect sizes reported on opt-in groups are often substantially diminished when studied on the general population (Davis et al. 2013).

Can we compare these figures to other research? A study involving 2 000 Swedish households found that participants who visited a website which provided user-friendly analysis of their aggregated electricity consumption reduced their electricity consumption by 15% on average (Vassileva et al. 2012). The savings sustained for the duration of the four-year study. But only 32% of those with access to the website visited the website. Households who did not visit the website did not reduce their energy consumption. Hence the average energy reduction across all households with access to the website was $32\% \times 15\% \approx 5\%$.

2.5 Is 'fine-grained' disaggregation necessary?

Much research into disaggregation aims to deliver 'fine-grained' estimated power demand for each appliance at relatively high temporal resolution (e.g. 0.1 Hz). Fine-grained disaggregation algorithms are complex to engineer and often computationally expensive to run. Is it worth the effort? Home Energy Analytics (HEA) do 'coarse-grained' disaggregation: they disaggregate energy usage into five broad categories at *monthly* temporal resolution (see Figure 2.2). Despite the coarse granularity of the feedback, HEA achieved significant average reductions in electricity usage of 6.1% (Schmidt 2012; HEA 2012; HEA 2013; Brown 2014; HEA 2015). HEA's results tell us that fine-grained feedback is certainly not *required*. Fine-grained feedback enables many

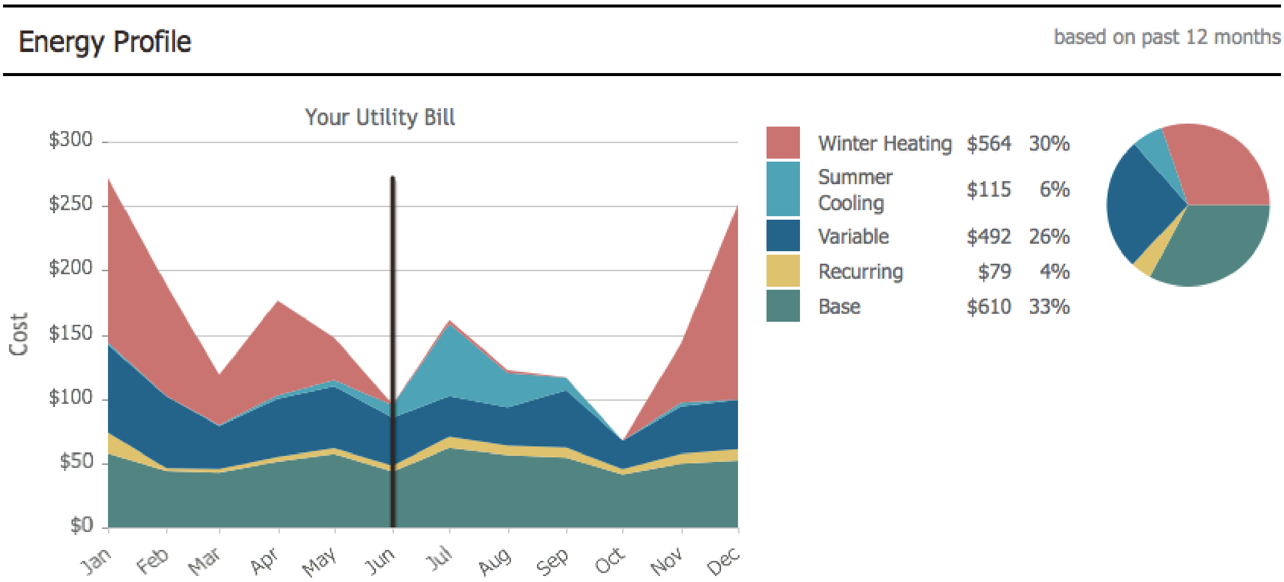


Figure 2.2: An example of the 'coarse-grained' disaggregation performed by HEA. Source: http://corp.hea.com/how-it-works (with permission).

use-cases not discussed here but, on the question of the efficacy of feedback to drive energy reductions, we simply do not know if fine-grained feedback is more effective because no studies compared fine-grained against coarse-grained feedback. Fine-grained feedback might be *less* effective because some users do not trust it (Churchwell et al. 2014).

2.6 Does aggregate or disaggregated feedback enable greater savings for the whole population?

Four studies directly compared aggregate feedback against disaggregated feedback. Three of these studies found aggregate feedback to be *more* effective than disaggregated feedback (Krishnamurti et al. 2013; Churchwell et al. 2014; Sokoloski 2015). The fourth study found disaggregated feedback and aggregate feedback to be equally effective (McCalley & Midden 2002). Are there any explanations for this counter-intuitive result?

Two of the four studies (McCalley & Midden 2002; Krishnamurti et al. 2013) were synthetic computer simulations and so may not generalise to 'real life'.

The other two studies were well controlled field studies (Churchwell et al. 2014; Sokoloski 2015). In both field studies, aggregate feedback was displayed on an always-on IHD whilst disaggregated data was displayed on Bidgely's website (which has since been redesigned (Bidgely 2015)). Participants in the disaggregation groups did not have an IHD. Sokoloski 2015 found that, on average, participants in the IHD condition viewed the IHD eight times per day whilst participants in the disaggregation condition viewed the website only once per day. Churchwell et al. 2014 found a similar pattern and also reported that some participants did not trust the fine-grained disaggregated data. Perhaps aggregate data is not *intrinsically* more effective than disaggregated data; instead, perhaps *IHDs* are more effective than *websites* or mobile apps.

Perhaps *dedicated* displays for disaggregated data may help enhance efficacy, although this adds costs. Or, as Sokoloski 2015 suggests, efficacy may be increased by *combining* disaggregated feedback presented on a website with aggregate feedback presented on an IHD.

Furthermore, a meta-analysis of the efficacy of *aggregate* energy feedback suggests it alone achieves 3% energy savings (Davis et al. 2013). This analysis adjusted for several (but not all) biases.

2.7 Suggestions for future research

There are several gaps in the existing literature. Below is a list of suggestions for new experiments.

No existing field studies compared aggregate feedback against disaggregated feedback *on the same type of display*. The studies which *did* compare aggregate feedback against disaggregated feedback used an IHD for aggregate feedback and a website for disaggregated feedback and found that the aggregate feedback was more effective at reducing energy demand. But we cannot rule out that this result is simply because users viewed the IHD more frequently than they viewed the website. Hence it would be valuable to run an experiment where both the 'aggregate' and 'disaggregated' groups received feedback on the same device (e.g. an IHD with a dot-matrix display to display disaggregated feedback).

A related study would explore the effectiveness of aggregate feedback presented on an IHD *combined with* access to disaggregated data on a website; compared to just the IHD. The IHD might pique users' interest and motivate them to explore their disaggregated energy usage on a website or smart phone. For example, Elburg 2015 state that *"In-home displays appear to be* *a crucial 'stepping stone' to kick-start consumer interest and engagement in accessing energy information, especially amongst less committed or experienced consumers. Sophisticated web-based services on PC, tablet and smart phone are potentially more powerful to help reduce energy demand, but in practice more so with already committed and technology minded subsets of the population. Therefore, opt-in websites or apps should not be considered as the contemporary substitute for in-home displays, but rather as a complementary option."* Furthermore, Snow et al. 2013 argue that disaggregated data may have a role to play in maintaining and sustaining the effect of feedback. And Ståhlberg 2010 surveyed energy users and found that the option to have *both* an in-home-display *and* access to a web portal was the second-favourite option. The UK Government's Department of Energy and Climate Change recently announced that they will allow utility companies be temporarily exempted from the requirement to supply IHDs to all customers if the utility company wishes to run a rigorous trial of an alternative feedback technology (DECC et al. 2016).

How 'granular' does the disaggregated energy feedback need to be to enable people to save energy? The HEA studies show a 6.1% electricity saving despite the fact that their platform provides relatively coarse feedback (it is coarse both in the temporal dimension and in terms of the categories that they disaggregate into). HEA do not publish full details of their algorithm but it may be similar to Batra et al. 2016c, which is much more straight-forward to implement and to compute than many NILM algorithms. Are highly-granular NILM algorithms worth the extra engineering effort?

How does the *general population* respond to disaggregated energy data? Do different sub-groups of the general population react differently? Evidence presented in both Churchwell et al. 2014 and Sokoloski 2015 suggest, with low statistical confidence, that sub-groups react differently. If sub-groups do react differently, what proportion of the population save energy when given access to disaggregated energy data and what features define this sub-population? For example, is a pre-existing, deep concern about climate change necessary to enable users to save energy when given disaggregated data? Do 'fuel-poor' users in the developed world or the developing world pay more attention to disaggregated feedback than rich people because saving money is more critical to them?

In my analysis, I grouped all studies together using quite a loose definition of 'disaggregated feedback'. The studies use a diverse set of disaggregation approaches. Some studies only provide feedback on a single appliance. Some were synthetic lab studies whilst other were field trials. Some provided disaggregation at the level of appliances; others at the level of load type. There are probably too few studies to allow a meta-analysis to meaningfully attribute variance to these different factors. But a Bayesian network meta-analysis may be able to provide additional insight if used with informative priors (Gelman & Hill 2007; J. Zhang et al. 2015).

Can disaggregated energy feedback drive *long-term* energy reductions? All but one study ran for five months or less and so cannot speak to this question. HEA's studies ran for up to 44 months and found an average electricity reduction of 6.1% but that was in a sub-population who were motivated to save energy. Manifest Mind, a consultancy, has a phrase for the rapid decay in enthusiasm that users have for home energy management devices: "Mean Time to Kitchen Drawer" (Walton 2015).

How important is immediate, real-time disaggregated feedback? None of the studies compared real-time disaggregated feedback to offline feedback. Real-time disaggregation is computationally harder and likely to be less accurate than 'offline' disaggregation because just the start of the appliance signature is available. Three studies provided real-time disaggregated feedback (Dobson & Griffin 1992; McCalley & Midden 2002; Wood & Newborough 2003). The average saving across these three studies is 9.4% and the average across the studies using 'offline' disaggregation is 11.6% but this comparison is too crude to draw any robust conclusions.

What happens if disaggregated energy data is displayed on one or more dedicated displays? This could be a single, central display. Or multiple displays, one near each appliance. See Wood & Newborough 2007 for further discussion. None of the studies that I reviewed compared different ways of presenting disaggregated energy data. Neither did any of the general studies on changing habits using digital technology reviewed by Hermsen et al. 2016. Sokoloski 2015 commented that the relatively low energy savings in her disaggregation group (3% compared with 10.6% in the IHD group) may, at least in part, be due to the fact the people viewed the disaggregation website only once a day whilst subjects in the IHD group viewed their IHD eight times a day, on average: a highly significant difference ($p < 0.001$). Displaying disaggregated data on an IHD may be possible. But, in the UK at least, IHDs are to be supplied to all customers, free of charge. Which means that utility companies are eager to absolutely minimise the cost of IHDs. Even a cost increase of a few pennies would be unattractive.

Does disaggregated energy feedback displayed on a paper energy bill drive energy savings more or less effectively than disaggregated feedback displayed on a website?

How many *unpublished* studies reached *negative* conclusions about the effectiveness of disaggregated energy data but remain unpublished because of the publication bias towards positive results?

Can disaggregated energy data *alone* (without recommendations) produce energy savings? Or is the power of disaggregation that it can enable better personalisation of recommendations? If recommendations are required then do they *have* to be *personalised* recommendations?

Does normative feedback on disaggregated data (e.g. "your fridge is using more energy than most fridges") help people to save energy? There is evidence that normative feedback on *aggregate* energy data is *not* effective. Negative normative feedback ("you are doing worse than most people") could be seen as a kind of high-tech bullying and just makes people feel uncomfortable without producing results. *Positive* normative feedback ("you are doing better than most people") might make people feel complacent and actually drive an *increase* in consumption.

Do users who already use a low amount of energy *increase* their energy consumption when given feedback? For example, Sokoloski 2015 found that 'low' energy users *increased* they energy consumption by 7.98% when given disaggregated energy data whilst 'high' energy users decreased their consumption by 8.84%. Similar trends were found for aggregate energy feedback by Bittle et al. 1979–1980 and Brandon & Lewis 1999.

What is the effect of different groupings of disaggregated energy data? For example, what is the effect if energy consumption is itemised by *activity* rather than by appliance? Wood & Newborough 2007 discuss several ways to group appliances and Rettie et al. 2014 argues that it is important to provide feedback in terms of activities rather than energy consumption or price; and to talk in terms of 'wastage'.

Pullinger et al. 2014 discusses the need to provide users with 'practice-based' advice (where a 'practice' is an action such as doing the washing or taking a shower) and user-friendly assistance to understand the data and motivating behaviour change by describing comfort and convenience benefits rather than just carbon or price benefits.

Bartram 2015 reviews and discusses many opportunities for visualising eco-feedback for domestic users. Could these techniques be used for visualising disaggregated data?

Below is a list of suggestions for how to make future papers on feedback as useful as possible: If possible, conduct a *randomised controlled trial*. Publish as much information as possible. How were subjects recruited? Were subjects selected from the *general population*? Did any subjects withdraw during the study period? Was there a control group? Did the study control for the Hawthorne effect and weather? What sources of bias may influence the result? How exactly was feedback presented; and how rapidly did the information update? How often did participants view the display? Was disaggregated data available from the very beginning of the experiment or did the disaggregation platform take time to adapt to each home? Publish the results of *all* the valid statistical analyses performed; not just the 'best' result. Crucially, please publish some measure of the *spread* of the result (e.g. the standard deviation). Ideally, publish online, full, anonymised results so researchers can collate your results into a meta-analysis.

2.8 Conclusions

Disaggregation has *many* use-cases beyond feedback. This paper specifically considers a single use-case of disaggregation: reducing energy consumption via feedback. Averaged across the population, there is evidence that disaggregated feedback *may* help to reduce electricity consumption by ~0.7-4.5%. But disaggregation might not be *necessary* to achieve this saving because aggregate feedback may be equally effective. Amongst 'energy enthusiasts', disaggregated feedback might save more energy but fine-grained disaggregation may not be necessary.

We must emphasise that all we can do is report the current state of the research. We *cannot* rule out the possibility that disaggregated feedback is, in fact, more effective than aggregate feedback. Neither can we rule out that fine-grained feedback is more effective than coarse-grained. All we can say is that current evidence contradicts the first hypothesis and that there is no evidence available to address the second hypothesis.

Perhaps users will become more interested in disaggregated data if energy prices increase or if concern about climate change deepens. Or perhaps users in fuel poverty will be more likely to act on disaggregated feedback in order to save money. Or perhaps users will trust disaggregation estimates more if accuracy improves or if designers find ways to communicate uncertain disaggregation estimates. Or perhaps real-time feedback or better recommendations will improve performance. Or perhaps disaggregating by behaviour rather than by appliance will make disaggregated feedback more effective.

Importantly, we conclude that the existing evidence-base is heterogeneous and has many gaps. Perhaps a large, well controlled, long-duration, randomised, international study will find that disaggregated feedback is more effective than aggregated feedback. This desire for more social science research into the effects of disaggregated energy feedback motivates the second and third parts of this thesis: in order to perform large-scale research into the effects of disaggregated energy feedback, it would be useful to have robust disaggregation algorithms and software.

Chapter 3

Threats and challenges to NILM

This chapter will address some common concerns about NILM.

3.1 Will ‘smart appliances’ & ‘smart plugs’ make NILM obsolete?

It is already possible for ‘smart appliances’ and ‘smart plugs’ to report appliance-level power demand to the network. It is also already possible for a single ‘energy dashboard’ to allow the user to compare the energy used by each appliance (Egarter et al. 2015). In fact, this has been possible for a while: smart home systems have been manufactured at least since 1985 (see Figure 3.1). But they have yet to achieve significant market penetration. If smart homes *do* become widespread then this could make NILM obsolete. So, to decide whether energy disaggregation is a worthwhile technology, we must ask whether widespread adoption of the ‘smart home’ is imminent.

The short answer is that smart home technology today is a little like computer networking technology in the 1980s. Back then, there were many competing options for networking hardware (Ethernet, Token ring, ARCNET, AppleTalk) and networking protocols (NetBEUI, IPX/SPX, AppleTalk, TCP/IP). Crucially, these systems did not work together. It required a lot of skill and care to build a network which ran efficiently. For users and developers alike, investing a lot of money in one networking standard was risky because an alternative networking system

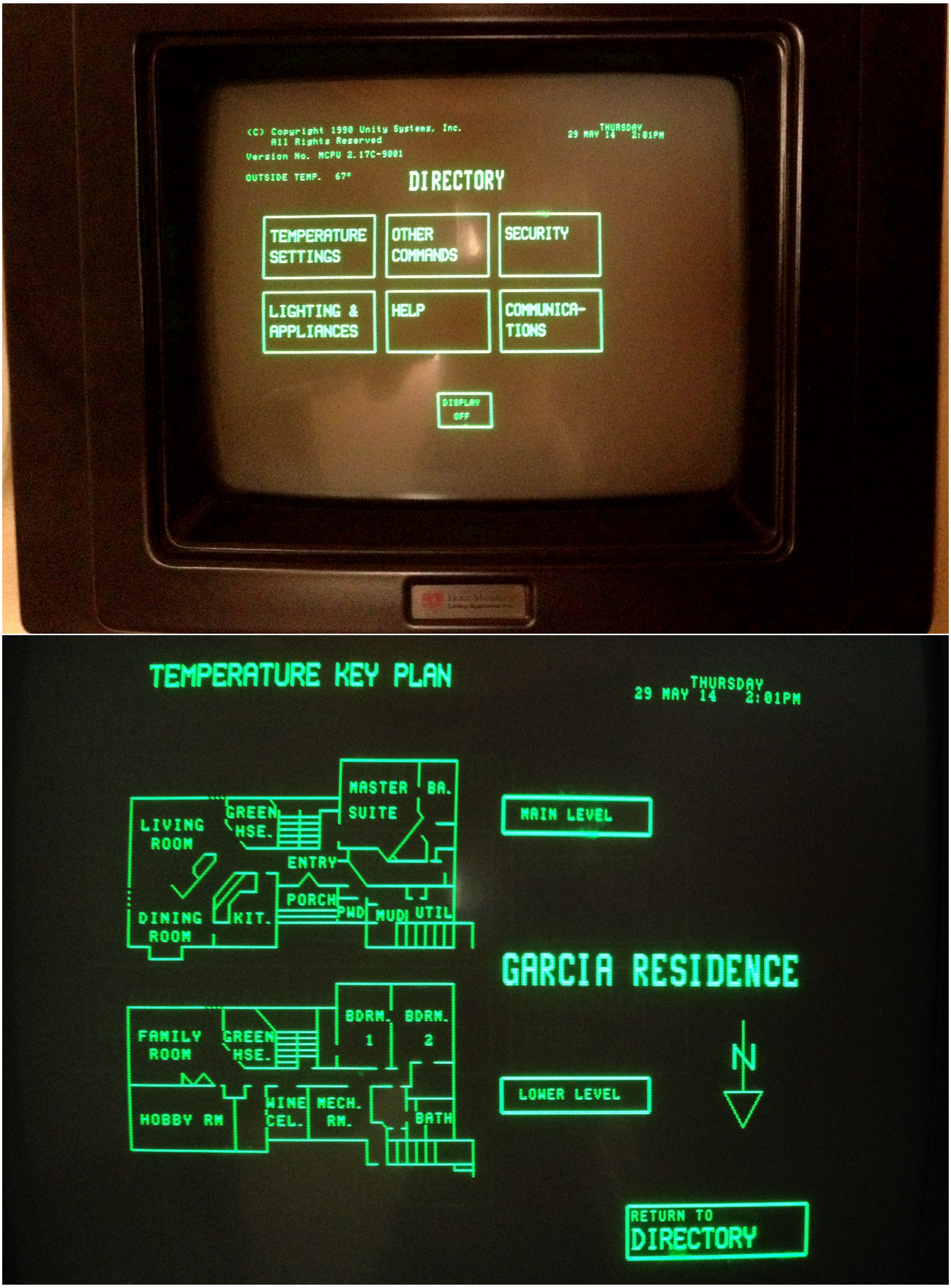


Figure 3.1: A touchscreen 'Home Manager' made by Unity Systems and installed in 1990. Unity Systems made Home Managers from 1985 until 1999. It controls sockets, switches, the sprinkler, security, temperature and more. Source: http://imgur.com/a/Jb6jW.

could soon become dominant. It was not until TCP/IP over Ethernet became dominant in the 1990s that computer networking took off.

As I will describe below, today's 'smart home' market has many of the problems that the PC networking market had thirty years ago.

At least three technologies are required for the smart home to provide disaggregated energy data to the user:

1. Appliances or power sockets which measure their own power demand and report this to the network.
2. A physical network to transfer data from each appliance to the internet.
3. A standard API or metadata schema for energy data. This is required to allow a single, third party energy dashboard application to acquire and understand energy data from each appliance, no matter who manufactured the appliance.

Below, I will examine each of these requirements:

3.1.1 Smart appliances

Today 'smart' appliances are available from manufacturers including Panasonic, LG, Electrolux, Bosch, Whirlpool, General Electric, Siemens and Samsung. Some smart appliances do report their energy consumption to the network. For example, the Whirlpool WEL98HEBU ($1,500 USD) has a 'smart energy' feature which tracks how much energy it uses. It is hard to be sure exactly how many of today's smart appliances report energy demand to the network because the published specifications for these appliances rarely mention energy at all and, if they do mention energy, the specifications give very few details.

Are manufacturers losing interest in energy?

There is evidence that manufacturers are losing interest in *energy* applications for smart appliances and are focusing more on features related to 'convenience', security and entertainment (Wilkenfeld & Harrington 2015). To see this trend *away* from energy, contrast product offerings from about five years ago with current products:

In 2009 Google launched 'PowerMeter' and Microsoft launched 'Hohm'. Both of these products were web applications designed to help users track their aggregate energy consumption. In 2010, General Electric (GE) unveiled their 'Nucleus Energy Manager - The Future of Home Energy Management' (General Electric 2010). GE's Nucleus Energy Manage was part of GE's 'Brillion' suite of home energy solutions (see Figure 3.2) which included several smart appliances, each of which could measure their own energy consumption and report this using ZigBee Smart Energy Profile to a Nucleus hub, which would send energy data to the internet to allow users to view disaggregated data on a web interface. GE's marketing for Nucleus placed 'energy' front and centre: their 2010 press release for Nucleus (General Electric 2010) mentioned 'energy' 30 times and many screen shots for the 'Nucleus' smart phone app showed appliance energy consumption (Figure 3.3).

However, manufacturers appear to have quickly moved *away* from energy applications. In June 2011, (just as I started my PhD on energy disaggregation), Google killed 'PowerMeter' (Kanellos 2011) and Microsoft killed 'Hohm' (LaMonica 2011). GE killed Nucleus some time around 2013 (the last software update for their Nucleus iPhone app was October 2012 (General Electric 2012)). GE's 2015 press release about their new range of connected appliances mentions 'energy' only once as an afterthought (General Electric 2015b). The app store pages for GE's most recent smart home apps do not show a single screen shot relating to energy (General Electric 2015a).

This move away from energy is not restricted to GE, Google and Microsoft. Whirlpool's global director of energy and sustainability recently gave an interview (St. John 2013) in which he stated that:

> "*The energy piece is an interesting story, but not the most compelling one for the consumer... because these new appliances are first and foremost consumer goods, making features like remote control and visibility, as well as ease of networking and use, far more important than energy savings.*".

On the other hand, there is evidence that appliance manufacturers *do* still have an interest in energy, although perhaps not for consumer-facing products. For example, Dr Robby Simpson is

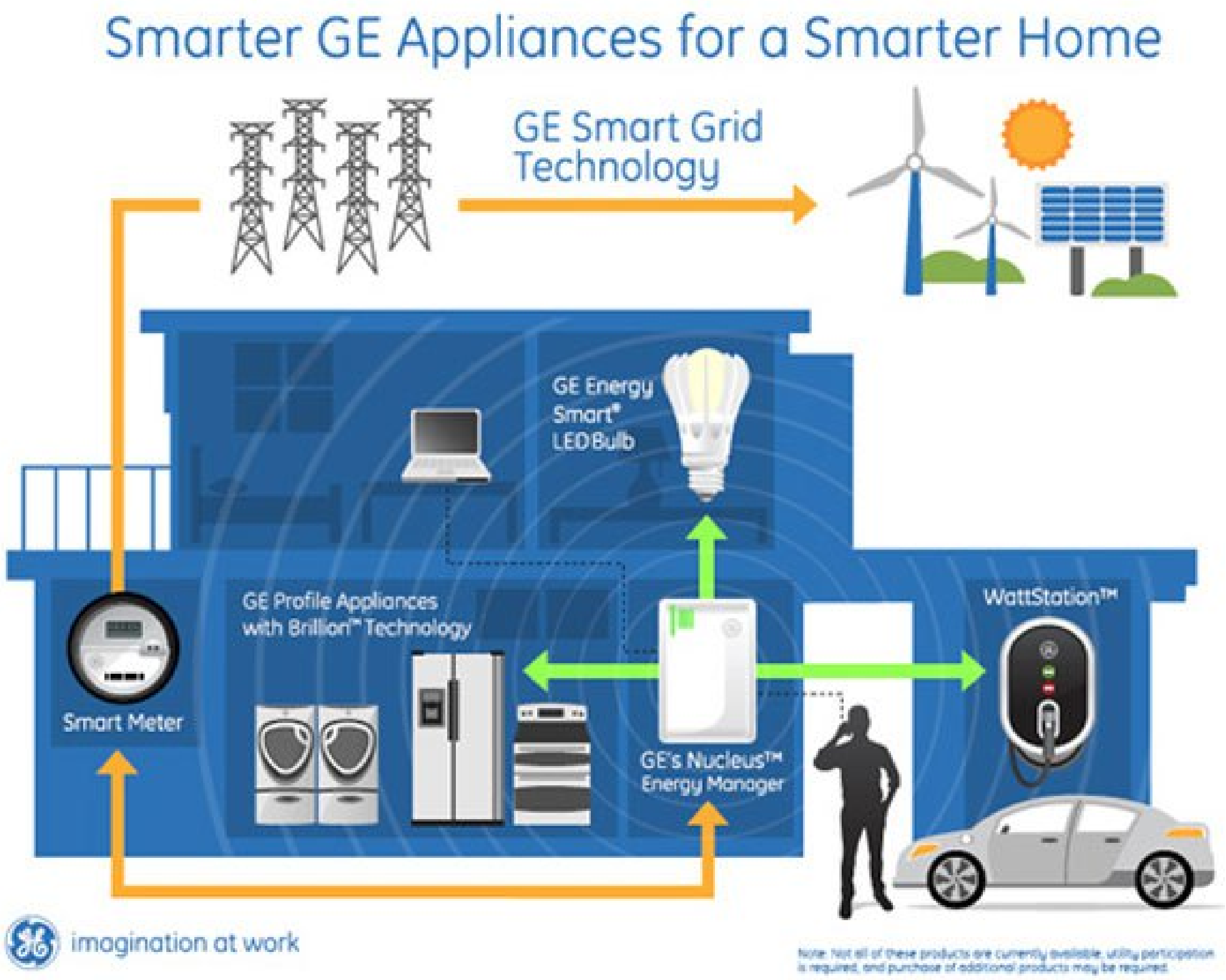


Figure 3.2: General Electric’s Nucleus energy manager. This represents GE’s vision of the smart home back in 2010. Source: Dahl 2011.

Figure 3.3: General Electric's Nucleus iPhone application showing energy usage for a single smart appliance. Source: General Electric 2012.

both System Architect at GE Energy *and* Vice Chair of the Smart Energy Profile 2.0 working group (Simpson 2013). Also, Dr Kannan Tinnium (Technology Leader, Electrical Technologies & Systems, GE Global Research) gave predictions for trends in 2016 (Electronics Maker 2016):

> "*We will see more and more integration of smart grids with renewable power sources such as wind and solar energy, pursuing the primary agenda of clean environment. Although the wind and solar energy sources that are derived from the environment are infinite, these sources are not under human control. When we have such a high penetration of renewables on the grid, there exist operational challenges such as intermittent supply and unpredictable availability of these energy sources. Integration of renewable energy into the smart grid, powered by innovative technology solutions for energy storage will be fundamental in addressing these challenges and achieving greater degrees of reliability and consistency in delivery.*"

3.1.2 Low-cost appliance power monitors ('smart plugs')

If appliances do not measure their own power consumption then their power consumption could be measured be an appliance power monitor. Power meters which measure the power demand of one or a small number of appliances have been available for several years. See Table 3.1 for a list of power monitors and see U.S. Energy Information Administration (EIA) 2015, pages 26–28 for a table of additional meters and details.

Kazmi et al. 2014 provides a recent review of appliance level energy monitoring using wireless sensor networks (WSNs).

Conventional individual appliance monitors (IAMs) are rather expensive and bulky to install in an average home. These limitations motivated DeBruin et al. 2015 to create the PowerBlade[1] meter which is small and cheap (£7.70). For commercial applications, Doyle et al. 2015 developed a small embedded energy sensor.

Even at only £7.70 per sensor, it would still be expensive to instrument *every* appliance in a house. For example, House 1 in UK-DALE has 54 appliances hence would cost £7.70 $\times$ 54 = £416 to measure every appliance using the PowerBlade. It might be more sensible to measure, say, the top-five energy consuming appliances in the home. But how would the user find the top-five energy using appliances without measuring every appliance?

An additional challenge with installing many sensors is that those sensors use energy and, with enough sensors, the energy consumption of the sensors might outweigh the energy reductions associated with having disaggregated energy data. For example, Louis et al. 2016 found that a full home energy management system (including individual sensors for many appliances) would result in an *increase* in electricity consumption for a one-person house of around 15%, largely due to the standby power consumption of the sensors and the home energy management system itself.

Smart plugs have been available for several years but market penetration appears to be low (although I could not find exact numbers). It is perhaps significant that EDF Energy discontinued their entire EcoManager range around 2015.

A lack of demand for smart plugs is both good and bad news for NILM. It is good news because

[1] `http://lab11.eecs.umich.edu/projects/powerblade`

Table 3.1: Example individual appliance meters (also called 'smart plugs'). US Dollars and Euros have been converted to GB Pounds at conversions rates of $1:£0.70 and €1:£0.78. Prices are per appliance meter, excluding the wireless base station.

Model	Manufacturer	Price (£)	Used to record	Sample period (secs)	Network connection
EcoManager Transmitter Plug[a]	EDF & Current Cost	15	UK-DALE	6	433 MHz TRX
IAM[b]	Current Cost	13	-	6	433 MHz TX
Kill A Watt[c]	P3 International	16	-	N/A	none
Circle[d]	Plugwise	28	Tracebase	1	ZigBee mesh
HomeMatic Radio Plug[e]	eQ-3	39	-	5 (best[f])	868 MHz TRX
PowerPort[g]	Enmetric	[n]	REDD	1	IEEE 802.15.4 wireless
nPlug[h]	IBM Research India	105	iAWE	1	Wi-Fi
PowerScout[i]	DENT Instuments	[n]	AMPds	60	serial or Ethernet
eGauge meters[j]	eGauge	350	Dataport	1	Ethernet or PLC or Wi-Fi
'Watts Up?' meters[k]	'Watts Up?'	100	Kelly 2011	1	Ethernet
WiFiPlug[l]	WiFi Plug	45	-	[n]	Wi-Fi
PowerBlade[m]	EECS at Uni. of Michigan	7·7	-	<1	BLE

[a] The EDF EcoManager products appear to have been discontinued some time around 2015.
[b] www.currentcost.com/product-iams.html
[c] www.p3international.com/products/p4400.html
[d] www.plugwise.com/products/energy-management/energy-meters-and-switches. 1 Hz sample rate only possible by polling Circles from custom software (as per Tracebase). PlugWise report [private communication] that the highest sample rate for logging using their Source software is 1 minute.
[e] www.eq-3.de/produkt-detail-aktoren/items/homematic-funk-schaltaktor-1-fach-zwischenstecker.317.html
[f] The HomeMatic gateway can only sustain a sample period of 5 seconds to a small number of sensors.
[g] www.enmetric.com/platform#Hardware
[h] Ganu et al. 2012. The nPlug is a research prototype. iAWE used a variant of nPlug called jPlug.
[i] www.dentinstruments.com/power-meter. The PowerScout is mainly for monitoring *circuits*.
[j] www.egauge.net/products
[k] www.wattsupmeters.com/secure/products.php?pn=0
[l] www.wifiplug.co.uk
[m] lab11.eecs.umich.edu/projects/powerblade and see DeBruin et al. 2015
[n] Information not available on manufacturer's website.

it means that NILM can provide information which is not yet available to users. It is bad news because it suggests that users are not very interested in disaggregated energy data! Perhaps the small proportion of the population who are especially interested in disaggregated energy data have already satisfied their hunger for data by buying smart plugs!

3.1.3 Networking for the smart home

There a many competing, incompatible wireless networking standards for the smart home.

‘Conventional’ Wi-Fi (IEEE 802.11a/b/g/n/ac) is very power-hungry and so is not suitable for battery-powered nodes. The most recent wireless standard for the internet of things is a version of Wi-Fi designed for the internet of things: the IEEE 802.11ah standard and its HaLow™ extension[2] (pronounced ‘halo’). It uses the unlicensed 900 MHz band, has a range of up to 1 km, good penetration through walls, low energy requirements and each access point will support thousands of HaLow devices. Data transfer speeds are 150 kbps to 18 Mbps and official product certification from the Wi-Fi Alliance will not begin until 2018, although uncertified products may appear earlier. Future Wi-Fi access points will likely be tri-band (900 MHz, 2.4 GHz and 5 GHz) and HaLow devices will be able to communicate directly to the home’s Wi-Fi access point (and hence to the internet). It is this ability of HaLow devices to connect directly to the internet that is a major advantage of HaLow over competing standards.

An alternative to HaLow is the IEEE 802.15.4 specification which defines the physical (PHY) and media access control (MAC) layers for low-cost radio communications which operates in one of three frequency bands (868/915/2450 MHz) depending on the region. Data transfer rates vary from 20 to 250 kbit/second. 802.15.4 only defines the PHY and MAC layers. Additional layers are required in order to define a full networking stack.

‘ZigBee’ is built on top of the IEEE 802.15.4 PHY and MAC layers. ZigBee defines the network layer and application layer to construct an open, wireless mesh-networking standard. One aim of ZigBee is to be simpler and less expensive than Bluetooth or Wi-Fi (although the new Wi-Fi HaLow competes more directly with ZigBee). ZigBee was first developed in 1998, standardised in 2003 and revised in 2006. Transmission distances are limited to 10-100 meters, depending on

[2] http://www.wi-fi.org/discover-wi-fi/wi-fi-halow

the power output and local conditions. Longer transmission distances are achieved by sending the message over multiple 'hops'.

'Thread' is another mesh networking standard built on the IEEE 802.15.4 physical networking layer (PHY) and medium access control (MAC). Thread's higher layers include 6LoWPAN (for using IPv6 in low-power, low-bandwidth devices). Thread includes AES encryption and, apparently, is sufficiently energy efficient to allow for 'sleepy' nodes to run for 'years' on a single AA battery. Google's Nest products already use Thread and many other organisations are in the Thread Group[3], including ARM, Analog Devices, Atmel, D-Link, Google, HTC, Huawei, Intel, Johnson Controls, LG, Logitech, Microsoft, Osram, Philips, Qualcomm, Samsung, Schneider Electric, Whirlpool and others. Thread was launched around 2014.

Z-Wave is a proprietary mesh protocol which operates around 900 MHz and is designed for low-energy transmission at data rates up to 100 kbit/s. The first Z-Wave products appears in the early 2000s and the Z-Wave Alliance was formed in 2005.

Bluetooth Low Energy (BLE), marketed as 'Bluetooth Smart', is a point-to-point wireless personal area network designed for low-energy and low-cost applications. It was first developed in 2006 by Nokia under the name 'Wibree' and became part of the Bluetooth Core Specification Version 4.0 in 2010.

DECT-ULE is an open wireless protocol based on the original Digital European Cordless Telephony (DECT) voice protocol operating at 1.9 GHz. The ultra-low energy (ULE) variant of DECT appeared in 2011 for home automation products.

X10 was the first general purpose home automation protocol, developed in 1975. It originally used power line wiring for signalling. A wireless (RF) protocol was later developed, operating at 310 MHz in the U.S. and 433 MHz in Europe.

Insteon is a home automation mesh networking for communication using power line networking or radio frequency (RF) comms, or both. Insteon products were launched in 2005 by Smartlabs, Inc. Insteon is X10-compatible.

WeMo, a brand owned by Belkin, is not a wireless networking standard in its own right but instead piggybacks on top of Wi-Fi.

[3] http://www.threadgroup.org/ABOUT/Our-Group

As well as the above standards, there are a myriad of proprietary wireless networking protocols.

Wi-Fi is emerging as dominant for *mains powered* smart appliances.

As outlined above, there are many networking standards for the internet of things (IoT) and it is not yet clear which (if any) standard will become dominant for *battery*-powered nodes. However, there are signs that 'conventional' Wi-Fi (IEEE 802.11a/b/g/n) is becoming the dominant networking standard for appliances which are plugged into mains power (because Wi-Fi's power requirements are not a problem when you have mains power). All the smart appliances on the market in early 2016 that I researched use Wi-Fi (this is in contrast to, for example, GE's Nucleus appliances back in 2010 which used ZigBee Smart Energy Profile).

If appliance manufacturers converge on Wi-Fi instead of ZigBee then this raises the question of whether the smart meter (which uses ZigBee Smart Energy Profile, not Wi-Fi) will remain at the centre of the smart home (for example, the UK Government in documents such as Richards et al. 2014 shows the smart meter as being at the heart of the smart home energy management system). How will Wi-Fi enabled appliances receive demand response signals or energy pricing signals if they are not attached to the smart meter's ZigBee network? Technically speaking, demand response and pricing signals can be sent over the internet and to appliances over Wi-Fi, perhaps using Smart Energy Profile 2 (see Section 3.1.3). But who will supply this data? The appliance manufacturers or the utility companies or the government? And how will appliances know which energy supplier to query to get relevant pricing information? I have emailed manufacturers to ask this question but have not received a reply.

A common API for energy

Standardising a way for appliances to exchange *bits* is only part of the challenge. Appliances also need the know what those bits *mean*: appliances need a standard 'language' to structure communications about energy consumption.

The ZigBee Alliance define a suite of application-layer protocols. This suite includes a 'Smart Energy Profile (SEP)'. The UK's SMETS2 smart meters (DECC 2014) will use ZigBee Smart Energy Profile (SEP) 1.2. In the USA, more than 40 million ZigBee electric meters are being deployed by more than 11 utility companies (ZigBee Alliance 2013).

SEP2 (ZigBee Alliance & HomePlug Alliance 2013) (standardised in 2013 as IEEE 2030.5-2013[4]) structures communication about energy usage information, energy pricing, demand response and load control, control of plug-in electric vehicle charging, distributed energy resources and more. SEP2 can run over *any* Internet Protocol (IP) network: it is not tied to ZigBee link layers. The Zibee Alliance worked with many partners to design SEP2 including the Wi-Fi Alliance and the HomePlug Alliance. SEP2 has been demonstrated over Wi-Fi, Ethernet and IEEE 1901 'broadband over power lines' networks (ZigBee Alliance & HomePlug Alliance 2013). At the time of writing, the Consortium for SEP2 Interoperability (CSEP[5]) are in the process of setting up a testing and certification body for SEP2. As time progresses, we are likely to see 'SEP2' branding less and less as the standard distances itself from its former home (the ZigBee Alliance) and emphasises its new home (IEEE) (Robby Simpson, personal communication, February 2016).

From 2009 to 2013, the Korea Smart Grid Association (KSGA) ran a smart houses test-bed project in Jeju Island (South Korea) with 1500 houses and more than 60 companies including LG and Samsung. They used SEP1.1 to test demand response and real-time pricing. More recently, KSGA have looked at using SEP2 with appliances connected directly to the internet via Wi-Fi (Kang & Park 2014).

SEP2 defines 21 categories of appliance such as 'pool pump', 'electric vehicle' and 'smart appliance' but SEP2 does not define a fine-grained controlled vocabulary for domestic appliances (e.g. 'kettle', 'toaster' etc.). Hence one manufacturer's washing machine may identify as 'washer' and another manufacturer's machine may identify as 'WM'. This lack of standardisation will make it harder to build energy management tools. Nordman & H. Y. Cheung 2013 wrote a draft appliance taxonomy and my NILM Metadata schema also defines a common appliance vocabulary (see chapter 6). SEP2 uses the Common Information Model (CIM) IEC 61968 / 61970 as its semantic model (Simpson 2013) but CIM appears to be mostly concerned with the power network *upstream* of the domestic consumer (McMorran 2007).

We are starting to see SEP2 adopted in the distributed energy resources (DER) space; for example in solar inverters and battery storage systems. This is particularly true in California with Rule 21[6] (for smart inverters) and in Hawaii (Robby Simpson, personal communication,

[4] https://standards.ieee.org/findstds/standard/2030.5-2013.html
[5] http://www.csep.org
[6] http://www.energy.ca.gov/electricity_analysis/rule21

February 2016; also see St. John 2015a).

SEP is not the only standard available for energy. Other standards include ANSI/CTA-2047 (Consumer Technology Association 2014) which enables "*consumer electronic devices to communicate their energy usage information for example over a home network as well as optionally respond to basic demand/response commands. The usage data may be a measured or estimated value or may use other methods to indicate energy usage*" and OpenADR (OpenADR Alliance 2016) which attempts to standardise, simplify and automate demand response.

The Internet Engineering Task Force (IETF)'s RFC6988 document on "Requirements for Energy Management" (Quittek et al. 2015) defines the requirements for standards specifications for energy management, mostly focused on networked computer systems. It is a standard to allow routers, servers etc. to report their power consumption, power state and other information over an IP network.

Too much diversity?

In the last two years year, the big manufacturers have defined their own *general* smart home 'frameworks': Google has Brillo[7] (with three components: an embedded OS based on Android, core services and a developer kit) and Weave[8] (a communication platform for IoT devices). Electrolux have announced that they will work with Brillo (Thompson 2016).

Samsung has the Artik[9] platform, Apple has HomeKit[10], ARM has mbed[11], Amazon has AWS IoT[12]. Microsoft is backing AllJoyn from the AllSeen Alliance[13], a consortium that proposes an open protocol for IoT and includes Philips, Sony, Faber, HoneyWell, HTC, and LG.

Hartog et al. 2014 review 43 semantic assets (frameworks) for smart appliances.

This diversity means that it is unclear which framework (if any) will become dominant. At the Consumer Electronics Show 2016 (CES), Frank Gillet (an analyst at Forrester) said of smart home networking that "*It's going to be extremely messy*" and Lee Ratliff (an analyst from

[7] `https://developers.google.com/brillo`
[8] `https://developers.google.com/weave`
[9] `https://www.artik.io`
[10] `https://developer.apple.com/homekit`
[11] `https://www.mbed.com`
[12] `https://aws.amazon.com/iot`
[13] `https://allseenalliance.org`

Figure 3.4: Smart homes can make life *more* complex. Source: Cate 2016 (with permission).

IHS) said "*I'm afraid we're looking at many, many, many years of many standards, consumer confusion, market confusion*" (Shankland 2016).

3.1.4 Conclusions regarding smart homes

Will smart appliances and smart plugs make NILM obsolete any time soon?

It has been technically possible for several years to build a smart home which can measure the power demand of each appliance. For example, Sweden recently announced an urban development with 150 new apartments all equipped with smart appliances (Electrolux 2015) and E.ON ran a smart homes trial in Milton Keynes from 2011-2015 called 'Thinking Energy' (E.ON 2014).

But the fact that it is *technically possible* to measure power demand with smart appliances does not mean that smart homes will be *widespread* imminently. There are several reasons to believe that energy-focused smart homes are many years away from widespread adoption:

Expense: Smart plugs and smart appliances are several times more expensive than their 'dumb' counterparts.

Low replacement rate: Household appliances are replaced slowly. The average age of major domestic appliances in Germany is 13.5 years (Gutberlet 2008).

Uncertain demand: Is there demand for 'smart' appliances? The concept of the 'smart' home can be traced back to the 1930s (Strengers 2013). The first commercial attempts at a 'smart fridge' came with Electrolux's Screenfridge (Electrolux 1999) and LG's 'Internet Digital DIOS' fridge (LG Electronics 2000). Both were a commercial flop (Baxter 2010). In fact, internet-enabled appliances have been somewhat of a running joke in some quarters[14]. Are users willing to pay large price premiums for appliances which introduce new security, privacy and reliability issues (Bright 2014; Fox-Brewster 2016), are more complex to configure and nag users via their phones? Traditional 'dumb' appliances just quietly get on with their job, cannot be hacked and do not crash! See Figure 3.4 for a recent example of a smart home making life *more* complex. Do users care about the 'benefits' of smart appliances publicised by manufacturers? Or do users see many of these 'benefits' as *solutions looking for problems*? Perhaps many users *do* have an appetite for these products given that, for example, in 2015 over 8 000 people pledged $600 000 (USD) to the KickStarter project HidrateSpark[15]: "*a connected water bottle that tracks your water intake and glows to make sure that you never forget to drink your water again*". I would suggest that four billion years of evolution has already solved the remembering-to-drink-water problem.

Fragmentation: There are *many* competing, incompatible networking protocols, APIs and frameworks for the smart home. Will users be happy to commit to one platform if it might be obsolete soon, especially given the large financial costs involved?

Motivation: Perhaps manufacturers are not interested in energy applications. Hence, even if smart appliances become widespread soon, appliances might not report their power demand. We cannot know the motivations of manufacturers for certain but we can speculate that traditional whitegoods manufacturers are trying to find a way to increase their profits in a stagnating market. Advertising companies such as Google are hungry to acquire as much data about users as possible to allow them to target adverts. Apple has profited handsomely in the past from locking users into its ecosystem so why not try to

[14] See the (rather rude) Tumblr blogs 'F**k Yeah Internet Fridge' and 'We put a chip in it' whose tag line is "*It was just a dumb thing. Then we put a chip in it. Now it's a smart thing.*"

[15] `http://www.hidratespark.com`

> lock users into Apple-compatible smart home products too? And Samsung is looking for a way to compensate for stagnating sales in smart phones and TVs. Perhaps manufacturers believe that consumers do not prioritise energy efficiency and hence manufacturers have little interest in prioritising energy efficiency in their consumer-facing products.

In conclusion, it is impossible to predict with confidence when (if ever) smart homes will give users disaggregated energy data. But we can put a lower limit on the amount of time it will take. Even if affordable, stable, secure, *compatible*, energy-aware smart appliances became available tomorrow, and if the general public showed demand for these products, it would still take over a decade for the population to replace their existing appliances with new appliances (Gutberlet 2008).

Smart appliances do not appear to present an imminent threat to NILM. As smart appliances trickle into homes, they can be integrated into NILM, as shown by Egarter et al. 2015 who built an energy management system which integrates smart appliances with NILM.

Concerns over smart meters

A vocal minority hold concerns about the privacy, health impacts and financial cost of the smart grid (Raimi & Carrico 2016). NILM potentially provides even more of a threat to privacy. Indeed, George Hart (G. W. Hart 1992) discussed the privacy concerns about NILM.

Barbosa et al. 2015 developed techniques to *fool* NILM techniques in order to preserve privacy.

Skjølsvold & Ryghaug 2015 provides evidence of people feeling alientated by the smart grid. A recent review of the British public's perception of the UK smart metering initiative is provided by Buchanan et al. 2016.

Utility companies are reportedly concerned about the large volume of data that smart meters will generate. Hence Tariq et al. 2015 adapted the LZMA compression algorithm to smart meter data and were able to reduce the data size by 98% on average.

3.2 Conclusions

Let us summarise the challenges facing NILM:

- There is little robust evidence that NILM helps people to save more energy than simpler forms of feedback such as IHDs (see chapter 2).
- NILM (with sample rates < 100 Hz) cannot disaggregate vampire loads (unless population averages are used), yet minimising vampire loads plays an important role in energy saving.
- How much 10-second data (or faster) will be available? UK SMETS2 smart meters will allow users to collect 10-second data using a consumer access device (CAD) but how many users will actually buy and install CADs? NILM can still be performed on the half-hourly data that smart meters will send to the utility company but the estimates will be considerably less accurate than if 10-second data were used for NILM.
- Smart appliances or smart plugs will, at some point, make NILM redundant. We cannot be sure when smart plugs and smart appliances will be adopted on a wide scale. It could be 10-20 years from now.
- We have no way to measure progress in NILM algorithm development (see chapter 5).
- Some people have negative feelings towards smart meters and NILM.
- At current energy prices, the energy cost of a typical appliance activation is low (pence), which may lead users to think that they should not bother to reduce their energy consumption.
- Some NILM companies are selling NILM on the basis of questionable 'evidence' on the efficacy of NILM to help people save energy. If it turns out that the actual energy savings are considerably lower then the NILM market might react badly.

And let us summarise the advantages of NILM:

- NILM *may* drive substantial energy savings in the general population. We simply do not know with confidence! (See chapter 2)

- There are *many* unanswered questions about the efficacy of disaggregated feedback to promote energy reductions (see chapter 2). NILM offers the most cost-effective technique to investigate these questions.
- Directly driving energy savings is not the only use-case for NILM: There are many other use-cases, some of which indirectly lead to energy savings (see section 1.3.5).

Part II

Tools to enable disaggregation research

Chapter 4

The UK domestic appliance-level electricity dataset (UK-DALE)

To conduct research on disaggregation algorithms, researchers require data describing not just the aggregate demand per building but also the 'ground truth' demand of individual appliances. In this context, we present UK-DALE: an open-access dataset from the UK recording Domestic Appliance-Level Electricity at a sample rate of 16 kHz for the whole-house and at $1/6$ Hz for individual appliances. This is the first open access UK dataset at this temporal resolution. We recorded from five houses, one of which was recorded for 3.5 years, the longest duration we are aware of for any energy dataset at this sample rate. We also describe the low-cost, open-source, wireless system we built for collecting our dataset.

This chapter is based on our paper published in Nature Publishing Group's journal 'Scientific Data', Kelly & Knottenbelt 2015b. Please also see Appendix A for additional details.

4.1 Introduction

Energy disaggregation researchers require access to large datasets recorded in the field to develop disaggregation algorithms but it is not practical for every researcher to record their own dataset. Hence the creation of open access datasets is key to promote a vibrant research community.

Researchers at MIT led the way by releasing The Reference Energy Disaggregation Data Set (REDD) in 2011 (J. Zico Kolter & M. J. Johnson 2011) and more datasets have subsequently been released.

To test the performance of a disaggregation algorithm for a specific country, it is important to have access to data from that country because electricity usage varies significantly between countries; both because different countries use different sets of appliances and also because different cultures show different usage patterns.

When we set out to record UK-DALE, the only open-access dataset recorded in the UK was the DECC/DEFRA Household Electricity Study (Zimmermann et al. 2012) which has a sample period of two minutes. This sample rate is 12 times slower than UK smart meters, which will sample once every 10 seconds (DECC 2014). It is this smart meter data which will provide the input to disaggregation algorithms so researchers require access to 10 second data to design disaggregation algorithms for the UK (some other countries will also use smart meters with similar sample periods). Since we released UK-DALE, a third UK dataset has been released: the REFIT dataset (Murray 2015), which samples once every 6-8 seconds.

We present the first open access UK dataset with a high temporal resolution. We recorded from five houses. Every six seconds we recorded the active power drawn by individual appliances and the whole-house apparent power demand. Additionally, in three houses, we sampled the whole-house voltage and current at 44.1 kHz (down-sampled to 16 kHz for storage) and also calculated the active power, apparent power and RMS voltage at 1 Hz. In House 1, we recorded for 3.5 years and individually recorded from almost every single appliance in the house resulting in a recording of 54 separate channels (although less channels were recorded towards the start of the dataset). We will continue to record from this house for the foreseeable future. We recorded from the four other houses for several months; each of these houses recorded between 5 to 26 channels of individual appliance data. Figure 4.2 provides an overview of the system design and Table 4.1 summarises the dataset.

This dataset may also be of use to researchers working on:

- modelling the electricity grid.
- exploring the potential for automated demand response.

- appliance usage behaviour.

4.2 Methods

Desirable attributes for a disaggregated electricity dataset include:

- Simultaneously record the power demand of each individual appliance in each house; or as many individual appliances as possible. This data can be used to validate the appliance-by-appliance estimates produced by a disaggregation system or to train the system.
- Record the whole-house active power. This will be the input to the disaggregation algorithm.
- Sample once every 10 seconds or faster.
- Record for as long as possible.

We first describe our approach to monitoring individual appliances once every 6 seconds and then describe how we recorded whole-house mains power at 44.1 kHz.

4.2.1 Individual appliance monitoring

In UK houses such as those in our dataset, mains 'rings' extend from the fuse box. Many sockets may share the same ring. Hence, in order to measure individual appliances in the UK, we must install plug-in individual appliance monitors (IAMs) between each appliance and its wall socket (see Figure 4.1).

We used EcoManager Transmitter Plugs developed by Current Cost and distributed by EDF Energy (see Figure 4.3). The standard base station for these IAMs is the EcoManager[1]. The EcoManager can only handle a maximum of 14 transmitter plugs and only provides data once per minute via its serial port. We needed up to 54 appliances monitored per house and data every 10 seconds or faster. To achieve this, we built our own base station. With the help of

[1] As of 2016, the EcoManager base station and transmitter plugs are no longer available to purchase.

Figure 4.1: Two EDF Transmitter Plugs installed in our kitchen.

others in the community (especially Graham Murphy, Matt Thorpe and Paul Cooper), we reconstructed the specification of the EcoManager protocol. The Current Cost RF protocol and the EDF EcoManager RF protocols are available on the GitHub wiki for my `rfm_edf_ecomanager` project[2].

We built our own base station by programming an open-source, rapid-development platform called the Nanode[3] (see Figure 4.3). The Nanode includes an Atmel ATmega328P microcontroller running at 16 MHz (the same microcontroller used on several Arduinos) and a HopeRF RFM12b[4] radio frequency (RF) module. The EcoManager products use the same (or similar) RF module as the Nanode, tuned to the 433 MHz ISM band (note that, in some countries, it is illegal to use the 433 MHz band without a license). Please see Appendix A.1 for the details of how we discovered the wireless configuration parameters.

Each IAM picks its own 32-bit ID at random when the IAM's 'pair' button is pressed. Each IAM stores its ID in non-volatile memory. Our base station maintains a list of these IDs and polls each IAM in order. Each IAM replies to its polling packet within 20 ms. The power demand measurement in this packet may be a few seconds old because the Transmitter Plugs

[2]https://github.com/JackKelly/rfm_edf_ecomanager/wiki
[3]http://www.nanode.eu
[4]http://www.hoperf.com/rf_transceiver/modules/RFM12B.html

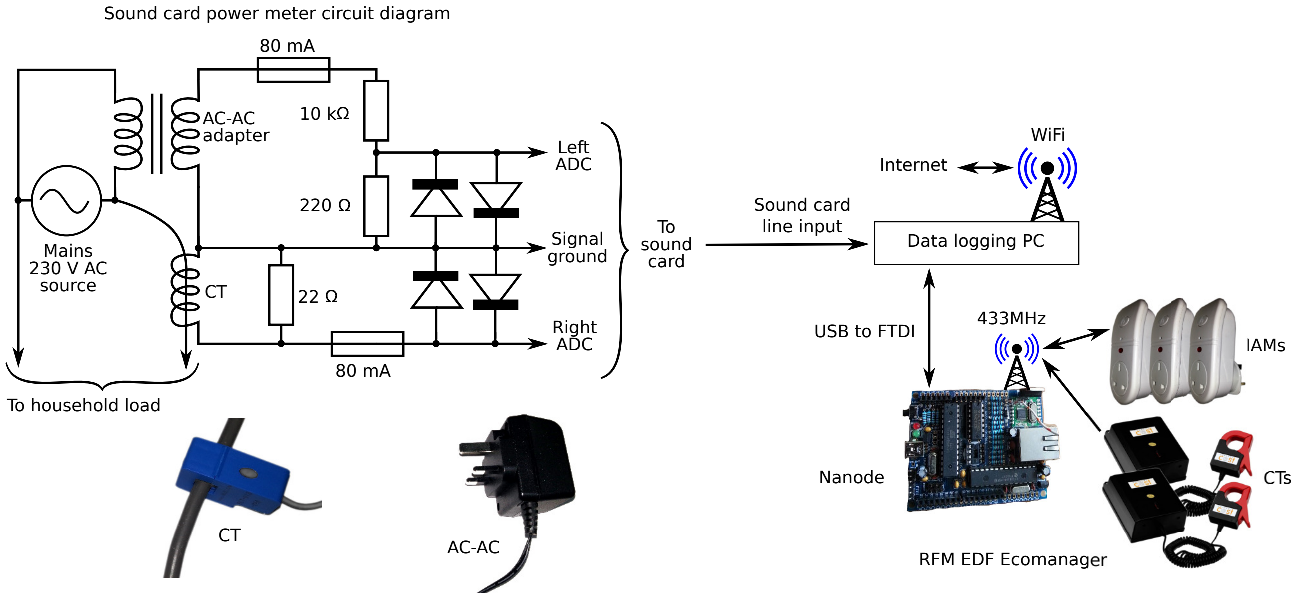


Figure 4.2: System diagram for the data collection system. The system has three major components: 1) the data logging PC; 2) the sound card power meter and 3) the 'RFM EDF Ecomanager' which uses a Nanode to communicate over the air with a set of individual appliance monitors (IAMs) and current transformer (CT) sensors. On the left is the circuit diagram for interfacing a sound card to a CT clamp and AC-AC adaptor to measure mains current and voltage, respectively. The circuit was adapted from Robert Wall's work (Wall 2012). Each diode is a 1N5282 (1.3 V forward voltage bias).

appear to sample the power demand every few seconds and store the last measurement in local memory. Measuring the power demand and responding to polling requests appear to happen asynchronously in the Transmitter Plugs.

The EcoManager RF protocol uses a modular sum checksum byte to provide some resilience against RF corruption. Power measurements are sent from our Nanode base station to a data logging PC over an FTDI-to-USB cable. It is also possible to turn IAMs on or off remotely.

To measure power demand from hard-wired appliances such as boilers and kitchen ceiling lights, we used Current Cost transmitters[5] (TX) with current transformer (CT) clamps. These transmitters use the same radio frequency as the EDF IAMs but a different protocol. In particular, the Current Cost transmitters cannot *receive* RF data. Instead they transmit a data packet once every 6 ± 0.3 seconds without first checking if the RF channel is clear. Hence RF collisions are inevitable and there is no mechanism to request re-transmission of lost data.

To minimise the chance of packet collisions, our base station learns the transmit period of each Current Cost TX and does not transmit for a short window of time prior to the expected arrival

[5] http://www.currentcost.com/product-iams.html

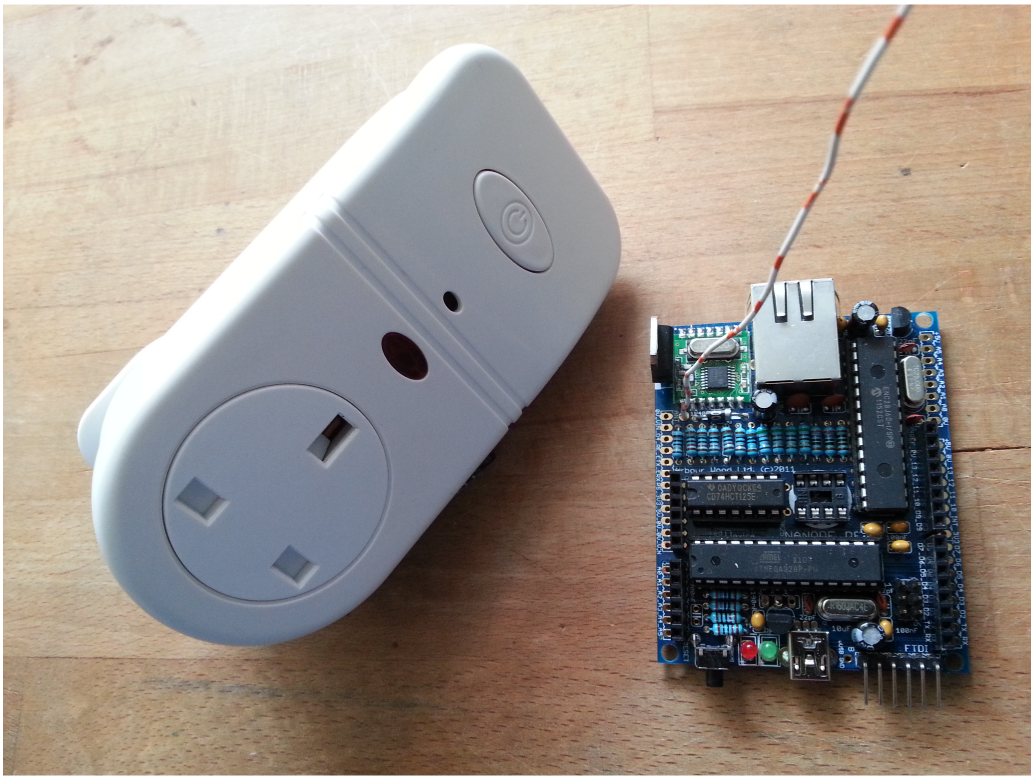


Figure 4.3: An EDF EcoManager Transmitter Plug (left) and a Nanode (right).

of a Current Cost TX packet.

Current Cost transmitters do not use a checksum. Instead they use Manchester encoding. RF corruption may result in an invalid Manchester code. If this happens then the receiver can detect the corrupt Manchester code and discard the packet. Unfortunately, corruption may damage the data payload *without* invalidating the Manchester encoding, hence corrupt packets are not *guaranteed* to be detected from Current Cost transmitters.

4.2.2 Measuring whole-house power demand

Given that there will be a flood of smart meter data in the near future, disaggregation researchers need access to data which is as similar as possible to the data that will be recorded by smart meters. Unfortunately, 'real' smart meters are not trivial to install or to acquire: installation requires an electrician from the utility company and, critically, back in 2012 when we were designing our data collection system, the UK smart meter engineering specifications had not yet been finalised (the SMETS2 document (DECC 2014) is not an engineering specification). This motivated us to build our own measurement system.

The iteration of the UK smart meter specification (DECC 2014) available when we started designing our system was detailed enough to allow us to build our own metering system which closely mimics what a UK smart meter is likely to provide[6]. The SMETS2 specifications (DECC 2014) state that smart meters are required to connect to the Home Area Network (HAN) using ZigBee Smart Energy Protocol v1. Presumably, a disaggregation system would access smart meter data by way of a 'Consumer Access Device' (CAD) connected to the HAN. CADs can request instantaneous active power and pricing data from the smart electricity meter once every 10 seconds[7].

We aimed to build a metering system that would collect active power once a second, as well as to sample the voltage and current waveforms at 44.1 kHz to allow researchers who are interested in this high frequency data to use it.

One solution was to use an off-the-shelf Current Cost whole-house transmitter with a current transformer (CT) clamp. These work with our wireless base station. We used this solution in several houses where our bespoke solution was impractical.

However, there are several disadvantages to using a CT clamp connected to a wireless transmitter (see Appendix A.2). As such, no existing home energy monitor that we were aware of provided an accurate proxy for UK smart meters. Expensive power quality monitors costing several hundred or thousand UK pounds can measure with the accuracy we require but these are prohibitively expensive and some require CT sensors *without* a split core, hence requiring the installer to disconnect the meter tails from the utility company's meter, which can only be done with permission from the utility company.

Hence we designed and built a low-cost, high resolution, easy to install system for recording whole-house mains power demand using a computer sound card, a CT clamp and an AC-AC adapter.

Typical sound cards have remarkably good analogue to digital converters (ADCs). Typical specifications of a modern sound card include:

[6]These specifications are subject to formal change control processes such that any changes are subject to analysis and stakeholder approval

[7]Additionally, CADs will be able to request data such as 13 months of half hourly active import data, 3 months of half hourly reactive import data and 3 months of half hourly active and reactive export data (DECC 2014 and personal communication with DECC, October 2013)

- 96 kHz sample rate.
- Simultaneous recording of at least 2 channels.
- 90 dB signal to noise.
- 20 bits per sample. Not all of these bits are 'signal' though. For example, in the ADC we used, each bit provides 6 dB of dynamic range, so we effectively have $90/6 = 15$ bits of 'signal' and $20 - 15 = 5$ bits of 'noise' per sample.
- Built-in high-pass filter.
- Built-in anti-alias filter.

To record mains voltage and current waveforms we built a simple circuit to connect the sound card to an AC-AC adapter and a CT clamp (see Figure 4.2). This circuit does not require the user to handle any hazardous voltages. We used the line-input of the sound card rather than the microphone input because the line-input should provide a lower noise signal path than the sound card's microphone pre-amplifier. The standard maximum peak-to-peak voltage for consumer audio equipment line-input is 0.89 volts. Hence our circuit reduces the output voltage of each sensor so that we never deliver more than 0.89 volts to the sound card.

To measure mains voltage as safely as possible, we used a standard AC-AC adapter (the 'Ideal Power 77DB-06-09'). This provides a peak open-circuit output voltage of approximately 11 volts. Research done by the Open Energy Monitor project (Wall 2012) suggests that the output of the AC-AC adapter should track the mains input voltage linearly over the range 185.5 V to 253 V. We reduced the AC-AC adapter's output voltage with a voltage divider circuit (we used two resistors: 10 kΩ and 220 Ω) to produce about 0.7 V peak-to-peak which is fed into one channel of the sound card's line input.

To measure mains current, we used a current transformer (CT) clamp (the 'YHDC SCT-013-000'). Both the CT clamp and AC-AC adapter were sourced from the Open Energy Monitor shop[8]. The CT is connected in parallel to a 22 Ω burden resistor. This configuration produces about 0.89 V peak-to-peak across the burden resistor when the CT is presented with a primary current of 30 amps RMS which, we believe, is the most current that any of the houses under study will pull.

[8] https://shop.openenergymonitor.com

To protect the sound card against overload, both channels include an 80 mA quick-blow fuse and a pair of 1N5282 diodes (with a 1.3 V forward voltage bias) to ensure that the circuit is unlikely to ever deliver more than 1.3 V to the sound card.

Our system has a measurement resolution of 150 mW (please see Appendix A.3 for the calculation of the measurement resolution).

We now describe the software for our sound card power meter. We use the following relations to calculate $|S|$ (apparent power) and P ('real' or 'active' power) from simultaneously recorded vectors of voltage and current readings (we record in chunks each with a duration of 1 second; this time period was chosen because REDD uses this sample period for mains data):

$$|S| = I_{\text{rms}} \times V_{\text{rms}} \tag{4.1}$$

$$P = \frac{1}{N} \sum_{i=1}^{N} I_i V_i \tag{4.2}$$

Where I_{rms} and V_{rms} are the root mean squared values for the current and voltage vectors respectively; N is the number of samples; I_i and V_i are the i^{th} samples of the current and voltage vectors respectively. The system does not guarantee that we always process chunks of length equal to precise integer multiples of the mains cycle period but, as demonstrated in Section 4.4, we still achieve relative errors consistently less than 2%.

The conclusion is that we achieve a resolution greater than that required to provide a good proxy for 'real' smart meters[9]. We save P, $|S|$ and V_{rms} to disk once a second with a precision of 2 decimal places in a CSV file.

We also save the raw ADC data to disk. To reduce the space required, the ADC data are down-sampled using the open-source audio tool `sox`[10] to 16 kHz[11]. The ADC is 20-bit but few audio processing tools can process 20-bit files so we pad each sample to produce a 24-bit file. The uncompressed 16 kHz 24-bit files would require 8.3 GBytes per day so we compress the files using the Free Lossless Audio Codec (FLAC)[12] to reduce the storage requirements to

[9] although we acknowledge that we do not know the precise resolution of 'real' smart meters. This decision is likely to be left to the manufacturers (personal communication with DECC, March 2013)

[10] `http://sox.sourceforge.net`

[11] REDD used 15 kHz and we originally wanted to use the standard defined by REDD but we found that support for 16 kHz is more common than for 15 kHz in processing tools

[12] `http://xiph.org/flac`

≈ 5 GBytes per day.

After publishing our UK-DALE dataset, we found that other authors have used a sound card to record electricity data (Quintal et al. 2010; Nunes et al. 2011; Pereira 2011; Englert et al. 2013; Kahl et al. 2016).

Calibration

To convert the raw ADC values to voltage and current readings, we must first find appropriate conversion constants. We calibrate each data collection system separately to compensate for manufacturing variability in the components. We calibrate each system once when it is first setup. We connect a 'Watts up? PRO meter[13]' to the data logging PC via USB during setup to automatically calibrate voltage and current conversion factors. We typically use a resistive load like a kettle to calibrate the system. If the 'Watts up?' meter reports a power factor greater than 0.97 then the calibration script also calibrates the phase shift introduced by the sensors.

Open source implementation

We have implemented our power monitoring system as five software projects. All software packages are available from `github.com/JackKelly/`<`package name`>. The packages are:

rfm_edf_ecomanager
: Nanode C++ code. This code allows the Nanode to talk directly to multiple Current Cost whole-house sensors (CC TXs) as well as to multiple EDF Transmitter Plugs (CC TRXs). Users talk to the Nanode over the serial port. Users send simple commands. It sends data back to the PC in a simple JSON format.

rfm_ecomanager_logger
: A Python script for communicating with the `rfm_edf_ecomanager` Nanode system. This provides a command-line tool for 'pairing' sensors with the logging system; assigning human-readable names to those sensors and then recording the data to disk in CSV files using the same format as MIT's REDD files. The emphasis is on reliable logging. `rfm_ecomanager_logger` attempts to restart the Nanode if the Nanode crashes.

[13] `https://www.wattsupmeters.com/secure/products.php?pn=0`

`rfm_ecomanager_logger` ensures, as far as possible, that recorded time stamps are correct (which is not trivial given that the Nanode does not have a real time clock and given that serial data could be kept in the operating system's buffer if the system is under heavy load). Data are recorded approximately once every six seconds for each channel.

powerstats

Produce statistics and graphs from REDD-formatted power data. Mainly used for checking the health of sensors.

babysitter

A Python module for 'babysitting' each logging system. Sends an email if a sensor stops working or if `rfm_ecomanager_logger` fails. Also sends a 'heartbeat' email once a day to the home owner containing statistics (created by `powerstats`) describing the last day's power data. Also provides useful 'health' information about the system such as remaining disk space.

snd_card_power_meter

System for recording voltage and current waveforms at 44.1 kHz, 20-bit per channel using a PC's sound card. Calculates and saves active power, apparent power and RMS voltage to a CSV file once a second. Records down-sampled ADC data to a FLAC file.

4.2.3 Complete metering setup

To collect our own dataset, we installed the following equipment in each house:

- Multiple EDF Individual Appliance Monitors.
- A CurrentCost CT clamp and transmitter to measure whole-house apparent power. House 1 used additional CC CT clamps to measure the lighting circuit, kitchen ceiling lights, boiler and solar hot water pump.
- Nanode running our `rfm_edf_ecomanager` code.
- A small-footprint Atom PC[14]. We used the Intel DN2800MT motherboard with a Realtek

[14]Full component listing of the Atom PCs we built can be found at jack-kelly.com/intel_atom_notes and a guide to setting up a complete data logging system can be found at github.com/JackKelly/rfm_ecomanager_logger/wiki/Build-a-complete-logging-system

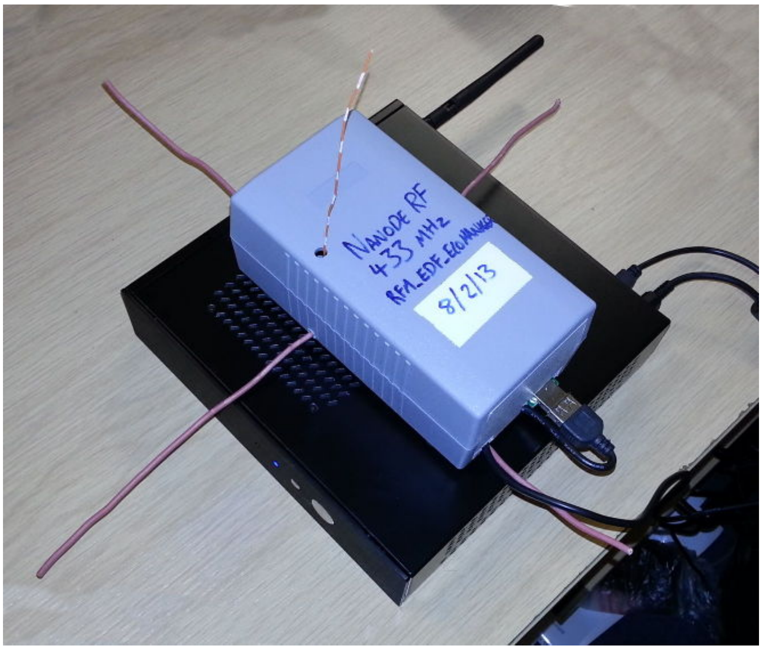


Figure 4.4: The data logging PC is the black case on the bottom. The Nanode is inside the box on the top. The four wires running in the plane of the ground are the RF ground plane. Each wire is 1/4-wavelength. The striped wire running upwards is the 1/4-wavelength antenna.

ALC888S audio codec capable of sampling at 96 kHz 20-bit resolution with a signal to noise ratio of 90 dB; and a line-input socket on the rear; and a 320 GB HDD; runs Ubuntu Linux Server; consumes 14 W active power.

- Houses 1, 2 and 5 had the sound card power meter system installed to measure whole-house active and reactive power and voltage.

A system diagram is shown in Figure 4.2. A Nanode and Atom PC can be seen in Figure 4.4.

CSV data files recorded by the data logging PC were transmitted to a remote server every morning using `rsync`[15]. FLAC files were transferred manually using an external hard disk every two months.

Please see Appendix A.4 for a list of known issues of the data recording system.

4.2.4 Selecting houses to record

The subjects were either MSc students or PhD students in the Computing Department at Imperial College London. The subjects chose to do a research project with the authors. To

[15] https://rsync.samba.org

assist in both their own project and in the collection of the UK-DALE dataset, the students kindly agreed to install metering hardware in their respective houses. The upper bound on the number of houses we could record from was set by a combination of a limited financial budget, limited time to assemble the metering hardware, and a limit in the number of students who volunteered to work on research projects relating to domestic energy consumption.

Within each house, the home owner selected which appliances to record, with the recommendation from the authors that the most energy hungry appliances should take priority.

We acknowledge that our participants are *not* sampled at random from the general population; and hence are unlikely to be representative of the general population.

4.3 Details of the recorded data

UK-DALE uses a data format similar to that used by the first public disaggregation dataset, the Reference Energy Disaggregation Data Set (REDD) (J. Zico Kolter & M. J. Johnson 2011).

There are five directories in UK-DALE, one per house. The directories are named `house_<x>` where x is an integer between 1 and 5.

Each directory contains a set of `channel_<i>.dat` CSV files (one file per electricity meter i) and a `labels.dat` file which is a CSV file which maps from channel number i to appliance name. All CSV files in UK-DALE use a single space as the column separator (as per REDD).

One way in which UK-DALE differs from REDD is that UK-DALE includes a set of detailed metadata files. These follow the NILM Metadata schema (see Chapter 6). The metadata files are in YAML text file format (YAML is a superset of the JSON format). This metadata describes properties such as the specifications of each appliance; the mains wiring between the meters and between meters and appliances; exactly which measurements are provided by each meter; which room each appliance belongs in etc. The `labels.dat` file in each directory is redundant and is only included to provide compatibility with REDD.

All data in UK-DALE as of January 2015 are available from the UK Energy Research Council's Energy Data Centre. The data are also available from `www.doc.ic.ac.uk/~dk3810/data`.

The latter source will be updated as we collect more data. There are three forms of data in UK-DALE:

- The 6 second data from the Current Cost meters (DOI:10.5286/UKERC.EDC.000001)
- The 1 second data from our sound card power meter (DOI:10.5286/UKERC.EDC.000001)
- The 16 kHz data recorded by our sound card power meter (DOI:10.5286/UKERC.EDC.000002). The complete set of 16 kHz files requires 6 TBytes of storage. The 16 kHz data is supplied as a set of 200 MByte files. Each file records 1 hour of data.

The 6 second data and 1 second data are stored in CSV files, one CSV file per meter. The first column is a UNIX timestamp (the number of seconds elapsed since 1970-01-01 00:00:00 UTC). The UNIX timestamp is UTC (Coordinated Universal Time) and hence ignores daylight saving transitions (the UK is `UTC+0` during winter and `UTC+1` during summer).

The 1 second data, 6 second data and metadata are also available as a single HDF5 binary file, ready for use with the open-source energy disaggregation toolkit NILMTK (see Chapter 5). The UK-DALE HDF5 binary file is available from `www.doc.ic.ac.uk/~dk3810/data` and from the UKERC EDC.

6 second data

For the 6 second data, the second column in each CSV file is a non-negative integer which records power demand of the downstream electrical load. The file names of the 6 second data take the form of `channel_<X>.dat` where `X` is a positive integer (with no leading zero). There are two types of 6 second resolution meters:

1. Individual appliance monitor transmitter plugs that record *active* power (in units of watts).
2. Current transformer meters that record *apparent* power (in units of volt-amperes).

Individual appliance monitors have a push-button switch to allow users to turn the connected appliance on and off. We record the activity of this switch in a `channel_<X>_button_press.dat` file. If the switch has just been toggled *on* then a '1' is recorded. If the switch has just been toggled *off* then a '0' is recorded. The motivation behind logging switch-press events is that these provide (imperfect) room occupancy information.

Switch *on* events should be a perfectly clean recording (i.e. the only possible reason for an on-switch event appearing in the data is that the user pressed the switch). Unfortunately, off-switch events may include false positives. Occasionally IAMs turn off spontaneously (an event which is impossible to distinguish from a genuine button press). Also, if power is lost and returned to the IAM within 12 seconds then this will be logged as an off-switch event. If the power is off for more than 12 seconds then the system assumes that the IAM was deliberately unplugged and hence the system will switch the IAM to its previous power state when it reappears; this automatic switch event is not recorded.

1 second data

There are four columns in each CSV file recording the whole-house power demand every second:

1. UNIX timestamp.
2. Active power (watts).
3. Apparent power (volt-amperes).
4. Mains RMS voltage.

All four columns record real numbers (not integers). The first column has one decimal place of precision; the other columns have two decimal places of precision. The 1 second data is in a CSV file called `mains.dat` in directories `house_1`, `house_2` and `house_5`.

16 kHz data

The 16 kHz data is compressed using the Free Lossless Audio Codec (FLAC). For houses 1, 2, and 5 UK-DALE records a stereo 16 kHz audio file of the whole-house current and voltage waveforms. The files are labelled `vi-<T>.flac` where T is a real number recording the UNIX timestamp with micro-second precision (using an underscore as the decimal place). This timestamp is the time at which the audio file began recording. The recordings are split into hour-sized chunks. We also include a `calibration.dat` file for each house. This is a text file specifying the multipliers required to convert the raw output of the analogue to digital converter to amps and volts.

To make use of the FLAC files (for processing in, for example, MATLAB or Python), first decompress the files to create WAV files. This decompression can be done with many audio tools. We use the audio tool `sox`.

With the WAV files in hand, the next task is to convert from the values in the WAV files (in the range $[-1, 1]$) to volts and amps. Use the `calibration.cfg` file for the house in question. This file specifies an `amps_per_adc_step` parameter and a `volts_per_adc_step` parameter. Users can safely ignore the `phase_difference` parameter and assume that the measurement hardware introduces no significant phase shift. Use the following formula to calculate volts from the WAV files:

$$\text{volts} = \text{value from WAV} \times \text{volts per ADC step} \times \text{ADC steps}$$

`ADC_steps` $= 2^{31}$ for Houses 1 and 2. `ADC_steps` $= 2^{15}$ for House 5.

Use a similar formula for amps. To explain the formula above: The recording software stores each sample as a 32 bit integer for Houses 1 and 2; and as a 16-bit integer for House 5. Hence, for Houses 1 and 2, there are 2^{32} ADC steps for the full range from $[-1, 1]$ and 2^{31} ADC steps for half the range.

4.4 Exploration of the recorded data

Table 4.1 summarises the UK-DALE dataset. The table includes some metadata (which is also recorded in the machine-readable metadata supplied with the dataset) including the type of building, the year of construction, the main heat source, whether the property is bought or rented, the number of occupants, a description of the occupants, the total number of meters,

House	**1**	**2**	**3**	**4**	**5**
Building type	end of terrace	end of terrace		mid-terrace	flat
Year of construction	1905	1900		1935	2009
Energy improvements	solar thermal & loft insulation & solid wall insulation & double glazing	cavity wall insulation & double glazing		loft insulation & double glazing	
Heating	natural gas	natural gas		natural gas	natural gas
Ownership	bought	bought		bought	bought
Number of occupants	4	2		2	2
Description of occupants	2 adults and 1 dog started living in the house in 2006. One child born in 2011. Second child born in 2014.	2 adults. 1 at work all day; the other sometimes home		1 adult and 1 pensioner	2 adults
Total number of meters	54	20	5	6	26
Number of site (mains) meters	2	2	1	1	2
Sample rate of mains meters	16 kHz & 1 Hz & 6 seconds	16 kHz & 1 Hz & 6 seconds	6 seconds	6 seconds	16 kHz & 1 Hz & 6 seconds
Date of first measurement	2012-11-09	2013-02-17	2013-02-27	2013-03-09	2014-06-29
Date finished installing all meters	2013-04-12	2013-05-22	"	"	"
Date of last measurement	2016-05-13	2013-10-10	2013-04-08	2013-10-01	2014-11-13
Date when some meters were removed					2014-09-06
Total duration (days)	1280	234	39	205	137
Total uptime for mains meter (days)	1244	140	36	155	131
Uptime proportion	0.97	0.6	0.93	0.75	0.96
Average mains energy consumption per day (active kWh)	7.64	7.17			13.75
Average mains energy consumption per day (apparent kVAh)	8.9	8	12.35	10.24	17.56
Following statistics calculated when all meters installed					
Correlation of sum of submeters with mains	0.96	0.86	0.47	0.55	0.9
Proportion of energy submetered	0.8	0.68	0.19	0.28	0.79
Mean dropout rate (ignoring large gaps)	0.02	0.02	0.02	0.02	0.02

Table 4.1: Summary statistics for each house.

the number of site meters, the sample rate of the mains meters and the start and end dates for the recordings. The table also includes summary statistics calculated using the open source energy disaggregation tool NILMTK (see Chapter 5): the average mains energy consumed per day, the correlation of the mains meter with the sum of all submeters, the proportion of energy submetered, and the dropout rate. The values for the average energy consumption per day are close to the value of 9.97 kWh per day reported in DECC's Household Electricity Survey (Zimmermann et al. 2012) (which surveyed 251 houses in the UK), hence we can have some confidence that our houses consumed a fairly typical amount of energy for a UK house.

To explain the rest of the row labels in Table 4.1: 'uptime' is the total time that the system was active and recording. The 'total duration' is 'date of last measurement' minus 'date of first measurement'. The correlation of the mains meter with the sum of all submeters gives an indication of how much of the variance in the mains signal is captured by the submeters. The proportion of energy submetered is the total energy captured by the submeters divided by the total energy captured by the mains meter. The dropout rate (ignoring large gaps) gives a measure of the rate at which packets were lost due to radio errors (large gaps are ignored because these are often caused by a meter being deliberately unplugged). Some metadata is not available because the occupants are no longer contactable.

Figures 4.5 to 4.9 (except panel 'a' in Figure 4.7) were produced using NILMTK. The scripts to generate these plots are available at github.com/JackKelly/ukdale_plots.

Figure 4.5 shows the power demand for a typical day for House 1. We show the individual power demand for the top-five appliances (ranked by energy consumption) and all other submeters summed together. We also show the whole-house mains power demand. The difference between the mains power demand and the top of the submetered power demand illustrates the small amount of energy which is not submetered.

The time periods when each meter was capturing data is shown in Figure 4.6. Note that the numerous gaps in the data from House 1 are almost all deliberate and not the result of an equipment failure. For example, some meters in House 1 are manually turned off if the attached appliance is unplugged.

An example of 16 kHz data captured by our sound card power meter is shown in Figure 4.7 panel 'a'. Note that the voltage is almost a pure 50 Hz sine wave but the current contains many

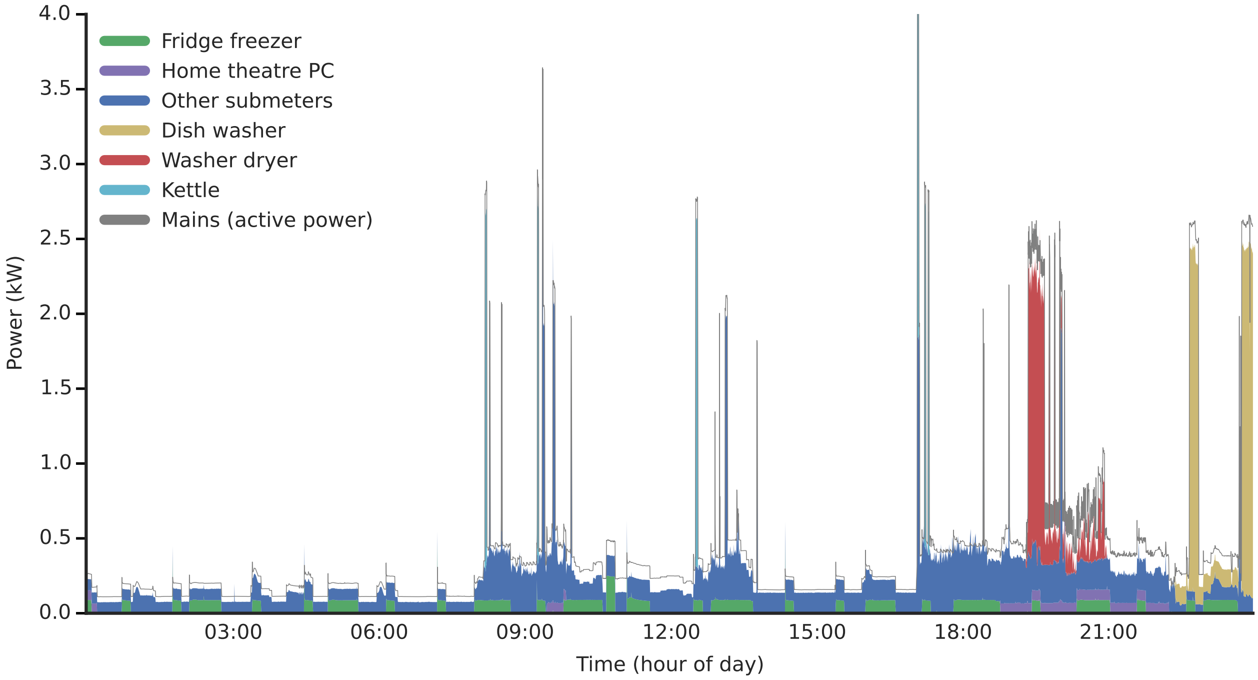


Figure 4.5: Power demand for a typical day (Sunday 2014-12-07) in House 1. The thin grey line shows the mains (whole-house) active power demand recorded using our sound card power meter. The stacked and filled coloured blocks show the power demand for the top five appliances (by energy consumption) and the dark blue block shows all the other submeters summed together. The thin white gap between the top of the coloured blocks and the mains plot line represents the power demand not captured by any submeter.

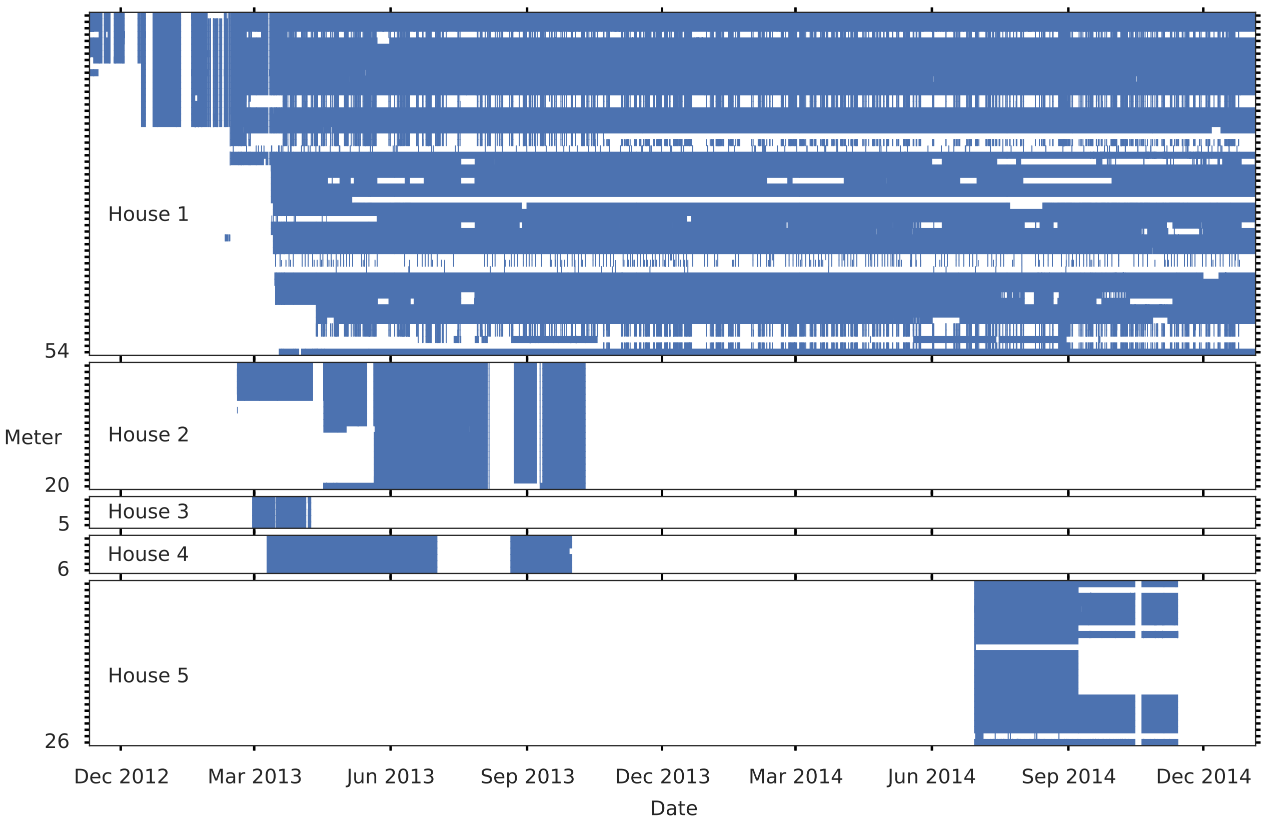


Figure 4.6: Time periods when meters were recording. The five houses in the dataset are represented by the five panels in this plot. The height of each panel is proportional to the number of meters installed in each house. Each thin row (marked by each y-axis tick mark) represents a meter. Blue areas indicate time periods when a meter was recording. White gaps indicate gaps in the dataset. The data for House 1 continues into 2015 and 2016 but that data has been left off this plot in order to provide enough space to see the detail in the plots for Houses 2-5.

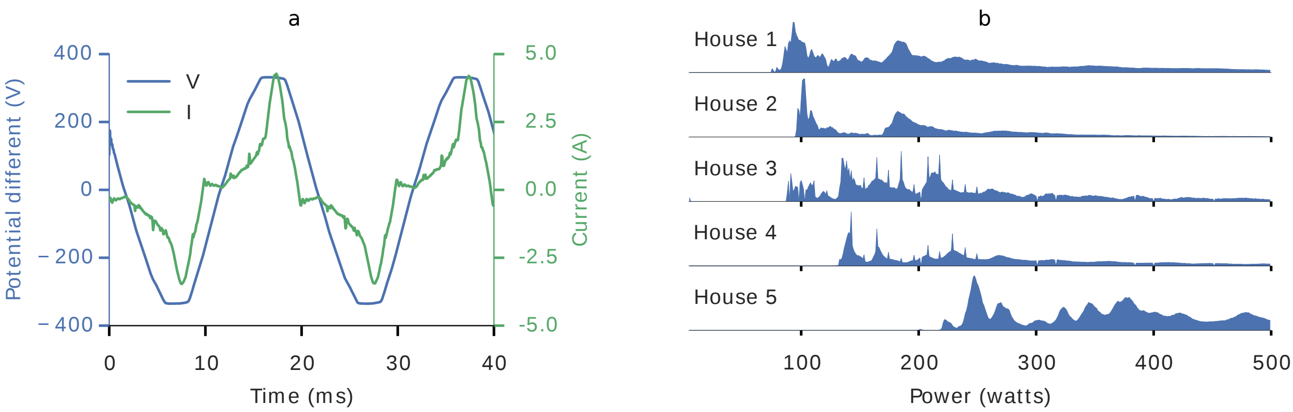


Figure 4.7: Mains electricity data. **a,** 16 kHz sampling of mains voltage and current using our sound card power meter from House 1 on 2014-09-03 21:00:00+01:00. The green line shows the current and the blue line shows the voltage. Panel **b** shows histograms of mains power demand for each house. The five subplots represent the five houses in the dataset. There is some density above 500 watts but this has been cropped from this plot to allow us to see detail in the range between 0 and 500 watts.

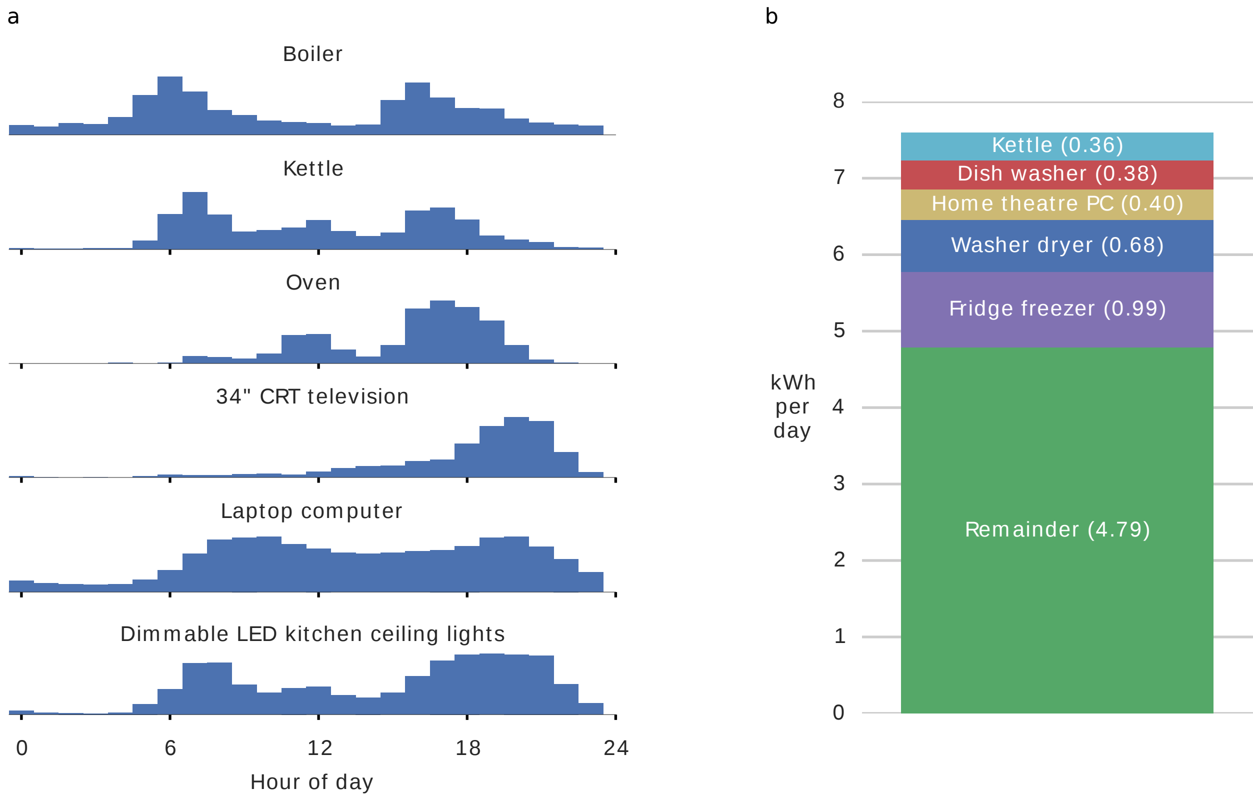


Figure 4.8: Electrical appliance usage in House 1. a, Histograms of daily appliance usage patterns. Panel **b** shows average daily energy consumption of the top-five appliances in House 1. All appliances were ranked by the amount of energy they consumed and the top-five are shown here. All lights were grouped together. The 'remainder' block at the bottom represents the difference between the total mains energy consumption and the sum of the energy consumption of the top five appliances. As such, the top edge of the bar shows the average daily total energy consumption for House 1.

harmonics.

The distribution of values for the mains power demand for each house is shown in Figure 4.7 panel 'b'. The left-most edge of each density represents the 'vampire power' of each house (i.e. the power demand when no one is using an appliance but power is still being drawn by always-on appliances and appliances in standby mode).

Figure 4.8 panel 'a' shows the hour per day that several appliances are used. For example, the oven shows two peaks in usage: one around midday (lunch) and one around 18:00 (dinner).

The energy consumed by the top five energy consuming appliances in House 1 is shown in

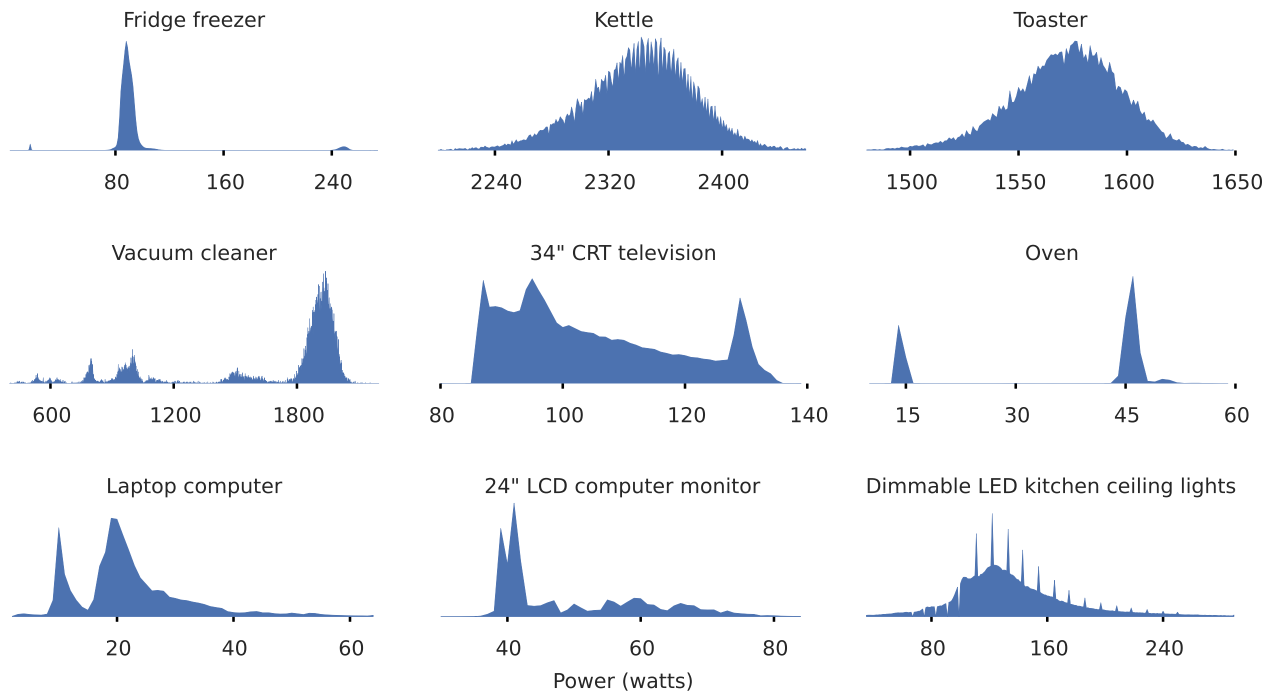


Figure 4.9: Histograms of appliance power demand from House 1.

Figure 4.8 panel 'b'. This is relevant because energy disaggregation researchers often prioritise the disaggregation of the appliances responsible for the largest energy consumption.

The distribution of values of power demand for individual appliances is shown in Figure 4.9. Some appliance information can be inferred from the histograms. For example, the top left panel shows a histogram for the power demand of the fridge: the main peak around 90 W is the normal compressor cycle, the peak around 17 W is the fridge lamp and the peak around 250 W is the defrosting cycle. The vacuum cleaner has six discrete power settings, all of which can be seen in its histogram.

Figure 4.10 shows the measurement errors for our sound card power meter and the Current Cost Current Transformer (CT) sensor across a range of resistive loads. The ground truth was measured using a 'Watts up? PRO' meter[16] (with a nominal accuracy of ±1.5%). The range of loads were created by using three incandescent lamps and by changing the number of primary turns on the CT from one to seven, in steps of one. For each load, we recorded one minute of data and took the largest error from that minute of data. The sound card power meter was calibrated six months prior to the test. This illustrates that our sound card power meter consistently produces a relative error of less than 2% and that the Current Cost CT meter

[16] https://www.wattsupmeters.com/secure/products.php?pn=0

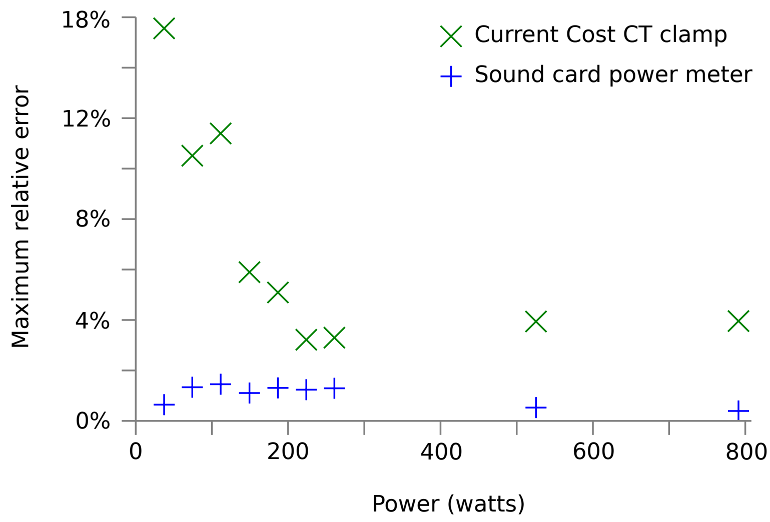


Figure 4.10: Maximum relative measurement error for power measurements across a range of loads. A 'Watts up? PRO' meter was used to record the ground-truth.

produces errors of less than 6% as long as the power is above 100 watts (the whole-house power demand very rarely drops below 100 watts).

We compared the total active energy recorded by our sound card power meter for House 1 with the utility-installed 'spinning disk' electricity meter in House 1. We selected the time period of 2013-05-22 20:36 to 2014-11-28 08:55 for the comparison because we have a continuous recording from our sound card power meter for this period, and we have readings of the utility meter at the start and end of this period. The total energy recorded by the utility meter for this period was 4030.60 kWh. The total recorded by the sound card power meter was 4142.93 kWh. The relative difference is 2.71%.

Houses 1, 2 and 5 had two mains meters: a sound card power meter and a Current Cost meter. For each of these houses, we compared the apparent energy recorded by the sound card power meter against the Current Cost whole-house meter. The relative difference was 1.79% in House 1; 7.30% in house 2 and 5.68% in house 5.

4.4.1 Correlations with weather

We expect the usage of some appliances to be correlated with some weather variables. Previous studies have demonstrated correlations between temperature and heating/cooling demand in Australia (de Dear & M. Hart 2002) and between temperature and total household electricity

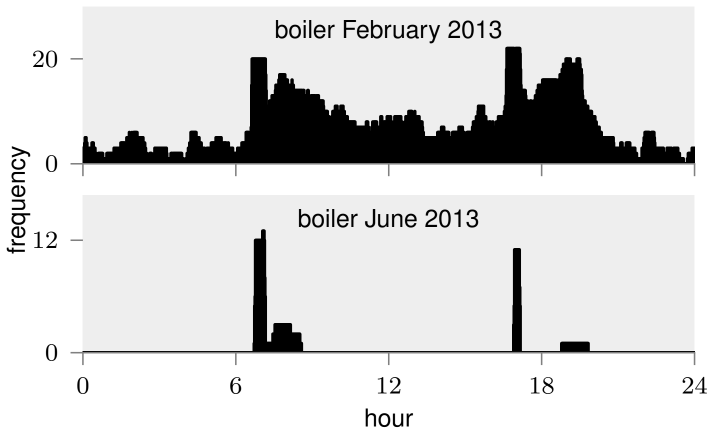


Figure 4.11: Seasonal variation in boiler usage. In June (bottom panel), the boiler usage histogram is dominated by the hot water program which runs twice a day. In February (top panel), considerably more energy is used on space heating than on hot water heating.

usage in America (Kavousian et al. 2013). We wanted to search for similar correlations in our UK data.

If robust correlations can be demonstrated then a disaggregation system could learn correlations between weather variables and appliance usage in order to refine its appliance usage estimates. Weather data are relatively easy to acquire programmatically: for example, the UK Metoffice provides free access to live weather data via their DataPoint API[17].

An example of seasonal variation in boiler usage is shown in Figure 4.11. During February, when the weather is relatively cold in the UK, the boiler is primarily used to heat to the radiators for the majority of the day. In May, when the weather is warmer, the space heating requirement drops away and the twice-daily hot water heating program dominates the usage histogram.

Figure 4.12 shows correlations between lighting usage and solar radiation; between solar thermal pump activity and solar radiation; between boiler usage and maximum temperature and between fridge activity and minimum external temperature. As one might expect, the strongest correlation is between solar radiation and the activity of the solar thermal pump (which turns on automatically when the temperature of the evacuated tube solar thermal panels on the roof exceed a threshold). Lighting circuit usage correlates with solar radiation although there is considerable variation in lighting usage which is not explained by solar radiation. The correlation between external maximum temperature and boiler usage is also strong; and it is noteworthy

[17]http://www.metoffice.gov.uk/datapoint

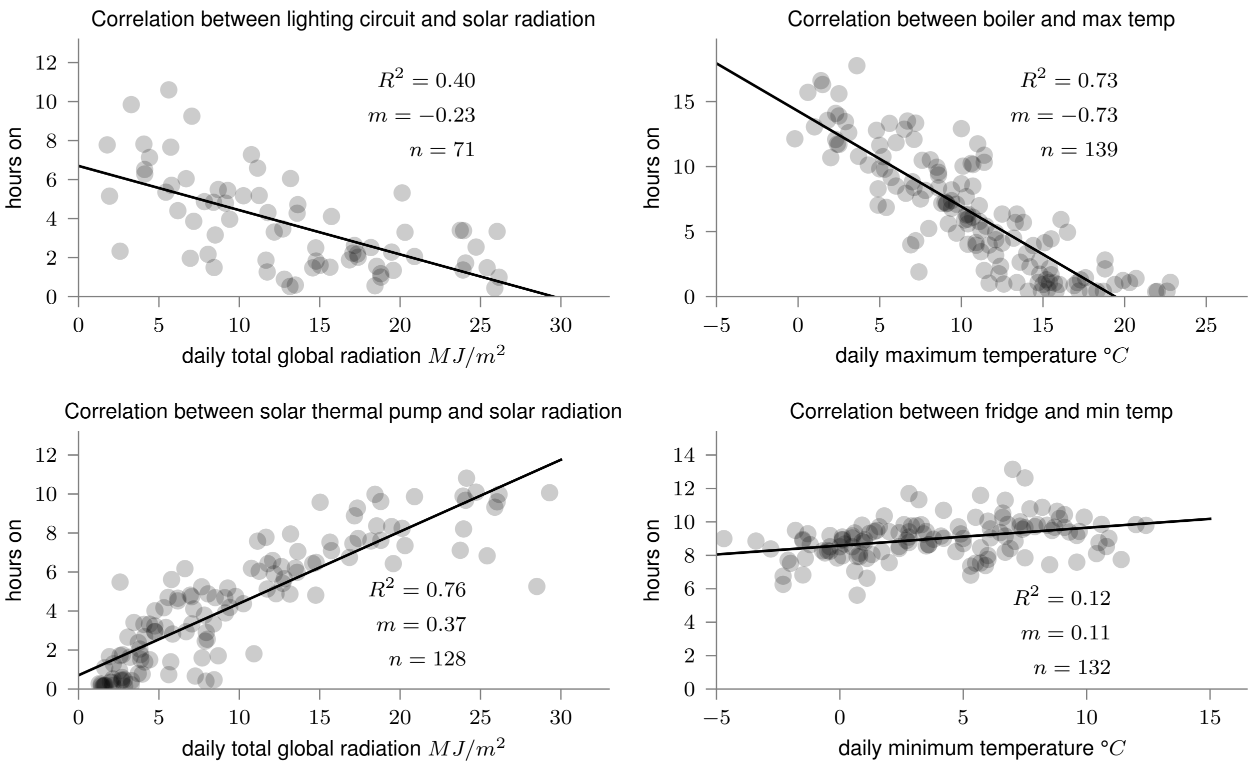


Figure 4.12: Linear regressions showing correlation between appliance activity and weather. R^2 denotes the coefficient of determination, m is the gradient of the regression line and n is the number of data-points used in each regression. Each data-point represents one day. Historical daily averages from Heathrow weather station (20 miles due west of the premises under consideration) were obtained from the UK Met Office under their Educational program. Days for which the appliance usage was zero were ignored because we assume that the house was unoccupied on these days.

that the X-axis intercept ($\approx$ 19℃) is approximately the set point for the boiler thermostat, as one might expect.

Please see Appendix A.5 for the usage notes.

4.5 Reception

According to Google Scholar[18], our UK-DALE paper (Kelly & Knottenbelt 2015b) has been cited 68 times, as of the 6th April 2017. The dataset has been downloaded many more times, although we do not have a precise number for the downloads because our FTP server does not

[18] https://scholar.google.co.uk/citations?view_op=view_citation&citation_for_view=Z9LOTgsAAAAJ:ufrVoPGSRksC

keep logs longer than one month old.

4.6 Acknowledgments for UK-DALE

- Geoff Dutton, the data manager for the UKERC Energy Data Centre.
- Dr Mark Bilton at Imperial's Low Carbon London Lab.
- Robert Wall from the Open Energy Monitor project; and the rest of the Open Energy Monitor community.
- Peter Morgan from DECC.
- Graham Murphy, Matt Thorpe and Paul Cooper who contributed hugely to the effort to decode the Current Cost and EDF RF protocols.
- The Current Cost company.
- This work was funded by the EPSRC and by Intel via their EU Doctoral Student Fellowship Programme.
- We thank the students whose homes we recorded from.

Chapter 5

An open-source tool kit for non-intrusive load monitoring (NILMTK)

This chapter is based on Batra et al. 2014a; Kelly et al. 2014 and NILMTK's documentation[1].

5.1 Why develop an open-source NILM toolkit?

George Hart, the 'inventor' of NILM, gave a keynote presentation[2] at the third International Workshop on NILM[3] in Vancouver in May 2016. The following evening, we invited Professor Hart to the programme committee dinner. Over dinner, Hart mentioned how excited he was to attend a NILM workshop 32 years since his first publication on NILM (G. W. Hart 1984) and 21 years since his *last* publications on NILM (G. W. Hart 1995; Rouvellou & G. W. Hart 1995).

Later that evening, Hart mentioned that, as far as he could tell, the field of energy disaggregation had not progressed as much as he had expected (Hart has not worked on NILM since the mid-1990s). This lack of progress had also been worrying me and some of my collaborators.

[1] http://nilmtk.github.io

[2] https://www.youtube.com/watch?v=hKy9UJpOFpU

[3] http://nilmworkshop.org/2016

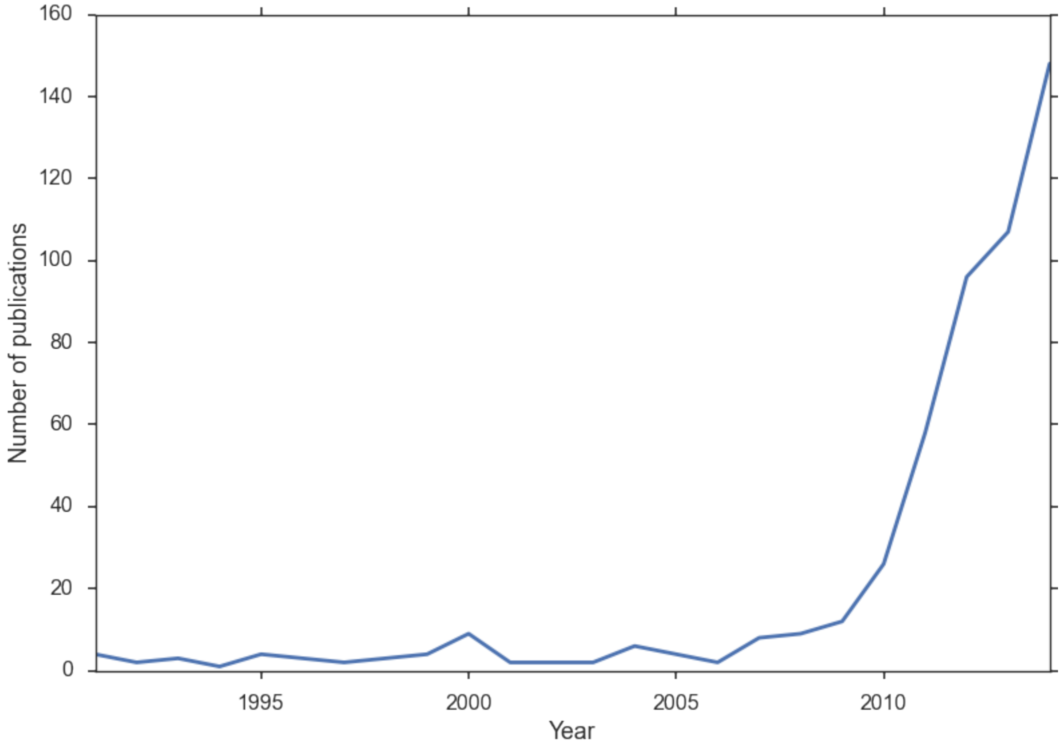


Figure 5.1: NILM papers published per year. Source: Parson 2015 (with permission).

Why has progress been slower than expected? Part of the explanation is that there was comparatively little NILM research done before about 2008 (see Figure 5.1). But there are, perhaps, deeper problems. Energy disaggregation research has several features which almost certainly slow the rate of progress. Three of these problems are:

1. It is currently impossible to compare the performance of NILM algorithms presented across any pair of papers because each paper uses different metrics, different datasets and different pre-processing steps. Hence we have no way to measure progress over time. And we have no way to decide which NILM approaches are most promising and hence most deserving of future research effort. Thus far, NILM research is a little like natural selection *without selection*: it is just a random walk!

2. It is hard to replicate algorithms described in papers because little code is shared and because papers often do not provide full implementation details. Replication is essential to the scientific process.

3. Each public dataset uses a different format hence importing multiple datasets requires tedious engineering effort. So, in practice, researchers rarely use more than one dataset. This reduces our ability to measure the generalisability of NILM algorithms.

Our aim with NILMTK was to help to fix these three issues by developing an open-source toolkit for NILM researchers. We wanted to make it easier for researchers to conduct high quality research. In particular, we wanted to:

1. Provide a common set of pre-processing tools, benchmark NILM algorithms and disaggregation metrics to help individual research teams to validate the performance of their algorithms in a standardised way.
2. Develop, with the community, a standardised way to publish NILM algorithm code.
3. Provide a standard dataset format and also provide converters for datasets to the standard format.

5.2 Overview of NILMTK

NILMTK is an open-source toolkit for energy disaggregation. NILMTK is written in Python and the code is freely available on GitHub at github.com/nilmtk/nilmtk. Anyone can propose changes to the code or documentation; or submit bug reports or feature requests.

The Python files include "docstrings" which document the classes and methods. This API documentation is automatically extracted from the Python files using the documentation tool Sphinx[4] and presented as HTML pages[5]. NILMTK also has a manual[6] and a set of example scripts presented as iPython notebooks[7]. I ran a one day hands-on NILMTK workshop at the 2014 London NILM Workshop[8] and an afternoon session on NILMTK at the 2015 London NILM Workshop[9].

An overview of the NILMTK pipeline is shown in Figure 5.2. Datasets are first converted to NILMTK's standard data format which is based on the HDF5 file format. Then users can invoke NILMTK's dataset statistics functions to identify issues with the dataset, or to compute informative statistics. NILMTK also provides a set of pre-processing functions to

[4]http://sphinx-doc.org
[5]http://nilmtk.github.io/nilmtk/master/index.html
[6]http://github.com/nilmtk/nilmtk/tree/master/docs/manual
[7]http://github.com/nilmtk/nilmtk/tree/master/notebooks
[8]http://www.oliverparson.co.uk/nilm-2014-london
[9]http://www.nilm.eu/energy_disaggregation_in_industry__academia

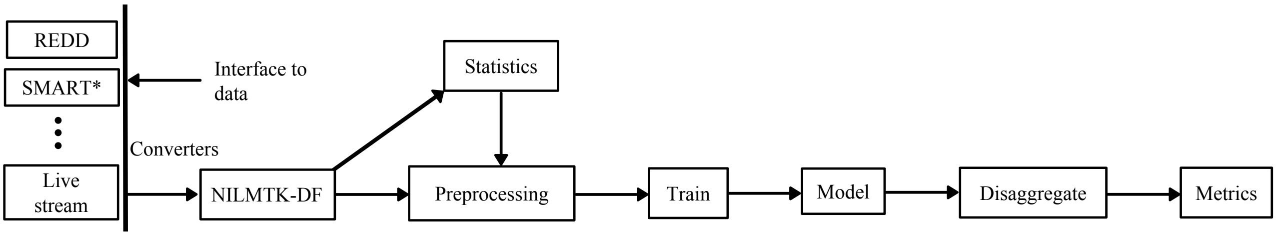


Figure 5.2: The NILMTK data processing pipeline (Figure created in collaboration with Oliver Parson and Nipun Batra).

clean up imperfections or to re-sample data to a different timebase (e.g. downsampling 1 Hz data to 1 minutely data). NILMTK has a number of benchmark NILM algorithms which can be trained on disaggregated data and then, once models are inferred, aggregate data can be disaggregated. Finally, NILMTK provides a set of standard metrics functions to compute the disaggregation performance.

NILMTK has gone through two major versions. NILMTK v0.1 was described in Batra et al. 2014a. NILMTK v0.2 was a complete rewrite of NILMTK and was described in Kelly et al. 2014. I designed most of the architecture and wrote most of the code for NILMTK v0.2.

5.3 The NILMTK developers

The majority of the code and documentation for NILMTK was written by a "core team" of three people: Nipun Batra[10] (who was doing his PhD at the Indraprastha Institute of Information Technology (IIIT) Delhi[11] when we started NILMTK; and moved to University of Virginia[12] in March 2017); Oliver Parson[13] (who was at the University of Southampton during the majority of NILMTK's development but is now at Centrica Connected Home[14]); and me. Work started on NILMTK towards the end of November 2013. The NILMTK project was first proposed by Nipun.

A substantial proportion of the code has been contributed by people not in the "core team". Several of NILMTK's eleven dataset converters were written by the dataset authors. Two of NILMTK's disaggregation algorithms were written by the algorithm developers.

10 http://nipunbatra.github.io
11 https://www.iiitd.ac.in
12 http://www.cs.virginia.edu/~nb2cz
13 http://www.oliverparson.co.uk
14 https://www.centrica.com/about-us/what-we-do/connected-home

5.4 Other open-source tool kits

Shortly after NILMTK was released, a research group at ETH Zurich released NILM-Eval[15], a NILM toolkit written in MATLAB (Cicchetti 2014; Beckel et al. 2014). To quote NILM-Eval's GitHub page:

> NILM-Eval is a MATLAB framework that allows to evaluate non-intrusive load monitoring algorithms in different scenarios to gain a comprehensive view on their performance. NILM-Eval makes it easy to evaluate algorithms on multiple data sets, households, data granularities, time periods, and specific algorithm parameters. By encapsulating those parameters in configurations, NILM-Eval further allows the user with little effort to repeat experiments performed by others, to evaluate an algorithm on a new data set, and to fine-tune configurations to improve the performance of an algorithm in a new setting.

Later in 2014, the Data Science for Social Good Summer Fellowship at the University of Chicago[16] ran an energy disaggregation project which resulted in an open source framework for disaggregated energy data[17].

As far as we can tell, NILMTK is had a larger impact on the NILM community than NILM-Eval or the DSSG project. One imperfect way to measure impact is to look at the number of "stars" each project has on GitHub. NILMTK has 186 stars; NILM-Eval has 21 stars and the DSSG project has 17 stars.

5.5 NILMTK's design and features

We will now discuss in detail the design and features of NILMTK v0.2. Please see Appendix B.1 for a list of software dependencies for NILMTK.

[15] https://github.com/beckel/nilm-eval
[16] https://dssg.uchicago.edu
[17] https://dssg.uchicago.edu/project/building-open-source-tools-to-analyze-smart-meter-data/

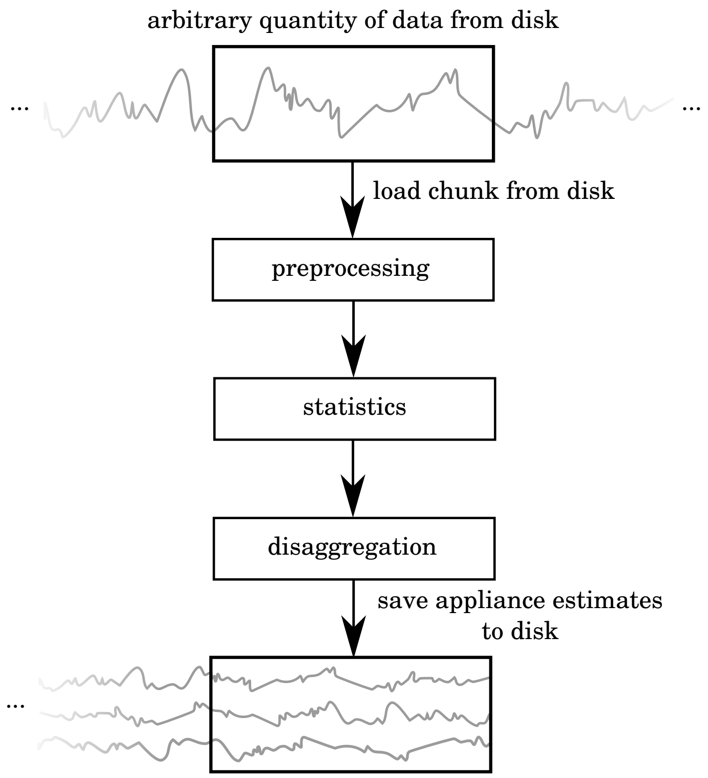


Figure 5.3: NILMTK v0.2 can process an arbitrary quantity of data by loading data from disk in chunks. This figure illustrates the loading of a chunk of aggregate data from disk (top) and then pushing this chunk through a processing pipeline which ends in saving appliance estimates to disk chunk-by-chunk.

5.5.1 NILMTK can handle an arbitrary amount of data

One of the main differences between NILMTK v0.1 and v0.2 is that v0.2 can handle an arbitrary amount of data. It does this by loading data from disk in chunks such that each chunk can fit into memory (see Figure 5.3).

This "out of core" processing is implemented using Python's `generator` functions. `Generators` provide a convenient way to define `iterators`. Each time `iterator.next()` is called, new data is returned. This provides a natural way to iterate through an arbitrary amount of data.

A simple generator example is provided by the Python wiki documentation on generators[18]:

[18] https://wiki.python.org/moin/Generators

```python
# a generator that yields items instead of returning a list
def firstn(n): # define a function which takes a single parameter, n
    num = 0
    while num < n:
        yield num  # The loop returns 'num'; pauses and returns
                   # control to the calling code until
                   # 'next()' is called again on the iterator object
        num += 1

sum_of_first_n = sum(firstn(1000000))
```

Iterators can be chained such that each step in the processing pathway takes an iterator as input and returns a new iterator. This mechanism is used in NILMTK to allow processing pipelines to be defined in a *lazy* way. In other words, the processing transformations can be defined and chained independently of the data. It is only when the user tries to pull data out of the very end of the processing pipeline that any data is loaded into memory from disk.

Out of core processing allows NILMTK to handle datasets which are too big to fit into system memory and also allows NILMTK to limit its memory usage. One problem with NILMTK v0.1 was that it would try to use more memory than was available to Python, which would result in the entire system becoming unresponsive.

5.5.2 Classes

A UML diagram for NILMTK v0.2 is shown in Figure 5.4. The following sections will briefly describe the main classes in NILMTK v0.2.

The design of NILMTK can be considered as being divided into layers of abstraction where the bottom layer is closest to the hardware and the top layer provides the highest level of abstraction in order to provide a convenient interface for users. These layers are loosely inspired by the abstraction layers in the TCP/IP stack.

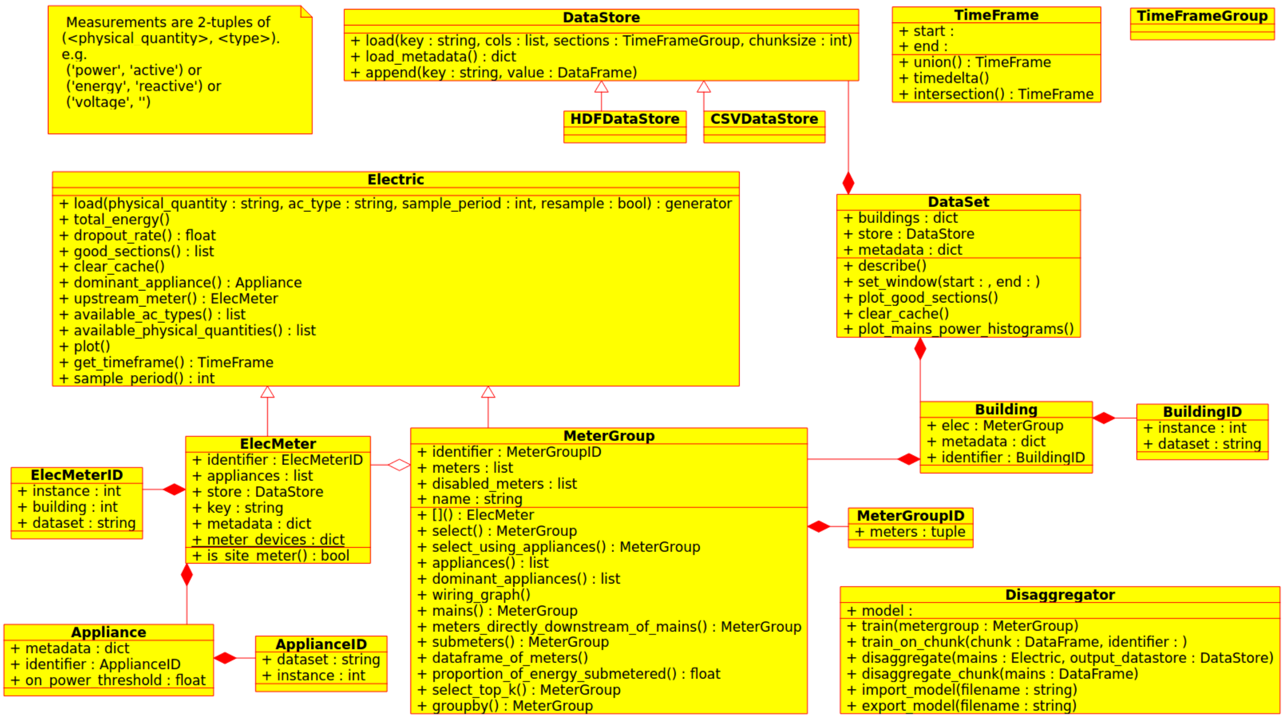


Figure 5.4: UML diagram for NILMTK v0.2.

The `DataStore` class

The `DataStore` class defines the interface for loading and writing data. The `DataStore` class is in NILMTK's bottom abstraction layer. It loads a single chunk of data at a time and returns a generator of Pandas DataFrame objects (an in-memory 2D matrix).

The `DataStore` class is totally agnostic about what the data 'means'. Each column could represent voltage, current, temperature, PIR readings etc. This is similar to how the Link layer in TCP/IP is totally agnostic about what the packet's payload 'means'.

We have implemented two `DataStore` subclasses: `HDFDataStore` and `CSVDataStore`. Other `DataStores` could, in principal, be implemented. These might include classes for pulling data from big data stores on the network, or from meters etc.

To minimise memory usage, `DataStore.load()` can be told exactly which columns to load and can be given a list of `TimeFrames` to load from disk.

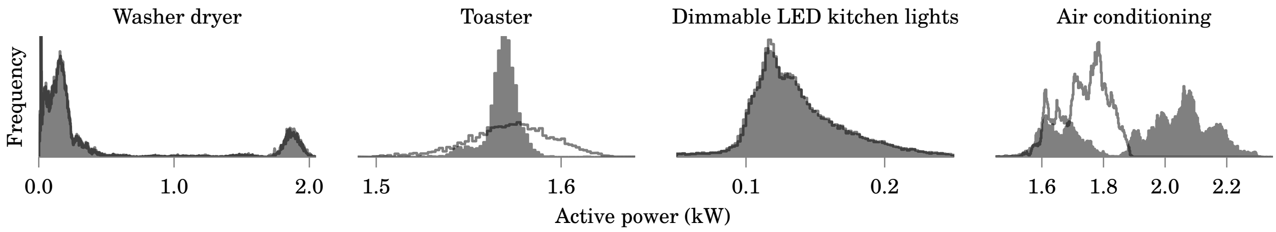


Figure 5.5: Histograms of power consumption. The filled grey plots show histograms of normalised power. The thin, grey, semi-transparent lines drawn over the filled plots show histograms of un-normalised power.

The `TimeFrame` class and the `TimeFrameGroup` class

A `TimeFrame` object represents a time period with an arbitrary start date and an arbitrary end date. It contains methods for manipulating time periods. For example, `tf1.check_for_overlap(tf2)` returns `True` if `tf1` and `tf2` overlap.

The `TimeFrameGroup` class exists to store multiple `TimeFrame` objects. It contains methods to, for example, find the intersection between two `TimeFrameGroup` objects.

The `Electric` superclass, `ElecMeter` class and `MeterGroup` class

The `Electric` class exists one level of abstraction above the `DataStore` class. The `Electric` class has two subclasses: the `ElecMeter` and `MeterGroup` classes.

The `Electric` class provides methods such as `plot_power_histogram()` which can be used to create histograms like the one shown in Figure 5.5.

The `ElecMeter` class represents a single electricity meter. The `MeterGroup` class represents multiple electricity meters (for example, all the meters in a single home; or all the meters in a dataset which measure lights).

`MeterGroup` contains a set of methods for selecting `ElecMeters`. For example, `MeterGroup.select_top_k(k=5)` will return a new `MeterGroup` which contains only the top five `ElecMeters`, ranked by energy consumption, allowing for the creation of graphs like that in Figure 5.6.

Another way to select from a `MeterGroup` is to search by the appliance type connected to the meter. For example, `metergroup['kettle']` will select the `ElecMeter` connected to the kettle.

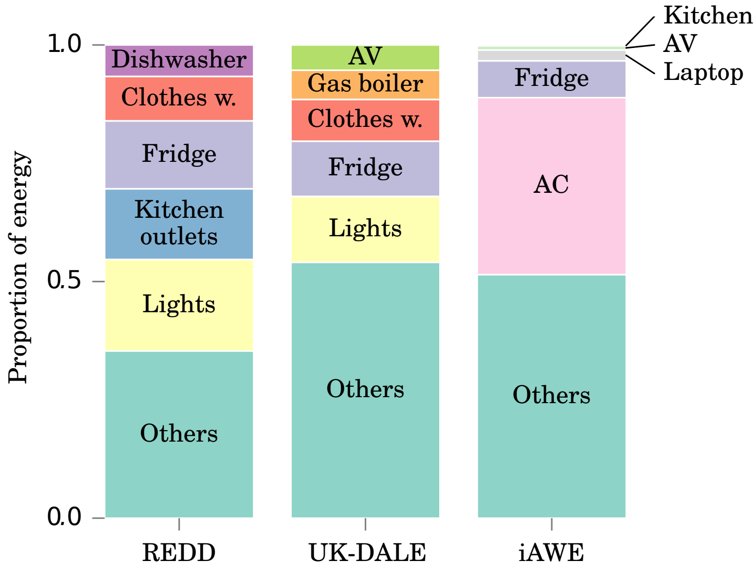


Figure 5.6: Top five appliances in terms of the proportion of the total energy used in a single house (house 1) in each of REDD (USA), iAWE (India) and UK-DALE.

`ElecMeters` can have a substantial amount of metadata associated with them, as per the NILM Metadata schema.

The `Appliance` class

Just as in our NILM Metadata schema, NILMTK distinguishes between *appliances* and *electricity meters* so that we can represent any arbitrary relationship between electricity meters and appliances. The `Appliance` class represents a single electric appliance, such as a kettle or toaster.

Appliances can have detailed metadata associated with them, as per the NILM Metadata schema. That metadata might come from the dataset itself (such as the make and model of the appliance) or might come from NILM Metadata's central store of metadata (such as the fact that a kettle is often found in the kitchen).

The `Building` class and the `DataSet` class

The `Building` class represents a physical building such as a house. Each `Building` object has an `elec` attribute which is a `MeterGroup` object which represents all the `ElecMeters` in that building. Each `Building` object also has a `metadata` attribute which is a Python dictionary

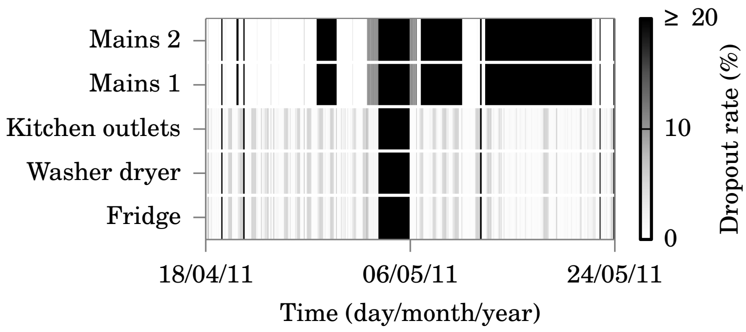


Figure 5.7: Lost samples per hour from a representative subset of channels in REDD house 1.

Data set	Number of appliances	Percentage energy sub-metered	Dropout rate (percent) ignoring gaps	Mains up-time per house (days)	Percentage up-time
REDD	9, 16, 23	58, 71, 89	0, 10, 16	4, 18, 19	8, 40, 79
Smart*	25	86	0	88	96
Pecan Street	13, 14, 22	75, 87, 150	0, 0, 0	7, 7, 7	100, 100, 100
AMPds	20	97	0	364	100
iAWE	10	48	8	47	93
UK-DALE	4, 12, 53	19, 48, 82	0, 7, 22	36, 102, 470	73, 84, 100

Table 5.1: Summary of dataset results calculated by the diagnostic and statistical functions in NILMTK. Each cell represents the range of values across all households per data set. The three numbers per cell are the minimum, median and maximum values. AMPds, Smart* and iAWE each contain just a single house, hence these rows have a single number per cell.

recording metadata about the building, using the NILM Metadata schema.

A `DataSet` object holds all the `Buildings` in a single dataset and provides some methods for computing statistics for the entire dataset such as `plot_good_sections` which can be used to create plots like the one shown in Figure 5.7. It is also possible to compute a range of statistics across the entire dataset, such as the statistics presented in Figure 5.1. `DataSet.set_window()` allows users to define a `TimeFrame` over which all subsequent computations will be performed.

The `Disaggregator` class

The `Disaggregator` class provides a common interface to train and test disaggregation algorithms. There are four implementations in NILMTK: `CombinatorialOptimisation`; `FHMM` for training and performing exact inference on a factorial hidden Markov model; `Hart85` for performing disaggregation in a similar way to that described in G. W. Hart 1985 (originally implemented by one of my 2014 MSc individual project students, Simon Leigh); and `MLE` for performing maximum likelihood estimation (originally implemented by José Alcalá working

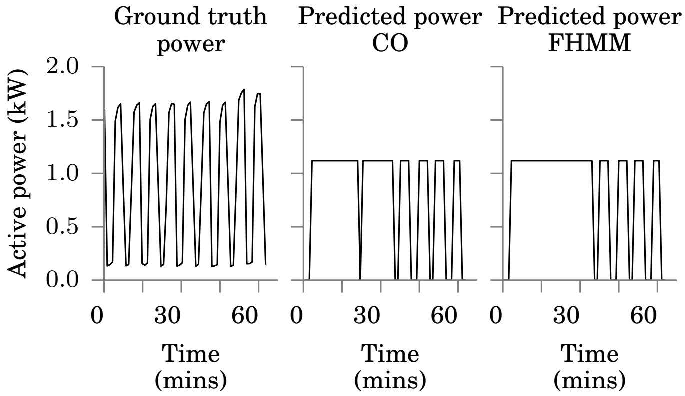


Figure 5.8: Predicted power estimates generated by the CO and FHMM algorithms and, for comparison, the ground truth for air conditioner 2 in the iAWE data set. Figure created by Nipun Batra.

with Oliver Parson).

We decided that third party, cutting-edge NILM algorithms should be maintained by the original developer (instead of becoming part of the core NILMTK project). A list of third-party NILM algorithms compatible with NILMTK are listed in the NILMTK wiki[19].

5.5.3 Data format

NILMTK stores disaggregated datasets and the output of disaggregation algorithms in a common format. We have designed a hierarchical data structure which can be implemented using either a set of CSV files in a folder hierarchy of or, more commonly, a single HDF5 file per dataset. The full details of NILMTK's dataset format are presented in NILMTK's manual[20].

5.5.4 Disaggregation

Once a `Disaggregator` object has been trained (or once a pre-trained model has been loaded), it is possible to use `Disaggregator.disaggregate()` to produce a set of appliance-by-appliance estimates. Figure 5.8 shows the ground truth power for an air conditioner and the predicted power using the combinatorial optimisation algorithm and the FHMM algorithm.

[19] `https://github.com/nilmtk/nilmtk/wiki/NILM-Algorithms`

[20] `https://github.com/nilmtk/nilmtk/blob/master/docs/manual/development_guide/writing_a_dataset_converter.md`

5.5.5 Automated unit testing

We have written a suite of unit tests. The tests cover 55% of the code. One of the main benefits of having a set of unit tests is that it allows us to quickly check if any new code has broken any functionality. We use the free Travis Continuous Integration (CI) service[21] to automatically run all unit tests whenever new code is committed to github or whenever a new pull request is submitted. This has allowed us to catch and fix multiple bugs.

5.6 NILMTK and the community

The wider community can interact with NILMTK and the NILMTK developers through several channels:

- The NILMTK issue queue on GitHub[22].
- Pull requests on github.
- The energy disaggregation mailing list[23] (which I set up).
- Direct emails to developers.
- Face to face at workshops and conferences.

The NILMTK issue queue on GitHub is the primary channel through which the community interacts with the NILMTK project. There are a total of 522 issues on the issue queue, of which 359 are closed and 130 are open (as of 2016-08-23).

Keeping up with the issue queue has been a substantial challenge for us. Sometimes multiple issues will be submitted per day, each of which might take several hours to fix.

A large proportion of the issues submitted to the issue queue are not bug reports but are requests for help. In retrospect, we should have put more effort towards writing comprehensive documentation near the beginning of the project. This would have saved a lot of time on the issue queue!

[21] https://travis-ci.org/nilmtk/nilmtk
[22] https://github.com/nilmtk/nilmtk/issues
[23] https://groups.google.com/forum/#!forum/energy-disaggregation

5.7 Conclusion

NILMTK is an open source toolkit and it certainly has the potential to satisfy the aims that we described at the start of this chapter. NILMTK has certainly achieved recognition in the NILM community: our first NILMTK paper (Batra et al. 2014a) has been cited 109 times (according to Google Scholar on 2017-04-07); and we frequently receive email correspondence about NILMTK. But it is not yet clear whether it has driven substantial change in the NILM research community. This might, in part, be because NILMTK is over-complicated. Ideas for simplifying NILMTK are presented in the next section.

5.7.1 Future work: simplifying NILMTK

Perhaps the most pressing need is to *simplify* the NILMTK code. No developers outside the "core team" have contributed code to the majority of NILMTK's classes. This is likely to be because NILMTK's code is complex.

For example, I designed and implemented a mechanism whereby, prior to executing any dataset statistics, NILMTK would check a dataset's metadata to see if the data was in the correct state (for example, had all the gaps been filled?). This metadata-checking feature has, as far as I can tell, never been used. Yet it adds a fair amount of complexity to the code.

The out-of-core processing in NILMTK complicates the majority of the code. Now that system memory is relatively cheap,[24] perhaps it would be prudent to go back to processing everything in-memory (but to have simple checks to make sure NILMTK did not try to load more data into memory than is available to Python). This would significantly simplify the code. Alternatively, NILMTK could make use of modern Python packages such as Blaze[25] and Dask[26] which handle out-of-core processing (this idea is discussed further in issue #248 on the issue queue[27]).

Another issue with NILMTK's current design is that it is a monolithic "walled garden". NILMTK wraps data in complex objects which carry lots of state. This makes it easy to pass data through

[24] 16 GBytes of DDR4 RAM is £120 including VAT, as of April 2017. And technologies like Intel's Optane (also known as "3D XPoint") might enable computers to have hundreds of gigabytes of core memory for relatively small amount of money within a few years.

[25] `http://blaze.pydata.org`

[26] `http://dask.readthedocs.io`

[27] `https://github.com/nilmtk/nilmtk/issues/248`

a pure-NILMTK tool chain. But it is hard to create a *hybrid* tool chain where NILMTK is used for only part of the processing. As such, we are considering redesigning NILMTK so that it more closely follows the "UNIX design philosophy" (McIlroy et al. 1978) and becomes a set of small, well defined tools; rather than a sprawling monolith (these ideas for simplifying NILMTK are further discussed in issue #479[28]).

Other ideas for simplifying NILMTK are tagged with the "simplify" label[29] on the NILMTK issue queue.

[28]https://github.com/nilmtk/nilmtk/issues/479
[29]https://github.com/nilmtk/nilmtk/labels/simplify

Chapter 6

A metadata schema for disaggregated energy data

Multiple disaggregated electricity datasets have been released over the last few years. Each dataset uses a different format to communicate essential characteristics of the dataset. For example, the metadata might be in a CSV file or a plain-text README. The lack of a standard metadata schema makes it unnecessarily time-consuming to write software (such as NILMTK) to process multiple datasets. Worse, the lack of a metadata standard means that crucial information is absent from some datasets.

This chapter describes a metadata schema for representing appliances, meters, buildings, datasets, prior knowledge about appliances and appliance models. The schema defines a graph-based representation and provides a simple but powerful inheritance mechanism. This chapter is based on a published conference paper, Kelly & Knottenbelt 2014.

In this chapter we first introduce the need for a metadata schema (Section 6.1), then we outline related work (Section 6.2); then we describe the design (Section 6.3) and implementation (Section C.1); then we discuss limitations of the schema (Section 6.4). Towards the end of the chapter we discuss the community's reaction to the metadata schema since its publication in 2014 (Section 6.5). This motivates a discussion of whether a simpler metadata schema is required (Section 6.4.1) and, finally, the chapter concludes (Section 6.6).

6.1 Introduction

In the quest to design and implement a high performance energy disaggregation system, researchers require several types of data:

The primary requirement is for datasets which record the power demand of whole homes as well as the ‘ground truth’ power demand of individual appliances within the home. In 2011, researchers at MIT released the first public dataset for energy disaggregation research (J. Zico Kolter & M. J. Johnson 2011). Since then, eleven more datasets have been released (Pecan Street 2011; Reinhardt et al. 2012; Holcomb 2012; K. Anderson et al. 2012; Batra et al. 2013; Sean Barker et al. 2012; Zimmermann et al. 2012; Makonin et al. 2013; Beckel et al. 2014; Kelly & Knottenbelt 2015b; Gao et al. 2015; Nambi et al. 2015; Kahl et al. 2016; Makonin et al. 2016).

These datasets have been well received but, because each dataset uses a different file format, it is time consuming to import multiple datasets. This is an issue because an important criteria for evaluating any machine learning algorithm is how well it generalises across multiple datasets. A further challenge with existing datasets is that machine-readable metadata is often minimal and uses a schema and vocabulary unique to that dataset. At best, the lack of a standard metadata schema makes it time-consuming to write software to process multiple datasets. At worse, some datasets simply lack sufficient metadata to allow the data to be properly interpreted. For example, the mains wiring connecting meters to each other and to appliances forms a tree (with the whole-house meter at the root and appliances at the leaves). This tree structure has more than two levels in some datasets yet the metadata rarely specifies the wiring tree.

A secondary requirement is for data describing the *behaviour* of appliances (e.g. a probability distribution describing the typical times per day that each appliance is used). This prior knowledge can be used to fine-tune the estimates produced by a disaggregation system. Such data is available in research papers and industry reports, but not in a machine-readable form.

Finally, consumers are unlikely to put effort into training a disaggregation tool. As such, if open-source disaggregation solutions such as NILMTK (see Chapter 5) are to be viable as consumer-facing disaggregation solutions then researchers must distribute pre-trained models for each appliance. If these models adhere to a standard metadata schema then multiple software systems can exchange models.

Against this background, we propose a hierarchical metadata schema for energy disaggregation. Specifically, our schema models electricity meters, appliances (including prior knowledge such as probability distributions describing typical times of use and parameters describing inferred models of appliances), buildings and datasets.

Although we have made every effort to ensure that our proposed schema and controlled vocabularies capture the information present in all the datasets we are aware of, our schema can undoubtedly be improved and so the schema is presented as an open-source project[1] (under a permissive Apache 2.0 license) to which contributions are most welcome!

6.2 Related work

The general principal of using metadata to describe research datasets is not new. For example, the not-for-profit organisation DataCite[2] (established in 2009) publishes the DataCite Metadata Schema for describing research datasets. Additional general schemas include SensorML[3] for describing general sensor data and the DDI CodeBook[4] for describing social science survey data. The OGC Observations and Measurements (O&M) standard[5] (also published as ISO/DIS 19156) defines an XML schema for sampling and making observations and there exists a specialisation of O&M for environmental data, the Metadata Objects for Linking Environmental Sciences (MOLES3) (Parton et al. 2015).

In late-2013, the Nature Publishing Group launched a journal called 'Scientific Data[6]' for describing datasets. The machine-readable description of each dataset is captured using the ISA_Tab metadata specification[7] which specifies a hierarchical schema consisting of the 'investigation' (the project context), the 'study' (a unit of research) and the 'assay' (an analytical measurement).

[1] https://github.com/nilmtk/nilm_metadata
[2] http://www.datacite.org
[3] http://www.opengeospatial.org/standards/sensorml
[4] http://www.ddialliance.org/Specification/DDI-Codebook
[5] http://www.opengeospatial.org/standards/om
[6] http://www.nature.com/scientificdata
[7] http://isa-tools.org

Biology researchers have embraced the need for metadata schemas and controlled vocabularies as demonstrated by, for example, The Open Biological and Biomedical Ontologies database[8] which aims to enable the creation of a suite of interoperable reference ontologies for the biomedical domain.

Project Haystack[9] is an open-source initiative to develop taxonomies and tagging conventions for a building's equipment and operational data. Amongst other achievements, Haystack defines a language for describing electricity meters, including which parameters each meter records and the relationships between meters, and between meters and loads. But Haystack is primarily targeted at large commercial buildings rather than domestic buildings, and does not define a controlled vocabulary for appliance names, let alone more granular detail about appliances.

The UK Energy Research Council's Energy Data Centre provides a simple schema[10] based on the Dublin Core Metadata Initiative[11] (DCMI).

The Power Consumption Database[12] is a community project which aims to collect a database of appliance power consumption information. As of April 2017, the power consumption database has 1,257 entries and is being actively added to. Also, Samsung have done research into building a common database of appliance power demand data for training NILM algorithms (Lai et al. 2012).

Bruce Nordman[13] at the Lawrence Berkeley National Laboratory (in California, USA) has written multiple reports on energy reporting over Internet Protocols (Nordman 2013; Nordman 2014; Nordman & I. Cheung 2016) and device classification. For example, he has assembled a taxonomy of Miscellaneous and Low Power Products (Nordman & Sanchez 2006).

To summarise the related work: there are many metadata projects for describing *general* datasets but only a small number of metadata projects for describing *energy* datasets. To the best of our knowledge, there are no existing metadata schemas specifically for describing objects relevant to energy disaggregation. Existing datasets for energy disaggregation do provide some metadata (e.g. a text file mapping appliance names to recording channels) but this

[8] http://www.obofoundry.org
[9] http://project-haystack.org
[10] http://ukedc.rl.ac.uk/format.html
[11] http://dublincore.org
[12] http://www.tpcdb.com
[13] http://nordman.lbl.gov

metadata does not use a controlled vocabulary and often provides scant details. Hence we set out to design a metadata schema specifically for disaggregated energy data.

6.3 Design

The NILM Metadata schema models several objects relevant to energy disaggregation: electricity meters; appliances; prior knowledge about appliances; appliance models; buildings; and datasets. The schema specifies property names for each object; the data type for each value; and controlled vocabularies (e.g. for appliance names and categories).

A UML Class diagram showing the relationship between classes is shown in Figure 6.1 and a brief example metadata instance is shown in Figure 6.2.

In the sections below, we describe our `Dataset` and `Building` schemas; the distinction between *meters* and *appliances*; the representation of electricity meters; the representation of the mains wiring; the inheritance mechanism for appliances; categorisation; the containment mechanism that allows an appliance to contain other appliances; prior knowledge; and finally our representation of learnt models.

6.3.1 Dataset

NILM Metadata places the primary objects of interest into a tree-shaped hierarchy (Figure 6.1). At the root is a `dataset` object. This contains `buildings`, each of which contains electricity meters, many of which measure the power demand of appliances.

This tree hierarchy captures all datasets we are aware of except one: the 'Tracebase' dataset (Reinhardt et al. 2012), which describes appliances *without* their building context. To handle tracebase, meter objects in NILM Metadata can be directly contained within a dataset object (without requiring a building object).

The dataset schema records properties such as `date`*, `rights_list`*, `geospatial_coverage`*, `timeframe`*, `funding`, `creators`*, `related_documents`*,

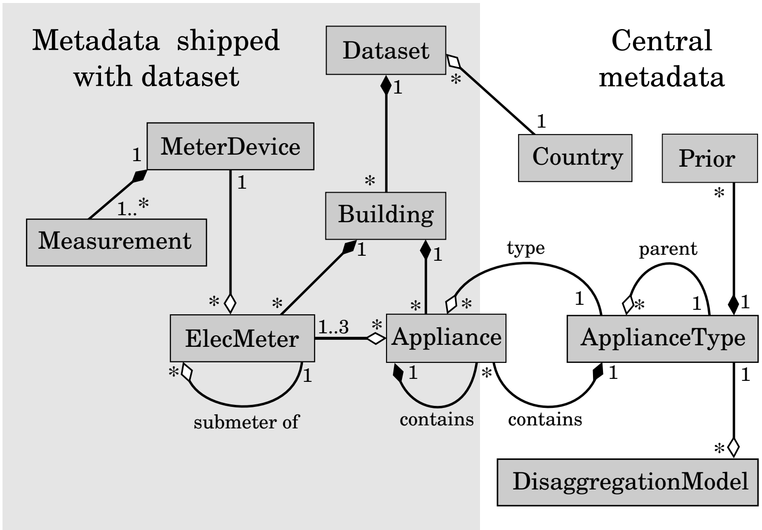
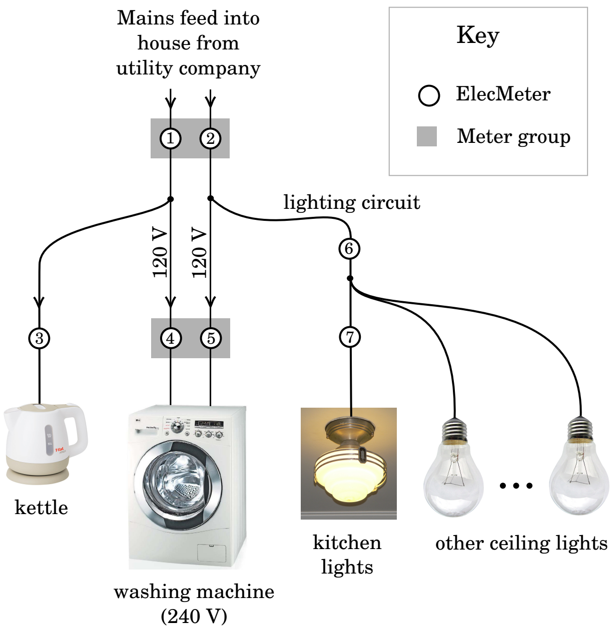


Figure 6.1: UML Class Diagram showing the relationships between classes. A dark black diamond indicates a 'composition' relationship whilst a hollow diamond indicates an 'aggregation' relationship. For example, the relationship between `Dataset` and `Building` is read as '*each `Dataset` contains any number of `Buildings` and each `Building` belongs to exactly one `Dataset`*'. We use hollow diamonds to mean that objects of one class *refer* to objects in another class. For example, each `Appliance` object refers to exactly one `ApplianceType`. Instances of the classes in the shaded area on the left are intended to be shipped with each dataset whilst objects of the classes on the right are common to all datasets and are stored within the NILM Metadata project. Some `ApplianceType`s contain `Appliance`s, hence the box representing the `Appliance` class slightly protrudes into the 'common metadata' area on the right.




```
# building1.yaml
instance: 1
elec_meters:
  1:
    site_meter: true,
    device_model: Whole-House Meter Model FOO
  2:
    site_meter: true,
    device_model: Whole-House Meter Model FOO
  3:
    submeter_of: 1
    device_model: Individual Appliance Monitor Model BAR
  4:
    submeter_of: 1
    device_model: Individual Appliance Monitor Model BAR
  5:
    submeter_of: 2
    device_model: Individual Appliance Monitor Model BAR
  6:
    submeter_of: 2
    device_model: Circuit Monitor Model BAZ
  7:
    submeter_of: 6
    device_model: Circuit Monitor Model BAZ

appliances:
- {type: kettle, instance: 1, room: kitchen, meters: [3]}
- {type: washing machine, instance: 1, meters: [4,5]}
- {type: light, instance: 1, room: kitchen, meters: [7]}
- {type: light, instance: 2, multiple: true, meters: [6]}
```

Figure 6.2: The illustration on the left shows a cartoon mains wiring diagram for a domestic building. Black lines indicate mains wires. This home has a split-phase mains supply (common in North America, for example). The washing machine draws power across both splits. All other appliances draw power from a single split. The text on the right shows a minimalistic description (using the NILM Metadata schema) of the wiring diagram on the left.

timezone and geo_location. The starred properties (*) are adapted from the Dublin Core metadata initiative (DCMI).

6.3.2 Building

Buildings are identified by an integer property instance (unique within the dataset). Each building may have a list of rooms (using a controlled vocabulary for room names), and some properties shared with dataset: timeframe, geo_location, timezone. These properties default to the values set in the parent dataset but can be overridden per building. Each building contains an elec_meters property which stores a dictionary of ElecMeter objects (each key is a meter instance integer).

6.3.3 Meters are distinct from appliances

A tempting simplification would be to assume a one-to-one relationship between electricity meters and appliances. But we often observe one-to-*many* relationships such as a single meter connected to a multi-way mains adapter which, in turn, feeds multiple appliances. And we occasionally observe many-to-one relationships: for example, in the US and Canada many large domestic appliances like washing machines draw a total of 240 volts from two 120 volt 'split-phase' supplies found in a typical house. Some datasets use one meter per 120 volt feed and hence use two meters for each 240 volt appliance. We also frequently observe situations where some appliances are not submetered. To handle the case where a single appliance receives more than one power supply (e.g. split-phase or three-phase power), we allow each Appliance object to contain a list of meter instances, each of which refers uniquely to one ElecMeter. To handle the case where a single meter is connected to multiple appliances, each ElecMeter can be referred to by any number of Appliance objects. The meter may also specify a dominant_appliance to specify if a single appliance is on more often than other appliances on that meter.

6.3.4 ElecMeters and MeterDevices

To record the specifications of each meter, we use one `MeterDevice` object per meter type. `MeterDevice` objects record properties which apply to a specific model of electricity meter. For example, it captures the `sample_period` in seconds; the `measurements` recorded by the meter (e.g. voltage, reactive power, active energy etc.); the meter `manufacturer` and `model`.

`ElecMeter` objects are distinct from `MeterDevice` objects. `ElecMeter` objects represent *each* physical meter installed in a building. Each `ElecMeter` references exactly one `MeterDevice`. `ElecMeter` is also the place where any pre-processing carried out on the data can be described (for example, have gaps been filled? Or unrealistic values been removed?)

6.3.5 Mains wiring

Each building in a typical dataset will have one meter which records the aggregate, whole-building mains power demand. Downstream of this meter might be meters which measure entire circuits within the building (e.g. the lighting circuit). Finally, there are often meters which measure individual appliances. An example wiring diagram is shown in Figure 6.2. Nordman 2014 also discusses the need for describing wiring trees.

As such, the mains wiring connecting meters with each other can be described as a tree. Each `ElecMeter` can specify either a `submeter_of` property (the numeric ID of the upstream meter) or a `site_meter` property (a boolean flag which is set to `true` if this meter measures the whole-building aggregate). The property names '`submeter_of`' and '`site_meter`' are adapted from Project Haystack. The wiring hierarchy can be any depth. In large, commercial installations, a meter in one building may be downstream of a meter in another building. This case can be handled by specifying the numeric ID of the other building using the `upstream_meter_in_building` property. If this property is absent then we assume the upstream meter is in the same building.

6.3.6 Appliance and ApplianceType

Each `Appliance` object represents an appliance *instance* in the real world. Each `Appliance` object has a `type` property which refers to an `ApplianceType` object. `ApplianceType` objects are not shipped with the dataset; instead they are stored within NILM Metadata. `ApplianceType` objects capture knowledge about appliance types (e.g. the categories each appliance type falls within; probability distributions describing the power demand for the appliance etc.). The set of names of each `ApplianceType` *is* the controlled vocabulary for appliance types.

6.3.7 Inheritance for ApplianceTypes

A 'wine cooler' can be considered a specialisation of a 'fridge'. As this example illustrates, electrical appliances can be described as a hierarchical tree.

Inheritance is a well-established technique in software engineering for maximising code re-use. NILM Metadata implements a simple but powerful form of inheritance known as prototype-based inheritance (first implemented in the Self programming language Chambers et al. 1989 and used in JavaScript). Objects in prototype-based programming languages are not instances of a class but, instead, inherit from any other object (the 'parent' or 'prototype' object). In NILM Metadata, each `ApplianceType` object has a `parent` from which it inherits properties. These properties can be modified by the child and the child can specify properties not specified by the parent. The inheritance tree can be any depth.

Inheritance follows a small number of rules. If a property is contained in the parent and absent in the child then it is copied to the child. If a property is present in both parent and child then it is handled differently depending on the type of the property:

1. **list (array)** objects become the union of the parent and child lists.
2. **scalar** objects in the child override ('shadow') properties in the parent.
3. **objects** (dictionaries) are recursively updated using the rules above.

Child objects can specify a `do_not_inherit` property to specify a list of property names which should not be inherited.

Subtypes versus inheritance. Appliance objects have a `subtype` property (which must be set to a member of the appliance type's `subtypes` set). What is the difference between a subtype and a child object? Subtypes are useful when two related appliances are so similar that we can safely ignore the differences for the purposes of energy disaggregation. For example, an analogue radio and a digital radio are sufficiently similar to mean that they can both be subtypes of the 'radio' object. On the other hand, an electric cooker has a significantly different electricity load profile compared to a cooker fuelled by natural gas, so these are separate objects.

Additional properties. Some appliances have rare properties. For example, a television might have a `screen_size` property. We do not want to pollute the common 'appliance' schema with these properties (because, for example, it makes no sense for a cooker to be able to specify a `screen_size` property!). Instead, appliance objects can define an `additional_properties` property. This specifies the schema for any additional properties (using JSON Schema). `additional_properties` is inherited using the same rules as any other property.

6.3.8 Appliance categorisation

When analysing domestic power consumption, we often want to group appliances into certain categories. For example, we might want to ask 'what is the total energy consumption for all consumer electronics?'.

Domestic appliances are traditionally classified as one of 'wet', 'cold', 'consumer electronics', 'ICT', 'cooking', 'lighting' or 'heating'.

An alternative classification is a simple binary classification of 'large appliances' (e.g. dish washer) versus 'small appliances' (e.g. a radio).

A more finely-grained classification based on the electrical properties of appliances was proposed by Tsagarakis et al. 2013. For example, the taxonomy proposed by Tsagarakis et al. splits lighting into general incandescent lamps, fluorescent lamps and light-emitting diode (LED) sources. Each appliance can have multiple classifications from this taxonomy.

Yet another taxonomy for domestic appliances is the Google product taxonomy[14] (used on Google Shopping). This taxonomy is a tree which we represent as list of classifications.

[14] https://support.google.com/merchants/answer/160081

NILM Metadata currently supports all four taxonomies listed above and it would be trivial to add more. We specify a controlled vocabulary for the category names. Our appliance schema specifies a `categories` property which is an object with the following properties: `traditional` (*string*), `size` (*string*), `electrical` (*array of strings*) and `google_shopping` (*array of strings*). At present, all `ApplianceTypes` the NILM Metadata have a 'traditional' classification and many have classifications for the other taxonomies.

6.3.9 Appliances can contain other appliances

Some appliances can be modelled as a container of other objects. For example, a washing machine can be modelled as a drum motor and a water heating element (and a few other components). `Appliance` and `ApplianceType` objects in NILM Metadata have a `components` property which stores an array of appliance objects. Containment is recursive and can be of any depth. Nordman 2014 also talks of Devices (the entire appliance) which can contain Components.

Of course, *all* appliances can be decomposed into components. Do we model each individual resistor and transistor? No; the end-goal is to model appliances only in sufficient detail to allow an energy disaggregation system to identify the whole appliance given prior knowledge of the components. As such, we only describe individual components if their electrical behaviour is observable from a typical mains electricity meter. It is also important that components be truly separate entities from an electrical perspective. For example, a fridge freezer should *not* be modelled as containing both a fridge and a freezer because that would imply that a fridge freezer has two separate compressors but - as far as we are aware - fridge freezers typically have one compressor.

If an appliance contains multiple instances of the same component then we use the `count` property in the component to specify the number of instances. If an appliance contains multiple instances of the same component but the exact number of components is unknown then set `multiple` to 'true'.

The container appliance inherits categories from each of its components. This is useful mostly for the 'electrical' taxonomy. For example, if we model a washing machine as a motor and a heater then the washing machine inherits the appropriate electrical classifications from both the motor and the heater.

Our representation of lighting exploits NILM Metadata's containment mechanism. We distinguish between the light fitting (also called the luminaire or fixture) and the electric lamp(s) within each fitting. We have a 'light' object which contains any number of 'lamps' (of which there are several kinds including 'LED lamp' and 'incandescent lamp'). Light objects can also contain a 'dimmer' object.

6.3.10 Prior knowledge

Disaggregation algorithms could refine their output by exploiting prior knowledge such as the typical time of day each appliance is used or correlations with other appliances (e.g. the computer monitor is often on when the computer is on).

NILM Metadata specifies a `Prior` class which holds several properties, including:

`distribution_of_data`: The distribution of the data expressed as normalised frequencies per discrete bin (for continuous variables) or per category (for categorical variables).

`model`: Describes a model fitted to describe the probability density function (for continuous variables) or the probability mass function (for discrete variables).

`source`: Where did the prior come from? Is it a subjective guess, or the result of primary data analysis, or taken from a published paper?

`specific_to`: Is the prior relevant to a defined set of countries?

`training_data`: What data was used to 'train' the prior?

The `ApplianceType` class specifies an optional `distributions` property which is a dictionary with the following keys (the value corresponding to each key is an array of `Prior` objects): `on_power`, `on_duration`, `off_duration`, `usage_hour_per_day`, `usage_day_per_week`, `usage_month_per_year`, `rooms`, `subtypes`, `appliance_correlations`, `ownership`, `ownership_per_country`, `ownership_per_continent`.

We store an *array* of priors (rather than a single prior) for each distribution. This allows us to store multiple beliefs about each distribution (which could be combined using Bayesian statistics). For example, we might find several published papers which provide evidence about the distribution over the power consumption of an appliance. Furthermore, NILM Metadata collects all relevant priors as it descends the inheritance hierarchy for each object (for example, a 'wine cooler' might not have any priors associated with it but it will inherit prior knowledge from its parent 'fridge' object). Of course, priors from a distant ancestor are less relevant than priors from a recent ancestor so, as we traverse the inheritance tree, we tag each prior with a `distance` property (a positive integer indicating the number of 'generations' away the prior is from the appliance in question).

6.3.11 Learnt models of appliances

End-users of domestic disaggregation software are unlikely to put any effort into training the system. This means that we must use either a supervised learning algorithm with pre-trained models or an unsupervised disaggregation algorithm (which must still have some form of prior appliance model to be able to provide human-readable names for each appliance). To store learnt model parameters for each appliance type, we specify a simple `DisaggregationModel` class. This has properties such as `model_type` (a controlled vocabulary with terms such as 'HMM' for hidden Markov model), `training_data, date_prepared` etc. The model's parameters are stored in a `parameters` object.

The `DisaggregationModel` schema can be used to fully specify an appliance model that has been learnt from the data. There can be one learnt model for each model type (e.g. 'HMM') and for each appliance type (e.g. 'fridge'). This allows the model parameters to be produced and shared by a researcher and then any user can take advantages of these models for disaggregation.

The end result is that only a small number of researchers need to put effort into generating models and then anyone with an internet connection can make use of the models in a disaggregation system.

Please see Appendix C for additional details regarding the Python implementation of the metadata schema; the file organisation; and a short example.

6.4 Limitations and future work

At present, the schema cannot accommodate micro-generation (e.g. solar panels).

The schema is not as easy to use as it could be. For example, an automatic validator for metadata has been requested by some users. As has an automatic 'wizard' to guide users, step-by-step, though the creation of a valid metadata description of their dataset. Finally, it would help if the schema defined some sensible defaults so users did not have to specify so many things in their metadata.

6.4.1 Simplification

On the one hand, we are yet to come across a NILM dataset which *cannot* be described using NILM Metadata. On the other hand, the majority of datasets that we have encountered do not require the full expressive power of the schema. The community would probably benefit from a 'simple profile' of NILM Metadata schema.

There are plenty of features of NILM Metadata which appear to have *never* been used in practice so these could simply be removed from the specification:

- `DisaggregationModel`
- `Prior`
- The ability for an `Appliance` to contain other `Appliance` objects (e.g. for a light fitting to contain multiple lamps)

Also, it turns out that people would prefer CSV files to contain the metadata rather than YAML files because CSV files can be read easily by every 'data science' platform and programming language.

Further discussion of how best to simplify the schema is available on the NILM Metadata's github issue queue: https://github.com/nilmtk/nilm_metadata/issues/16

6.5 Usage within the community

The schema was first released in 2014. It is now an integral part of NILMTK and, as such, is used by all 11 dataset converters in NILMTK, many of which have been contributed by the dataset authors themselves. Judging from emails and conversations at workshops, it seems likely that the schema has also been used 'internally' by several researchers, although it is hard to quantify this.

6.6 Conclusions

We have proposed a metadata schema for representing objects relevant to energy disaggregation. The schema adapts ideas from DCMI, Project Haystack, ISA_Tab and DataCite; and adds new elements relevant to energy disaggregation. We also propose a simple but powerful inheritance mechanism to minimise duplication of information and effort. The schema has successfully been used to capture metadata for at least eleven datasets.

Whilst NILM Metadata is fit for use now, there will inevitably be use-cases that we have neglected hence we warmly welcome contributions from the community! NILM Metadata is open-source to facilitate collaboration and is available at github.com/nilmtk/nilm_metadata

Part III

Disaggregation algorithms

Chapter 7

Review of NILM algorithms

In this chapter, we discuss some prior NILM algorithms and why deep neural networks might be expected to perform better than these previous approaches.

There are two broad approaches to disaggregation: optimisation ("non-event-based NILM") and pattern recognition ("event-based NILM").

7.1 Disaggregation framed as an optimisation problem

The optimisation problem can be described as follows: a smart meter signal is a time-series $Y = \{y_1, y_2, ..., y_k\}$ where k is the number of samples and y_t is the meter reading at time t. We describe the state of each appliance with a boolean vector $A = \{a_1, a_2, ..., a_n\}$ for n appliances. If appliance i is *on* then $a_i = 1$; else $a_i = 0$. The power consumption of each appliance is described in vector $P = \{p_1, p_2, ..., p_n\}$. For example, if appliance 1 is a fridge which uses 100 watts and is currently *on* then $a_1 = 1$ and $p_1 = 100$. The total power consumption y_t at time t is the sum the power consumption of all active appliances:

$$y_t = \sum_{i=1}^{n} a_i p_i + e_t \tag{7.1}$$

Where e_t is an error term.

If the power consumption of every appliance is know then disaggregation can be stated as a combinatorial optimisation problem where, for every time slice t we try to find a state vector A_t^* such that:

$$A_t^* = \arg\min_A \left| y_t - \sum_{i=1}^{n} a_i p_i \right| \tag{7.2}$$

7.1.1 Optimisation is computationally intractable

George Hart, one of the early pioneers of disaggregation research, points out that the optimisation problem specified in equation 7.2 is an NP-complete "weighted set" problem and that a precise solution is only achievable by enumerating every possible state (G. W. Hart 1992). This is computationally impractical because n appliances, each of which can occupy any one of s states, can be configured in s^n combinations so the computational complexity blows up exponentially as $\mathcal{O}(s^n)$. Say we have thirty appliances, each of which can be in one of four states, and we have a month of data sampled once every five seconds. That is approximately 10^{24} operations[1], which would take 5×10^{10} seconds ($\sim$ 1 700 years) on NVIDIA's top-of-the-line GPU at the time of writing[2].

Whilst the optimisation problem specified in equation 7.2 is a succinct description of the problem, it fails to capture many of the challenges present in practical systems. These problems include (but are not limited to):

1. We are unlikely to know the power consumption of every appliance.
2. We are unlikely to know the total number of appliances.
3. Many appliances do not draw tidy, discrete levels of power; instead their power consumption may spike, undershoot, oscillate or ramp over time.
4. A smart meter may sample less frequently than is required to faithfully capture rapid changes. In other words, the meter may sample at sub-Nyquist rates. This results in considerable distortion of the digital recording.

[1] Each sample y_t requires $s^n = 4^{30}$ multiplications and $s^n = 4^{30}$ additions. Total number of samples $k = 30$ days per month $\times$ 24 hrs per day $\times$ 60 mins per hour $\times$ 60 secs per min $\div$ 1 sample every 5 seconds. Total number of operations $= 2 \times 4^{30} \times k \approx 10^{24}$.

[2] The NVIDIA GTX 1080 Ti, a top-end gaming GPU released at the end of March 2017, can do 11.3 TFLOPS (fused multiply-adds).

5. Many appliances (like washing machines and tumble driers) have multiple internal states. Transitions between these states may be non-deterministic. Each run of the appliance may produce a different waveform (see Figure 7.1).

6. Different appliances of the same class produce different waveforms (which is a problem if we want to build a common database of appliances for multiple users).

7. Some appliances generate identical waveforms (e.g. a kettle and the water heater in a washing machine generate very similar waveforms).

8. Appliance signatures overlap and occlude each other in the aggregate smart meter signal.

9. The mains voltage in the UK is nominally 230 volts but can range from 216 volts to 253 volts which is -6%, +10% of the nominal 230 volt supply voltage[3]. Assuming a linear load, we can expect the power consumption to vary by -12%, +20%. Home energy meters do not measure voltage, but utility-installed smart meters do.

10. Ultimately users care more about *how much energy* each appliance uses rather than *when* each appliance is on. Estimating energy consumption for a simple two-state appliance like a toaster is trivial if we know how long the appliance has run for. But estimating power consumption of complex appliances like washing machines is less trivial.

These challenges mean that conventional optimisation approaches such as combinatorial optimisation are not feasible for anything other than toy scenarios.

7.2 Blind source separation

Blind source separation (BSS) refers to a group of methods which includes principal component analysis (PCA), singular value decomposition (SVD), independent component analysis (ICA) and non-negative matrix factorisation (NNMF).

In general, BSS aims to recover a set of individual source signals $s(t) = (s_1(t), ..., s_n(t))^T$ (where t is the time step from 1 to T and n is the number of source signals) from a set of mixed signals

[3] www.decc.gov.uk/en/content/cms/meeting_energy/en_security/gas_electric/electricity/electricity.aspx Note that the rather ugly formulation of "230 volts -6%, +10%" is set to change to "230 volts ±10%".

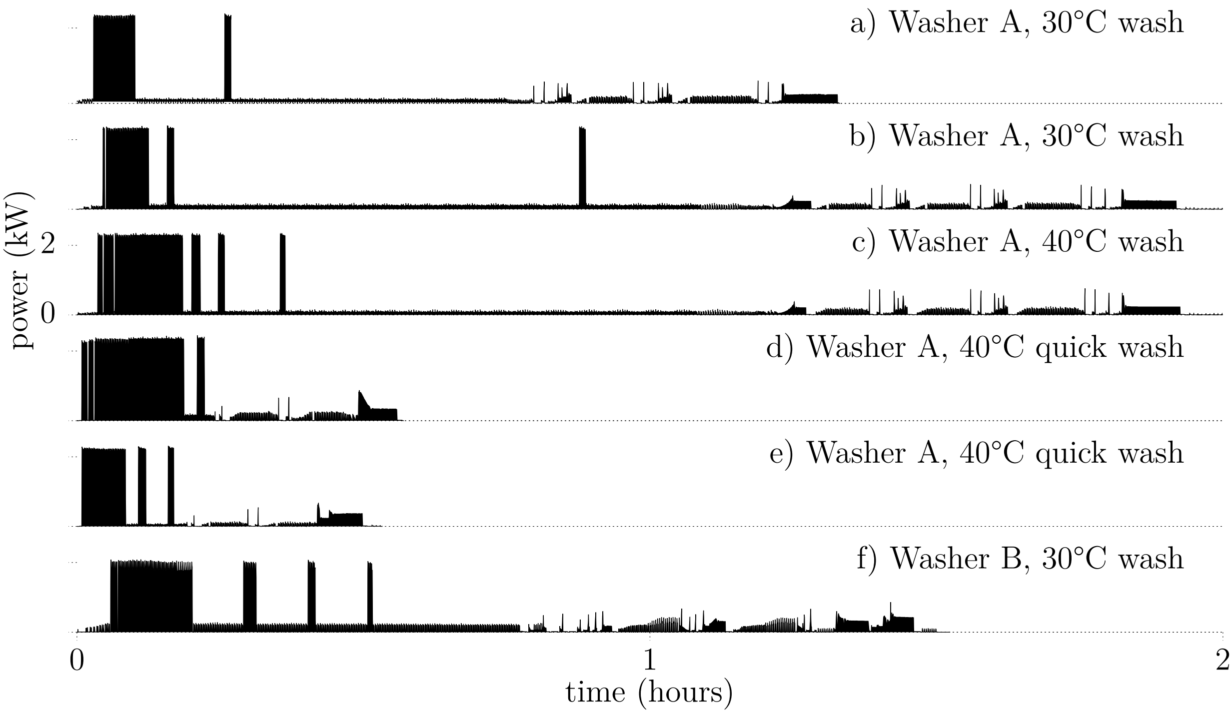


Figure 7.1: Six washing machine waveforms. *a* - *e* were produced by the same washing machine.

$x(t) = (x_1(t), ..., x_m(t))^T$ (where m is the number of mixed signals). Typically in BSS, the number of mixed signals m is equal to the number of source signals n. For example, in the "cocktail party" problem, we have a number of people talking at the same time and we also have the *same* number of microphones dotted around the room. In this setting, BSS can un-mix the audio to produce one audio track for each person.

In NILM, we typically have $m < n$ (for example, we might have a single smart meter which measures 50 appliances). Hence the classic formulation of BSS may not be suitable for NILM.

7.3 Extracting steady states from a smart meter signal

Most disaggregation systems within our scope share a common first step (and I will argue later that our system should *not* start with this step because it throws away valuable information). This first step is to identify "steady states" in the smart meter signal (see figure 7.2). The aim is to deliberately remove rapid transients in the signal such as, for example, when a motor first turns on it draws a large amount of power for a short period as it accelerates, and then the power consumption drops to a steady plateau. After steady states have been identified,

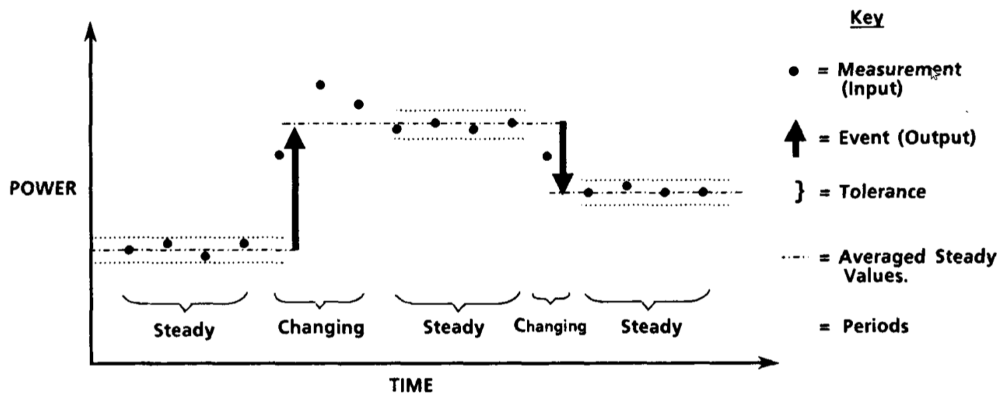


Figure 7.2: Hart's "steady-state" detection. Taken from G. W. Hart 1992.

changes in steady states are identified. These "change events" are output to the next step in the signal processing pathway.

What exactly qualifies as a "steady state"? Hart describes the criteria he used in some detail (G. W. Hart 1985; G. W. Hart 1992). First, the signal is partitioned into "steady" and "changing" sections, as illustrated in figure 7.2. A "steady" section must have a certain minimum length (Hart used three samples, which for his 1 Hz system is equivalent to three seconds) and the signal must not vary by more than 15 watts or VAR in any component. All sections not labelled "steady" are labelled "changing". The samples in each "steady" section are averaged. Finally, changes in "steady" states are identified. The first sample in the corresponding "changing" period provides the time stamp. Changes below a certain minimum threshold are discarded. The final output is a list of <*timestamp, value*> pairs.

An alternative algorithm for determining steady-states based on the ratio value between rectangular areas defined by the signal samples is developed in M. Figueiredo et al. 2010 and tested in M. B. Figueiredo et al. 2011; M. Figueiredo et al. 2012.

One of my central hypotheses is that converting a smart meter signal to a sequence of steady states looses information and that it would be better to take advantage of transient and frequency-domain features to increase disaggregation performance. Figure 7.3 shows the power consumed by a tumble drier, sampled at 1 Hz. Note the low-amplitude ($\approx$ 200 W) repetitive spikes during the first forty minutes (these correspond to the drum spinning one direction for about one minute; stopping; and reversing for another minute etc.). Then, after forty minutes, the heating element cycles on and off to prevent the drier from exceeding some maximum temperature; a behaviour which produces high-amplitude ($\approx$ 2 kW) spikes in the power con-

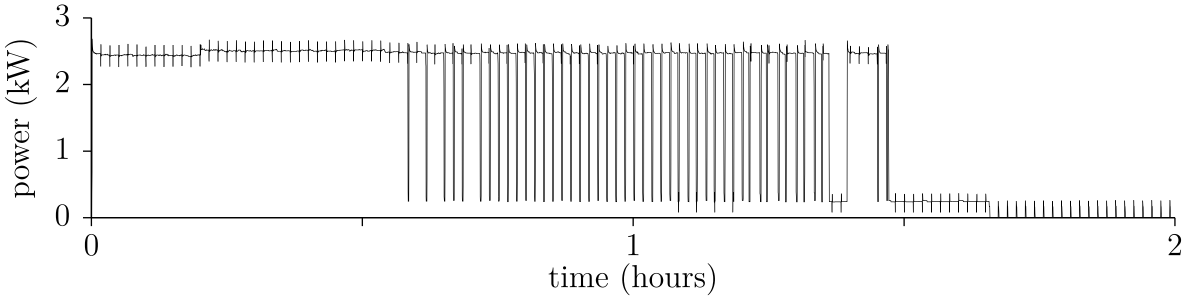


Figure 7.3: Tumble drier signature sampled at 1 Hz

sumption.

Why are these rapid changes a problem? Steady-state algorithms have two options when faced with an appliance like a tumble drier: either smooth out the rapid changes (hence losing a lot of information: those high frequency features are rather distinctive) or attempt to track the high frequency changes (hence violating the design assumption that a steady state is *steady*). We could also consider the TV waveform in figure 8.1 where the signature's "noise" is a decidedly distinctive feature yet this would be completely smoothed out by a steady-state algorithm. A further issue with rapidly-changing waveforms is that the smart meter may sample the aggregate signal at sub-Nyquist rates, hence aliasing the signal.

7.4 Hart's non-intrusive load monitoring algorithm

The "non-intrusive load monitoring" (NILM[4]) technique was developed primarily by George Hart between the mid-1980s to mid-1990s at various institutions, starting at MIT (G. W. Hart 1984; G. W. Hart 1992; G. W. Hart 1995). Hart, Kern and Schweppe of MIT patented the technique in 1989 (G. W. Hart et al. 1989). George Hart is often cited as the "godfather" of disaggregation.

G. W. Hart 1992 describes a 'signature taxonomy' of features (see Figure 7.4) which might be useful for distinguishing between appliances. His earliest work from 1984 described experiments of extracting more detailed features[5]. However, Hart subsequently decided to focus on extracting only transitions between steady-states. Many NILM algorithms designed for low

[4]Hart originally called this technique "Nonintrusive Appliance Load Monitoring" which he abbreviated to NALM, but the term "NILM" is more common today.

[5]This claim is taken from G. W. Hart 1992 because no copy of George Hart's 1984 technical report was available.

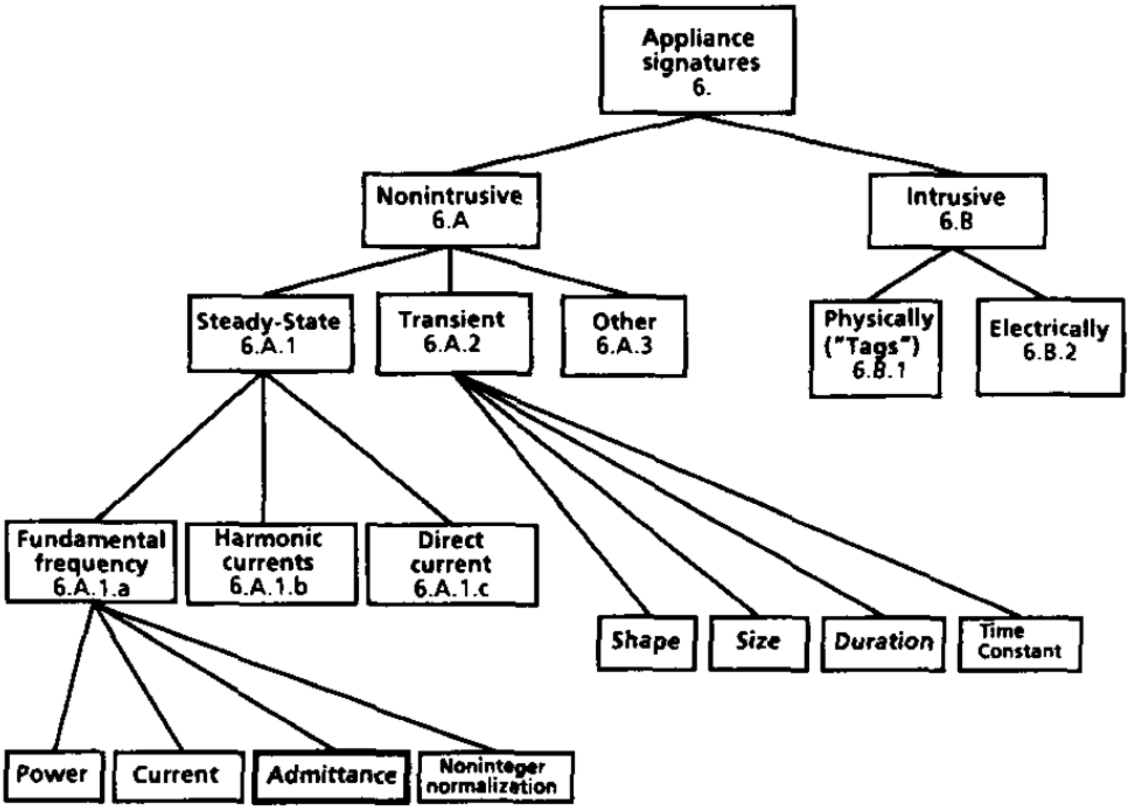


Figure 7.4: George Hart's 'signature taxonomy'. Source: G. W. Hart 1992 (with permission).

frequency data (1 Hz or slower) follow Hart's lead and only extract a small number of features. In contrast, in high frequency NILM (sampling at kHz or MHz), there are numerous examples in the literature of manually engineering rich feature extractors (e.g. Steven B Leeb et al. 1995; Amirach et al. 2014).

George Hart's NILM algorithm takes as its input readings of *real* power, *reactive* power and voltage sampled once a second (for a brief introduction to the difference between *real* and *reactive* power please see Appendix D but for the following discussion it is only necessary to know that real power and reactive power are two different parameters we can measure for any alternating current load). Variations in mains voltage are corrected by normalising the power readings. Steady states are identified and then *changes* between steady states are located.

For example, consider a 500 watt heater turning *on* and *off*. The real power reading will *increase* by 500 watts the moment the heater turns *on* and will *decrease* by 500 watts the moment the heater turns *off*.

Changes with equal magnitudes and opposite signs are then paired. In our example, the +500 watt event corresponding to the heater turning *on* would be paired to the -500 watt event

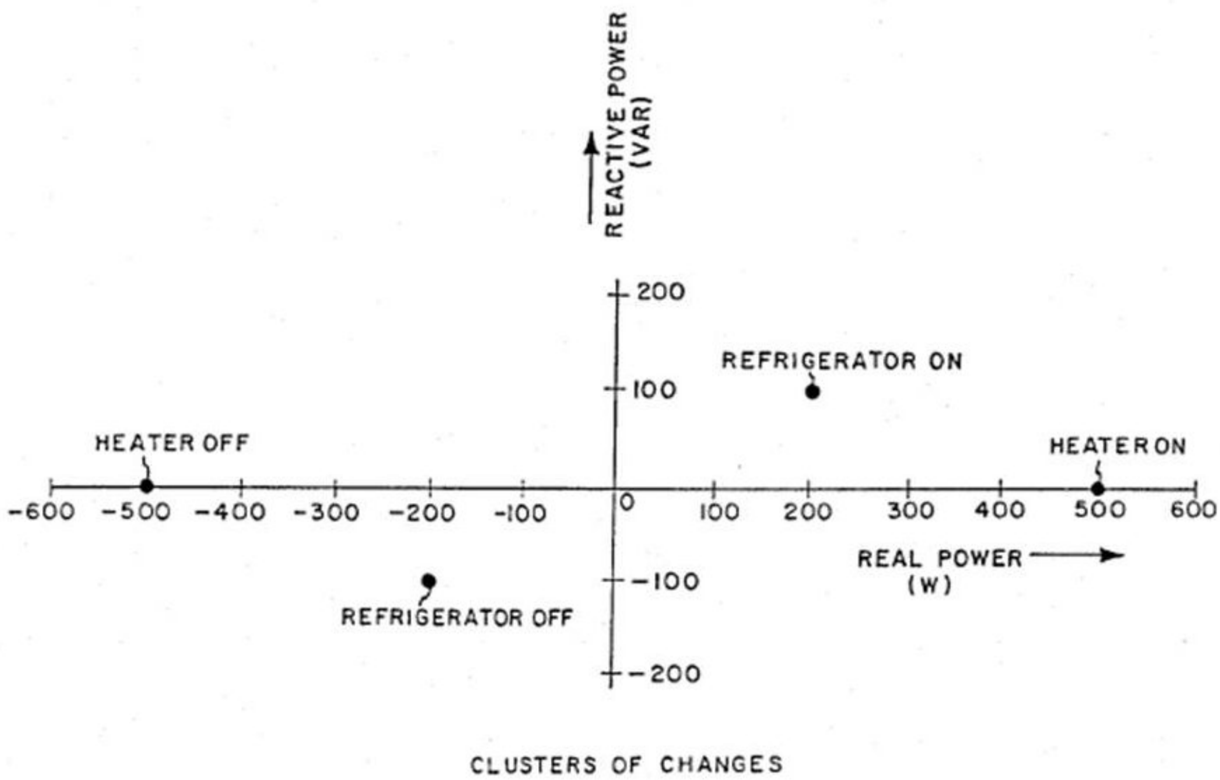


Figure 7.5: Distinguishing between a heater and a fridge by comparing real and reactive power consumption. The heater is a purely resistive load and hence pulls no reactive power. The refrigerator mostly pulls real power but also pulls some reactive power. These two variables allow us to discriminate between most devices. This diagram is taken from US patent 4858141, filed by George Hart and colleagues from MIT in 1989 (G. W. Hart et al. 1989).

corresponding to the heater turning *off* and hence the NILM algorithm can deduce the length of time the device has been active and the total energy consumption for each device.

Most devices draw both *real power* and *reactive power*. Different devices draw different proportions of real and reactive power. Hence, if we measure both real and reactive power then we can plot devices on a two-dimensional plot like the one in figure 7.5. This allows us to tease apart some appliances which may be indistinguishable when measuring just real power. The draft specifications for UK smart meters (DECC 2014) requires that the UK's smart meters must measure both real and reactive power but home energy monitors like the Current Cost only measure apparent power.

The original NILM algorithm consists of these five steps:

Hart identified three classes of devices (G. W. Hart 1992):

1. *on / off* (e.g. toaster, light bulb, kettle)

Algorithm 1 The original NILM algorithm

1. An "edge detector" identifies changes in steady-state levels in the normalised aggregate power consumption (see figure 7.5).
2. Cluster analysis is performed to locate these changes in a two-dimensional *signature space* of real and reactive power (figure 7.5). Hart et al. developed a clustering algorithm which works in a single pass through the data. The algorithm involves "splitting" and "merging" clusters as new data-points are encountered (G. W. Hart 1985; G. W. Hart 1995).
3. Clusters of similar magnitude and opposite sign are paired. This catches simple, "two-state" loads like heaters which can only be either *on* or *off* but fails to fully pair clusters attributable to complex devices like dish washers which have many states.
4. Unmatched clusters are associated with existing or new clusters according to a best likelihood algorithm. This step is known as *anomaly resolution*.
5. Events are assigned human-readable labels (e.g. "kettle" rather than "load#213") by matching events to a database of known device power consumption learnt during a training phase.

---

2. finite state machines (e.g. washing machine, tumble drier)
3. continuously variable (e.g. dimmable lights)

The NILM algorithm models each multi-state appliance (like washing machines) as a finite state machine (G. W. Hart 1995; Rouvellou & G. W. Hart 1995). A washer typically has a water heater and a drum motor. The NILM algorithm attempts to identify that the heater and motor always come on together so that "washing machine" power consumption can be reported to the user rather than "heater" and "motor" power consumption. As we will discuss shortly, the NILM algorithm did not entirely succeed in modelling multi-state appliances.

How does the NILM approach disaggregate an aggregate signal given a set of learnt finite state models? G. W. Hart 1992 notes that the Viterbi algorithm is a promising starting point but that the Viterbi algorithm cannot correct errors in which symbols (changes between steady states) are inserted into or deleted from a message sequence. Hence Hart and colleagues developed a generalised version of the Viterbi algorithm which optimally corrects insertions, deletions, mergers and many other types of error (A. Bouloutas et al. 1991; G. W. Hart 1992; G. W. Hart & A. T. Bouloutas 1993).

How does the NILM approach learn finite state models in the first place? A brief overview is given in G. W. Hart 1992 and the details are given in G. W. Hart 1994; G. W. Hart 1995 but unfortunately I have not yet been able to locate copies of these references. Rouvellou & G. W. Hart 1995 describes a solution to a similar problem.

There are two classes of NILM algorithm (G. W. Hart 1992):

1. Manual-Setup NILM (MS-NILM). This requires a manual setup stage where the system asks the installer to turn each appliance on and off in turn.

2. Automatic-Setup NILM (AS-NILM). This uses *a priori* knowledge about the characteristics of appliances. No manual setup is required. As far as the user and installer are concerned, the AS-NILM is plug-and-play.

The first MS-NILM system was built and field tested in in 1984 (G. W. Hart 1984) and the first AS-NILM system a year later (G. W. Hart 1985).

An especially fascinating section of G. W. Hart 1992 discusses disaggregation from an information theoretic perspective in which appliances are considered to be emitting symbols (signature features) across a common bus (the mains wiring). Under this framework, the job of the disaggregation system is to act as a receiver / decoder.

For reviews of NILM techniques, see G. W. Hart 1992; Laughman et al. 2003; Zeifman & Roth 2011; Froehlich et al. 2011; Zoha et al. 2012; Makonin 2012; Ying & Peng 2013; Wong et al. 2013; Armel et al. 2013; A. Ridi et al. 2014; Bonfigli et al. 2015; Adabi et al. 2015.

7.4.1 Performance of Hart's NILM algorithm

How well does Hart's NILM algorithm perform in field tests? A report prepared for the Electric Power Research Institute (EPRI) (Paragon Consulting Services LLC 1998) (Hart worked at EPRI for a few years) reported the results of a field test of commercial NILM devices. It is not clear whether these NILM devices implemented every one of Hart's ideas but it is likely that they implemented a sizable subset of Hart's ideas given that Hart appears to have finished working on NILM around 1995 and the EPRI field tests were conducted in the late 1990s. The report states (my emphasis):

> "*The conclusions of the beta test were based on 26 sites and 128 appliances monitored... the [NILM] software was able to determine two-state loads ('on' or 'off') in the mid 90% range accuracy (using the parallel metering data as the baseline). Refrigerators were measured, on average, in the mid 80% range. Multi-state appliances (e.g. dishwashers, clothes washers, heat pumps) registered lower results, indicating the need for improvement in the algorithm.*
>
> *...Although effective as a load research tool for single-state appliances,* ***enhancements must be made to the [NILM] algorithm to improve the monitoring for multi-state appliances and variable-speed loads. Without the ability to monitor all types of appliances within a residence, [NILM] does not provide a full-featured monitoring system.***"

The EPRI also published a report detailing the results of a beta test of a NILM device in a commercial setting (Hadden 1999):

> *"In several cases, the instrument was able to detect the presence of over twenty individual loads. Serious limitations were identified in the current version, however. The [commercial NILM] units were unable to:*
>
> - *Identify individual loads if they turned on or off simultaneously.*
> - *Successfully identify and properly calculate the energy consumption of multiple identical loads.*
> - *Combine the individual loads in 'multi-state' devices such as refrigerators into one load identifiable as 'refrigerator', although the individual constituent loads were often recognizable.*
> - *Correctly calculate the energy consumption of loads with variable thermal characteristics such as refrigerators.*
> - *Identify a load signal and correlate that signal to a named load. A trained technician must manually assign a load name to the loads identified by the instrument."*

7.4.2 Limitations of Hart's NILM algorithm

- Only uses steady-state features (deliberately ignores transient events).
- Appliances consuming similar power may not be separated correctly.
- Requires measurements of both *real* and *reactive* power at 1 Hz or faster.
- Deliberately ignores appliances with power consumption below 150 watts (modern appliances like game consoles, PCs etc. use less than 150 watts, as do most electric lights).
- Cannot process continually variable devices; and has poor performance on multi-state appliances.
- Many modern appliances have similar power factors hence it may be hard to tease them apart using *real* and *reactive* power.

7.4.3 Extending Hart's NILM algorithm

Transients

Consider two different types of load: 1) a washing machine's motor accelerating from standstill to full-speed; and 2) a large TV. Both loads might draw 200 watts when running. One difference between them is that the washing machine motor controller gently accelerates the motor over 30 seconds so the power consumption ramps from 0 watts to 200 watts over a 30 second period; whilst the TV instantly draws 200 watts. The washing machine's "transient" power ramp can be used as an identifying feature. Can these "transients" be used to increase disaggregation performance?

G. W. Hart 1992 discussed the difference between "steady state" and "transient" signature features. For example, an electric motor draws a large "in-rush" current the moment it starts up but the motor's power draw quickly stabilises to a steady state power draw. G. W. Hart 1992 contains a fascinating discussion on the incorporation of "transients" into the NILM framework. Hart decided not to use transients for a number of reasons, including the fact that they are rather intermittent.

Norford and Leeb, both at MIT, extended the NILM algorithm to process start-up transients (Steven B Leeb et al. 1995; Norford & Steven B. Leeb 1996; Steven B. Leeb & Kirtley 1996; Steven B. Leeb et al. 1998; Shaw et al. 1998; Shaw & Steven B. Leeb 1999). They observed that, when using sample rates considerably higher than 1 Hz, most loads have repeatable transient profiles. Disaggregation based on recognition of transients permits near-real-time identification of devices. Transients in the aggregate data are identified by comparing them to a set of *exemplar* transients learnt during a training phase. To enable transients to be identified even if multiple transients overlap, transients are broken down into a series of segments and these segments are matched using a traversal filter. This approach is described in detail in Steven B Leeb et al. 1995. Other studies which use transient features extracted from high-frequency sample rate smart meter data include Wilkinson & M. D. Cox 1996; Khan et al. 1997; Cole & Albicki 1998b; Shaw & Steven B. Leeb 1999; C. H. Kim & Aggarwal 2000; H.-H. Chang et al. 2010; R. Cox et al. 2006; H.-T. Yang et al. 2007; Reeg & Overbye 2010; Davies 2011; Lin & Tsai 2011.

Norford and Leeb used sample rates well in excess of 1 Hz. Yet the UK's smart meters will sample once every ten seconds. Hence these smart meters cannot reliably detect transients which last less than 20 seconds. This fact appears to have prevented researchers from using transient detection with low-sample rate meters. However, I argue that we *should* explore ways to take advantage of transients. Intermittent transients can be handled in a probabilistic fashion. Some appliances exhibit ramps which last for many seconds (e.g. a washing machine motor might gently accelerate over half a minute) and many appliances exhibit oscillations which could be used as discriminative features.

Using additional features

Hart's algorithm only extracts a single feature from the meter signal: changes between steady states. Cole and Albicki extend Hart's algorithm to take advantage of two additional features of appliances with large power draw: "edges" (the initial spike observed at start-up) and "slopes" (the slower variation that occurs during turn-on events) (Cole & Albicki 1998b; Cole & Albicki 1998a). See Figure 7.6 for examples of an "edge" and a "slope".

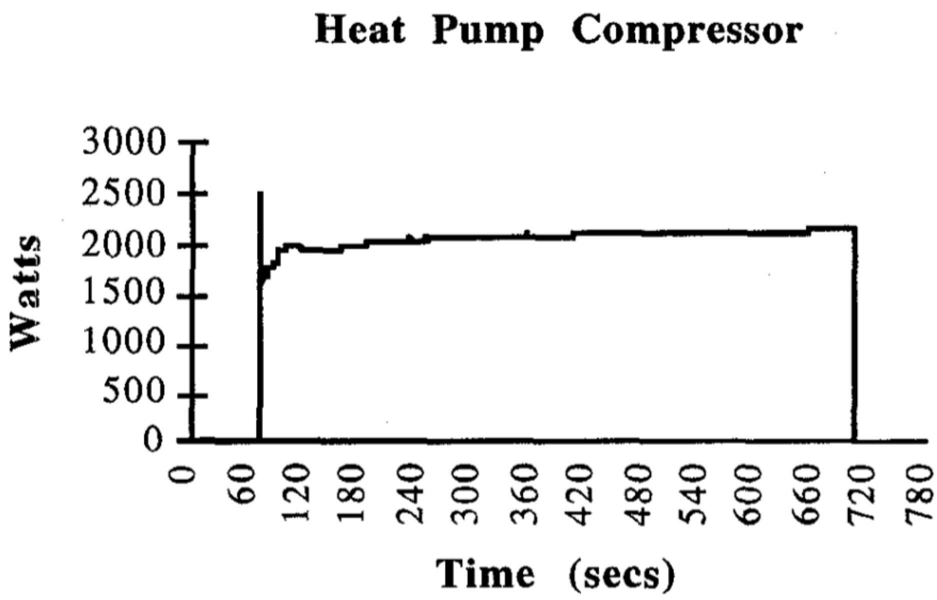


Figure 7.6: A heat pump compressor waveform. The "edge" is the initial spike at about 70 seconds. The "slope" is the increase from 70 seconds onwards. Taken from Cole & Albicki 1998b.

7.5 Sparse coding

J. Z. Kolter et al. 2010 developed a novel extension to a machine learning technique known as *sparse coding* to disaggregate home energy monitor data with a temporal resolution of one hour. Their method uses "structured prediction" to train sparse coding algorithms to maximise disaggregation performance.

This approach builds upon sparse coding methods developed for single-channel source separation. A sparse coding algorithm is used to learn a model of each device's power consumption over a typical week from a large corpus of training data. These learnt models are combined to predict the power consumption of devices in previously unseen homes. Given the very low temporal resolution of the aggregate data, it is impressive that this technique achieves a test accuracy of 55%.

7.6 Hidden Markov models

Hidden Markov models (L. E. Baum & Petrie 1966; L. E. Baum & Eagon 1967; L. E. Baum & Sell 1968; L. E. Baum et al. 1970; L. Baum 1972) have been well-studied in the NILM literature. The idea is to use the hidden state as the state of the appliance in question, and to use the power demand as the observation. We learn a transition matrix to describe the probability of the appliance transitioning between states. An extension to the hidden Markov model, the *factorial* hidden Markov model (FHMM) uses multiple hidden Markov chains (Ghahramani & Jordan 1997). At each time step, the observation is some aggregation of the observations from each individual Markov chain. In NILM, we typically consider FHMMs where the observation is the *sum* of the output from each individual Markov chain. Hence each individual Markov chain represents each appliance in the home; and the observation is the aggregate power demand.

7.6.1 Conditional factorial hidden semi-Markov models

In 2011, Kim & Han at the University of Illinois collaborated with Marwah, Arlitt and Lyon from HP Labs Palo Alto to develop an unsupervised disaggregation approach (H. Kim et al. 2011) which shares many of our design aims: they assume only *low frequency measurements* are available and they use *empirical data* collected from seven homes over a six month period. However, Kim et al.'s approach is dissimilar to ours in that theirs is an *unsupervised* learning system: it trains itself from the aggregate signal. Besides describing a novel approach to NILM, Kim et al.'s paper is also a treasure-trove of data analysis: notable figures include histograms of appliance power consumption; histograms of appliance *on*- and *off*-durations; Pearson's coefficients of all pairs of appliances displayed as a heatmap; and daily and weekly usage patterns.

The authors model the data using four HMM variants and compare the performance of the four models. The four models are:

1. a Factorial HMM (Ghahramani & Jordan 1997) where the single hidden state variable in a conventional HMM is replaced by multiple hidden state variables, all influencing the observation variable. In Kim et al.'s work each hidden state variable represents an appliance.

2. a *Conditional* FHMM which extends FHMM to incorporate additional features (e.g. time of day, other sensor measurements, dependency between appliances).

3. a factorial hidden *semi*-Markov model (FHSMM) which extends FHMM to allow for alternative probability density functions to be used for the state occupancy durations of the appliances (the authors use a gamma distribution).

4. a merger of the FHSMM and CFHMM which the authors label a Conditional Factorial Hidden Semi-Markov Model (CFHSMM).

Parameters are estimated using an EM algorithm in which Gibbs sampling is used as the E-step. Hidden states are estimated using simulated annealing (the authors state that dynamic programming (e.g. the Viterbi algorithm) is computationally intractable for CFHSMMs). This approach models the stable steady-state real power consumption of each device, although the authors point out that the approach could be modified to accept additional measurements such as reactive power.

The authors demonstrate that the CFHSMM outperforms the other HMM variants at disaggregation but that CFHMM comes a very close second, suggesting that the inclusion of extra parameters such as time-of-day and dependencies between appliances are more important than swapping the Gaussian probability distribution function for a delta function in the semi-Markov model.

This approach does not attempt to model multi-state appliances: the system is only capable of explicitly modelling two-state appliances. In their conclusion, the authors note that "*our results revealed that the tested methods work well for appliances with simple or modestly complex power signatures, but less well for more complex signatures.*" Having said this, Kim et al.'s approach does explicitly model dependencies between appliances and so should be able to model some multi-state appliances as collections of highly dependent components. So this paper presents an intriguing approach but it is not clear how its performance compares to Hart et al.'s "AS-NALM" approach (G. W. Hart 1992), which also does a good job of disaggregating simple devices but performs less well for complex appliances (although Kim et al.'s approach appears perfectly capable of disaggregating appliances with power consumptions of less than 150 watts using only a real power measurement taken at 1 Hz or less whilst Hart's approach ignored appliances < 150 watts and required real, reactive and voltage measurements). The authors report an accuracy of up to 78% for households with eight active appliances.

Kim et al. also comment that their approach needs to be explicitly told how many appliances are active in the aggregate signal; it cannot automatically determine this. This is unattractive for two reasons: firstly, most users will not want to spend time counting the number of appliances they own and secondly the number of appliances varies over time.

7.6.2 Generic models and house-specific models

Parson et al. (Parson et al. 2011; Parson et al. 2012) built a system capable of disaggregating individual appliances from an aggregate load without requiring prior knowledge about the number or type of appliances within the home, or access to appliance-level metering data.

A fascinating and novel contribution of Parson et al.'s paper is that training is done by tuning prior, generic appliance models to specific appliance instances within the household currently being disaggregated. This house-specific training is done using only the household's aggregate signal. The transition matrix for each generic model consists of a sparse binary matrix, where ones represent possible state transitions. Each appliance's expected demand is represented as values of the Gaussian emission function's mean and variance.

When exposed to aggregate data from a previously unseen house, the system first uses the Expectation-Maximisation (EM) algorithm (initialised with the transition matrix and power demand from the generic model) to locate clean appliance signatures within the aggregate data. This is done by applying the EM algorithm to small overlapping windows of aggregate data which effectively locates windows of aggregate data generated by *only* the modelled appliance. The system then tunes each hidden Markov model using the signatures teased from the aggregate data.

Once the HMMs have been tuned, disaggregation is performed using an extension of the Viterbi algorithm which filters the aggregate signal to remove all previously disaggregated appliances. The Viterbi algorithm is robust against situations in which the modelled appliance's observations have been filtered out in a previous step. Interference from unmodelled appliances is minimised by allowing the forward pass to filter out observations for which the point probability is below a predefined threshold.

These steps are repeated until all known devices have been disaggregated. Inference is done using Gibbs sampling.

Unlike the HMMs used by Kim et al., Parson et al.'s HMMs use hidden states to represent two concepts: 1) changes in steady state; and 2) the steady state. For example, a fridge state transition model has four hidden states: {off (0 W), turn on (+80 W), on (80 W), turn off (-80 W)}. Hence hidden states only emit non-zero observations when the appliance's power demand changes. Another difference between Parson's HMMs and Kim's HMMs is that Kim et al. used the steady-state power consumption as the observed sequence whilst Parson et al. use the changes between steady states as the observed sequence.

To quantify the performance of their system, the authors used the REDD data set (J. Zico Kolter & M. J. Johnson 2011) downsampled to one sample every minute. The authors modelled three appliances: the fridge, the microwave and the clothes drier which together constitute 35% of the household energy consumption. The approach disaggregated these appliances to an accuracy of 83%. Interestingly, the greatest disaggregation error was seen for the clothes dryer (the most complex device they attempted to disaggregate).

Parson et al. state that "*we also aim to investigate the gain of disaggregating appliances in parallel, instead of iteratively, to resolve conflicts caused by two or more appliances changing state simultaneously*".

7.6.3 Computationally efficient methods for inference in FHMMs

Tuplets of overlapping appliances

Zeifman 2012 points out that the FHMM disaggregation techniques developed by H. Kim et al. 2011 are of limited practical use: "*the most severe limitations are computational complexity and heavy reliance on the iterative optimization methods for both estimation and disaggregation*". The computational complexity grows exponentially with the number of appliances.

Zeifman also describes a problem with Hart's method: it struggles to distinguish between appliances with similar power draw.

To address these two issues, Zeifman proposes a disaggregation system where *tuplets* of overlapping appliances are modelled as a Markov Chain and are disaggregated together using a modified Viterbi algorithm. State durations are used to calculate transition probabilities.

Zeifman reports an accuracy of at least 80-90% for two-state appliances.

7.6.4 Approximate inference in additive factorial HMMs

J. Zico Kolter & Jaakkola 2012, both at MIT's Computer Science and AI lab in 2012, describe an innovative and efficient technique for approximate inference in additive FHMMs (additive FHMMs are used by H. Kim et al. 2011 and Parson et al. 2012). An "additive" FHMM is a factorial HMM where each observation is the sum of the (unobserved) real-valued emissions from each hidden state. Exact inference is computationally intractable in such models. Existing approximate inference techniques for additive FHMMs are prone to local optima (J. Zico Kolter & Jaakkola 2012) and rather computationally expensive.

Kolter & Jaakkola exploit the structure of additive FHMMs in several ways. Firstly, they use the forward difference of the aggregate signal as the observation. Secondly, they use a "robust" mixture component to handle unmodelled observations and, finally, they constrain the posterior so that at most one hidden state can change at a time (as a side-effect, this system cannot handle overlapping state transitions). The resulting method is computationally efficient (scales almost linearly in the number of HMMs) and can handle non-IID noise. It is also free from local optima. The end result is a convex approximate inference method which can be solved rapidly for hundreds of thousands of variables.

The system learns in an unsupervised fashion from the aggregate data. It does this by extracting all candidate device "snippets" from the signal (where a "snippet" is a clean cycle of just one appliance in the aggregate data) and then uses spectral clustering to compute similarities between these snippets (hence, as the authors acknowledge, it can only learn devices with relatively short durations). The system can cope with appliances with more than two states. Their system, when applied to two weeks of real meter data, achieves an average precision score of 87.2% on seven circuits.

7.6.5 Disadvantages of using HMMs to model multi-state appliances

1. **Several types of appliance violate the Markov assumption** (that the next state depends only on the current state and not on the preceding states). For example, consider a simplified model of a washing machine with three states: *wash*, *heat* and *spin*. A typical sequence might be wash→heat→wash→heat→wash→spin. The transition matrix for a first-order Markov model would encode that the washer can transition from *wash* to either *heat* or *spin*. But this is an oversimplification: the washer can transition to spin *only if* it has previously heated the water several times. So, first-order HMMs are not a perfect fit to our problem domain. Does this matter? Might a first-order HMM be *good enough*? Other researchers have had some success with HMMs (e.g. H. Kim et al. 2011; Parson et al. 2012).

2. **HMMs do not explicitly encode state *duration*.** HMMs are well suited to application domains like speech recognition where the duration of each state does not vary much (even so, some researchers have explored explicitly encoding duration in speech recognition HMMs, e.g. Vaseghi 1995). But are HMMs well suited to applications where state duration can vary by orders of magnitude? Some appliance states might last one minute (e.g. a kettle) whilst other states might last for hours (e.g. the office computer might be on for many hours during the day). Traditional HMMs cannot explicitly represent state duration. Instead they implicitly encode state duration in the self-loop probabilities where the probability of staying in state i for d consecutive time slices $p_i(d)$ is equal to the probability of $d-1$ self-loop transitions and a state exit transition: $p_i(d) = a_{ii}^{d-1}(1-a_{ii})$. In other words, the probability of staying in one state decreases exponentially with time. This is definitely not what we want. If we know that, say, the computer is usually on for eight hours then we want our model to expect that the computer is most likely to turn off eight hours after turning on, with the probability dropping away either side of the eight-hour mark. Researchers have explored various ways to encode state duration into an HMM. For example, Vaseghi 1995 replaces the state transition probabilities $a_{ij}$with duration-dependent variables $a_{ij}(d)$; Hauberg & Sloth 2008 describe a system where the discrete Markov model in an HMM is replaced by a state-space model with a continuous hidden variable variable determining the discrete state of the system, this system is implemented as a particle filter. M. T. Johnson 2005 provides an analysis of the capacity

and complexity of HMM duration modelling techniques. The bottom line is that yes, an HMM can be modified to represent state duration but this comes with considerable implications for time-complexity and implementation-complexity. And, because HMMs do not explicitly encode state duration, they cannot take advantage of state duration information to speed up disaggregation (for example, if we know that a state is very likely to finish an hour after starting then it makes sense to start our search for evidence of the state finishing an hour after it starts, rather than starting one time-slice after the start).

3. **HMMs typically require that *every* appliance is modelled.**

4. **HMMs typically start by "denoising" the input signal.** One of my hypotheses is that we can do better by *exploiting* the "texture" in the raw signal.

7.7 Other approaches

Other approaches described in the literature include a rule-based pattern recognition system (Farinaccio & Zmeureanu 1999) for disaggregating aggregate data sampled once every 16 seconds from a clamp-on sensor. Each appliance is recognised by checking candidate waveforms in the aggregate data against a set of rules. It is assumed that simultaneous events do not occur. The system takes advantage of both steady state features and some transient features. Training is done from individual metering of appliances for about a week each. It would appear that rules are hand-built for each appliance.

Other approaches include:

Neural networks: Prudenzi 2002; Srinivasan et al. 2006; H.-T. Yang et al. 2007; Aydinalp-Koksal & Ugursal 2008; Ruzzelli et al. 2010

Genetic algorithms: Baranski & Voss 2004b; Baranski & Voss 2004a; H.-H. Chang et al. 2011; Egarter & Elmenreich 2013; Egarter et al. 2013; G. Zhang et al. 2015

Support vector machines: T. Onoda et al. 2000; Murata & Takashi Onoda 2001; Srinivasan et al. 2006; Lin & Tsai 2011; Jiang et al. 2013; Altrabalsi et al. 2014; Jiang et al. 2014; Kleiminger et al. 2015

Chapter 8

Manual feature extraction experiments

In the previous chapter, we argued that a disadvantage of many existing NILM algorithms is that they deliberately filter out many of the rich features of the raw aggregate power demand signal. One of our primary hypotheses is that we can improve disaggregation performance by extracting a rich set of features from the aggregate power demand signal. In Chapter 9, we describe how we used deep neural networks to extract rich features from the raw aggregate power demand signal.

As a proof of the concept that rich features can be extracted from low-frequency smart meter data, we developed and experimented with a novel feature extraction algorithm for NILM: our “spike histogram” algorithm.

Many appliances emit a waveform with distinctive time-domain patterns. For example, most appliances which contain electric heating elements control temperature *not* by varying the power delivered to the heating element (which would require lots of power to be dissipated by the control circuitry) but rather by turning the heating element on-and-off (a technique called “pulse-width modulation”).

For example, our bread maker turns its 570 watt heater on and off roughly every 30 seconds (see Figure 8.1), and the heater’s power demand when on is fairly constant (it ranges from 555-580 watts). TVs[1] draw more power when displaying bright images than when displaying dark images; for example our TV consumes between 90 and 130 watts (see Figure 8.1). The

[1]With the possible exception of LCD TVs because they leave their backlight on constantly.

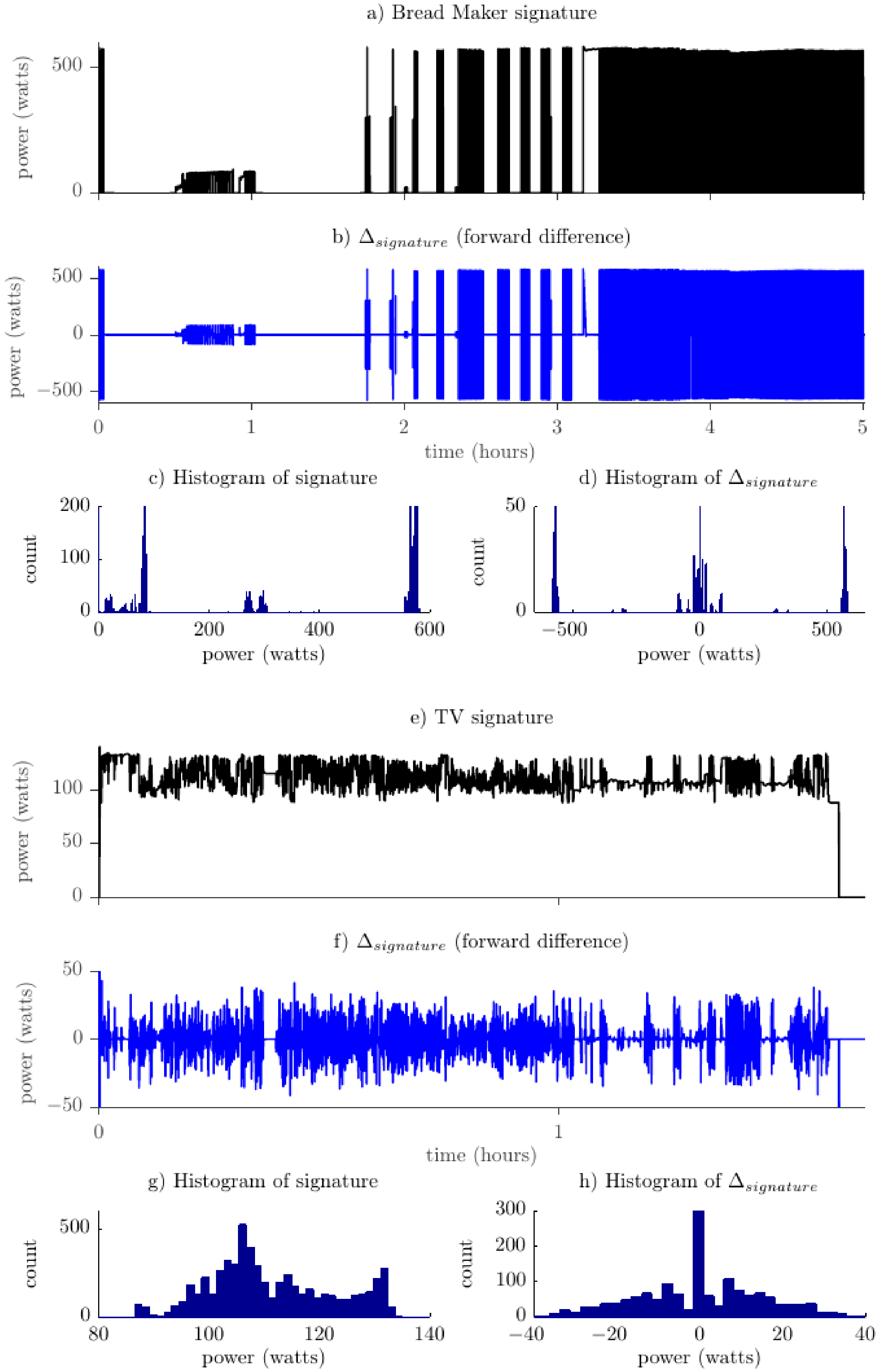


Figure 8.1: Analysis of a bread maker (panels *a* - *d*) and a CRT TV (panels *e* - *h*). The forward difference of the signature (denoted by $\Delta_{signature}$) at each time t = *signature(t+1) - signature(t)*.

signatures, forward differences of the signatures, histograms of the signatures and histograms of the forward difference are shown for our TV and breadmaker in Figure 8.1.

Can we build a feature detector that can efficiently discriminate between, say, the time-domain patterns produced by a bread maker and a TV?

8.1 Spectrogram

A tool commonly used in signal analysis is the spectrogram which is a 2D plot with time represented along the x-axis and frequency along the y-axis. For each point on the spectrogram, the amount of energy carried by that frequency at that time is represented by the pixel colour. A spectrogram is created by first chopping the input timeseries into overlapping chunks. The frequency content of each chunk is determined using short-time Fourier transform (STFT). A simple example of a spectrogram is show in figure 8.2.

Fourier transformation attempts to decompose a timeseries into a set of sine waves. But our smart meter signals are composed mostly of horizontal steady states and vertical transitions. The result of decomposing a *rectangular* wave into a set of sine waves is, for our purposes at least, rather meaningless. A single rectangular pulse produces a “spray” of energy across the frequency spectrum. More precisely: an instantaneous edge in the time-domain requires an infinite bandwidth in the frequency-domain (hence ideal square waves are impossible to achieve in practice). The top panel of Figure 8.3 illustrates the problem. The spectrogram was produced from the synthetic rectangular wave shown in the bottom panel. The first half of the synthetic wave maintains the same amplitude but increases in frequency over time; the second half maintains the same frequency but decreases in amplitude. The spectrogram fails to capture these trends.

8.2 Spike histogram

Our “spike histogram” algorithm detects high-frequency changes of a similar magnitude, such as a bread maker turning its heater on and off every 30 seconds, or the random variation present in a TV’s signature. As far as I am aware, this is a novel proposal. The motivation

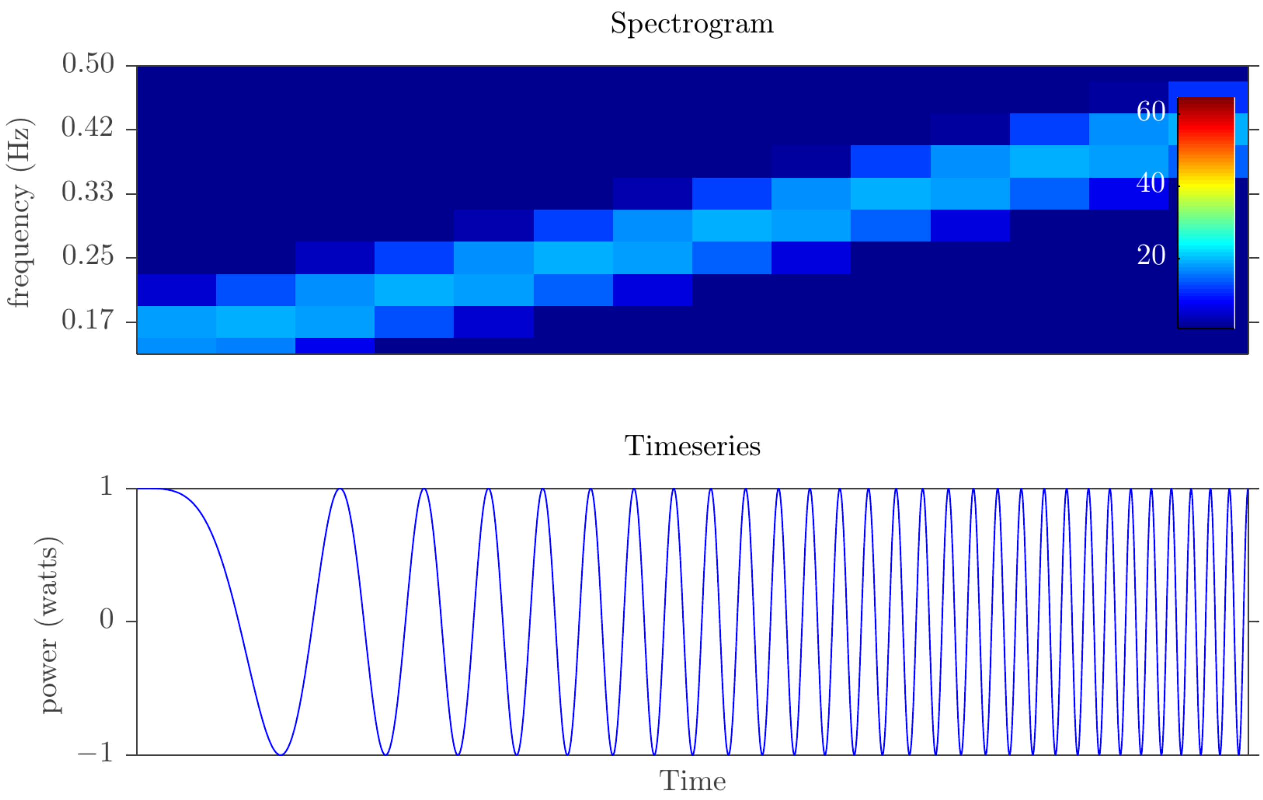


Figure 8.2: Simple example illustrating the spectrogram (top panel) which results from a "chirp" (bottom panel) which is a sinusoidal signal whose frequency increases over time. The diagonal line in the spectrogram indicates that the chirp increases in frequency over time. The two panels share the same x-axis (time).

is that traditional steady-state detectors developed for NILM struggle to take advantages of rapid changes.

Can we do better than the spectrogram? My proposed solution is, for want of a better name, a "spike histogram". An example is shown in the middle panel of Figure 8.3.

The process is simple. First, for each time t, we calculate the forward difference between the timeseries value at time *t+1* and the value at time t to produce a vector that we denote with Δ. Next a set of ten "power bins" is created: for example if $\max(\Delta) = 1000$ then $powerBin_1$ represents spikes between 1-99 watts, $powerBin_2$ represents spikes between 100-199 watts etc. Then the forwards difference vector Δ is broken into overlapping chunks in time where each chunk is, say, 3 minutes in length. For each chunk, each value of Δ is assigned to a single powerBin. The result is a 2D plot where time is represented on the x-axis and each powerBin is represented on the y-axis. The pixel colour represents the quantity of values in each powerBin.

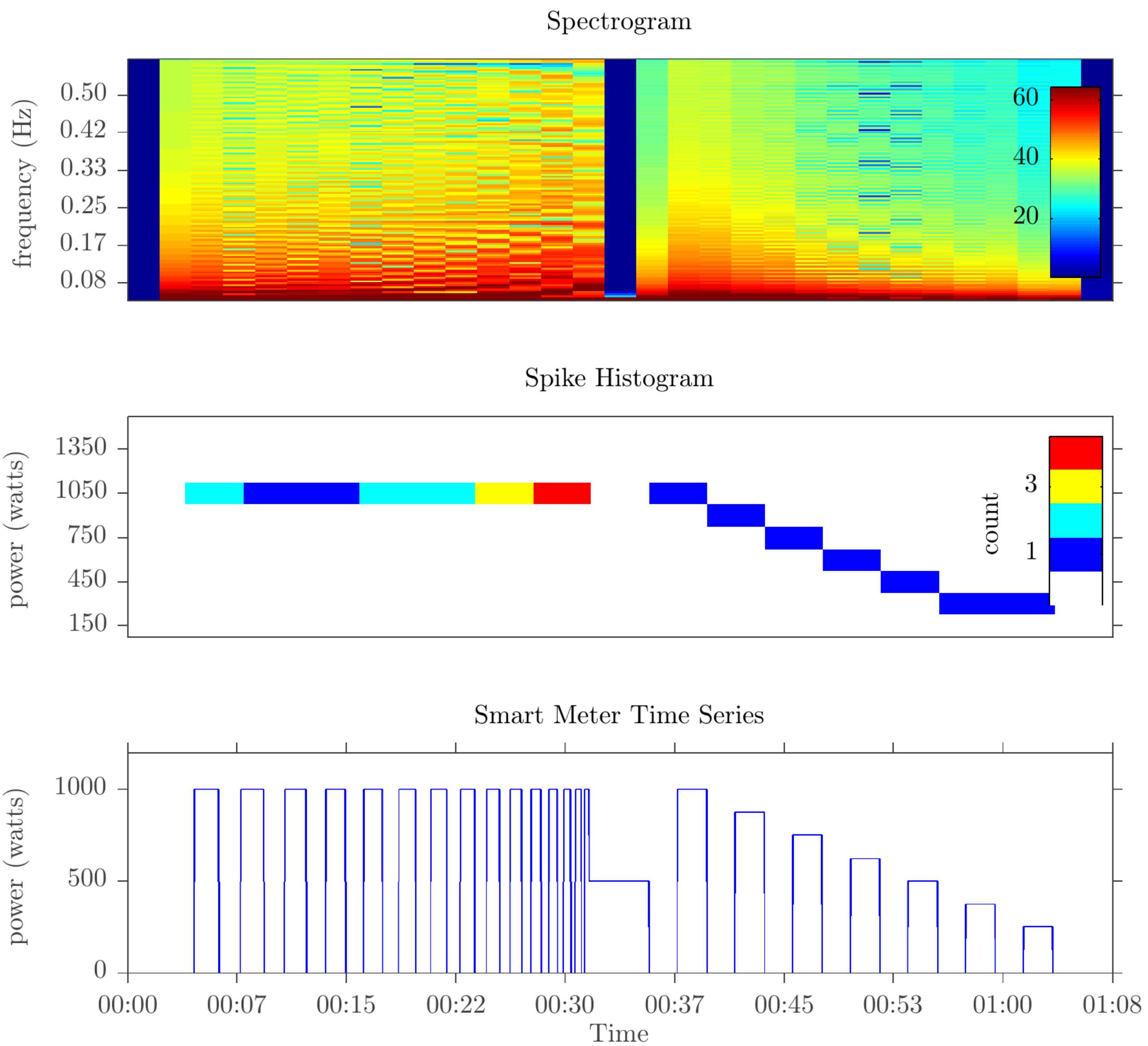


Figure 8.3: Synthetic test data (bottom panel) analysed using a spectrogram (top panel) and my "spike histogram" algorithm (middle panel). All three panels share the same x-axis (which represents time). During the first half of the synthetic data the amplitude of each rectangular wave remains the same but the frequency increases. During the second half of the synthetic data the frequency remains constant but the amplitude decreases. The pixel colour in the spectrogram represents the amount of energy attributed to the frequency indicated on the y-axis at the time indicated on the x-axis.

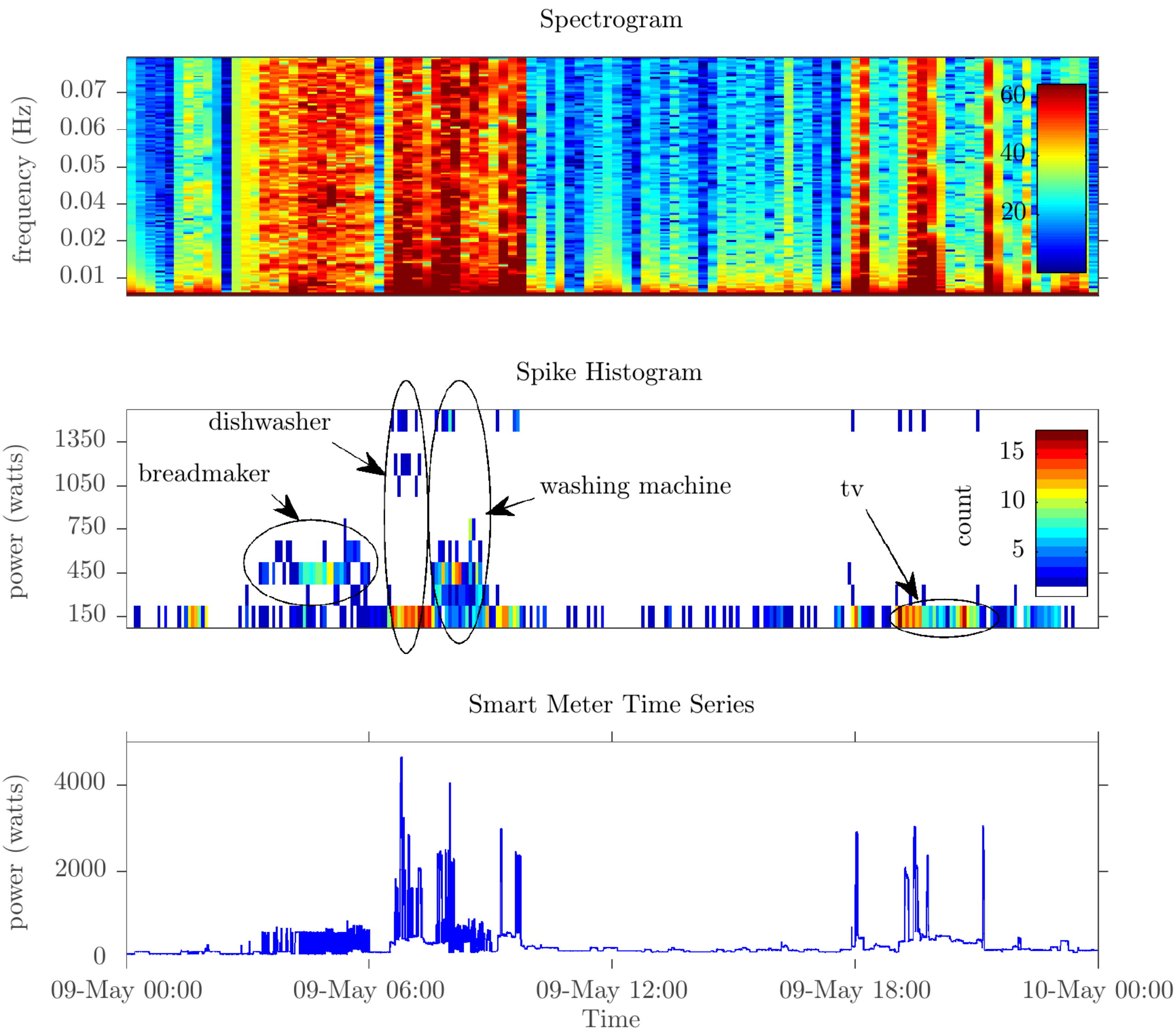


Figure 8.4: Real smart meter data (bottom panel) analysed using a spectrogram (top panel) and my "spike histogram" algorithm (middle panel). All three panels share the same x-axis (which represents time). The middle panel has been manually annotated to illustrate that the spike histogram feature detector can discriminate between the time-domain patterns emitted by the breadmaker, dishwasher, washing machine and TV.

Figure 8.3 illustrates how to "read" a spike histogram plot. The frequency of spikes in the timeseries is represented by colour in the spike histogram. The magnitude is represented by vertical position.

In practice, tiny values of Δ (less than 5 watts) are ignored because they are almost certainly just the result of random noise.

8.2.1 Tests on real smart meter data

How do the spectrogram and the spike histogram perform on real data? Figure 8.4 shows twenty-four hours of real smart meter data from my house. I have manually annotated the spike histogram to show which features correspond to which appliances. Note that the only information discernible from the spectrogram is whether or not any vertical transitions are present in the smart meter data. In contrast, the spike histogram shows distinct patterns for the breadmaker, dishwasher, washing machine and television.

8.3 Discussion

The spectrogram is useful for high sample-rate smart meter data (tens or hundreds of kHz) because, at these high sample-rates, the sinusoidal nature of the mains AC signal[2] is faithfully captured by the analogue-to-digital conversion. But standard smart meters only sample once every ten seconds (0.1 Hz) hence the sinusoidal nature of the analogue signal is lost and the digital signal contains many sharp edges, each of which causes Fourier analysis to show a spray of energy across the frequency spectrum. Hence any technique based on Fourier analysis is unlikely to provide a useful foundation on which to build an informative feature detector for standard smart meters.

In light of these limitations, I experimented with a "spike histogram" algorithm. My experiments suggest that extraction of "rich" features can work well to detect some appliances, even with low temporal resolution input data.

[2] In the UK, mains AC power oscillates at 50 Hz. There is information in many of the higher harmonics.

Manually engineering multiple feature detectors may be viable for a small number of appliances and a small number of types of smart meter data. But it would not be viable for the large and diverse set of appliances available in practice; and for the large and diverse set of smart meter types. This motivates the use of deep neural networks, which can learn to extract complex features from raw input data. We discuss deep neural networks applied to NILM in the following chapter.

Chapter 9

Deep neural networks for energy disaggregation

Recently, deep neural networks have driven remarkable improvements in classification performance in machine learning fields such as image classification and automatic speech recognition. One of the main benefits of deep neural networks is that they can automatically learn a hierarchy of feature extractors from raw input data (provided that enough training data is available).

The hypothesis we test in this chapter is that disaggregation performance can be improved by automatically learning a hierarchy of feature extractors from raw smart meter data, using deep neural networks.

In this chapter, we adapt three deep neural network architectures to energy disaggregation: 1) a form of recurrent neural network called 'long short-term memory' (LSTM); 2) denoising autoencoders; and 3) a network which regresses the start time, end time and average power demand of each appliance activation.

We use seven metrics to test the performance of these algorithms on real aggregate power data from five appliances. Tests are performed against a house not seen during training and against houses seen during training. We find that all three neural nets achieve better F1 scores (averaged over all five appliances) than either combinatorial optimisation or factorial hidden Markov models and that our neural net algorithms generalise well to an unseen house.

This chapter is based on Kelly & Knottenbelt 2015a.

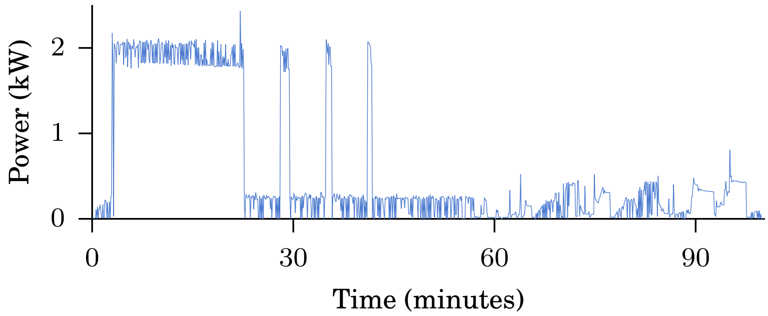


Figure 9.1: Example washing machine power demand (from UK-DALE House 1).

9.1 Motivation

Our main hypothesis for this chapter is that recent techniques from machine learning fields such as image classification can be adapted to *automatically learn* which features to extract from smart meter signals; and that this will lead to an increase in NILM performance. As discussed in Chapter 7, the majority of existing NILM algorithms designed for ten-second data cannot extract rich features from the raw data. In fact, many algorithms begin by throwing away much of the texture in the raw data. Our claim is that we can improve disaggregation performance by exploiting these rich features.

Humans can learn to detect appliances in aggregate data by eye, especially appliances with feature-rich signatures such as the washing machine signature shown in Figure 9.1. Humans almost certainly make use of a variety of features such as the rapid on-off cycling of the motor (which produces the rapid $\sim$ 200 watt oscillations), the ramps towards the end as the washer starts to spin the clothes etc. As discussed in the previous chapter, we *could* consider hand-engineering feature extractors for these rich features. But this would be time consuming and the resulting feature detectors may not be robust to noise and artefacts; and may not work for the wide range of appliances available; and the wide range of smart meter data available.

Before 2012, the dominant approach to extracting features for image classification was to hand-engineer feature detectors such as scale-invariant feature transform (SIFT) (Lowe 1999) and difference of Gaussians (DoG). Then, in 2012, Krizhevsky et al.'s winning algorithm in the ImageNet Large Scale Visual Recognition Challenge achieved a substantially lower error score (15%) than the second-best approach (26%) (Krizhevsky et al. 2012). Krizhevsky et al.'s approach did not use hand-engineered feature detectors. Instead they used a deep neural network which automatically learnt to extract a *hierarchy of features* from the raw image. Deep learning is now a dominant approach not only in image classification but also in fields such as automatic speech recognition (Graves & Jaitly 2014), machine translation (Sutskever et al. 2014), even learning to play computer games from scratch (Mnih et al. 2015)!

In this chapter, we investigate whether deep neural nets can be applied to energy disaggregation. The use of 'small' neural nets on NILM dates back at least to Roos et al. 1994 (although that paper was just a proposal) and continued with H.-T. Yang et al. 2007; Lin & Tsai 2010; Ruzzelli et al. 2010; and H.-H. Chang et al. 2011. But these small nets do not appear to learn a hierarchy of feature detectors. A big breakthrough in image classification came when the compute power (courtesy of GPUs) became available to train *deep* neural networks on large amounts of data. In the present research, we want to see if deep neural nets can deliver good performance on energy disaggregation.

Our main contribution is to adapt three deep neural network architectures to NILM. For each architecture, we train one network per target appliance. We compare two benchmark disaggregation algorithms (combinatorial optimisation and factorial hidden Markov models) to the disaggregation performance of our three deep neural nets using seven metrics. We also examine how well our neural nets generalise to appliances in houses not seen during training because, ultimately, when NILM is used 'in the field' we very rarely have ground truth appliance data for the houses for which we want to disaggregate. So it is essential that NILM algorithms can generalise to unseen houses.

Please note that, once trained, our neural nets *do not* need ground truth appliance data from each house. End-users would only need to provide aggregate data. This is because each neural network should learn the 'essence' of its target appliance such that it can generalise to unseen instances of that appliance. In a similar fashion, neural networks trained to do image classification are trained on many examples of each category (dogs, cats, etc.) and generalise to unseen examples of each category.

To provide more context, we will briefly sketch how our neural networks could be deployed at scale, in the wild. Each net would undergo *supervised* training on *many* examples of its target appliance type so each network learns to generalise well to unseen appliances.

Training is computationally expensive (days of processing on a fast GPU). But training does not have to be performed often. Once these networks are trained, inference is much cheaper (around a second of processing per network on a fast GPU for a week of aggregate data). Aggregate data from unseen houses would be fed through each network. Each network should filter out the power demand for its target appliance. This processing would probably be too computationally expensive to run on an embedded processor inside a smart meter or in-home-display. Instead, the aggregate data could be sent from the smart meter to a server in the cloud.

The storage requirements for one 16 bit integer sample (0–64 kW in 1 watt steps) every ten seconds is 17 kilobytes per day uncompressed. This signal should be easily compressible because there are numerous periods in domestic aggregate power demand with little or no change. With a compression ratio of 5:1, and ignoring the datetime index, the total storage requirements for a year of data from 10 million users would be 13 terabytes. If one week of aggregate data can be processed in one second per home (which should be possible given further optimisation) then data from 10 million users could be processed by 16 GPU compute nodes. Alternatively, disaggregation could be performed on a compute device within each home (a modern laptop or mobile phone or a dedicated 'disaggregation hub' could handle the disaggregation). A GPU is not *required* for disaggregation, although it makes it faster.

This chapter is structured as follows: In Section 9.1 we discuss why – in principal – deep neural nets might perform better on NILM than previous NILM algorithms. In Section 9.2 we provide a very brief introduction to artificial neural nets. In Section 9.3 we describe how we prepare the training data for our nets and how we 'augment' the training data by synthesising additional data. In Section 9.4 we describe how we adapted three neural net architectures to NILM. In Section 9.5 we describe how we do disaggregation with our nets. In Section 9.6 we present the disaggregation results of our three neural nets and two benchmark NILM algorithms. Finally, in Section 9.7 we discuss our results, offer our conclusions and describe some possible future directions for research.

9.2 Introduction to artificial neural nets

An artificial neural network (ANN) is a directed graph where the nodes are artificial neurons and the edges allow information from one neuron to pass to another neuron (or the same neuron in a future time step). Neurons are typically arranged into layers such that each neuron in layer l connects to every neuron in layer $l+1$. Connections are weighted and it is through modification of these weights that ANNs learn. ANNs have an *input layer* and an *output layer*. Any layers in between are called *hidden layers*. The *forward pass* of an ANN is where information flows from the input layer, through any hidden layers, to the output. Learning (updating the weights) happens during the *backwards pass*.

9.2.1 Forwards pass

Each artificial neuron calculates a weighted sum of its inputs, adds a bias and passes this sum through an activation function. Consider a neuron which receives I inputs. The value of each input is represented by input vector x. The weight on the connection from input i to neuron h is denoted by w_{ih} (so w is the 'weights matrix'). The weighted sum (also called the 'network input') of the inputs into neuron h can be written $a_h = \sum_{i=1}^{I} x_i w_{ih}$. The network input a_h is then passed through an activation function θ to produce the neuron's final output b_h where $b_h = \theta(a_h)$. In this chapter, we use the following activation functions: linear: $\theta(x) = x$; rectified linear (ReLU): $\theta(x) = \max(0, x)$; hyperbolic tangent (tanh): $\theta(x) = \frac{\sinh x}{\cosh x} = \frac{e^x - e^{-x}}{e^x + e^{-x}}$.

Multiple nonlinear hidden layers can be used to represent the input data (hopefully by learning a hierarchy of feature detectors), which gives deep nonlinear networks a lot of expressive power (Geoffrey E Hinton et al. 2006; Bengio; LeCun et al. 2007).

9.2.2 Backwards pass

After computing a forwards pass through the entire network to get the network's output for a specific network input, we then compute the error of the output relative to the target (in all our experiments we use the mean squared error (MSE) as the objective function). We then modify the network's weights in the direction which should reduce the error.

In practice, the forward pass is often computed over a *batch* of randomly selected input vectors. In our work, we use a batch size of 64 sequences per batch for all but the largest recurrent neural network (RNN) experiments. In our largest RNNs we use a batch size of 16 (to allow the network to fit into the 3GB of RAM on our GPU).

How do we modify each weight to reduce the error? It would be computationally intractable to enumerate the entire error surface. MSE gives a smooth error surface and the activation functions are differentiable hence we can use gradient descent. The first step is to compute the gradient of the error surface at the position for the current batch by calculating the derivative of the objective function with respect to each weight. Then we modify each weight by adding the gradient multiplied by a 'learning rate' scalar parameter. To efficiently compute the gradient (in $O(W)$ time) we use the backpropagation algorithm (Rumelhart et al. 1985; Werbos 1988; Williams & Zipser 1995). In all our experiments we use stochastic gradient descent (SGD) with Nesterov momentum of 0.9.

9.2.3 Convolutional neural nets

Consider the task of identifying objects in a photograph. No matter if we hand engineer feature detectors or learn feature detectors from the data, it turns out that useful 'low level' features concern small patches of the image and include features such as edges of different orientations, corners, blobs etc. To extract these features, we want to build a small number of feature detectors (one for horizontal lines, one for blobs etc.) with small receptive fields (overlapping sub-regions of the input image) and slide these feature detectors across the entire image. Convolutional neural nets (CNNs) (Fukushima 1980; Atlas et al. 1988; LeCun et al. 1998) build a small number of filters, each with a small receptive field, and these filters are duplicated (with shared weights) across the entire input.

Similarly to computer vision tasks, in time series problems we often want to extract a small number of low level features with a small receptive field across the entire input. All of our nets use at least one 1D convolutional layer at the input.

9.3 Training data

Deep neural nets need a lot of training data because they have a large number of trainable parameters (the network weights and biases). The nets described in this chapter have between 1 million to 150 million trainable parameters. Large training datasets are important. It is also common practice in deep learning to increase the effective size of the training set by duplicating the training data many times and applying realistic transformations to each copy. For example, in image classification, we might flip the image horizontally or apply slight affine transformations.

A related approach to creating a large training dataset is to generate simulated data. For example, Google DeepMind train their algorithms (Mnih et al. 2015) on computer games because they can generate an effectively infinite amount of training data. Realistic synthetic speech audio data or natural images are harder to produce.

In energy disaggregation, we have the advantage that generating effectively infinite amounts of synthetic aggregate data is relatively easy by randomly combining real appliance activations. (We define an 'appliance activation' to be the power drawn by a single appliance over one complete cycle of that appliance. For example, Figure 9.1 shows a single activation for a washing machine.) We trained our nets on both synthetic aggregate data and real aggregate data in a 50:50 ratio. We found that synthetic data acts as a regulariser. In other words, training on a mix of synthetic and real aggregate data rather than just real data appears to improve the net's ability to generalise to unseen houses. For validation and testing we use only real data (not synthetic).

We used UK-DALE (see Chapter 4) as our source dataset. Each submeter in UK-DALE samples once every 6 seconds. All houses record aggregate apparent mains power once every 6 seconds. Houses 1, 2 and 5 also record active and reactive mains power once a second. In these houses, we downsampled the 1 second active mains power to 6 seconds to align with the submetered data and used this as the real aggregate data from these houses. Any gaps in appliance data shorter than 3 minutes are assumed to be due to RF issues and so are filled by forward-filling. Any gaps longer than 3 minutes are assumed to be due to the appliance and meter being switched off and so are filled with zeros.

We manually checked a random selection of appliance activations from every house. The UK-DALE metadata shows that House 4's microwave and washing machine share a single meter (a fact that we manually verified) and hence these appliances from House 4 are not used in our training data.

We train one network per target appliance. The target (i.e. the desired output of the net) is the power demand of the target appliance. The input to every net we describe in this chapter is a

Table 9.1: Number of training activations per house.

	1	2	3	4	5
Kettle	2836	543	44	716	176
Fridge	16 336	3526	0	4681	1488
Washing machine	530	53	0	0	51
Microwave	3266	387	0	0	28
Dish washer	197	98	0	23	0

Table 9.2: Number of testing activations per house.

	1	2	3	4	5
Kettle	54	29	40	50	18
Fridge	168	277	0	145	140
Washing machine	10	4	0	0	2
Microwave	90	9	0	0	4
Dish washer	3	7	0	3	

window of aggregate power demand. The window width is decided on an appliance-by-appliance basis and varies from 128 samples (13 minutes) for the kettle to 1536 samples (2.5 hours) for the dish washer. We found that increasing the window size hurts disaggregation performance for short-duration appliances (for example, using a sequence length of 1024 for the fridge resulted in the autoencoder (AE) failing to learn anything useful and the 'rectangles' net achieved an F1 score of 0.68; reducing the sequence length to 512 allowed the AE to get an F1 score of 0.87 and the 'rectangles' net got a score of 0.82). On the other hand, it is important to ensure that the window width is long enough to capture the majority of the appliance activations.

For each house, we reserved the last week of data for testing and used the rest of the data for training. The number of appliance training activations is show in Table 9.1 and the number of testing activations is shown in Table 9.2. The specific houses used for training and testing is shown in Table 9.3. Note that it took around 24 hours to train each network (one network per appliance and per architecture), so we did not have time to do full cross-validation. Instead we trained each network once; and tested each network once.

9.3.1 Choice of appliances

We used five target appliances in all our experiments: the fridge, washing machine, dish washer, kettle and microwave. We chose these appliances because each is present in at least three houses

Table 9.3: Houses used for training and testing.

	Training	Testing
Kettle	1, 2, 3, 4	5
Fridge	1, 2, 4	5
Washing machine	1, 5	2
Microwave	1, 2	5
Dish washer	1, 2	5

Table 9.4: Arguments passed to `get_activations()`.

Appliance	Max power (watts)	On power threshold (watts)	Min. on duration (secs)	Min. off duration (secs)
Kettle	3100	2000	12	0
Fridge	300	50	60	12
Washing m.	2500	20	1800	160
Microwave	3000	200	12	30
Dish washer	2500	10	1800	1800

in UK-DALE. This means that, for each appliance, we can train our nets on at least two houses and test on a different house. These five appliances consume a significant proportion of energy and the five appliances represent a range of different power 'signatures' from the simple on/off of the kettle to the complex pattern shown by the washing machine (Figure 9.1).

'Small' appliances such as games consoles and phone chargers are problematic for many NILM algorithms because the effect of small appliances on aggregate power demand tends to get lost in the noise. By definition, small appliances do not consume much energy individually but modern homes tend to have a large number of such appliances so their combined consumption can be significant. Hence it would be useful to detect small appliances using NILM. We have not explored whether our neural nets perform well on 'small' appliances.

9.3.2 Extract activations

Appliance activations are extracted using NILMTK's `Electric.get_activations()` method. The arguments we passed to `get_activations()` for each appliance are shown in Table 9.4. On simple appliances such as toasters, we extract activations by finding strictly consecutive samples above some threshold power. We then throw away any activations shorter than some threshold duration (to ignore spurious spikes). For more complex appliances such as washing machines whose power demand can drop below threshold for short periods during a cycle, NILMTK ignores short periods of sub-threshold power demand.

The August 2015 release of UK-DALE added the metadata listed in Table 9.4 into UK-DALE's metadata. This was done so that NILMTK's `get_activations` method can be called without any explicit arguments and NILMTK will pull these parameters from UK-DALE's metadata.

9.3.3 Select windows of real aggregate data

First we locate all the activations of the target appliance in the home's submeter data for the target appliance. Then, for each training example, the code decides with 50% probability whether this example should include the target appliance or not. If the code decides not include the target appliance then it finds a random window of aggregate data in which there are no activations of the target appliance. Otherwise, the code randomly selects a target appliance activation and randomly positions this activation within the window of data that will be shown to the net as the target (with the constraint that the activation must be captured completely in the window of data shown to the net, unless the window is too short to contain the entire activation). The corresponding time window of real aggregate data is also loaded and shown to the net as its input. If other activations of the target appliance happen to appear in the aggregate data then these are not included in the target sequence. The net is trained to focus on the first complete target appliance activation in the aggregate data.

9.3.4 Synthetic aggregate data

To create synthetic aggregate data we start by extracting a set of appliance activations for five appliances across all training houses: kettle, washing machine, dish washer, microwave and fridge. To create a single sequence of synthetic data, we start with two vectors of zeros: one vector will become the input to the net; the other will become the target. The length of each vector defines the 'window width' of data that the network sees. We go through the five appliance classes and decide whether or not to add an activation of that class to the training sequence. There is a 50% chance that the target appliance will appear in the sequence and a 25% chance for each other 'distractor' appliance. For each selected appliance class, we randomly select an appliance activation and then randomly pick where to add that activation on the input vector. Distractor appliances can appear anywhere in the sequence (even if this means that only part of the activation will be included in the sequence). The target appliance activation must be completely contained within the sequence (unless it is too large to fit).

Of course, this relatively naïve approach to synthesising aggregate data ignores a lot of structure that appears in real aggregate data. For example, the kettle and toaster might often appear within a few minutes of each other in real data, but our simple 'simulator' is completely unaware of this sort of structure. We expect that a more realistic simulator might increase the performance of deep neural nets on energy disaggregation.

9.3.5 Implementation of data processing

All our code is written in Python and we make use Pandas, Numpy and NILMTK for data preparation. Each network receives data in a mini-batch of 64 sequences (except for the large RNN sequences, in which case we use a batch size of 16 sequences). The code is multi-threaded so the CPU can be busy preparing one batch of data on the fly whilst the GPU is busy training on the previous batch.

9.3.6 Standardisation

In general, neural nets learn most efficiently if the input data has zero mean. First, the mean of each sequence is subtracted from the sequence to give each sequence a mean of zero. Every input sequence is divided by the standard deviation of a random sample of the training set. We do not divide each sequence by its *own* standard deviation because that would change the scaling and the scaling is likely to be important for NILM.

Forcing each sequence to have zero mean throws away information. Information that NILM algorithms such as combinatorial optimisation and factorial hidden Markov models rely on. We have done some preliminary experiments and found that neural nets appear to be able to generalise better if we independently centre each sequence. But there are likely to be ways to have the best of both worlds: i.e. to give the network information about the absolute power whilst also allowing the network to generalise well.

One big advantage of training our nets on sequences which have been independently centred is that our nets do not need to consider vampire (always on) loads.

Targets are divided by a hand-coded 'maximum power demand' for each appliance to put the target power demand into the range [0, 1].

9.4 Neural network architectures

In this section we describe how we adapted three different neural net architectures to do NILM.

In total, we performed 574 experiments with a range of neural network architectures and training regimes. The Python scripts for specifying these experiments are available online[1]. Across the 574 experiments, we tried a range of configurations and training schemes. Unfortunately, as we developed the experiments we discovered (and fixed) issues with the data, which somewhat invalidated the previous results. As such, we cannot directly present a quantitative comparison of all these experiments. But, below, we list the main experiments we tried and provide some brief information about the performance implications.

- We tried configuring the nets to output the power demand for multiple appliances per network. This appeared to work well for simple synthetic data but failed for realistic data.
- Output a boolean vector specifying whether each appliance is *on* or *off*, rather than outputting the real-valued power demand for the appliance. Boolean vectors did not appear to work as well as outputting the real-valued power demand for the appliance.
- We tried convolutional layers versus fully-connected layers. Convolutional layers at the input appear to improve performance.
- We tried implementing batch normalisation (Ioffe & Szegedy 2015) but it did not appear to improve performance.
- We tried implementing mixture density networks (Bishop 1994) to output a Gaussian mixture model specifying a probability distribution over the estimated power demand at each time step for the target appliance. This did not appear to improve performance.

[1] https://github.com/JackKelly/neuralnilm_prototype

- We tried a range of activation functions including linear, rectified linear, sigmoid, and tanh.
- We tried LSTMs with and without peepholes.
- We tried computing the forwards difference of the input before passing the forwards difference to the network. This hurt performance.
- We tried a range of weight initialisation schemes.
- For the denoising autoencoder, we tried tying symmetric weights, but this hurt performance slightly.
- Tried outputting a polygon with multiple segments. This did not work as well as the simpler approach of just outputting a single rectangle over the target appliance activation.
- Tried dropout (Geoffrey E. Hinton et al. 2012). This appeared to hurt performance.
- Tried a range of learning rates and momentum settings.
- Tried putting an RNN onto the output of an autoencoder. This did not help.
- Tried skip-connections. This did not help performance.
- We tried varying the training regime. For example:
  - We tried training with measured aggregate data versus a mixture of synthetic aggregate data and measured aggregate data. Adding synthetic aggregate data appears to improve performance and works as a regularizer.
  - We tried forcing the target appliance activation to always be fully captured in the network's input window; and we tried allowing the activation to start or end outside of the network's input window. We got slightly better performance by forcing the appliance activation to be within the network's input window.

The three architectures described below offered the best performance.

9.4.1 Recurrent neural networks

In Section 9.2 we described *feed forward* neural networks which map from a single input vector to a single output vector. During inference, when the network is shown a second input vector, it has no memory of the previous input.

Recurrent neural networks (RNNs) allow cycles in the network graph such that the output from neuron i in layer l at time step t is fed via weighted connections to every neuron in layer l (including neuron i) at time step $t+1$. This allows RNNs, in principal, to map from the *entire history* of the inputs to an output vector. This makes RNNs especially well suited to sequential data. In our work, we train RNNs using backpropagation through time (BPTT) (Werbos 1990).

In practice, RNNs can suffer from the 'vanishing gradient' problem (Hochreiter & Schmidhuber 1997) where gradient information disappears or explodes as it is propagated back through time. This can limit an RNN's memory. One solution to this problem is the 'long short-term memory' (LSTM) architecture (Hochreiter & Schmidhuber 1997) which uses a 'memory cell' with a gated input, gated output and gated feedback loop. The intuition behind LSTM is that it is a differentiable latch (where a 'latch' is the fundamental unit of a digital computer's RAM). LSTMs have been used with success on a wide variety of sequence tasks including automatic speech recognition (Graves & Jaitly 2014; Chorowski et al. 2014) and machine translation (Sutskever et al. 2014).

An additional enhancement to RNNs is to use *bidirectional* layers. In a bidirectional RNN, there are effectively two parallel RNNs, one reads the input sequence forwards and the other reads the input sequence backwards. The output from the forwards and backwards halves of the network are combined either by concatenating them or doing an element-wise sum (we experimented with both and settled on concatenation, although element-wise sum appeared to work almost as well and is computationally cheaper).

We should note that bidirectional RNNs are not naturally suited to doing online disaggregation. Bidirectional RNNs could still be used for online disaggregation if we frame 'online disaggregation' as doing *frequent, small batches* of offline disaggregation.

We experimented with both RNNs and LSTMs and settled on the following architecture for energy disaggregation:

1. Input (length determined by appliance duration)
2. 1D convolution (filter size=4, stride=1, number of filters=16, activation function=linear, border mode=same)
3. bidirectional LSTM (N=128, with peepholes)
4. bidirectional LSTM (N=256, with peepholes)
5. Fully connected (N=128, activation function=TanH)
6. Fully connected (N=1, activation function=linear)

At each time step, the network sees a single sample of aggregate power data and outputs a single sample of power data for the target appliance.

In principal, the convolutional layer should not be necessary (because the LSTMs should be able to remember all the context). But we found the addition of a convolution layer to slightly increase performance (the convolutional layer convolves over the time axis). We also experimented with adding a convolutional layer *between* the two LSTM layers with a stride > 1 to implement hierarchical subsampling (Graves 2012). This showed promise but we did not use it for our final experiments.

On the backwards pass, we clip the gradient at [-10, 10] as per Alex Graves in Graves 2013. To speed up computation, we propagate the gradient backwards a maximum of 500 time steps. Figure 9.2 shows an example output of our LSTM network in the two 'RNN' rows.

9.4.2 Denoising autoencoders

In this section, we frame energy disaggregation as a 'denoising' task. Typical denoising tasks include removing grain from an old photograph; or removing reverb from an audio recording; or in-filling a masked part of an image. Energy disaggregation can be viewed as an attempt to recover the 'clean' power demand signal of the target appliance from the background 'noise' produced by the other appliances. A successful neural network architecture for denoising tasks is the 'denoising autoencoder'.

An autoencoder (AE) is simply a network which tries to reconstruct the input. Described like this, AEs might not sound very useful! The key is that AEs first *encode* the input to a *compact* vector representation (in the 'code layer') and then *decode* to reconstruct the input. The simplest way of forcing the net to discover a *compact* representation of the data is to have a code layer with less dimensions than the input. In this case, the AE is doing dimensionality reduction. Indeed, a linear AE with a single hidden layer is almost equivalent to PCA. But AEs can be deep and non-linear.

A denoising autoencoder (dAE) (Vincent et al. 2008) is an autoencoder which attempts to reconstruct a clean target from a noisy input. dAEs are typically trained by artificially corrupting a signal before it goes into the net's input, and using the clean signal as the net's target. In NILM, we consider the corruption as being the power demand from the other appliances. So we do not add noise artificially. Instead we use the aggregate power demand as the (noisy) input to the net and ask the net to reconstruct the clean power demand of the target appliance.

The first and last layers of our NILM dAEs are 1D convolutional layers. We use convolutional layers because we want the network to learn low level feature detectors which are applied equally across the *entire* input window (for example, a step change of 1000 watts might be a useful feature to extract, no matter where it is found in the input). The aim is to provide some invariance to where exactly the activation is positioned within the input window. The last layer does a 'deconvolution'.

The exact architecture is as follows:

1. Input (length determined by appliance duration)
2. 1D conv (filter size=4, stride=1, number of filters=8, activation function=linear, border mode=valid)
3. Fully connected (N=(sequence length - 3) $\times$ 8, activation function=ReLU)
4. Fully connected (N=128; activation function=ReLU)
5. Fully connected (N=(sequence length - 3) $\times$ 8, activation function=ReLU)
6. 1D conv (filter size=4, stride=1, number of filters=1, activation function=linear, border mode=valid)

Layer 4 is the middle, code layer. The entire dAE is trained end-to-end in one go (we do not do layer-wise pre-training as we found it did not increase performance). We do not tie the weights as we found this also appears to not enhance NILM performance. An example output of our NILM dAE is shown in Figure 9.2 in the two 'Autoencoder' rows.

9.4.3 Regress start time, end time & power

Many applications of energy disaggregation do not require a detailed second-by-second reconstruction of the appliance power demand. Instead, most energy disaggregation use-cases require, for each appliance activation, the identification of the start time, end time and energy consumed. In other words, we want to draw a rectangle around each appliance activation in the aggregate data where the left side of the rectangle is the start time, the right side is the end time and the height is the average power demand of the appliance between the start and end times.

Deep neural networks have been used with great success on related tasks. For example, Nouri used deep neural networks to estimate the 2D location of 'facial keypoints' in images of faces (Nouri 2014). Example 'keypoints' are 'left eye centre' or 'mouth centre top lip'. The input to Nouri's neural net is the raw image of a face. The output of the network is a set of x, y coordinates for each keypoint.

Our idea was to train a neural network to estimate three scalar, real-valued outputs: the start time, the end time and mean power demand of the first appliance activation to appear in the aggregate power signal. If there is no target appliance in the aggregate data then all three outputs should be zero. If there is more than one activation in the aggregate signal then the network should ignore all but the first activation. All outputs are in the range [0, 1]. The start and end times are encoded as a proportion of the input's time window. For example, the start of the time window is encoded as 0, the end is encoded as 1 and half way through the time window is encoded as 0.5. For example, consider a scenario where the input window width is 10 minutes and an appliance activation starts 1 minute into the window and ends 1 minute before the end of the window. This activation would be encoded as having a start location of 0.1 and an end location of 0.9. Example output is shown in Figure 9.2 in the two 'Rectangles' rows.

The three target values for each sequence are calculated during data pre-processing. As for all of our other networks, the network's objective is to minimise the mean squared error. The exact architecture is as follows:

1. Input (length determined by appliance duration)
2. 1D conv (filter size=4, stride=1, number of filters=16, activation function=linear, border mode=valid)
3. 1D conv (filter size=4, stride=1, number of filters=16, activation function=linear, border mode=valid)
4. Fully connected (N=4096, activation function=ReLU)
5. Fully connected (N=3072; activation function=ReLU)
6. Fully connected (N=2048, activation function=ReLU)
7. Fully connected (N=512, activation function=ReLU)
8. Fully connected (N=3, activation function=linear)

9.4.4 Neural net implementation

We implemented our neural nets in Python using the Lasagne library[2]. Lasagne is built on top of Theano (Bergstra et al. 2010; Bastien et al. 2012). We trained our nets on an nVidia GTX 780Ti GPU with 3 GB of RAM (but note that Theano also allows code to be run on the CPU without requiring any changes to the user's code). On this GPU, our nets typically took between 1 and 12 hours to train per appliance. The exact code used to create the results in this chapter is available in our 'NeuralNILM Prototype' repository[3] and a more elegant re-write is available in our 'NeuralNILM' repository[4].

We manually defined the number of weight updates to perform during training for each experiment. For the RNNs we performed 10 000 updates, for the denoising autoencoders we performed 100 000 and for the regression network we performed 300 000 updates. Neither the RNNs nor the AEs appeared to continue learning past this number of updates. The regression networks appear to keep learning no matter how many updates we perform!

[2] github.com/Lasagne/Lasagne
[3] github.com/JackKelly/neuralnilm_prototype
[4] github.com/JackKelly/neuralnilm

The nets have a wide variation in the number of trainable parameters. The largest dAE nets range from 1M to 150M (depending on the input size); the RNNs all had 1M parameters and the regression nets varied from 28M to 120M parameters (depending on the input size).

All our network weights were initialised randomly using Lasagne's default initialisation. All of the experiments presented in this chapter trained end-to-end from random initialisation (no layerwise pre-training).

9.5 Disaggregation

How do we disaggregate arbitrarily long sequences of aggregate data given that each net has an input window duration of, at most, a few hours? We first pad the beginning and end of the input with zeros. Then we slide the net along the input sequence. As such, the first sequence we show to the network will be all zeros. Then we shift the input window `STRIDE` samples to the right, where `STRIDE` is a manually defined positive, non-zero integer. If `STRIDE` is less than the length of the net's input window then the net will see overlapping input sequences. This allows the network to have multiple attempts at processing each appliance activation in the aggregate signal, and on each attempt each activation will be shifted to the left by `STRIDE` samples.

Over the course of disaggregation, the network produces multiple estimated values for each time step because we give the network overlapping segments of the input. For our first two network architectures, we combine the multiple values per timestep simply by taking the mean.

Combing the output from our third network is a little more complex. We layer every predicted 'appliance rectangle' on top of each other. We measure the overlap and normalise the overlap to [0, 1]. This gives a probabilistic output for each appliance's power demand. To convert this to a single vector per appliance, we threshold the power and probability.

9.6 Results

The disaggregation results on an unseen house are shown in Figure 9.3. The results on houses seen during training are shown in Figure 9.4.

We used benchmark implementations from NILMTK (see Chapter 5) of the combinatorial optimisation (CO) and factorial hidden Markov model (FHMM) algorithms.

On the unseen house (Figure 9.3), both the denoising autoencoder and the net which regresses the start time, end time and power demand (the 'rectangles' architecture) out-perform CO and FHMM on every appliance on F1 score, precision score, proportion of total energy correctly assigned and mean absolute error. The LSTM out-performs CO and FHMM on two-state appliances (kettle, fridge and microwave) but falls behind CO and FHMM on multi-state appliances (dish washer and washing machine).

On the houses seen during training (Figure 9.4), the dAE outperforms CO and FHMM on every appliance on every metric except relative error in total energy. The 'rectangles' architecture outperforms CO and FHMM on every appliance (except the microwave) on F1, precision, accuracy, proportion of total energy correctly assigned and mean absolute error.

The full disaggregated time series for all our algorithms and the aggregate data and appliance ground truth data are available at www.doc.ic.ac.uk/~dk3810/neuralnilm

The classification metrics we used are:

$$\mathbf{TP} = \text{number of true positives} \tag{9.1}$$
$$\mathbf{FP} = \text{number of false positives} \tag{9.2}$$
$$\mathbf{FN} = \text{number of false negatives} \tag{9.3}$$
$$\mathbf{P} = \text{number of positives in ground truth} \tag{9.4}$$
$$\mathbf{N} = \text{number of negatives in ground truth} \tag{9.5}$$
$$\mathbf{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{9.6}$$
$$\mathbf{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{9.7}$$
$$\mathbf{F1} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{9.8}$$
$$\mathbf{accuracy} = \frac{\text{TP} + \text{TN}}{\text{P} + \text{N}} \tag{9.9}$$

For the classification metrics, we thresholded the power demand using the `on_power_threshold` parameter defined in Table 9.4 to create a binary *on/off* vector for the ground-truth power demand and the estimated power demand.

The disaggregation metrics we used are:

$$\mathbf{E} = \text{total actual energy} \tag{9.10}$$

$$\hat{\mathbf{E}} = \text{total predicted energy} \tag{9.11}$$

$$\mathbf{y_t^{(i)}} = \text{appliance } i \text{ actual power at time } t \tag{9.12}$$

$$\hat{\mathbf{y}}_\mathbf{t}^{(\mathbf{i})} = \text{appliance } i \text{ estimated power at time } t \tag{9.13}$$

$$\bar{\mathbf{y}}_\mathbf{t} = \text{aggregate actual power at time } t \tag{9.14}$$

$$\textbf{relative error in total energy} = \frac{\hat{E} - E}{\max(E, \hat{E})} \tag{9.15}$$

$$\textbf{mean absolute error} = {}^1\!/_T \sum_{t=1}^{T} |\hat{y}_t - y_t| \tag{9.16}$$

proportion of total energy correctly assigned =

$$1 - \frac{\sum_{t=1}^{T} \sum_{i=1}^{n} |\hat{y}_t^{(i)} - y_t^{(i)}|}{2 \sum_{t=1}^{T} \bar{y}_t} \tag{9.17}$$

The proportion of total energy correctly assigned is taken from (J. Zico Kolter & M. J. Johnson 2011).

We used multiple metrics because, in our view, no single metric is informative for all use-cases. For example, consider the use-case where we want to provide users with a weekly breakdown of the energy used by each appliance. In this case, we care about getting the estimated energy correct; but we do not care about getting the timing correct. Hence, in this case, a metric such as the mean absolute error would be informative. On the other hand, if our use-case is informing users which appliances are currently switched *on*, then we care about timing but we do not care about energy estimation. Hence a metric such as F1-score might be appropriate.

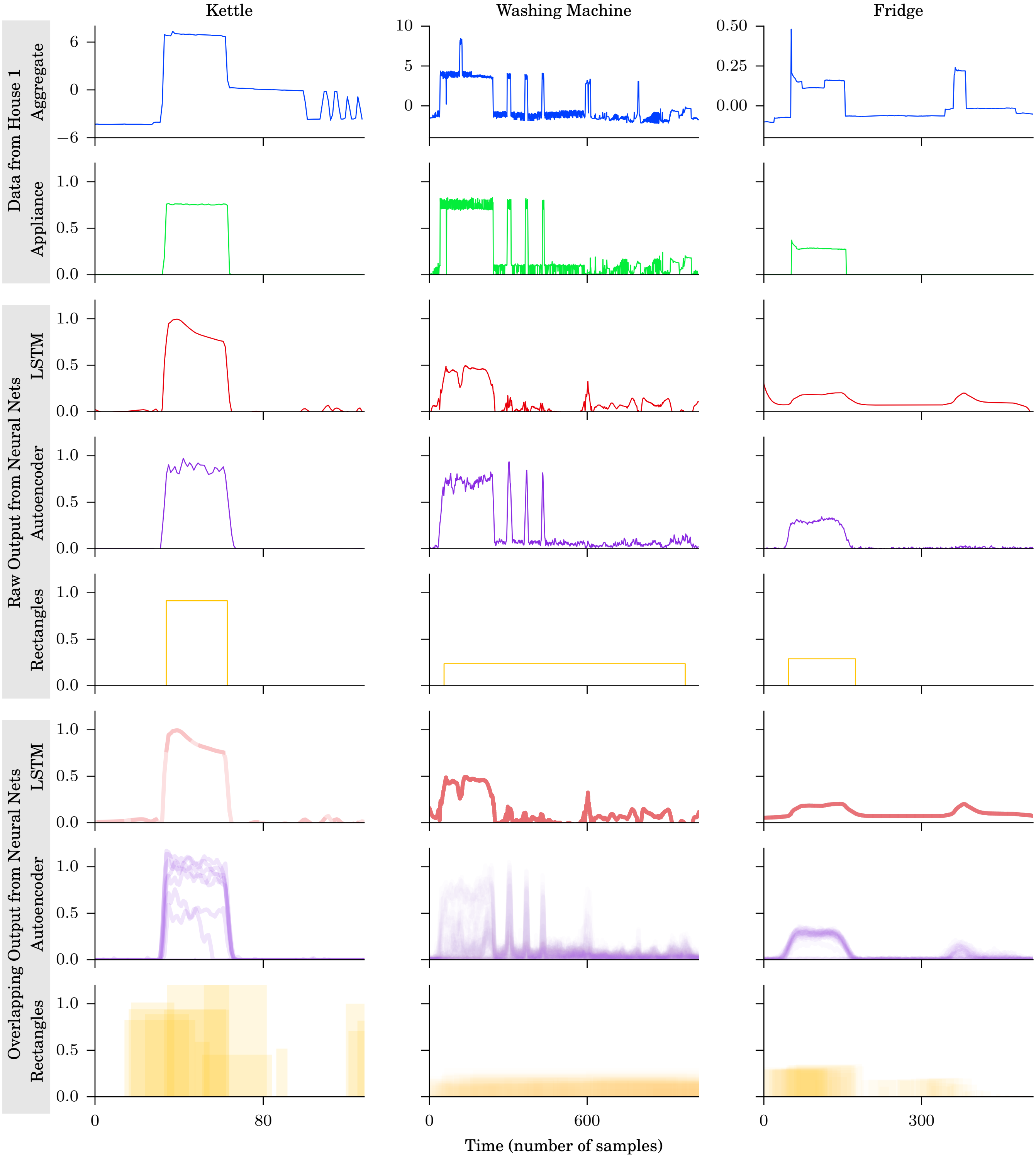


Figure 9.2: Example outputs produced by all three neural network architectures for three appliances. Each column shows data for a different appliance. The rows are in three groups (the tall grey rectangles on the far left). The top group shows measured data from House 1. The top row shows the measured aggregate power data from House 1 (the input to the neural nets). The Y-axis scale for the aggregate data is standardised such that its mean is 0 and its standard deviation is 1 across the data set. The Y-axis range for all other subplots is $[0, 1]$. The second row shows the single-appliance power demand (i.e. what the neural nets are trying to estimate). The middle group of rows shows the raw output from each neural network (one pass through each network). The bottom rows show the result of sliding the network over the aggregate data with `STRIDE=16` and overlapping the output. For the 'rectangles' net, the the rectangle height should be the *mean* power demand over the duration of the identified activation.

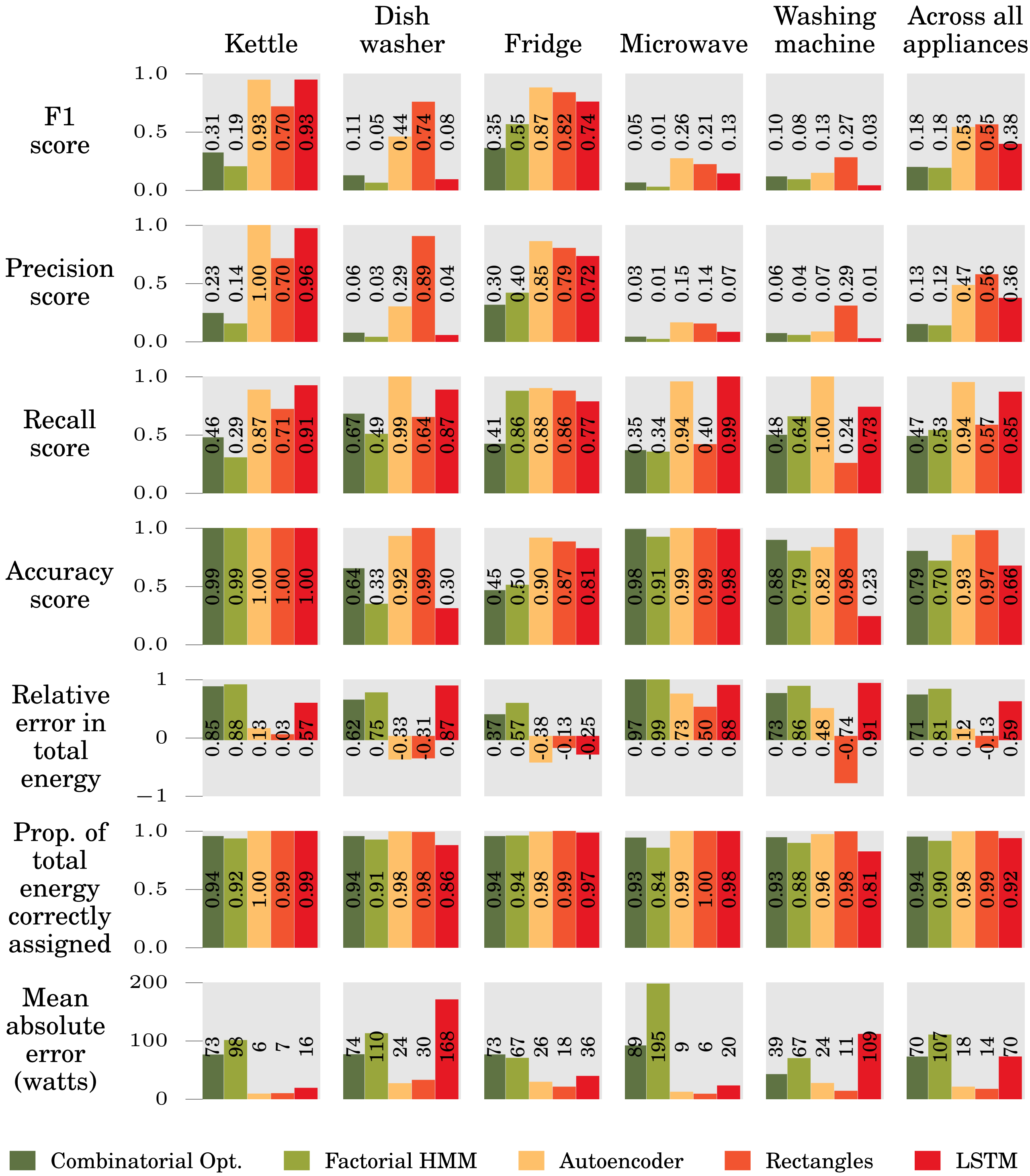


Figure 9.3: Disaggregation performance on a house not seen during training.

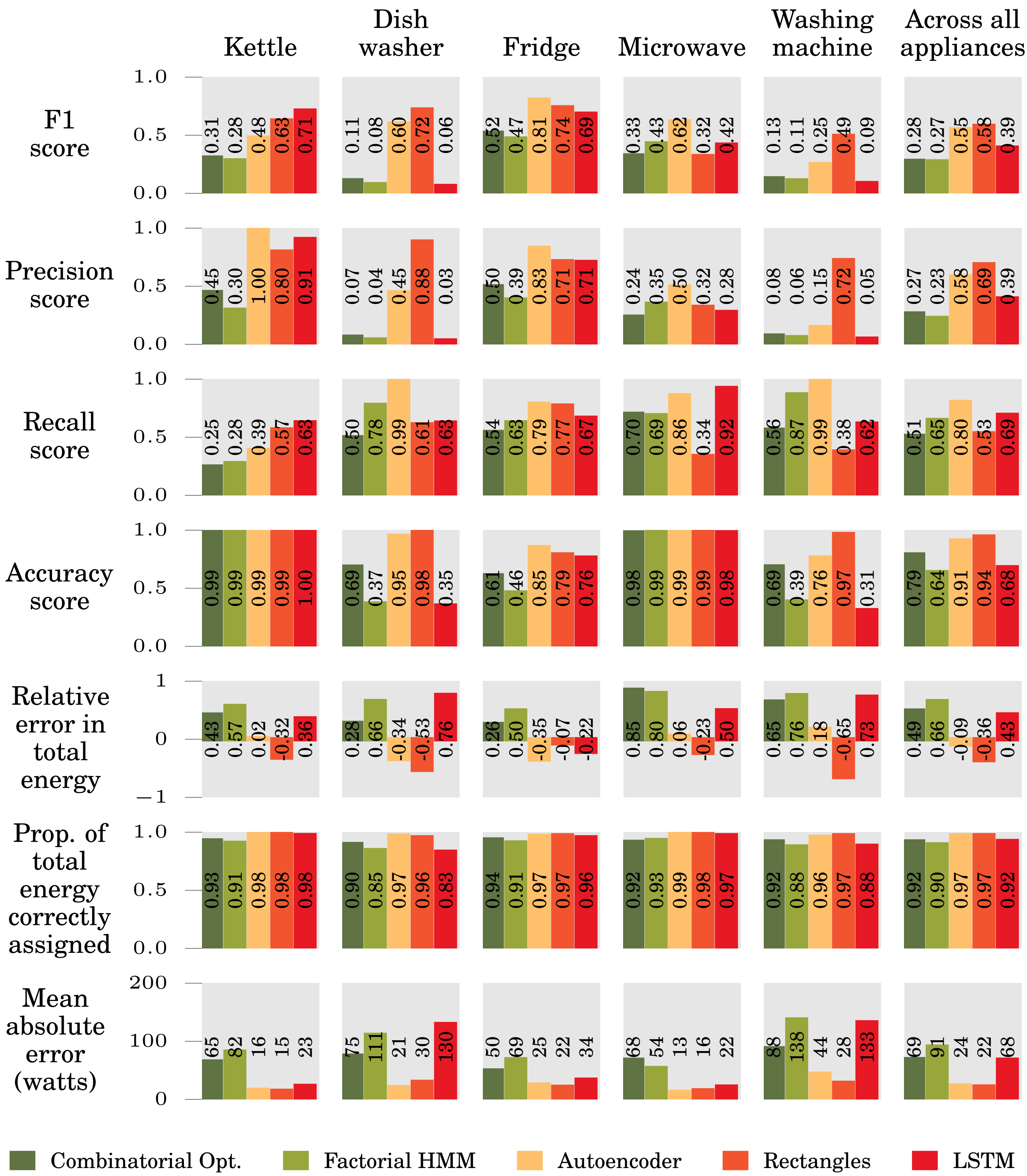


Figure 9.4: Disaggregation performance on houses seen during training (the time window used for testing is different to that used for training).

9.7 Conclusions & future work

We have adapted three neural network architectures to NILM. The denoising autoencoder and the 'rectangles' architectures perform well, especially on unseen houses. We believe that deep neural nets show great promise for NILM. But there is plenty of work still to do!

It is worth noting that our comparison between each architecture is not entirely fair because the architectures have a wide range of trainable parameters. For example, every LSTM we used had 1M parameters whilst the larger dAE and rectangles nets had over 150M parameters (we did try training an LSTM with more parameters but it did not appear to improve performance).

Such large networks are likely to be prone to overfitting. We attempted to protect against overfitting by training on an effectively infinitely large dataset of synthetic aggregate data. We also validated the performance against homes which were not seen by the networks during training. However, more work could be done to train and test on *large* numbers of houses.

The advantage of such large networks is that they are highly "expressive" and appear able to learn to detect and reconstruct the nuanced features of each appliance we tested on. Our original hypothesis was that deep neural networks would learn to extract complex features from the data. This appears to have happened; although future work would benefit from a detailed examination of exactly which features the networks are learning to extract.

Our LSTM results suggest that LSTMs work best for two-state appliances but do not perform well on multi-state appliances such as the dish washer and washing machine. One possible reason is that, for these appliances, informative 'events' in the power signal can be many time steps apart (e.g. for the washing machine there might be over 1 000 time steps between the first heater activation and the spin cycle). In principal, LSTMs have an arbitrarily long memory. But these long gaps between informative events may present a challenge for LSTMs. Further work is required to understand exactly why LSTMs struggle on multi-state appliances. One aspect of our LSTM results that we *did* expect was that processing overlapping windows of aggregate data would not be necessary for LSTMs because they always output the same estimates, no matter what the offset of the input window (see Figure 9.2).

However, LSTMs are by no means a dead end for NILM. Mauch & B. Yang 2015 (published shortly after our Neural NILM paper) achieve good NILM performance using LSTMs.

Another study (Nascimento 2016) published after our Neural NILM work compares multiple deep learning architectures on NILM. Nascimento compares CNNs, recurrent convolutional neural nets (RCNN), LSTMs, gated recurrent units (GRU) and a residual architecture. Nascimento finds that the GRU architecture performed the best and LSTMs performed the worst (which is consistent with our findings). Whilst direct comparisons are not possible, it is likely that Nascimento achieved better performance than we did. The differences between our work and his include: he used deeper networks than we did; he framed NILM as a classification problem (instead of the regression approach that we used) and hence his networks estimated the *state* of each appliance instead of directly estimating the power demand; he used categorical cross-entropy as the loss function (we used MSE); he used L2 regularisation and batch normalisation.

We must also note that the FHMM implementation used in our work is not 'state of the art' and neither is it especially tuned. Other FHMM implementations are likely to perform better. We encourage other researchers to download[5] our disaggregation estimates and ground truth data and directly compare against our algorithms!

This work represents just a first step towards adapting the vast number of techniques from the deep learning community to NILM, for example:

9.7.1 Train on more data

UK-DALE has many hundreds of days of data but only from five houses. Any machine learning algorithm is only able to generalise if given enough variety in the training set. For example, House 5's dish washer sometimes has four activations of its heater but the dish washers in the two training houses (1 and 2) only ever have two peaks. Hence the autoencoder completely ignores the first two peaks of House 5's dish washer! If neural nets are to learn to generalise well then we must train on much larger numbers of appliances (hundreds or thousands). This should help the networks to generalise across the wide variation seen in some classes of appliance.

[5] Data available from www.doc.ic.ac.uk/~dk3810/neuralnilm

9.7.2 Unsupervised pre-training

In NILM, we generally have access to much more *unlabelled* data than *labelled* data. One advantage of neural nets is that they could, in principal, be 'pre-trained' on unlabelled data before being fine-tuned on labelled data. 'Pre-training' should allow the networks to start to identify useful features from the data but does not allow the nets to learn to *label* appliances. (Pre-training is rarely used in modern image classification tasks because very large labelled datasets are available for image classification. But in NILM we have much more unlabelled data than labelled data, so pre-training is likely to be useful.) After unsupervised pre-training, each net would undergo *supervised* training. Instead of (or as well as) pre-training on all available unlabelled data, it may also be interesting to try pre-training largely on unlabelled data from each house that we wish to disaggregate.

9.7.3 Additional ideas for future work

1. Perform a grid search to find the optimal hyper-parameters for each architecture.
2. Train with both unlabelled data and labelled data: either by performing unsupervised pre-training or ladder networks (Rasmus et al. 2015).
3. Combine all three approaches: pre-train a 'rectangles' net on unlabelled data as an autoencoder. Then attach an RNN to the output to capture detailed temporal patterns. Or use an ensemble of multiple different approaches.
4. Experiment with more permutations of the nets.
5. Experiment with dropout and batch normalisation, especially batch normalisation modified for RNNs (Cooijmans et al. 2016).
6. Try training one large net to do multiple appliances.
7. Improve the 'rectangle' method to output multiple states per appliance.
8. Exploit external features such as the time of day that appliances are often used; or correlations with external temperature (see Chapter 4 for examples of these patterns).
9. Build more sophisticated synthesiser of aggregate data.

10. Experiment with ways to give the network information about the absolute power (instead of independently centring each input sequence) whilst also allowing the network to generalise well.

11. Try variational autoencoders.

12. Generate a probabilistic output (either using existing 'layering' approach or mixture density networks or variational approaches).

13. Perform fully integrated, multi-appliance disaggregation: use discrete optimisation to find most likely set of appliances. Or an RNN which sees aggregate data as well as output of upstream appliance classifier.

14. Try adapting Spatial Transformer Networks (Jaderberg et al. 2015; Sønderby et al. 2015) to NILM. i.e. allow explicit invariance to where in the input window the appliance activation appears.

15. We need to handle multiple instances of the same class of appliance (e.g. a house with two fridges). If using AEs, perhaps each instance would have a different code layer activation, which would allow for some separation.

16. Try adapting attention-based RNNs to NILM (Bahdanau et al. 2016).

17. Try regularising RNNs with temporal coherence (Jonschkowski & Brock 2015) or, perhaps better, the 'stabilising activations' regulariser developed by Krueger & Memisevic 2015.

18. Deep Residual Learning for Image Recognition (He et al. 2015).

19. Combine Neighbourhood NILM (Batra et al. 2016a) with Neural NILM. Or train a net to exploit similar patterns found in Batra's work. My expectation is that the best performance (when 10-second data is available) would be achieved by combining Neighbourhood-NILM with Neural NILM. i.e. use neural nets to identify signatures, then use Neighbourhood-NILM to exploit prior information over the population. Or perhaps train a (smallish) neural net to replace the KNN in Neighbourhood-NILM (i.e. where the neural net learns to map from monthly aggregate data (and any available metadata) to monthly appliance break down).

20. Try using *Grid* long short-term memory cells (Kalchbrenner et al. 2016).

Chapter 10

Conclusion

This thesis addressed several questions on the topic of non-intrusive load monitoring. We have contributed to multiple sub-fields within NILM including datasets, metadata, algorithm development and the social science of disaggregated energy feedback.

10.1 Summary of thesis achievements

The thesis achievements are outlined below:

- We built a cost-effective and accurate hardware and software system for collecting electricity demand data from homes. We demonstrated that the system has a high measurement accuracy.
- Our data collection system was deployed to five homes to record the UK-DALE dataset. This dataset is the only UK dataset recorded at high temporal resolution.
- We collaborated with other NILM researchers to build NILMTK, an open-source NILM toolkit. This has many features including the ability to work with datasets too large to fit into system memory and the implementation of multiple reference NILM algorithms and metrics.
- To make it easier for researchers to describe and consume datasets, we developed the first metadata schema for disaggregated electricity data, the NILM Metadata schema.

- We presented the first application of deep neural networks to energy disaggregation and we demonstrated that all three of our neural Neural NILM algorithms achieve better F1 scores (averaged over all five appliances) than the benchmark algorithms that we tested against.

- Finally, we presented the first systematic review of the social science literature on whether disaggregated energy data actually helps people to reduce their energy consumption.

10.2 Future work

In this section, we discuss general ways to extend the work discussed in this thesis. Please note that chapters 5, 6, 9 and 2 each discussed future work relevant to the chapter in question.

10.2.1 A NILM algorithm competition

One problem with NILM research which is holding back innovation is that it is currently impossible to compare the performance of NILM algorithms across researchers. We developed NILMTK, in part, to help researchers to validate their NILM algorithms using a common set of metrics and datasets; but researchers are still free to cherry pick which metrics to publish.

A more robust approach to compare the performance of NILM algorithms may be to run a regular NILM algorithm competition. All teams would download a common dataset against which to compare their algorithms. Part of the downloaded dataset would include 'ground truth' data to train the algorithms. A second part of the dataset for testing would be have the ground truth hidden. Each team would upload their appliance-by-appliance estimates to the competition's web platform which would automatically compare each team's estimates against the hidden ground truth and would compute a range of metrics.

Regular competitions have been used to great effect in some other fields of machine learning. For example, the ImageNet Large Scale Visual Recognition Challenge (Russakovsky et al. 2015) has played a crucial role in the recent dramatic increase in performance of visual recognition algorithms.

I wrote a detailed proposal for a NILM competition[1] and submitted it to the Energy Disaggregation mailing list[2] for discussion (Kelly 2016).

10.2.2 Build a NILM web service

Building a NILM web service means running an Internet server which can accept aggregate power data and returns disaggregated estimates.

The ultimate aim would be to make disaggregation as easy as possible.

There could be two interfaces: 1) a simple web interface for humans to use and 2) a simple API to allow other machines to interact with the disaggregation service and hence allow other developers to build new services based on my disaggregation web service. There are several advantages to providing an open-source web service:

- **Lower the barrier to entry**: Allow other disaggregation researchers to dive straight into working on an open-source state-of-the-art disaggregation algorithm without having to reinvent the wheel.
- **Provide a benchmark against which other algorithms can be compared**: It is currently practically impossible to compare any pair of disaggregation papers. Each paper uses a different dataset, different metrics, different pre-processing etc. As such, we cannot measure progress. Having a high performance open-source algorithm would allow other researchers to directly compare their approaches to ours, and hence we can start to measure progress.
- **Existing commercial disaggregation services only provide very sparse output** (e.g. just summary statistics for a week of estimates). This greatly reduces the usefulness of the output.

Tang et al. 2014 have built a disaggregation web service but it requires manually entering each appliance's rated power demand. The aim with our web service would be to pre-train our Neural NILM algorithm (see chapter 9) to automatically recognise appliances, without requiring input from users.

[1] https://docs.google.com/document/d/1CGoiNNkcnAFpo7Lci0Dv7AIliK3xz7Rjq_mBBh0WwrQ
[2] https://groups.google.com/forum/#!topic/energy-disaggregation/4I9rHMpkMTY

10.2.3 Run a large randomised controlled trial

As discussed in chapter 2, the only two field trials that we are aware of which directly compare disaggregated feedback against aggregate feedback (Churchwell et al. 2014; Sokoloski 2015) compare aggregate feedback on an *in home display* against disaggregate feedback on a *website*. Hence these studies do not 'cleanly' compare disaggregated feedback against aggregate feedback because the trial is confounded by the effect of using different display media in the two conditions. Hence it would be interesting to run a large, randomised controlled trial which compares disaggregated feedback against aggregate feedback using the *same* display media in both conditions.

Appendix A

UK-DALE

A.1 Radio frequency (RF) details

The RFM12b module has a large number of configuration settings. In order to communicate with the EDF Transmitter Plugs, we give our own base station the same RF configuration settings used by the EDF equipment. The RMF12b has far too many parameters to search manually. So we found the appropriate starting point for our RF configuration settings by 'sniffing' configuration packets from the Serial Peripheral Interface (SPI) which connects the EcoManager's microcontroller to its RF module (see Figure A.1). We used a Bus Pirate[1] to sniff the SPI bus.

To maximise the distance over which we can transmit, we experimented with several antenna and RF module configurations. We settled on a 1/4-wavelength antenna combined with a ground plane composed of four 1/4-wavelength wires in a cross shape running in the plane of the ground, originating from just below the point at which the antenna connects to the Nanode's printed circuit board.

[1] http://dangerousprototypes.com/docs/Bus_Pirate

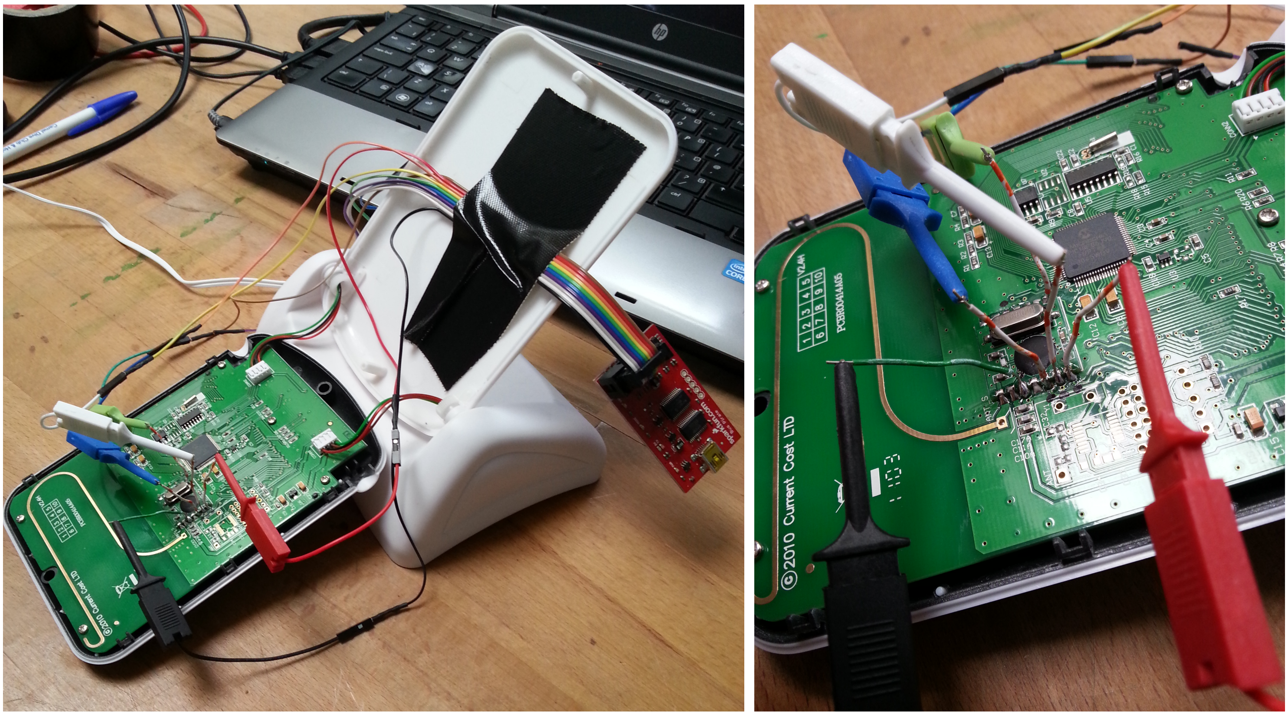


Figure A.1: Sniffing the wireless configuration parameters from the Current Cost base station. The image on the left shows a Bus Pirate (the small red circuit board on the far right of the image) connected to a Current Cost base station, ready to 'sniff' configuration data from the serial bus on the Current Cost base station. The image on the right shows a close up of the wireless module on the Current Cost base station.

A.2 Disadvantages of using a CT clamp connected to a wireless transmitter

- CT clamps measure current (I). The transmitter usually has no way to measure voltage and so must use a hard-coded value for voltage (V) to calculate a power reading (P) using $P = I \times V$. However, mains voltage in the UK is allowed to vary by +10% to -6% (sometimes quite abruptly) so power readings for a linear resistive load can vary by +20% to -12% (as noted by G. W. Hart 1992). These abrupt changes due to external noise are problematic for disaggregation algorithms because disaggregation algorithms tend to rely on *changes* in power demand to detect appliance state changes. However, not all appliances are affected by voltage variations. The power demand for 'constant power' devices remains constant across the legal voltage range. So, even if we have a measurement of voltage, it is not desirable to normalise the power demand by the measured voltage signal because different appliances respond differently to variation in supply voltage.

- Battery powered transmitters tend to sparsely sample from their CT clamp in order to minimise battery usage. Hence rapid changes may not be captured.

- Without instantaneous measurements of both voltage and current, it is not possible to measure active power or reactive power. Hence CT clamps without voltage measurements can only *estimate* apparent power.

- The OpenEnergyMonitor[2] emonTx is unaffected by all three disadvantages mentioned above. However, the version of the emonTx available in 2012 when we designed our system used an analogue to digital converter with only 10 bits of resolution. If we want to measure a primary current which varies from, say, 0 to 30 amps then the emonTx can only resolve changes larger than 14 watts. (The emonTx uses 10 bits of resolution to capture both the positive and negative sides of the AC signal so in effect it uses only 9 bits to cover a 30 amp range. $30\,\text{A} \div 2^9$ ADC steps $= 0.06\,\text{A}$ per ADC step so it can resolve changes in current larger than or equal to $0.06\,\text{A}$. And $0.06\,\text{A} \times 230\,\text{V} = 13.8\,\text{W}$). 'Real' smart meters will almost certainly have considerably higher resolution so, unfortunately, the emonTx available in 2012 when we were designing our system was not a suitable proxy for a 'real' smart meter.

A.3 Calculation of measurement resolution

Let us calculate a rough estimate for our measurement resolution. If we want to measure a primary current with a range of 0 to $30\,\text{A}_{\text{rms}}$ then we should be able to resolve changes in primary current of approximately $3\,\text{mA}$ per sample ($30\,\text{A}_{\text{rms}} \times \sqrt{2} \times 2 \approx 85\,\text{A}_{\text{peak-to-peak}}$ and $85\,\text{A}_{\text{peak-to-peak}} \div 2^{15}$ADC steps $\approx 3\,\text{mA}$). For the voltage measurement, if we want a range of 0 to $253\,\text{V}_{\text{rms}}$ ($230\,\text{V}_{\text{rms}} + 10\,\%$) then we should be able to resolve changes of approximately $22\,\text{mV}$ per sample ($253\,\text{volt}_{\text{rms}} \times \sqrt{2} \times 2 \approx 716\,\text{V}_{\text{peak-to-peak}}$ and $716\,\text{V}_{\text{peak-to-peak}} \div 2^{15}$ADC steps $\approx 22\,\text{mV}$). Given that the sensors are likely to be noisy and given that we are only providing $0.7\,\text{V}_{\text{peak-to-peak}}$ to the ADC for the voltage measurements, we should downgrade our resolution per sample to about $30\,\text{mV}$ and $5\,\text{mA}$ for voltage and current respectively. This gives us a resolution for power of approximately $30\,\text{mV} \times 5\,\text{mA} = 150\,\text{mW}$.

[2] https://openenergymonitor.org

A.4 Known issues

- Each IAM draws a little power (active power $\approx$ 0.9 W; apparent power $\approx$ 2.4 VA). House 1 has IAMs installed on almost every appliance and the *correlation* of the sum of all submeters in House 1 with the mains is 0.96. Yet the proportion of energy submetered in House 1 is only 80%. This reasonably low value for the proportion of energy submetered is likely due in large part to the fact that the 52 EDF IAMs installed in House 1 draw approximately 50 W in total, yet this power is not measured by the individual appliance meters.

- The IAMs and the Current Cost transmitters occasionally report spurious readings. `rfm_ecomanager_logger` filters out any readings above 4 kW for IAMs and above 20 kW for whole-house readings. 4 kW is above the safe maximum power draw for a UK mains appliance (2.99 kW = 13 amps $\times$ 230 volts). 20 kW is more than twice the maximum whole house power reading we have recorded (8.765 kW) across all houses in our dataset.

- Whilst the Atom motherboard's ADC is capable of sampling at 96 kHz, we had to use 44.1 kHz because the system produced buffer overflow errors if the sample rate was above 44.1 kHz. There remain some missing samples in the 16 kHz data due to buffer overflow errors during recording.

A.5 Usage notes

Any software designed to process REDD files should be able to open UK-DALE data (although the extra metadata provided by UK-DALE will be ignored).

The open-source energy disaggregation toolkit NILMTK (see Chapter 5) includes an importer for UK-DALE data. NILMTK can handle the metadata provided with UK-DALE. An HDF5 version of the UK-DALE dataset for use with NILMTK is available for download (please see Section 4.3 of this chapter).

There are several aspects of the dataset that might need to be addressed using appropriate pre-processing:

Data packets from the wireless meters are occasionally lost in transmission. Around 6% of packets are lost from the Current Transformer sensors and around 0.02% of packets are lost from Individual Appliance Monitor plugs. The sample period for the 6 second data may drift up or down by a second.

Some individual appliance monitors are switched off (and hence do not transmit any data) when the appliance is switched off from the mains. We switch them off to reduce the risk of an electrical fault causing a fire, and to save energy. Some appliances are unplugged for the majority of the time. For example, the vacuum cleaner is physically unplugged from the wall socket when not in use and the vacuum cleaner's meter is left attached to the cleaner's power plug (and hence is not powered). As a rule of thumb, any gap in the data longer than two minutes can be assumed to be caused by the appliance (and monitor) being switched off from the mains. Hence gaps longer than two minutes can safely be filled with zeros. The threshold of two minutes was chosen because we observed gaps less than two minutes caused by a succession of radio transmission errors. Any gap shorter than two minutes can be forward-filled from the previous reading.

Some appliances draw more than 0 watts when nominally 'off'. We typically use 5 watts as the threshold between 'on' and 'off'. The metadata includes an `on_power_threshold` property for each appliance. This property is present if the on power threshold for that appliance is not 5 watts.

NILMTK contains preprocessing tools for handling these scenarios.

Appendix B

NILMTK

B.1 Dependencies

One reason that we decided to use the Python language is because Python has a large number of high quality, open-source libraries for data analysis, plotting and machine learning (T. E. Oliphant 2007; Millman & Aivazis 2011). In the following list, we will briefly describe some of NILMTK's main dependencies:

The following packages are part of the SciPy Stack[1] (pronounced 'sigh-pie'):

NumPy (pronounced 'num-pie') provides N-dimensional arrays and many powerful operators to manipulate those arrays (van der Walt et al. 2011).

pandas builds on top of Numpy to provide, amongst other features, excellent support for time series data. For example, `Pandas.DataFrame.resample()` can elegantly up-sample or down-sample a time series (McKinney 2010).

Matplotlib is a plotting library (Hunter 2007).

IPython is an advanced, powerful interactive shell for Python, which is particularly well suited to scientific applications (Perez & Granger 2007).

SciPy (the library) provides efficient numerical routines such as numerical integration and optimisation (Jones et al. 2001–).

[1] https://www.scipy.org

The following packages are also used by NILMTK:

scikit-learn is a machine learning library (Pedregosa et al. 2011; Buitinck et al. 2013).

NILM Metadata is the implementation of our NILM Metadata schema (see Chapter 6). NILMTK v0.2 is tightly integrated with the NILM Metadata schema and uses many of the same concepts.

Appendix C

NILM Metadata

C.1 Implementation

We initially used defined the syntactic elements of the schema using JSON Schema Draft 4[1]. But specifying a formal schema with all the complexity of NILM Metadata quickly became impractical. So the schema does not currently have a machine-readable definition. Instead the schema is specified using extensive human-readable documentation available at `http://nilm-metadata.readthedocs.io`.

The companion code which implements the inheritance mechanism in NILM Metadata and performs validation is written in Python. We make use of the `jsonschema`[2] package for validation and `PyYAML`[3] for loading YAML files. Metadata instances can be written in JSON or YAML.

Prior to validating each appliance, the `properties` object specified by the 'appliance' schema is updated with concatenated `additional_properties` specified by the appliance's ancestors.

C.2 File organisation

To make the metadata reasonably easy for a human to navigate, we split the metadata into separate files, all contained within a `metadata` folder. Each `metadata` folder will have exactly

[1] `http://json-schema.org/`
[2] `https://github.com/Julian/jsonschema`
[3] `http://pyyaml.org/wiki/PyYAML`

one `dataset.yaml` file and some number of `building<I>.yaml` files (where `I` is an integer).

C.3 Example

`dataset.yaml`

```
name: UK-DALE
long_name: >
  UK Domestic Appliance-Level Electricity
```

`meter_devices.yaml`

```
- model: EnviR
  manufacturer: Current Cost
  measurements:
  - physical_quantity: power
    ac_type: apparent
    lower_limit: 0
    upper_limit: 30000
```

`building1.yaml`

```
instance: 1
rooms:
- {name: kitchen, instance: 1}
- {name: lounge, instance: 1}
elec_meters:
  1:
    device_model: EnviR
    site_meter: true
    sensors:
    - data_location: house1/channel_1.dat
  2:
    device_model: EnviR
```

```
submeter_of: 1
sensors:
- data_location: house1/channel_2.dat
preprocessing_applied:
  clip: {maximum: 4000}
appliances:
- type: light
  components:
  - type: LED lamp
    count: 10
    nominal_consumption: {on_power: 10}
    manufacturer: Philips
    year_of_manufacture: 2011
  - type: dimmer
  on_power_threshold: 10
  main_room_light: true
  dates_active:
  - {start: 2012, end: 2013}
```

Appendix D

Reactive power versus real power

Some utility-installed smart meters can measure several variables at once: voltage, *reactive power* and *real power*. What is the difference between *reactive power* and *real power*?

Mains is an *alternating current* source. If the load is a simple *resistive* load like a heater then current and voltage are perfectly in phase (as you might expect). Resistive loads passively accept changes in the supply voltage without opposition because they have no ability to store charge.

Some loads oppose the change inherent in an AC supply. These loads have a *capacitance* or *inductance* which allows them to store charge. This stored charge causes them to *react* to change in the supply. If the load is purely *reactive* then voltage and current will be 90 degrees out of phase hence the product of voltage and current will be positive for half a cycle and negative for the other half, hence no net energy flows to the device (but energy is lost in the wiring). No practical devices are *purely* reactive.

Most devices draw both *real power* and *reactive power*. Different devices draw different proportions of real and reactive power. Hence, if we measure both real and reactive power then we can plot devices on a two-dimensional plot like the one in Figure 7.5.

Appendix E

Copyright permissions

Table E.1: Copyright permissions table

Figure or Table #	Caption	Source	Copyright holder	Permission requested on	Permission granted?	Notes
Fig 1.1 and 5.1	Number of NILM publications per year	Parson 2015	Oliver Parson	2016-08-18	Yes	-
Fig 1.2	Screenshot of Smappee's itemised bill	St. John 2015b	Greentech Media	2016-08-31	Not yet	-
Tables 1.1 & 1.2	NILM companies	Adapted and extended from Parson 2012–2016	Oliver Parson	2016-08-31	Yes	-
Fig 1.3	Fossil-fuel emissions estimated to be compatible with a global temperature rise of 2°C	Gasser et al. 2015	Nature Publishing Group / Macmillan Publishers Ltd.	2016-08-31	Yes	Automatically cleared using RightsLink
Fig 1.4	Past and future changes in global mean sea level	Clark et al. 2016	Nature Publishing Group / Macmillan Publishers Ltd.	2016-08-31	Yes	Automatically cleared using RightsLink
Table 1.3	Responses to the question "Do you think accuracte information..."	Mansouri et al. 1996	Elsevier	2016-08-31	Yes	Automatically cleared using RightsLink
Fig 1.5	GB electricity demand profiles	Gavin 2014	Claire Gavin	2016-08-31	Not yet	-
Fig 4.2, Table 4.1, Fig 4.5, Fig 4.6, Fig 4.7, Fig 4.8, Fig 4.9, Fig 4.10, Fig 4.9	Various	Kelly & Knottenbelt 2015b (my work)	Nature Publishing Group / Macmillan Publishers Ltd.	2016-08-31	Yes	distributed under a Creative Commons CC-BY license

Continued on next page

Table E.1 – continued from previous page

Figure or Table #	Caption	Source	Copyright holder	Permission requested on	Permission granted?	Notes
Fig 5.3, Fig 5.5, Fig 5.6, Fig 5.7, Table 5.1	Various	Batra et al. 2014a and Kelly et al. 2014 (I created these figures)	ACM	2016-08-31	Yes	ACM permit authors to re-use their own work
Fig 5.8	Predicted power estimates generated by the CO and FHMM algorithms	Batra et al. 2014a (Nipun Batra's figure)	ACM	2016-08-31	Yes	ACM permit authors to re-use their own work
Fig 6.1, Fig 6.2	Various	Kelly & Knottenbelt 2014 (I created these figures)	IEEE	2016-08-31	Yes	"IEEE does not require individuals working on a thesis to obtain a formal reuse license"
Fig 7.4	George Hart's 'signature taxonomy'	G. W. Hart 1992	IEEE	2016-08-31	Yes	"IEEE does not require individuals working on a thesis to obtain a formal reuse license"
Fig 9.1, Table 9.1, Table 9.2, Table 9.3, Table 9.4, Figure 9.2, Figure 9.3, Figure 9.4	Various	Kelly & Knottenbelt 2015a (I created these figures)	ACM	2016-08-31	Yes	ACM permit authors to re-use their own work
Fig 2.1	The desktop PC used to display disaggregated energy data	Dobson & Griffin 1992	ACEEE	2016-08-31	Not yet	-
Fig 2.2	An example of the 'coarse-grained' disaggregation performed by HEA.	`http://corp.hea.com/how-it-works`	HEA	2016-08-31	Yes	-
Fig 3.1	A touchscreen 'Home Manager' made by Unity Systems and installed in 1990	`http://imgur.com/a/Jb6jW`	avboden	2016-08-31	Not yet	Contacted used via imgur
Fig 3.2	General Electric's Nucleus energy manager	Dahl 2011	CharlesHudson.com ?	2016-09-13	Not yet	-
Fig 3.3	General Electric's Nucleus iPhone application showing energy usage for a single smart appliance	General Electric 2012	?	-	Not yet	No contact information provided
Fig 3.4	Smart homes can make life more complex	Cate 2016	cate	2016-08-31	Yes	Tweeted her

Bibliography

Adabi, A.; Mantey, P.; Holmegaard, E. & Kjaergaard, M.B. (2015). 'Status and Challenges of Residential and Industrial Non-Intrusive Load Monitoring'. In: *Conference on Technologies for Sustainability*. SusTech. IEEE. Ogden, UT, pages 181–188. DOI: 10.1109/SusTech.2015.7314344 (cited on page 132).

Alcalá, José; Parson, Oliver & Rogers, Alex (2015). 'Detecting Anomalies in Activities of Daily Living of Elderly Residents via Energy Disaggregation and Cox Processes'. In: *2nd International Conference on Embedded Systems for Energy-Efficient Built Environments*. BuildSys. ACM. Seoul, South Korea, pages 225–234. DOI: 10.1145/2821650.2821654 (cited on page 26).

Altrabalsi, Hana; Liao, Jing; Stankovic, Lina & Stankovic, Vladimir (2014). 'A low-complexity energy disaggregation method: Performance and robustness'. In: *Symposium on Computational Intelligence Applications in Smart Grid*. CIASG. IEEE. Orlando, FL, pages 1–8. DOI: 10.1109/CIASG.2014.7011569 (cited on page 143).

Amirach, Nabil; Xerri, Bernard; Borloz, Bruno & Jauffret, Claude (2014). 'A new approach for event detection and feature extraction for NILM'. In: *21st International Conference on Electronics, Circuits and Systems*. ICECS. IEEE, pages 287–290. DOI: 10.1109/ICECS.2014.7049978 (cited on page 129).

Anderson, Kyle; Ocneanu, Adrian; Benítez, Diego; Carlson, Derrick; Rowe, Anthony & Bergés, Mario (2012). 'BLUED: A Fully Labeled Public Dataset for Event-Based Non-Intrusive Load Monitoring Research'. In: *2nd KDD Workshop on Data Mining Applications in Sustainability*. SustKDD. Beijing, China, pages 1–5 (cited on page 108).

Anderson, Will & White, Vicki (2009). *Exploring consumer preferences for home energy display functionality*. Technical report. UK: Energy Saving Trust (cited on page 16).

Armel, K Carrie; Gupta, Abhay; Shrimali, Gireesh & Albert, Adrian (2013). 'Is disaggregation the holy grail of energy efficiency? The case of electricity'. In: *Energy Policy* 52, pages 213–234. DOI: 10.1016/j.enpol.2012.08.062 (cited on pages v, 1, 132).

Atlas, Les E.; Homma, Toshiteru & II, Robert J. Marks (1988). 'An Artificial Neural Network for Spatio-Temporal Bipolar Patterns: Application to Phoneme Classification'. In: *Neural Information Processing Systems*. Edited by D.Z. Anderson. NIPS. American Institute of Physics, page 31 (cited on page 157).

Aydinalp-Koksal, Merih & Ugursal, V. Ismet (2008). 'Comparison of neural network, conditional demand analysis, and engineering approaches for modeling end-use energy consumption in the residential sector'. In: *Applied Energy* 85.4, pages 271–296. DOI: 10.1016/j.apenergy.2006.09.012 (cited on page 143).

Bahdanau, Dzmitry; Chorowski, Jan; Serdyuk, Dmitriy; Brakel, Philemon & Bengio, Yoshua (2016). 'End-to-end attention-based large vocabulary speech recognition'. In: *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Institute of Electrical & Electronics Engineers (IEEE). DOI: 10.1109/icassp.2016.7472618 (cited on page 179).

Baker, Monya (2015). 'Over half of psychology studies fail reproducibility test'. In: *Nature*. DOI: 10.1038/nature.2015.18248 (cited on page 33).

Baranski, Michael & Voss, Jürgen (2004a). 'Detecting patterns of appliances from total load data using a dynamic programming approach'. In: *4th International Conference on Data Mining*. ICDM. IEEE, pages 327–330. DOI: 10.1109/ICDM.2004.10003 (cited on page 143).

Baranski, Michael & Voss, Jürgen (2004b). 'Genetic algorithm for pattern detection in NIALM systems'. In: *International Conference on Systems, Man and Cybernetics*. Volume 4. ICSMC. IEEE, pages 3462–3468. DOI: 10.1109/ICSMC.2004.1400878 (cited on page 143).

Barbosa, Pedro; Brito, Andrey & Almeida, Hyggo (2015). 'Defending Against Load Monitoring in Smart Metering Data Through Noise Addition'. In: *30th Annual Symposium on Applied Computing*. SAC. ACM. Salamanca, Spain, pages 2218–2224. DOI: 10.1145/2695664.2695800 (cited on page 62).

Barker, S.; Musthag, M.; Irwin, D. & Shenoy, P. (2014). 'Non-intrusive load identification for smart outlets'. In: *International Conference on Smart Grid Communications*. SmartGridComm. IEEE. Venice, pages 548–553. DOI: 10.1109/SmartGridComm.2014.7007704 (cited on page 25).

Barker, Sean; Mishra, Aditya; Irwin, David; Cecchet, Emmanuel; Shenoy, Prashant & Albrecht, Jeannie (2012). 'Smart*: An open data set and tools for enabling research in sustainable homes'. In: *1st KDD Workshop on Data Mining Applications in Sustainability*. SustKDD. CiteSeerX DOI: 10.1.1.259.8712 (cited on page 108).

Bartram, Lyn (2015). 'Design Challenges and Opportunities for Eco-Feedback in the Home'. In: *IEEE Computer Graphics and Applications* 35.4, pages 52–62. DOI: 10.1109/mcg.2015.69 (cited on page 44).

Bastien, Frédéric; Lamblin, Pascal; Pascanu, Razvan; Bergstra, James; Goodfellow, Ian J.; Bergeron, Arnaud; Bouchard, Nicolas & Bengio, Yoshua (2012). 'Theano: new features and speed improvements'. In: *arXiv*. arXiv: 1211.5590 (cited on page 169).

Batra, Nipun; Gulati, Manoj; Singh, Amarjeet & Srivastava, Mani B. (2013). 'It's Different: Insights into Home Energy Consumption in India'. In: *5th Workshop on Embedded Systems For Energy-Efficient Buildings*. BuildSys. ACM. Roma, Italy, 3:1–3:8. DOI: 10.1145/2528282.2528293 (cited on pages 108, 223).

Batra, Nipun; Kelly, Jack; Parson, Oliver; Dutta, Haimonti; Knottenbelt, William; Rogers, Alex; Singh, Amarjeet & Srivastava, Mani (2014a). 'NILMTK: An Open Source Toolkit for Non-intrusive Load Monitoring'. In: *5th International Conference on Future Energy Systems*. e-Energy. ACM. Cambridge, UK. DOI: 10.1145/2602044.2602051 (cited on pages 29, 92, 95, 105, 196).

Batra, Nipun; Parson, Oliver; Bergés, Mario; Singh, Amarjeet & Rogers, Alex (2014b). 'A comparison of non-intrusive load monitoring methods for commercial and residential buildings'. In: *arXiv*. arXiv: 1408.6595 (cited on page 19).

Batra, Nipun; Singh, Amarjeet & Whitehouse, Kamin (2016a). 'Creating a Detailed Energy Breakdown from just the Monthly Electricity Bill'. In: *3rd International NILM Workshop*. Vancouver, Canada (cited on page 179).

Batra, Nipun; Singh, Amarjeet & Whitehouse, Kamin (2016b). 'Exploring The Value of Energy Disaggregation Through Actionable Feedback'. In: *3rd International NILM Workshop*. Vancouver, Canada (cited on page 19).

Batra, Nipun; Singh, Amarjeet & Whitehouse, Kamin (2016c). 'Gemello: Creating a Detailed Energy Breakdown from just the Monthly Electricity Bill'. In: *KDD*. Submitted (cited on page 42).

Baum, L.E. (1972). 'An Inequality and Associated Maximization Technique in Statistical Estimation of Probabilistic Functions of a Markov Process'. In: *Inequalities* (3), pages 1–8 (cited on page 137).

Baum, Leonard E. & Eagon, J. A. (1967). 'An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology'. In: *Bulletin of the American Mathematical Society* 73.3, pages 360–364. DOI: 10.1090/s0002-9904-1967-11751-8 (cited on page 137).

Baum, Leonard E. & Petrie, Ted (1966). 'Statistical Inference for Probabilistic Functions of Finite State Markov Chains'. In: *The Annals of Mathematical Statistics* 37.6, pages 1554–1563. DOI: 10.1214/aoms/1177699147 (cited on page 137).

Baum, Leonard E.; Petrie, Ted; Soules, George & Weiss, Norman (1970). 'A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains'. In: *The Annals of Mathematical Statistics* 41.1, pages 164–171. DOI: 10.1214/aoms/1177697196 (cited on page 137).

Baum, Leonard E. & Sell, George (1968). 'Growth transformations for functions on manifolds'. In: *Pacific Journal of Mathematics* 27.2, pages 211–227. DOI: 10.2140/pjm.1968.27.211 (cited on page 137).

Baxter, Elissa (2010). 'Failure to launch'. In: *The Sydney Morning Herald* (cited on page 61).

Beckel, Christian; Kleiminger, Wilhelm; Cicchetti, Romano; Staake, Thorsten & Santini, Silvia (2014). 'The ECO Data Set and the Performance of Non-intrusive Load Monitoring Algorithms'. In: *1st International Conference on Embedded Systems for Energy-Efficient Buildings*. BuildSys. ACM. Memphis, Tennessee, pages 80–89. DOI: 10.1145/2674061.2674064 (cited on pages 96, 108).

Bedwell, Ben; Leygue, Caroline; Goulden, Murray; McAuley, Derek; Colley, James; Ferguson, Eamonn; Banks, Nick & Spence, Alexa (2014). 'Apportioning energy consumption in the workplace: a review of issues in using metering data to motivate staff to save energy'. In: *Technology Analysis & Strategic Management* 26.10, pages 1196–1211. DOI: 10.1080/09537325.2014.978276 (cited on page 25).

Begley, C. Glenn & Ellis, Lee M. (2012). 'Drug development: Raise standards for preclinical cancer research'. In: *Nature* 483.7391, pages 531–533. DOI: 10.1038/483531a (cited on page 33).

Belley, Corinne; Gaboury, Sebastien; Bouchard, Bruno & Bouzouane, Abdenour (2013). 'Activity Recognition in Smart Homes Based on Electrical Devices Identification'. In: *6th International Conference on PErvasive Technologies Related to Assistive Environments*. PETRA. ACM. Rhodes, Greece, 7:1–7:8. DOI: 10.1145/2504335.2504342 (cited on page 25).

Bengio, Yoshua; LeCun, Yann et al. (2007). 'Scaling learning algorithms towards AI'. In: *Large-scale kernel machines* 34.5 (cited on page 156).

Bergstra, James; Breuleux, Olivier; Bastien, Frédéric; Lamblin, Pascal; Pascanu, Razvan; Desjardins, Guillaume; Turian, Joseph; Warde-Farley, David & Bengio, Yoshua (2010). 'Theano: a CPU and GPU Math Expression Compiler'. In: *Python for Scientific Computing Conference*. SciPy. Oral Presentation. Austin, Texas, USA (cited on page 169).

Bertagnolio, Stéphane (2012). 'Evidence-Based Model Calibration for Efficient Building Energy Services'. PhD thesis. Université de Liège, Liège, Belgium, page 303 (cited on page 21).

Bertoldo, Raquel; Poumadère, Marc & Jr., Luis Carlos Rodrigues (2015). 'When meters start to talk: The public's encounter with smart meters in France'. In: *Energy Research & Social Science* 9. Special Issue on Smart Grids and the Social Sciences, pages 146–156. DOI: 10.1016/j.erss.2015.08.014 (cited on page 15).

Bhatt, Jignesh; Shah, Vipul & Jani, Omkar (2014). 'An instrumentation engineer's review on smart grid: Critical applications and parameters'. In: *Renewable and Sustainable Energy Reviews* 40, pages 1217–1239. DOI: 10.1016/j.rser.2014.07.187 (cited on page 15).

Bidgely (2015). 'PG&E Pilot Yields 7.7% Energy Savings'. URL: http://www.bidgely.com/blog/pge-pilot-yields-7-7-energy-savings/ (cited on pages 18, 37, 38, 41).

Bishop, Christopher M. (1994). *Mixture density networks*. Technical report. Aston University, Birmingham, UK (cited on page 163).

Bittle, Ronald G.; Valesano, Robert M. & Thaler, Greg M. (1979–1980). 'The effects of daily feedback on residential electricity usage as a function of usage level and type of feedback information'. In: *Journal of Environmental Systems* 9.3, pages 275–287 (cited on page 44).

Bonfigli, Roberto; Squartini, Stefano; Fagiani, Marco & Piazza, Francesco (2015). 'Unsupervised algorithms for non-intrusive load monitoring: An up-to-date overview'. In: *15th International Conference on Environment and Electrical Engineering*. EEEIC. IEEE. Rome, Italy. DOI: 10.1109/eeeic.2015.7165334 (cited on pages 1, 132).

Bouloutas, A.; Hart, George William & Schwartz, M. (1991). 'Two extensions of the Viterbi algorithm'. In: *IEEE Transactions on Information Theory* 37.2, pages 430–436. DOI: 10.1109/18.75270 (cited on page 131).

Bradley, Peter; Leach, Matthew & Torriti, Jacopo (2013). 'A review of the costs and benefits of demand response for electricity in the UK'. In: *Energy Policy* 52. Special Section: Transition Pathways to a Low Carbon Economy, pages 312–327. DOI: 10.1016/j.enpol.2012.09.039 (cited on page 23).

Brandon, Gwendolyn & Lewis, Alan (1999). 'Reducing household energy consumption: a qualitative and quantitative field study'. In: *Journal of Environmental Psychology* 19.1, pages 75–85. DOI: 10.1006/jevp.1998.0105 (cited on page 44).

Bright, Peter (2014). 'Smart TVs, smart fridges, smart washing machines? Disaster waiting to happen'. In: *Ars Technica* (cited on page 61).

Brown, Rebecca (2014). 'Bringing It All Together: Design and Evaluation Innovations in the Alameda County Residential Behavior Pilot'. In: *Behavior, Energy, and Climate Change Conference*. BECC (cited on pages 35, 36, 38, 39).

Buchanan, Kathryn; Banks, Nick; Preston, Ian & Russo, Riccardo (2016). 'The British public's perception of the UK smart metering initiative: Threats and opportunities'. In: *Energy Policy* 91, pages 87–97. DOI: 10.1016/j.enpol.2016.01.003 (cited on pages 24, 62).

Buitinck, Lars; Louppe, Gilles; Blondel, Mathieu; Pedregosa, Fabian; Mueller, Andreas; Grisel, Olivier; Niculae, Vlad; Prettenhofer, Peter; Gramfort, Alexandre; Grobler, Jaques; Layton, Robert; Vanderplas, Jake; Joly, Arnaud; Holt, Brian & Varoquaux, Gaël (2013). 'API design for machine learning software: experiences from the scikit-learn project'. In: *European Conference on Machine Learning and Principles and Practices of Knowledge Discovery in Databases*. Prague, Czech Republic. arXiv: 1309.0238 (cited on page 190).

Cambridge Architectural Research Ltd. & Loughborough University (2013). *The Potential for Smart Meters in a National Household Energy Survey. Further Analysis of the Household Electricity Use Survey*. Technical report (cited on page 25).

Cate (2016). 'Tweet: When you're house sitting for millennials and ask how the lights work'. URL: https://twitter.com/c8ters/status/699701086656077825 (cited on pages 60, 196).

Chakravarty, Prateek & Gupta, Abhay (2013). 'Impact of Energy Disaggregation on Consumer Behavior'. In: *Behavior, Energy and Climate Change Conference*. BECC. UC Berkeley. Sacramento CA (cited on pages 26, 38).

Chambers, Craig; Ungar, David & Lee, Elgin (1989). 'An Efficient Implementation of SELF a Dynamically-typed Object-oriented Language Based on Prototypes'. In: *Object-oriented Programming Systems, Languages and Applications*. OOPSLA. ACM. New Orleans, Louisiana, USA, pages 49–70. DOI: 10.1145/74877.74884 (cited on page 115).

Chang, Hsueh-Hsien; Chien, Po-Ching; Lin, Lung-Shu & Chen, Nanming (2011). 'Feature Extraction of Non-intrusive Load-Monitoring System Using Genetic Algorithm in Smart Meters'. In: *8th International Conference on e-Business Engineering*. ICEBE. IEEE. Beijing, China, pages 299–304. DOI: 10.1109/ICEBE.2011.48 (cited on pages 143, 154).

Chang, Hsueh-Hsien; Lee, Meng-Chien; Chen, Nanming; Chien, Chao-Lin & Lee, Wei-Jen (2015). 'Feature Extraction Based Hellinger Distance Algorithm for Non-Intrusive Aging Load Identification in Residential Buildings'. In: *Industry Applications Society Annual Meeting*. IEEE. Addison, TX, pages 1–8. DOI: 10.1109/IAS.2015.7356778 (cited on page 19).

Chang, Hsueh-Hsien; Lin, Ching-Lung & Lee, Jin-Kwei (2010). 'Load Identification in Non-intrusive Load Monitoring Using Steady-State and Turn-on Transient Energy Algorithms'. In: *14th International Conference on Computer Supported Cooperative Work in Design*. CSCWD. IEEE. Shanghai, China, pages 27–32. DOI: 10.1109/CSCWD.2010.5472008 (cited on page 135).

Chang, Martin (2015). *15 Ways to Run Your Facility Better with Energy Data*. Technical report. Verdigris (cited on page 20).

Cheng, Vicky & Steemers, Koen (2011). 'Modelling domestic energy consumption at district scale: A tool to support national and local energy policies'. In: *Environmental Modelling & Software* 26.10, pages 1186–1198. DOI: 10.1016/j.envsoft.2011.04.005 (cited on page 22).

Chorowski, Jan; Bahdanau, Dzmitry; Cho, Kyunghyun & Bengio, Yoshua (2014). 'End-to-end Continuous Speech Recognition using Attention-based Recurrent NN: First Results'. In: arXiv: 1412.1602 (cited on page 165).

Christensen, Dane; Earle, Lieko & Sparn, Bethany (2012). *NILM Applications for the Energy-Efficient Home*. Technical report NREL/TP-5500-55291. National Renewable Energy Laboratory (cited on page 1).

Churchwell, Candice; Sullivan, Michael; Thompson, Dan & Oh, Jeeheh (2014). *HAN Phase 3 Impact and Process Evaluation Report*. Technical report. Nexant (cited on pages 36–38, 40–42, 183).

Cicchetti, R. (2014). 'NILM-Eval: Disaggregation of real-world electricity consumption data'. MSc. ETH Zurich (cited on page 96).

Clark, Peter U.; Shakun, Jeremy D.; Marcott, Shaun A.; Mix, Alan C.; Eby, Michael; Kulp, Scott; Levermann, Anders; Milne, Glenn A.; Pfister, Patrik L.; Santer, Benjamin D.; Schrag, Daniel P.; Solomon, Susan; Stocker, Thomas F.; Strauss, Benjamin H.; Weaver, Andrew J.; Winkelmann, Ricarda; Archer, David; Bard, Edouard; Goldner, Aaron; Lambeck, Kurt; Pierrehumbert, Raymond T. & Plattner, Gian-Kasper (2016). 'Consequences of twenty-first-century policy for multi-millennial climate and sea-level change'. In: *Nature Climate Change* 6.4, pages 360–369. DOI: 10.1038/nclimate2923 (cited on pages 13, 195).

Coakley, Daniel; Raftery, Paul & Keane, Marcus (2014). 'A review of methods to match building energy simulation models to measured data'. In: *Renewable and Sustainable Energy Reviews* 37, pages 123–141. DOI: 10.1016/j.rser.2014.05.007 (cited on page 21).

Coakley, Daniel; Raftery, Paul & Molloy, Padraig (2012). 'Calibration of whole building energy simulation models: Detailed case study of a naturally ventilated building using hourly measured data'. In: *1st Building Simulation and Optimization Conference*. Loughborough, UK, pages 57–64 (cited on page 21).

Colak, Ilhami; Fulli, Gianluca; Sagiroglu, Seref; Yesilbudak, Mehmet & Covrig, Catalin-Felix (2015). 'Smart grid projects in Europe: Current status, maturity and future scenarios'. In: *Applied Energy* 152, pages 58–70. DOI: 10.1016/j.apenergy.2015.04.098 (cited on page 15).

Cole, Agnim I. & Albicki, Alexander (1998a). 'Algorithm for nonintrusive identification of residential appliances'. In: *International Symposium on Circuits and Systems*. Volume 3. ISCAS. IEEE. Monterey, CA, 338–341 vol.3. DOI: 10.1109/ISCAS.1998.704019 (cited on page 135).

Cole, Agnim I. & Albicki, Alexander (1998b). 'Data Extraction for Effective Non-Intrusive Identification of Residential Power Loads'. In: *Instrumentation and Measurement Technology Conference*. Volume 2. IMTC. IEEE. St. Paul Minnesota, USA, 812–815 vol.2. DOI: 10.1109/IMTC.1998.676838 (cited on pages 135, 136).

Consumer Technology Association (2014). *CE Energy Usage Information (CE-EUI) ANSI/CTA-2047*. Technical report (cited on page 59).

Cooijmans, Tim; Ballas, Nicolas; Laurent, César; Gülçehre, Çağlar & Courville, Aaron (2016). 'Recurrent Batch Normalization'. In: *arXiv*. arXiv: 1603.09025 (cited on page 178).

Cox, Robert; Leeb, Steven B.; Shaw, Steven R. & Norford, Leslie K. (2006). 'Transient event detection for nonintrusive load monitoring and demand side management using voltage distortion'. In: *21st Annual Applied Power Electronics Conference and Exposition*. APEC. IEEE. DOI: 10.1109/APEC.2006.1620777 (cited on page 135).

Dahl, Timothy (2011). 'GE Introduces Their Smart Home Technologies at CES'. In: *Charles Hudson* (cited on pages 51, 196).

Dahlgreen, Will (2016). 'YouGov global survey: Britain among least concerned in the world about climate change'. URL: https://yougov.co.uk/news/2016/01/29/global-issues/ (cited on page 13).

Darby, Sarah (2006). *The effectiveness of feedback on energy consumption. A review for DEFRA of the literature on metering, billing and direct displays*. Technical report. Environmental Change Institute, University of Oxford (cited on page 16).

Davies, Peter (2011). 'Non-intrusive load identification and monitoring using its unique power signature'. GB2475172 (A) (cited on page 135).

Davis, Alexander L.; Krishnamurti, Tamar; Fischhoff, Baruch & Bruin, Wandi Bruine de (2013). 'Setting a standard for electricity pilot studies'. In: *Energy Policy* 62, pages 401–409. DOI: 10.1016/j.enpol.2013.07.093 (cited on pages 15, 35, 39, 41).

de Dear, Richard & Hart, Melissa (2002). *Appliance Electricity End-Use: Weather and Climate Sensitivity*. Technical report. Sustainable Energy Group, Australian Greenhouse Office, page 69 (cited on page 88).

DeBruin, Samuel; Ghena, Branden; Kuo, Ye-Sheng & Dutta, Prabal (2015). 'PowerBlade: A Low-Profile, True-Power, Plug-Through Energy Meter'. In: *13th Conference on Embedded Networked Sensor Systems*. SenSys. ACM. Seoul, South Korea, pages 17–29. DOI: 10.1145/2809695.2809716 (cited on pages 53, 54).

DECC (2014). *Smart Metering Equipment Technical Specifications: second version (SMETS2)*. Technical report. Version 1.58. UK: Department of Energy & Climate Change, UK Government (cited on pages 15, 57, 67, 71, 72, 130).

DECC (2016). *Provisional estimates of UK Greenhouse Gas emissions for 2015, including quarterly emissions for 4th quarter 2015*. Technical report. UK Government (cited on page 14).

DECC; Behavioural Insights Team & Madano (2016). *SMART METERS: DEROGATION GUIDANCE. Supporting energy supplier applications for trials of in-home display alternatives*. Technical report URN 16D/072. Department of Energy and Climate Change, UK Government (cited on page 42).

Dobson, John K. & Griffin, J. D. Anthony (1992). 'Conservation Effect of Immediate Electricity Cost Feedback on Residential Consumption Behavior'. In: *Summer Study on Energy Efficiency in Buildings*. Volume 10. American Council for an Energy-Efficient Economy (ACEEE). Washington, D.C., pages 33–35 (cited on pages 35, 36, 38, 43, 196).

Doyle, Frank; Duarte, Maria-Jose Rivas & Cosgrove, John (2015). 'Design of an Embedded Sensor Network for Application in Energy Monitoring of Commercial and Industrial Facilities'. In: *Energy Procedia* 83. Sustainability in Energy and Buildings: Proceedings of the 7th International Conference SEB-15, pages 504–514. DOI: 10.1016/j.egypro.2015.12.170 (cited on page 53).

Egarter, Dominik & Elmenreich, Wilfried (2013). 'EvoNILM: Evolutionary Appliance Detection for Miscellaneous Household Appliances'. In: *15th Conference Companion on Genetic and Evolutionary Computation*. GECCO. ACM. Amsterdam, The Netherlands, pages 1537–1544. DOI: 10.1145/2464576.2482733 (cited on page 143).

Egarter, Dominik; Monacchi, Andrea; Khatib, Tamer & Elmenreich, Wilfried (2015). 'Integration of legacy appliances into home energy management systems'. In: *Journal of Ambient Intelligence and Humanized Computing*, pages 1–15. DOI: 10.1007/s12652-015-0312-9 (cited on pages 47, 62).

Egarter, Dominik; Sobe, Anita & Elmenreich, Wilfried (2013). 'Evolving Non-Intrusive Load Monitoring'. In: *Applications of Evolutionary Computation*. Edited by AnnaI. Esparcia-Alcázar. Volume 7835. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pages 182–191. DOI: 10.1007/978-3-642-37192-9_19 (cited on page 143).

Ehrhardt-Martinez, Karen; Donnelly, Kat A. & Laitner, John A. (2010). *Advanced Metering Initiatives and Residential Feedback Programs: A Meta-Review for Household Electricity-Saving Opportunities*. Technical report E105. Washington, D.C.: American Council for an Energy-Efficient Economy (ACEEE) (cited on page 34).

Elburg, Henk van (2015). 'USmartConsumer: real-time smart meter feedback to kick-start consumer interest'. In: *8th International Conference on Energy Efficiency in Domestic Appliances and Lighting*. EEDAL. Lucerne, Switzerland, pages 1280–1292 (cited on page 41).

Electrolux (1999). 'Press Release: A refrigerator that 'thinks' – intelligent refrigerator will simplify homes'. URL: http://www.electroluxgroup.com/en/a-refrigerator-that-thinks-intelligent-refrigerator-will-simplify-homes-4349/ (cited on page 61).

Electrolux (2015). 'Press Release: Electrolux contributes smart appliances for next generation sustainable housing in Stockholm'. URL: http://www.electroluxgroup.com/en/electrolux-contributes-smart-appliances-for-next-generation-sustainable-housing-in-stockholm-20467/ (cited on page 60).

Electronics Maker (2016). 'IEEE Experts Recognize Deployment of Clean & Renewable Energy as a Dominant Technology Trend of 2016'. In: *Electronics Maker* (cited on page 52).

Englert, Frank; Schmitt, Till; Kößler, Sebastian; Reinhardt, Andreas & Steinmetz, Ralf (2013). 'How to Auto-configure Your Smart Home? High-resolution Power Measurements to the Rescue'. In: *4th International Conference on Future Energy Systems*. e-Energy.

ACM. Berkeley, California, USA, pages 215–224. DOI: 10.1145/2487166.2487191 (cited on page 75).

E.ON (2014). 'UK Press Release: E.ON expands testing of ground breaking smart home technologies in Thinking Energy trial'. URL: https://pressreleases.eon-uk.com/blogs/eonukpressreleases/archive/2014/04/02/2352.aspx (cited on page 60).

Farinaccio, Linda & Zmeureanu, Radu (1999). 'Using a pattern recognition approach to disaggregate the total electricity consumption in a house into the major end-uses'. In: *Energy and Buildings* 30.3, pages 245–259. DOI: 10.1016/S0378-7788(99)00007-9 (cited on page 143).

Figueiredo, Marisa B.; Almeida, Ana & Ribeiro, Bernardete (2011). 'An Experimental Study on Electrical Signature Identification of Non-Intrusive Load Monitoring (NILM) Systems'. In: *10th International Conference on Adaptive and Natural Computing Algorithms (ICANNGA)*. Edited by Andrej Dobnikar; Uroš Lotrič & Branko Šter. Volume 6594. Lecture Notes in Computer Science. Springer Berlin Heidelberg. Ljubljana, Slovenia, pages 31–40. DOI: 10.1007/978-3-642-20267-4_4 (cited on page 127).

Figueiredo, Marisa; de Almeida, Ana & Ribeiro, Bernardete (2012). 'Home electrical signal disaggregation for non-intrusive load monitoring (NILM) systems'. In: *Neurocomputing* 96, pages 66–73. DOI: 10.1016/j.neucom.2011.10.037 (cited on page 127).

Figueiredo, Marisa; de Almeida, Ana; Ribeiro, Bernardete & Martins, António (2010). 'Extracting Features from an Electrical Signal of a Non-Intrusive Load Monitoring System'. In: *11th International Conference on Intelligent Data Engineering and Automated Learning*. Edited by Colin Fyfe; Peter Tino; Darryl Charles; Cesar Garcia-Osorio & Hujun Yin. Volume 6283. IDEAL. Springer Berlin / Heidelberg, pages 210–217. DOI: 10.1007/978-3-642-15381-5_26 (cited on page 127).

Fischer, Joel E.; Ramchurn, Sarvapali D.; Osborne, Michael A.; Parson, Oliver; Huynh, Trung Dong; Alam, Muddasser; Pantidi, Nadia; Moran, Stuart; Bachour, Khaled; Reece, Steven; Costanza, Enrico; Rodden, Tom & Jennings, Nicholas R. (2013). 'Recommending Energy Tariffs and Load Shifting Based on Smart Household Usage Profiling'. In: *International conference on Intelligent User Interfaces*. IUI. DOI: 10.1145/2449396.2449446 (cited on page 19).

Fitzpatrick, Geraldine & Smith, Greg (2009). 'Technology-Enabled Feedback on Domestic Energy Consumption: Articulating a Set of Design Concerns'. In: *IEEE Pervasive Computing* 8.1, pages 37–44. DOI: 10.1109/MPRV.2009.17 (cited on page 16).

Fox-Brewster, Thomas (2016). 'You're Told 'Smart' Tech Will Make Your Home Safer – It's A Lie'. In: *Forbes* (cited on page 61).

Franco, A.; Malhotra, N. & Simonovits, G. (2014). 'Publication bias in the social sciences: Unlocking the file drawer'. In: *Science* 345.6203, pages 1502–1505. DOI: 10.1126/science.1255484 (cited on page 36).

Froehlich, Jon; Larson, Eric; Gupta, Sidhant; Cohn, Gabe; Reynolds, Matthew S. & Patel, Shwetak N. (2011). 'Disaggregated End-Use Energy Sensing for the Smart Grid'. In: *IEEE Pervasive Computing* 10.1, pages 28–39. DOI: 10.1109/MPRV.2010.74 (cited on page 132).

Fukushima, Kunihiko (1980). 'Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position'. In: *Biological cybernetics* 36.4, pages 193–202 (cited on page 157).

Gamberini, Luciano; Corradi, Nicola; Zamboni, Luca; Perotti, Michela; Cadenazzi, Camilla; Mandressi, Stefano; Jacucci, Giulio; Tusa, Giovanni; Spagnolli, Anna; Björkskog, Christoffer; Salo, Marja & Aman, Pirkka (2011). 'Saving is Fun: Designing a Persuasive Game for Power Conservation'. In: *8th International Conference on Advances in Computer Enter-*

tainment Technology. ACE 16. ACM. Lisbon, Portugal, 16:1–16:7. DOI: 10.1145/2071423.2071443 (cited on page 38).

Gamberini, Luciano; Spagnolli, Anna; Corradi, Nicola; Jacucci, Giulio; Tusa, Giovanni; Mikkola, Topi; Zamboni, Luca & Hoggan, Eve (2012). 'Tailoring Feedback to Users' Actions in a Persuasive Game for Household Electricity Conservation'. In: *7th International Conference on Persuasive Technology: Design for Health and Safety*. Edited by Magnus Bang & Eva L. Ragnemalm. PERSUASIVE. Springer Berlin Heidelberg. Linköping, Sweden, pages 100–111. DOI: 10.1007/978-3-642-31037-9_9 (cited on page 38).

Ganu, Tanuja; Seetharam, Deva P.; Arya, Vijay; Kunnath, Rajesh; Hazra, Jagabondhu; Husain, Saiful A; De Silva, Liyanage Chandratilake & Kalyanaraman, Shivkumar (2012). 'nPlug: A Smart Plug for Alleviating Peak Loads'. In: *3rd International Conference on Future Energy Systems*. e-Energy. ACM. Madrid, Spain. DOI: 10.1145/2208828.2208858 (cited on page 54).

Gao, Jingkun; Kara, Emre C.; Giri, Suman & Bergés, Mario (2015). 'A feasibility study of automated plug-load identification from high-frequency measurements'. In: *Global Conference on Signal and Information Processing*. GlobalSIP. IEEE. Orlando, Florida, USA (cited on page 108).

Garg, A. X.; Hackam, D. & Tonelli, M. (2008). 'Systematic Review and Meta-analysis: When One Study Is Just not Enough'. In: *Clinical Journal of the American Society of Nephrology* 3.1, pages 253–260. DOI: 10.2215/cjn.01430307 (cited on pages 33, 35).

Gasser, T.; Guivarch, C.; Tachiiri, K.; Jones, C. D. & Ciais, P. (2015). 'Negative emissions physically needed to keep global warming below 2 °C'. In: *Nature Communications* 6, page 7958. DOI: 10.1038/ncomms8958 (cited on pages 12, 195).

Gavin, Claire (2014). *Seasonal variations in electricity demand*. Technical report. DECC, UK Government (cited on pages 23, 195).

Gelman, Andrew & Hill, Jennifer (2007). 'Data Analysis Using Regression and Multilevel/Hierarchical Models' (cited on page 43).

General Electric (2010). 'GE Appliances Creates Home Energy Management Business'. URL: http://www.businesswire.com/news/home/20101130006276/en/GE-Appliances-Creates-Home-Energy-Management-Business (cited on page 50).

General Electric (2012). 'GE Nucleus iPhone app'. URL: https://itunes.apple.com/us/app/ge-nucleus/id445477211?mt=8 (cited on pages 50, 52, 196).

General Electric (2015a). 'GE Brillion™ Android app'. URL: https://play.google.com/store/apps/details?id=com.geappliances.brillion.wifi (cited on page 50).

General Electric (2015b). 'Press release: PUTTING THE APP IN APPLIANCES: GE ANNOUNCES A FULL SUITE OF CONNECTED PROFILE™ APPLIANCES IN 2015'. URL: http://pressroom.geappliances.com/news/putting-the-app-in-appliances:-ge-announces-a-full-suite-of-connected-profileTM-appliances-in-2015 (cited on page 50).

Ghahramani, Zoubin & Jordan, Michael I. (1997). 'Factorial Hidden Markov Models'. In: *Machine Learning* 29.2, pages 245–273. DOI: 10.1023/A:1007425814087 (cited on page 137).

Graves, Alex (2012). 'Supervised sequence labelling with recurrent neural networks'. Volume 385. Springer (cited on page 166).

Graves, Alex (2013). 'Generating sequences with recurrent neural networks'. In: arXiv: 1308.0850 (cited on page 166).

Graves, Alex & Jaitly, Navdeep (2014). 'Towards End-To-End Speech Recognition with Recurrent Neural Networks'. In: *31st International Conference on Machine Learning*. Edited by Tony Jebara & Eric P. Xing. ICML, pages 1764–1772 (cited on pages 154, 165).

Gupta, Abhay & Chakravarty, Prateek (2014). *White Paper: Impact of Energy Disaggregation on Consumer Behavior*. Technical report. Bidgely (cited on page 38).

Gutberlet, Kurt-Ludwig (2008). 'Energy efficiency in the household'. In: *GfK Conference 2008 - Climate-friendly consumption*. Nuremberg (cited on pages 61, 62).

Hadden, S. (1999). *Commercial Nonintrusive Load Monitoring System Beta Test Results. Final Report*. Technical report TR-114236. Palo Alto, California: Electric Power Research Institute (EPRI), page 90 (cited on page 133).

Hargreaves, Tom; Nye, Michael & Burgess, Jacquelin (2010). 'Making energy visible: A qualitative field study of how householders interact with feedback from smart energy monitors'. In: *Energy Policy* 38.10. The socio-economic transition towards a hydrogen economy - findings from European research, with regular papers, pages 6111–6119. DOI: 10.1016/j.enpol.2010.05.068 (cited on page 18).

Hargreaves, Tom; Nye, Michael & Burgess, Jacquelin (2013). 'Keeping energy visible? Exploring how householders interact with feedback from smart energy monitors in the longer term'. In: *Energy Policy* 52. Special Section: Transition Pathways to a Low Carbon Economy, pages 126–134. DOI: 10.1016/j.enpol.2012.03.027 (cited on page 18).

Harris, A. R.; Rogers, Michelle Marinich; Miller, Carol J.; McElmurry, Shawn P. & Wang, Caisheng (2015). 'Residential emissions reductions through variable timing of electricity consumption'. In: *Applied Energy* 158, pages 484–489. DOI: 10.1016/j.apenergy.2015.08.042 (cited on page 23).

Hart, George William (1984). *Nonintrusive Appliance Load Data Acquisition Method*. Technical report. MIT Energy Laboratory (cited on pages 92, 128, 132).

Hart, George William (1985). *Prototype Nonintrusive Appliance Load Monitor. Progress Report #2*. Technical report. MIT Energy Laboratory and Electric Power Research Institute (cited on pages 1, 102, 127, 131, 132).

Hart, George William (1992). 'Nonintrusive Appliance Load Monitoring'. In: *Proceedings of the IEEE* 80.12, pages 1870–1891. DOI: 10.1109/5.192069 (cited on pages 1, 2, 62, 124, 127–132, 134, 138, 185, 196).

Hart, George William (1994). 'Automatic Construction of Finite State Load Behavior Models'. In: *4th International Synopsium on Distribution Automation and Demand-Side Management*. Orlando, Florida (cited on page 131).

Hart, George William (1995). 'Nonintrusive Appliance Load Monitoring with Finite-state Appliance Models'. Electric Power Research Institute. Google Books: OeJfMQAACAAJ (cited on pages 92, 128, 131).

Hart, George William & Bouloutas, A. T (1993). 'Correcting Dependent Errors in Sequences Generated by Finite-State Processes'. In: *IEEE Transactions on Information Theory* 39.4, pages 1249–1260. DOI: 10.1109/18.243442 (cited on page 131).

Hart, George William; Kern, Edward C. & Schweppe, Fred C. (1989). 'Non-intrusive appliance monitor'. US4858141 A (cited on pages 128, 130).

Hartog, Frank den; Daniele, Laura & Roes, Jasper (2014). *Study on Semantic Assets for Smart Appliances Interoperability. First interim report*. Technical report D-S1. European Commission (cited on page 59).

Hauberg, Søren & Sloth, Jakob (2008). 'An Efficient Algorithm for Modelling Duration in Hidden Markov Models, with a Dramatic Application'. In: *Journal of Mathematical Imaging and Vision* 31.2-3, pages 165–170. DOI: 10.1007/s10851-007-0059-9 (cited on page 142).

He, Kaiming; Zhang, Xiangyu; Ren, Shaoqing & Sun, Jian (2015). 'Deep Residual Learning for Image Recognition'. In: *arXiv*. arXiv: 1512.03385 (cited on page 179).

HEA (2012). *Home Energy Disaggregation Utilizing Smart Meter Data*. Technical report. Home Energy Analytics (HEA) (cited on pages 36, 38, 39).

HEA (2013). *Mountain View Reduces GHG Emissions - Energy Upgrade Mountain View (EUMV) Phase 1 Summary*. Technical report. HEA & Energy Upgrade California in Mountain View (cited on pages 36, 38, 39).

HEA (2015). *Energy Upgrade Mountain View - EUMV Final Report*. Technical report. City of Mountain View & Acterra & HEA (cited on pages 35–39).

Hermsen, Sander; Frost, Jeana; Renes, Reint Jan & Kerkhof, Peter (2016). 'Using feedback through digital technology to disrupt and change habitual behavior: A critical review of current literature'. In: *Computers in Human Behavior* 57, pages 61–74. DOI: 10.1016/j.chb.2015.12.023 (cited on page 43).

Hinton, Geoffrey E; Osindero, Simon & Teh, Yee-Whye (2006). 'A fast learning algorithm for deep belief nets'. In: *Neural Computation* 18.7, pages 1527–1554. DOI: 10.1162/neco.2006.18.7.1527 (cited on page 156).

Hinton, Geoffrey E.; Srivastava, Nitish; Krizhevsky, Alex; Sutskever, Ilya & Salakhutdinov, Ruslan R. (2012). 'Improving neural networks by preventing co-adaptation of feature detectors'. In: arXiv: 1207.0580v1 [cs.NE] (cited on page 164).

Hochedez, Jean-François; Benassi, Romain; Richard, Florence & Lefevbre-Naré, Frédéric (2015). *Creating value from energy data: the disaggregation challenge*. Technical report. HOMEpulse (cited on page 7).

Hochreiter, Sepp & Schmidhuber, Jürgen (1997). 'Long short-term memory'. In: *Neural Computation* 9.8, pages 1735–1780. DOI: 10.1162/neco.1997.9.8.1735 (cited on page 165).

Holcomb, Chris (2012). 'Pecan Street Inc.: A Test-bed for NILM'. In: *1st International Workshop on Non-Intrusive Load Monitoring* (cited on page 108).

Holmegaard, Emil & Kjærgaard, Mikkel Baun (2015). 'Towards NILM for Industrial Settings'. In: *6th International Conference on Future Energy Systems*. e-Energy. Association for Computing Machinery (ACM). DOI: 10.1145/2768510.2770943 (cited on page 19).

Hu, Jianjun & Karava, Panagiota (2014). 'A state-space modeling approach and multi-level optimization algorithm for predictive control of multi-zone buildings with mixed-mode cooling'. In: *Building and Environment* 80, pages 259–273. DOI: 10.1016/j.buildenv.2014.05.003 (cited on page 20).

Hunter, John D. (2007). 'Matplotlib: A 2D Graphics Environment'. In: *Computing in Science & Engineering* 9.3, pages 90–95. DOI: 10.1109/mcse.2007.55 (cited on page 189).

Intelligent Energy Europe (2009). *Smart Domestic Appliances (Smart-A) Supporting the System Integration of Renewable Energy*. Technical report. European Commission (cited on page 24).

International Energy Agency (2015). *Energy Efficiency Market Report. Market Trends and Medium-Term Prospects*. Technical report. OECD (cited on page 14).

Ioffe, Sergey & Szegedy, Christian (2015). 'Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift'. In: arXiv: 1502.03167 (cited on page 163).

Jacucci, Giulio; Spagnolli, Anna; Gamberini, Luciano; Chalambalakis, Alessandro; Björkskog, Christoffer; Bertoncini, Massimo; Torstensson, Carin & Monti, Pasquale (2009). 'Designing Effective Feedback of Electricity Consumption for Mobile User Interfaces.' In: *PsychNology Journal* 7.3, pages 265–289 (cited on page 38).

Jaderberg, Max; Simonyan, Karen; Zisserman, Andrew & Kavukcuoglu, Koray (2015). 'Spatial Transformer Networks'. In: arXiv: 1506.02025 (cited on page 179).

Jiang, Lei; Li, Jiaming; Luo, Suhuai; Jin, J. & West, S. (2011). 'Literature Review of Power Disaggregation'. In: *International Conference on Modelling, Identification and Control.*

ICMIC. IEEE. Shanghai, pages 38–42. DOI: 10.1109/ICMIC.2011.5973672 (cited on page 1).

Jiang, Lei; Luo, Su Huai & Li, Jia Ming (2013). 'Intelligent Electrical Appliance Event Recognition Using Multi-Load Decomposition'. In: *Advanced Materials Research* 805-806. Edited by Xinwei Yu; Hongbing Ji; Shengzhou Chen; Xiaoguo Liu & Qingzhu Zeng, pages 1039–1045. DOI: 10.4028/www.scientific.net/AMR.805-806.1039 (cited on page 143).

Jiang, Lei; Luo, Suhuai & Li, Jiaming (2014). 'Intelligent electrical event recognition on general household power appliances'. In: *15th Workshop on Control and Modeling for Power Electronics*. COMPEL. IEEE. Santander, pages 1–3. DOI: 10.1109/COMPEL.2014.6877183 (cited on page 143).

Johnson, Michael T. (2005). 'Capacity and Complexity of HMM Duration Modeling Techniques'. In: *IEEE Signal Processing Letters* 12.5, pages 407–410. DOI: 10.1109/LSP.2005.845598 (cited on page 142).

Jones, Eric; Oliphant, Travis; Peterson, Pearu et al. (2001–). 'SciPy: Open source scientific tools for Python'. [Online; accessed 2016-08-19] (cited on page 189).

Jonschkowski, Rico & Brock, Oliver (2015). 'Learning state representations with robotic priors'. In: *Autonomous Robots* 39.3, pages 407–428. DOI: 10.1007/s10514-015-9459-7 (cited on page 179).

Kahl, Matthias; Haq, Anwar Ul; Kriechbaumer, Thomas & Jacobsen, Hans-Arno (2016). 'WHITED - A Worldwide Household and Industry Transient Energy Data Set'. In: *3rd International NILM Workshop*. Vancouver, Canada (cited on pages 75, 108).

Kalchbrenner, Nal; Danihelka, Ivo & Graves, Alex (2016). 'Grid Long Short-Term Memory'. In: *International Conference on Learning Representations*. ICLR. San Juan, Puerto Rico. arXiv: 1507.01526. Forthcoming (cited on page 179).

Kalogridis, Georgios & Dave, Saraansh (2014). 'Privacy and eHealth-enabled smart meter Informatics'. In: *1st International Workshop on Secure and Privacy-Aware Information Management in eHealth at the IEEE 16th International Conference on e-Health Networking, Applications and Services*. HEALTHCOM. IEEE. Natal, pages 116–121. DOI: 10.1109/HealthCom.2014.7001824 (cited on page 25).

Kanellos, Michael (2011). 'Google Kills PowerMeter'. In: *Greentech Media* (cited on page 50).

Kang, Se-il & Park, Byung-seok (2014). *Smart Appliances Interface with Demand Response*. Technical report. Korea Smart Grid Association (KSGA) and Smart Grid Standardization Forum (cited on page 58).

Karjalainen, Sami (2011). 'Consumer preferences for feedback on household electricity consumption'. In: *Energy and Buildings* 43.2–3, pages 458–467. DOI: 10.1016/j.enbuild.2010.10.010 (cited on page 16).

Kavgic, M.; Mavrogianni, A.; Mumovic, D.; Summerfield, A.; Stevanovic, Z. & Djurovic-Petrovic, M. (2010). 'A review of bottom-up building stock models for energy consumption in the residential sector'. In: *Building and Environment* 45.7, pages 1683–1697. DOI: 10.1016/j.buildenv.2010.01.021 (cited on page 22).

Kavousian, Amir; Rajagopal, Ram & Fischer, Martin (2013). 'Determinants of residential electricity consumption: Using smart meter data to examine the effect of climate, building characteristics, appliance stock, and occupants' behavior'. In: *Energy* 55, pages 184–194. DOI: 10.1016/j.energy.2013.03.086 (cited on page 89).

Kazmi, Aqeel H.; O'grady, Michael J.; Delaney, Declan T.; Ruzzelli, Antonio G. & O'hare, Gregory M. P. (2014). 'A Review of Wireless-Sensor-Network-Enabled Building Energy

Management Systems'. In: *ACM Transactions on Sensor Networks (TOSN)* 10.4, 66:1–66:43. DOI: 10.1145/2532644 (cited on page 53).

Kelly, Jack (2011). 'Disaggregating Smart Meter Readings Using Device Signatures'. MSc Thesis. Department of Computing, Imperial College London (cited on page 54).

Kelly, Jack (2016). 'Jack-Kelly.com blog post: Please help design a competition for energy disaggregation algorithms!' URL: http://jack-kelly.com/please_help_design_a_competition_for_energy_disaggregation (cited on page 182).

Kelly, Jack; Batra, Nipun; Parson, Oliver; Dutta, Haimonti; Knottenbelt, William; Rogers, Alex; Singh, Amarjeet & Srivastava, Mani (2014). 'NILMTK v0.2: A Non-intrusive Load Monitoring Toolkit for Large Scale Data Sets'. In: *1st International Conference on Embedded Systems For Energy-Efficient Buildings*. BuildSys. ACM. Memphis, TN, USA. DOI: 10.1145/2674061.2675024. arXiv: 1409.5908 (cited on pages 29, 92, 95, 196).

Kelly, Jack & Knottenbelt, William (2012a). 'Disaggregating Multi-State Appliances From Smart Meter Data'. In: *Imperial College Energy and Performance Colloquium*. London, UK (cited on page 29).

Kelly, Jack & Knottenbelt, William (2012b). 'Disaggregating Multi-State Appliances From Smart Meter Data'. In: *SIGMETRICS*. ACM. London, UK (cited on page 29).

Kelly, Jack & Knottenbelt, William (2014). 'Metadata for Energy Disaggregation'. In: *Computer Software and Applications Conference Workshop at the 2nd International Workshop on Consumer Devices and Systems*. CDS. IEEE. Västerås, Sweden. DOI: 10.1109/COMPSACW.2014.97. arXiv: 1403.5946 (cited on pages 29, 107, 196).

Kelly, Jack & Knottenbelt, William (2015a). 'Neural NILM: Deep Neural Networks Applied to Energy Disaggregation'. In: *2nd Workshop On Embedded Systems For Energy-Efficient Buildings*. BuildSys. ACM. Seoul, South Korea, pages 55–64. DOI: 10.1145/2821650.2821672. arXiv: 1507.06594 (cited on pages 29, 152, 196).

Kelly, Jack & Knottenbelt, William (2015b). 'The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes'. In: *Scientific Data* 2.150007. DOI: 10.1038/sdata.2015.7 (cited on pages 29, 66, 90, 108, 195, 223).

Kelly, Jack & Knottenbelt, William (2016). 'Does disaggregated electricity feedback reduce domestic electricity consumption? A systematic review of the literature'. In: *3rd International NILM Workshop*. Vancouver, Canada. arXiv: 1605.00962 (cited on pages 30, 32).

Kempton, Willett & Montgomery, Laura (1982). 'Folk Quantification of Energy'. In: *Energy* 7.10, pages 817–827. DOI: 10.1016/0360-5442(82)90030-5 (cited on page 17).

Khan, Umair A.; Leeb, Steven B. & Lee, Mark C. (1997). 'A Multiprocessor for Transient Event Detection'. In: *IEEE Transactions on Power Delivery* 12.1, pages 51–60. DOI: 10.1109/61.568225 (cited on page 135).

Kim, Chul Hwan & Aggarwal, Raj (2000). 'Wavelet transforms in power systems. Part 1: General introduction to the wavelet transforms'. In: *Power Engineering Journal* 14.2, pages 81–87. DOI: 10.1049/pe:20000210 (cited on page 135).

Kim, Hyungsul; Marwah, Manish; Arlitt, Martin; Lyon, Geoff & Han, Jiawei (2011). 'Unsupervised Disaggregation of Low Frequency Power Measurements'. In: *11th International Conference on Data Mining*. SDM. SIAM. Mesa, Arizona, USA. Chapter 64, pages 747–758. DOI: 10.1137/1.9781611972818.64 (cited on pages 137, 140–142).

Kleiminger, Wilhelm; Beckel, Christian & Santini, Silvia (2015). 'Household Occupancy Monitoring Using Electricity Meters'. In: *International Joint Conference on Pervasive and Ubiquitous Computing*. UbiComp. ACM. Osaka, Japan, pages 975–986. DOI: 10.1145/2750858.2807538 (cited on page 143).

Kolter, J. Z.; Batra, S. & Ng, A. Y. (2010). 'Energy Disaggregation via Discriminative Sparse Coding'. In: *24th Annual Conference on Neural Information Processing Systems*. Edited by J.D. Lafferty; C.K.I. Williams; J. Shawe-Taylor; R.S. Zemel & A. Culotta. NIPS. Curran Associates, Inc.. Vancouver, BC, Canada, pages 1153–1161 (cited on page 136).

Kolter, J. Zico & Jaakkola, Tommi (2012). 'Approximate Inference in Additive Factorial HMMs with Application to Energy Disaggregation'. In: *International Conference on Artificial Intelligence and Statistics*. AISTATS. La Palma, Canary Islands, pages 1472–1482 (cited on page 141).

Kolter, J. Zico & Johnson, Matthew J. (2011). 'REDD: A Public Data Set for Energy Disaggregation Research'. In: *SustKDD workshop on Data Mining Applications in Sustainability*. Volume 25. ACM. San Diego, California, USA, pages 59–62 (cited on pages 67, 78, 108, 140, 172, 223).

Krishnamurti, Tamar; Davis, Alexander L.; Wong-Parodi, Gabrielle; Wang, Jack & Canfield, Casey (2013). 'Creating an in-home display: Experimental evidence and guidelines for design'. In: *Applied Energy* 108, pages 448–458. DOI: 10.1016/j.apenergy.2013.03.048 (cited on pages 16, 40).

Krizhevsky, Alex; Sutskever, Ilya & Hinton, Geoffrey E. (2012). 'ImageNet Classification with Deep Convolutional Neural Networks'. In: *Advances in Neural Information Processing Systems 25*. Edited by F. Pereira; C.J.C. Burges; L. Bottou & K.Q. Weinberger. NIPS. Curran Associates, Inc., pages 1097–1105 (cited on page 153).

Krueger, David & Memisevic, Roland (2015). 'Regularizing RNNs by Stabilizing Activations'. In: arXiv: 1511.08400 (cited on page 179).

Lai, Po-hsiang; Trayer, Mark; Ramakrishna, Sudhir & Li, Ying (2012). 'Database Establishment for Machine Learning in NILM'. In: *1st International Non-Intrusive Load Monitoring Workshop* (cited on page 110).

LaMonica, Martin (2011). 'Microsoft kills Hohm energy app'. In: *CNET* (cited on page 50).

Laughman, Christopher; Lee, Kwangduk; Cox, Robert; Shaw, Steven; Leeb, Steven; Norford, Les & Armstrong, Peter (2003). 'Power Signature Analysis'. In: *IEEE Power and Energy Magazine* 1.2, pages 56–63. DOI: 10.1109/MPAE.2003.1192027 (cited on page 132).

LeCun, Yann; Bottou, Léon; Bengio, Yoshua & Haffner, Patrick (1998). 'Gradient-based learning applied to document recognition'. In: *Proceedings of the IEEE* 86.11, pages 2278–2324 (cited on page 157).

Leeb, Steven B.; Khan, Umair A. & Shaw, Steven R. (1998). 'Multiprocessing transient event detector for use in a nonintrusive electrical load monitoring system'. US5717325 A. Massachusetts Institute Of Technology (cited on page 135).

Leeb, Steven B. & Kirtley, James L. (1996). 'Transient event detector for use in nonintrusive load monitoring systems'. US5483153 A. Massachusetts Institute Of Technology (cited on page 135).

Leeb, Steven B; Shaw, Steven R & Kirtley Jr, James L (1995). 'Transient event detection in spectral envelope estimates for nonintrusive load monitoring'. In: *IEEE Transactions on Power Delivery* 10.3, pages 1200–1210. DOI: 10.1109/61.400897 (cited on pages 129, 135).

LG Electronics (2000). 'LG Unveils Internet-ready refrigerator'. URL: http://www.telecompaper.com/news/lg-unveils-internetready-refrigerator--221266 (cited on page 61).

Lin, Yu-Hsiu & Tsai, Men-Shen (2010). 'A novel feature extraction method for the development of nonintrusive load monitoring system based on BP-ANN'. In: *International Symposium on Computer Communication Control and Automation*. Volume 2. 3CA. IEEE, pages 215–218. DOI: 10.1109/3CA.2010.5533571 (cited on page 154).

Lin, Yu-Hsiu & Tsai, Men-Shen (2011). 'Applications of Hierarchical Support Vector Machines for Identifying Load Operation in Nonintrusive Load Monitoring Systems'. In: *9th World Congress on Intelligent Control and Automation*. WCICA. IEEE, pages 688–693. DOI: 10.1109/WCICA.2011.5970603 (cited on pages 135, 143).

Liu, Yan & Chen, Mei (2014). 'A review of nonintrusive load monitoring and its application in commercial building'. In: *4th Annual International Conference on Cyber Technology in Automation, Control, and Intelligent Systems*. CYBER. IEEE. Hong Kong, pages 623–629. DOI: 10.1109/CYBER.2014.6917536 (cited on page 20).

Louis, Jean-Nicolas; Caló, Antonio; Leiviskä, Kauko & Pongrácz, Eva (2016). 'Modelling home electricity management for sustainability: The impact of response levels, technological deployment & occupancy'. In: *Energy and Buildings* 119, pages 218–232. DOI: 10.1016/j.enbuild.2016.03.012 (cited on page 53).

Lowe, David G (1999). 'Object Recognition from Local Scale-Invariant Features'. In: *7th International Conference on Computer Vision*. Volume 2. IEEE, pages 1150–1157. DOI: 10.1109/ICCV.1999.790410 (cited on page 153).

Mahendra, Singh; Stéphane, Ploix & Frederic, Wurtz (2015). 'Modeling for Reactive Building Energy Management'. In: *Energy Procedia* 83. Sustainability in Energy and Buildings: Proceedings of the 7th International Conference SEB-15, pages 207–215. DOI: 10.1016/j.egypro.2015.12.175 (cited on page 20).

Makonin, Stephen (2012). *Approaches to Non-Intrusive Load Monitoring (NILM) in the Home*. Technical report. Simon Fraser University (cited on page 132).

Makonin, Stephen; Ellert, Bradley; Bajić, Ivan V. & Popowich, Fred (2016). 'Electricity, water, and natural gas consumption of a residential house in Canada from 2012 to 2014'. In: *Scientific Data* 3, page 160037. DOI: 10.1038/sdata.2016.37 (cited on page 108).

Makonin, Stephen; Popowich, Fred; Bartram, Lyn; Gill, Bob & Bajić, Ivan V. (2013). 'AMPds: A Public Dataset for Load Disaggregation and Eco-Feedback Research'. In: *Electrical Power and Energy Conference*. EPEC. IEEE. Halifax, NS. DOI: 10.1109/EPEC.2013.6802949 (cited on pages 108, 223).

Makriyiannis, Menelaos; Lung, Tudor; Craven, Robert; Toni, Francesca & Kelly, Jack (2014). 'Smarter Electricity through Argumentation'. In: *4th International Workshop on Combinations of Intelligent Methods and Applications in conjunction with the IEEE International Conference on Tools with AI*. IEEE (cited on pages 19, 29).

Makriyiannis, Menelaos; Lung, Tudor; Craven, Robert; Toni, Francesca & Kelly, Jack (2016). 'Smarter Electricity and Argumentation Theory'. In: *Combinations of Intelligent Methods and Applications: Proceedings of the 4th International Workshop, CIMA 2014, Limassol, Cyprus, November 2014 (at ICTAI 2014)*. Edited by Ioannis Hatzilygeroudis; Vasile Palade & Jim Prentzas. Volume 46. Springer, pages 79–95. DOI: 10.1007/978-3-319-26860-6_5 (cited on pages 19, 30).

Mansouri, Iman & Newborough, Marcus (1999). 'Dynamics of Energy Use in UK Households: End-use monitoring of Electric Cookers'. In: *Summer Study on Energy Efficiency in Buildings*. American Council for an Energy-Efficient Economy (ACEEE) (cited on page 38).

Mansouri, Iman; Newborough, Marcus & Probert, Douglas (1996). 'Energy consumption in UK households: Impact of domestic electrical appliances'. In: *Applied Energy* 54.3. Domestic Demand-Side Management, pages 211–285. DOI: 10.1016/0306-2619(96)00001-3 (cited on pages 16, 17, 195).

Martin, Rodney A & Poll, Scott (2014). 'Energy Analysis of Multi-Function Devices in an Office Environment.' In: *ASHRAE Transactions* 120.1 (cited on page 19).

Mauch, Lukas & Yang, Bin (2015). 'A new approach for supervised power disaggregation by using a deep recurrent LSTM network'. In: *Global Conference on Signal and Information Processing*. GlobalSIP. IEEE. Orlando, FL, USA (cited on page 176).

McCalley, L. T. & Midden, Cees J.H (2002). 'Energy conservation through product-integrated feedback: The roles of goal-setting and social orientation'. In: *Journal of Economic Psychology* 23.5, pages 589–603. DOI: 10.1016/s0167-4870(02)00119-8 (cited on pages 38, 40, 43).

McIlroy, Doug; Pinson, E. N. & Tague, B. A. (1978). 'Unix Time-Sharing System Forward'. In: *The Bell System Technical Journal*, pages 1902–1903 (cited on page 106).

McKinney, Wes (2010). 'Data Structures for Statistical Computing in Python'. In: *9th Python in Science Conference*. Edited by Stéfan van der Walt & Jarrod Millman, pages 51–56 (cited on page 189).

McMorran, Alan W. (2007). *An Introduction to IEC 61970-301 & 61968-11: The Common Information Model*. Technical report. University of Strathclyde. CiteSeerX DOI: 10.1.1.112.4648 (cited on page 58).

Millman, K. Jarrod & Aivazis, Michael (2011). 'Python for Scientists and Engineers'. In: *Computing in Science & Engineering* 13.2, pages 9–12. DOI: 10.1109/mcse.2011.36 (cited on page 189).

Mnih, Volodymyr; Kavukcuoglu, Koray; Silver, David; Rusu, Andrei A.; Veness, Joel; Bellemare, Marc G.; Graves, Alex; Riedmiller, Martin; Fidjeland, Andreas K.; Ostrovski, Georg; Petersen, Stig; Beattie, Charles; Sadik, Amir; Antonoglou, Ioannis; King, Helen; Kumaran, Dharshan; Wierstra, Daan; Legg, Shane & Hassabis, Demis (2015). 'Human-level control through deep reinforcement learning'. In: *Nature* 518.7540, pages 529–533. DOI: 10.1038/nature14236 (cited on pages 154, 158).

Murata, Hiroshi & Onoda, Takashi (2001). 'Applying Kernel Based Subspace Classification to a Non-intrusive Monitoring for Household Electric Appliances'. In: *International Conference on Artificial Neural Networks*. Edited by Georg Dorffner; Horst Bischof & Kurt Hornik. Volume 2130. ICANN. Springer Berlin Heidelberg. Berlin, Heidelberg, pages 692–698. DOI: 10.1007/3-540-44668-0_96 (cited on page 143).

Murray, David (2015). 'REFIT: Electrical Load Measurements'. DOI: 10.15129/31da3ece-f902-4e95-a093-e0a9536983c4 (cited on page 67).

Murtagh, Niamh; Gatersleben, Birgitta & Uzzell, David (2014). '20:60:20 - Differences in Energy Behaviour and Conservation between and within Households with Electricity Monitors'. In: *PLoS ONE* 9.3. DOI: 10.1371/journal.pone.0092019 (cited on page 37).

Nambi, S. N. Akshay Uttama; Reyes Lua, Antonio & Prasad, Venkatesha R. (2015). 'LocED: Location-aware Energy Disaggregation Framework'. In: *2nd International Conference on Embedded Systems for Energy-Efficient Built Environments*. BuildSys. ACM. Seoul, South Korea, pages 45–54. DOI: 10.1145/2821650.2821659 (cited on page 108).

Nascimento, Pedro Paulo Marques do (2016). 'Applications of deep learning techniques on NILM'. Master's thesis. COPPE/UFRJ - Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia, Rio de Janeiro (cited on page 177).

Natarajan, Sukumar; Padget, Julian & Elliott, Liam (2011). 'Modelling UK domestic energy and carbon emissions: an agent-based approach'. In: *Energy and Buildings* 43.10, pages 2602–2612. DOI: 10.1016/j.enbuild.2011.05.013 (cited on page 22).

Naus, Joeri; Vliet, Bas J.M. van & Hendriksen, Astrid (2015). 'Households as change agents in a Dutch smart energy transition: On power, privacy and participation'. In: *Energy Research & Social Science* 9. Special Issue on Smart Grids and the Social Sciences, pages 125–136. DOI: 10.1016/j.erss.2015.08.025 (cited on page 15).

Nordman, Bruce (2013). *Draft Report: Energy Reporting*. Technical report. Lawrence Berkeley National Laboratory (cited on page 110).

Nordman, Bruce (2014). *Energy Reporting Framework*. Internet-Draft. Internet Engineering Task Force (IETF) (cited on pages 110, 114, 117).

Nordman, Bruce & Cheung, Hoi Y. (2013). *Draft Report: A Basic Classification System for Energy-Using Products - Universal Device Classification*. Technical report. Lawrence Berkeley National Laboratory (cited on page 58).

Nordman, Bruce & Cheung, Iris (2016). *Data Model Needs for Building Device Interoperation*. Technical report. Lawrence Berkeley National Laboratory (cited on page 110).

Nordman, Bruce & Sanchez, Marla (2006). *Electronics Come of Age: A Taxonomy for Miscellaneous and Low Power Products*. Technical report. Lawrence Berkeley National Laboratory (cited on page 110).

Norford, Leslie K. & Leeb, Steven B. (1996). 'Non-intrusive electrical load monitoring in commercial buildings based on steady-state and transient load-detection algorithms'. In: *Energy and Buildings* 24.1, pages 51–64. DOI: 10.1016/0378-7788(95)00958-2 (cited on page 135).

Nouri, Daniel (2014). 'Using convolutional neural nets to detect facial keypoints tutorial'. URL: http://bit.ly/1OduG83 (cited on page 168).

Nunes, Nuno J; Pereira, Lucas; Quintal, Filipe & Bergés, Mario (2011). 'Deploying and evaluating the effectiveness of energy eco-feedback through a low-cost NILM solution'. In: *6th International Conference on Persuasive Technology* (cited on page 75).

Oliphant, Travis E. (2007). 'Python for Scientific Computing'. In: *Computing in Science & Engineering* 9.3, pages 10–20. DOI: 10.1109/mcse.2007.58 (cited on page 189).

Onoda, T.; Rätsch, Gunnar & Müller, Klaus-Robert (2000). 'A Non-Intrusive Monitoring System for Household Electric Appliances with Inverters'. In: *Symposium on Neural Computation*. Edited by H. Bothe & R. Rojas. ICSC Academic Press Canada/Switzerland. Berlin (cited on page 143).

Open Science Collaboration (2015). 'Estimating the reproducibility of psychological science'. In: *Science* 349.6251. DOI: 10.1126/science.aac4716 (cited on page 33).

OpenADR Alliance (2016). *OpenADR 2.0 Demand Response Program Guide*. Technical report 20140701 (cited on page 59).

Ozoh, Patrick & Apperley, Mark (2015). 'Simulating Electricity Consumption Pattern for Household Appliances Using Demand Side Strategies: A Review'. In: *15th New Zealand Conference on Human-Computer Interaction*. CHINZ. ACM. Hamilton, New Zealand, pages 65–71. DOI: 10.1145/2808047.2808057 (cited on page 24).

Paragon Consulting Services LLC (1998). *Low-Cost NIALMS Technology: Technical Assessment*. Technical report TR-108918-V2. Electric Power Research Institute (EPRI) (cited on page 132).

Parson, Oliver (2012–2016). 'NIALM in industry'. URL: http://blog.oliverparson.co.uk/2012/05/nialm-in-industry.html (cited on pages 7, 8, 195).

Parson, Oliver (2015). 'Overview of the NILM field'. URL: http://blog.oliverparson.co.uk/2015/03/overview-of-nilm-field.html (cited on pages 1, 2, 93, 195).

Parson, Oliver; Fisher, Grant; Hersey, April; Batra, Nipun; Kelly, Jack; Singh, Amarjeet; Knottenbelt, William & Rogers, Alex (2015). 'Dataport and NILMTK: A building data set designed for non-intrusive load monitoring'. In: *1st International Symposium on Signal Processing Applications in Smart Buildings at 3rd Global Conference on Signal & Information Processing*. GlobalSIP. IEEE. Orlando, Florida, USA (cited on page 30).

Parson, Oliver; Ghosh, Siddhartha; Weal, Mark & Rogers, Alex (2011). 'Using Hidden Markov Models for Iterative Non-intrusive Appliance Monitoring'. In: *Workshop on Machine Learning for Sustainability at the conference on Neural Information Processing Systems*. Sierra Nevada, Spain (cited on page 139).

Parson, Oliver; Ghosh, Siddhartha; Weal, Mark & Rogers, Alex (2012). 'Non-intrusive Load Monitoring using Prior Models of General Appliance Types'. In: *26th Conference on Artificial Intelligence*. AAAI. Toronto, ON, Canada, pages 356–362 (cited on pages 139, 141, 142).

Parton, Graham A; Donegan, Steven; Pascoe, Stephen; Stephens, Ag; Ventouras, Spiros & Lawrence, Bryan N (2015). 'MOLES3: Implementing an ISO standards driven data catalogue'. In: *International Journal of Digital Curation* 10.1. DOI: 10.2218/ijdc.v10i1.365 (cited on page 109).

Pecan Street (2011). 'Dataport'. URL: https://dataport.pecanstreet.org (cited on page 108).

Pedregosa, Fabian; Varoquaux, Gaël; Gramfort, Alexandre; Michel, Vincent; Thirion, Bertrand; Grisel, Olivier; Blondel, Mathieu; Prettenhofer, Peter; Weiss, Ron; Dubourg, Vincent; Vanderplas, Jake; Passos, Alexandre; Cournapeau, David; Brucher, Matthieu; Perrot, Matthieu & Duchesnay, Édouard (2011). 'Scikit-Learn: Machine Learning in Python'. In: *Journal of Machine Learning Research* 12, pages 2825–2830 (cited on page 190).

Pereira, Lucas (2011). 'Low cost non-intrusive home energy monitoring'. Masters thesis. Madeira Interactive Technologies Institute, University of Madeira, Portugal (cited on page 75).

Perez, Fernando & Granger, Brian E. (2007). 'IPython: A System for Interactive Scientific Computing'. In: *Computing in Science & Engineering* 9.3, pages 21–29. DOI: 10.1109/mcse.2007.53 (cited on page 189).

Populus (2015). *Smart energy outlook*. Technical report. Smart Energy GB (cited on page 15).

Prudenzi, A. (2002). 'A Neuron Nets Based Procedure for Identifying Domestic Appliances Pattern-of-Use from Energy Recordings at Meter Panel'. In: *Power Engineering Society Winter Meeting*. Volume 2. IEEE, pages 941–946. DOI: 10.1109/PESW.2002.985144 (cited on page 143).

Pullinger, Martin; Lovell, Heather & Webb, Janette (2014). 'Influencing household energy practices: a critical review of UK smart metering standards and commercial feedback devices'. In: *Technology Analysis & Strategic Management* 26.10, pages 1144–1162. DOI: 10.1080/09537325.2014.977245 (cited on page 44).

Quintal, Filipe; Nunes, Nuno J.; Ocneanu, Adrian & Bergés, Mario (2010). 'SINAIS: Home Consumption Package: A Low-cost Eco-feedback Energy-monitoring Research Platform'. In: *8th Conference on Designing Interactive Systems*. DIS. ACM. Aarhus, Denmark, pages 419–421. DOI: 10.1145/1858171.1858252 (cited on page 75).

Quittek, Juergen; Winter, Rolf; Dietz, Thomas; Claise, Benoit & Chandramouli, Mouli (2015). *RFC6988: Requirements for Energy Management*. Technical report 6988. Internet Engineering Task Force (IETF), page 28. DOI: 10.17487/rfc6988 (cited on page 59).

Raimi, Kaitlin T. & Carrico, Amanda R. (2016). 'Understanding and beliefs about smart energy technology'. In: *Energy Research & Social Science* 12, pages 68–74. DOI: 10.1016/j.erss.2015.12.018 (cited on page 62).

Rasmus, Antti; Berglund, Mathias; Honkala, Mikko; Valpola, Harri & Raiko, Tapani (2015). 'Semi-Supervised Learning with Ladder Networks'. In: *Advances in Neural Information*

Processing Systems 28. Edited by C. Cortes; N.D. Lawrence; D.D. Lee; M. Sugiyama & R. Garnett. Curran Associates, Inc., pages 3532–3540 (cited on page 178).

Reddy, T. Agami; Maor, Itzhak & Panjapornpon, Chanin (2007). 'Calibrating Detailed Building Energy Simulation Programs with Measured Data—Part I: General Methodology (RP-1051)'. In: *HVAC&R Research* 13.2, pages 221–241. DOI: 10.1080/10789669.2007.10390952 (cited on page 21).

Reeg, Christopher E. & Overbye, Thomas J. (2010). 'Algorithm development for Non-Intrusive Load Monitoring for Verification and Diagnostics'. In: *North American Power Symposium*. NAPS. IEEE, pages 1–5. DOI: 10.1109/NAPS.2010.5619600 (cited on page 135).

Reinhardt, Andreas; Bauman, Paul; Burgstahler, Daniel; Hollick, Matthias; Chonov, Hristo; Werner, Marc & Steinmetz, Ralf (2012). 'On the Accuracy of Appliance Identification Based on Distributed Load Metering Data'. In: *2nd IFIP Conference on Sustainable Internet and ICT for Sustainability*. SustainIT. Pisa, Italy, pages 1–9 (cited on pages 108, 111, 223).

Rettie, Ruth; Burchell, Kevin & Harries, Tim (2014). 'Energy Consumption Feedback: Engagement by Design'. In: *Design, User Experience, and Usability. User Experience Design for Everyday Life Applications and Services*. Edited by Aaron Marcus. Volume 8519. Lecture Notes in Computer Science. Springer International Publishing, pages 594–604. DOI: 10.1007/978-3-319-07635-5_57 (cited on pages 16, 44).

Richards, Patsy; Fell, Mike & White, Edward (2014). *Smart meters briefing*. Technical report. UK: House Of Commons Library, UK Government (cited on page 57).

Richardson, Jake (2015). 'Energy Management Startup Bidgely Raises $16.6 Million In B Round'. URL: http://cleantechnica.com/2015/11/13/energy-management-startup-bidgely-raises-16-6-million-b-round/ (cited on page 1).

Ridi, A.; Gisler, C. & Hennebert, J. (2014). 'A Survey on Intrusive Load Monitoring for Appliance Recognition'. In: *22nd International Conference on Pattern Recognition*. ICPR. Stockholm, pages 3702–3707. DOI: 10.1109/ICPR.2014.636 (cited on page 132).

Ridi, Antonio; Gisler, Christophe & Hennebert, Jean (2013). 'Automatic identification of electrical appliances using smart plugs'. In: *8th International Workshop on Systems, Signal Processing and their Applications*. WoSSPA. Institute of Electrical & Electronics Engineers (IEEE). Algiers. DOI: 10.1109/wosspa.2013.6602380 (cited on page 25).

Roos, JG; Lane, IE; Botha, EC & Hancke, Gerhard P (1994). 'Using neural networks for non-intrusive monitoring of industrial electrical loads'. In: *Instrumentation and Measurement Technology Conference*. IMTC. IEEE, pages 1115–1118. DOI: 10.1109/IMTC.1994.351862 (cited on page 154).

Rouvellou, I. & Hart, George William (1995). 'Inference of a Probabilistic Finite State Machine from its Output'. In: *IEEE Transactions on Systems, Man and Cybernetics* 25.3, pages 424–437. DOI: 10.1109/21.364856 (cited on pages 92, 131, 132).

Royapoor, Mohammad & Roskilly, Tony (2015). 'Building model calibration using energy and environmental data'. In: *Energy and Buildings* 94, pages 109–120. DOI: 10.1016/j.enbuild.2015.02.050 (cited on page 21).

Rumelhart, David E; Hinton, Geoffrey E & Williams, Ronald J (1985). *Learning internal representations by error propagation*. DTIC Document 8506. Institute for Cognitive Science, University of California, San Diego (cited on page 157).

Russakovsky, Olga; Deng, Jia; Su, Hao; Krause, Jonathan; Satheesh, Sanjeev; Ma, Sean; Huang, Zhiheng; Karpathy, Andrej; Khosla, Aditya; Bernstein, Michael; Berg, Alexander C. & Fei-Fei, Li (2015). 'ImageNet Large Scale Visual Recognition Challenge'. In: *Interna-*

tional Journal of Computer Vision (IJCV) 115.3, pages 211–252. DOI: 10.1007/s11263-015-0816-y (cited on page 181).

Ruzzelli, Antonio G; Nicolas, C; Schoofs, Anthony & O'Hare, Gregory MP (2010). 'Real-Time Recognition and Profiling of Appliances through a Single Electricity Sensor'. In: *7th Annual Communications Society Conference on Sensor Mesh and Ad Hoc Communications and Networks*. SECON. IEEE, pages 1–9. DOI: 10.1109/SECON.2010.5508244 (cited on pages 143, 154).

Sataøen, Hogne Lerøy; Brekke, Ole Andreas; Batel, Susana & Albrecht, Martin (2015). 'Towards a sustainable grid development regime? A comparison of British, Norwegian, and Swedish grid development'. In: *Energy Research & Social Science* 9. Special Issue on Smart Grids and the Social Sciences, pages 178–187. DOI: 10.1016/j.erss.2015.08.011 (cited on page 15).

Schmidt, Lisa (2012). 'Online smart meter analysis achieves sustained energy reductions: Results from five communities'. In: *Summer Study on Energy Efficiency in Buildings*. American Council for an Energy-Efficient Economy (ACEEE) (cited on pages 36, 39).

Schwartz, Daniel; Fischhoff, Baruch; Krishnamurti, Tamar & Sowell, Fallaw (2013). 'The Hawthorne effect and energy awareness'. In: *Proceedings of the National Academy of Sciences* 110.38. DOI: 10.1073/pnas.1301687110 (cited on page 35).

Schwartz, Tobias; Denef, Sebastian; Stevens, Gunnar; Ramirez, Leonardo & Wulf, Volker (2013). 'Cultivating Energy Literacy: Results from a Longitudinal Living Lab Study of a Home Energy Management System'. In: *SIGCHI Conference on Human Factors in Computing Systems*. CHI. ACM. Paris, France, pages 1193–1202. DOI: 10.1145/2470654.2466154 (cited on page 18).

Schwartz, Tobias; Stevens, Gunnar; Jakobi, Timo; Denef, Sebastian; Ramirez, Leonardo; Wulf, Volker & Randall, Dave (2015). 'What People Do with Consumption Feedback: A Long-Term Living Lab Study of a Home Energy Management System'. In: *Interacting with Computers* 27.6, pages 551–576. DOI: 10.1093/iwc/iwu009 (cited on pages 35, 38).

Seryak, John & Kissock, Kelly (2003). 'Occupancy and Behavioral Affects on Residential Energy Use'. In: *American Solar Energy Society Solar Conference*. Austin, Texas, pages 717–722 (cited on page 17).

Shankland, Stephen (2016). 'Your smart-home network will be a mess'. In: *CNET* (cited on page 60).

Sharma, Konark & Saini, Lalit Mohan (2015). 'Performance analysis of smart metering for smart grid: An overview'. In: *Renewable and Sustainable Energy Reviews* 49, pages 720–735. DOI: 10.1016/j.rser.2015.04.170 (cited on page 15).

Shaw, Steven R.; Abler, C. B.; Lepard, R. F.; Luo, D.; Leeb, Steven B. & Norford, Leslie K. (1998). 'Instrumentation for High Performance Nonintrusive Electrical Load Monitoring'. In: *Journal of Solar Energy Engineering* 120.3, pages 224–229. DOI: 10.1115/1.2888073 (cited on page 135).

Shaw, Steven R. & Leeb, Steven B. (1999). 'Identification of Induction Motor Parameters from Transient Stator Current Measurements'. In: *IEEE Transactions on Industrial Electronics* 46.1, pages 139–149. DOI: 10.1109/41.744405 (cited on page 135).

Simpson, Robby (2013). *Smart Energy Profile 2.0 Overview*. Technical report. GE Digital Energy (cited on pages 52, 58).

Skjølsvold, Tomas Moe & Ryghaug, Marianne (2015). 'Embedding smart energy technology in built environments: A comparative study of four smart grid demonstration projects'. In: *Indoor and Built Environment* 24.7, pages 878–890. DOI: 10.1177/1420326X15596210 (cited on page 62).

Snow, Stephen; Buys, Laurie; Roe, Paul & Brereton, Margot (2013). 'Curiosity to cupboard'. In: *25th Australian Computer-Human Interaction Conference on Augmentation, Application, Innovation, Collaboration*. OzCHI. Association for Computing Machinery (ACM). DOI: 10.1145/2541016.2541025 (cited on pages 16, 42).

Socolow, Robert H. (1977–1978). 'The twin rivers program on energy conservation in housing: Highlights and conclusions'. In: *Energy and Buildings* 1.3, pages 207–242. DOI: 10.1016/0378-7788(78)90003-8 (cited on pages 16, 17).

Sokoloski, Rebecca (2015). 'Disaggregated Electricity Consumption: Using Appliance-Specific Feedback to Promote Energy Conservation'. M.A. California State University San Marcos (cited on pages v, 37, 38, 40–44, 183).

Sønderby, Søren Kaae; Sønderby, Casper Kaae; Maaløe, Lars & Winther, Ole (2015). 'Recurrent Spatial Transformer Networks'. In: arXiv: 1509.05329 (cited on page 179).

Spagnolli, Anna; Corradi, Nicola; Gamberini, Luciano; Hoggan, Eve; Jacucci, Giulio; Katzeff, Cecilia; Broms, Loove & Jönsson, Li (2011). 'Eco-Feedback on the Go: Motivating Energy Awareness'. In: *Computer* 44.5, pages 38–45. DOI: 10.1109/MC.2011.125 (cited on page 38).

Spiegel, Stephan (2015). 'Optimization of In-House Energy Demand'. In: *Smart Information Systems*. Edited by Frank Hopfgartner. Advances in Computer Vision and Pattern Recognition. Springer International Publishing, pages 271–289. DOI: 10.1007/978-3-319-14178-7_10 (cited on page 26).

Spiegel, Stephan & Albayrak, Sahin (2014). 'Energy Disaggregation Meets Heating Control'. In: *29th Annual Symposium on Applied Computing*. SAC. ACM. Gyeongju, Republic of Korea, pages 559–566. DOI: 10.1145/2554850.2555088 (cited on page 26).

Spradlin, Justin; Vint, John & Fischer, Barry (2014). 'This neat data algorithm unlocks the power of smart grid technology - without using smart meters'. In: *Opower blog* (cited on page 5).

Srinivasan, D.; Ng, W.S. & Liew, A.C. (2006). 'Neural-Network-Based Signature Recognition for Harmonic Source Identification'. In: *IEEE Transactions on Power Delivery* 21.1, pages 398–405. DOI: 10.1109/TPWRD.2005.852370 (cited on page 143).

St. John, Jeff (2013). 'Whirlpool Launches the Wi-Fi Smart Appliance'. In: *Greentech Media* (cited on page 50).

St. John, Jeff (2015a). 'A State-by-State Snapshot of Utility Smart Solar Inverter Plans'. In: *Green Tech Media* (cited on page 59).

St. John, Jeff (2015b). 'Startup Smappee Goes Deep With Its Energy Disaggregation'. URL: http://www.greentechmedia.com/articles/read/startup-smappee-goes-deep-with-its-energy-disaggregation (cited on pages 6, 195).

Ståhlberg, Jonatan (2010). 'Disaggregated Electricity Feedback: An analysis of the conditions and needs for improved electricity feedback in houses'. Uppsala University (cited on pages 16, 42).

Stankovic, Lina; Wilson, Charlie; Liao, Jing; Stankovic, Vladimir; Hauxwell-Baldwin, Richard; Murray, David & Coleman, Mike (2015). 'Understanding domestic appliance use through their linkages to common activities'. In: *8th International Conference on Energy Efficiency in Domestic Appliances and Lighting*. EEDAL. Lucerne-Horw, Switzerland (cited on page 25).

Strbac, Goran (2008). 'Demand side management: Benefits and challenges'. In: *Energy Policy* 36.12, pages 4419–4426. DOI: 10.1016/j.enpol.2008.09.030 (cited on page 22).

Strengers, Yolande (2013). 'Smart Energy Technologies in Everyday Life: Smart Utopia?' DOI: 10.1057/9781137267054 (cited on pages 18, 61).

Su, Man; Ji, Jianting; Che, Yulin; Liu, Ting; Chen, Siyun & Xu, Zhanbo (2015). 'An Appliance Classification Method for Residential Appliance Scheduling'. In: *Adjunct 2015 International Joint Conference on Pervasive and Ubiquitous Computing and 2015 International Symposium on Wearable Computers*. UbiComp/ISWC. ACM. Osaka, Japan, pages 1567–1570. DOI: 10.1145/2800835.2801641 (cited on page 24).

Sutskever, Ilya; Vinyals, Oriol & Le, Quoc V. (2014). 'Sequence to Sequence Learning with Neural Networks'. In: *Advances in Neural Information Processing Systems 27*. Edited by Z. Ghahramani; M. Welling; C. Cortes; N.D. Lawrence & K.Q. Weinberger. NIPS. Curran Associates, Inc., pages 3104–3112. arXiv: 1409.3215 (cited on pages 154, 165).

Tang, Guoming; Chen, Jie; Chen, Cheng & Wu, Kui (2014). 'Smart Saver: A Consumer-Oriented Web Service for Energy Disaggregation'. In: *International Conference on Data Mining Workshop*. ICDMW. IEEE, pages 1235–1238. DOI: 10.1109/ICDMW.2014.19 (cited on page 182).

Tariq, Zaid Bin; Arshad, Naveed & Nabeel, Muhammad (2015). 'Enhanced LZMA and BZIP2 for Improved Energy Data Compression'. In: *International Conference on Smart Cities and Green ICT Systems*. SMARTGREENS. IEEE. Lisbon, Portugal (cited on page 62).

Telefónica (2014). *The Smart Meter Revolution - Towards a Smarter Future*. Technical report (cited on page 14).

Thompson, Ashlee Clark (2016). 'Electrolux appliances will work with Google's smart home platform'. In: *CNET* (cited on page 59).

Tong, Darren (2014). 'Smappee Review: Will it make you "Smart About Energy"?' In: *Alpha Efficiency website* (cited on page 5).

Torriti, Jacopo (2014). 'A review of time use models of residential electricity demand'. In: *Renewable and Sustainable Energy Reviews* 37, pages 265–272. DOI: 10.1016/j.rser.2014.05.034 (cited on page 21).

Tricoire, Aurélie (2015). 'Uncertainty, vision, and the vitality of the emerging smart grid'. In: *Energy Research & Social Science* 9. Special Issue on Smart Grids and the Social Sciences, pages 21–34. DOI: 10.1016/j.erss.2015.08.028 (cited on page 15).

Tsagarakis, George; Collin, Adam & Kiprakis, Aristides E. (2013). 'A Statistical Survey of the UK Residential Sector Electrical Loads'. In: *International Journal of Emerging Electric Power Systems* 14, pages 509–523. DOI: 10.1515/ijeeps-2013-0078. arXiv: 1306.0802 (cited on page 116).

Tweed, Katherine (2014). 'Bidgely Lands a Major Deal With TXU Energy to Offer Energy Analytics in Texas'. URL: http://www.greentechmedia.com/articles/read/bidgely-goes-big-time-with-txu-energy-win (cited on page 4).

Tweed, Katherine (2015). 'Bidgely Raises $16 Million for Energy Disaggregation'. URL: https://www.greentechmedia.com/articles/read/bidgely-raises-16m-for-energy-disaggregation (cited on page 4).

Tweed, Katherine (2016). 'Looking to Restructure, RWE Is Building a $140M Venture Fund for Cleantech'. URL: https://www.greentechmedia.com/articles/read/rwe-builds-venture-fund-for-clean-tech (cited on page 4).

Ueno, Tsuyoshi; Inada, Ryo; Saeki, Osamu & Tsuji, Kiichiro (2005). 'Effectiveness of displaying energy consumption data in residential houses – Analysis on how the residents respond'. In: *Summer Study. Energy savings: What works & who delivers?* European Council for an Energy-Efficient Economy (ECEEE). Mandelieu La Napoule, France, pages 1289–1299 (cited on page 38).

Ueno, Tsuyoshi; Inada, Ryo; Saeki, Osamu & Tsuji, Kiichiro (2006a). 'Effectiveness of an energy-consumption information system for residential buildings'. In: *Applied Energy* 83.8, pages 868–883. DOI: 10.1016/j.apenergy.2005.09.004 (cited on page 38).

Ueno, Tsuyoshi; Sano, Fuminori; Saeki, Osamu & Tsuji, Kiichiro (2006b). 'Effectiveness of an energy-consumption information system on energy savings in residential houses based on monitored data'. In: *Applied Energy* 83.2, pages 166–183. DOI: 10.1016/j.apenergy.2005.02.002 (cited on page 38).

Ueno, Tsuyoshi; Tsuji, Kiichiro & Nakano, Yukio (2006c). 'Effectiveness of Displaying Energy Consumption Data in Residential Buildings: To Know Is to Change'. In: *Summer Study on Energy Efficiency in Buildings*. American Council for an Energy-Efficient Economy (ACEEE). Pacific Grove, California (cited on page 38).

U.S. Energy Information Administration (EIA) (2015). *An Assessment of Interval Data and Their Potential Application to Residential Electricity End-Use Modeling*. Technical report. U.S. Department of Energy (DoE) (cited on pages 22, 53).

van der Walt, Stéfan; Colbert, S Chris & Varoquaux, Gaël (2011). 'The NumPy Array: A Structure for Efficient Numerical Computation'. In: *Computing in Science & Engineering* 13.2, pages 22–30. DOI: 10.1109/mcse.2011.37 (cited on page 189).

Vaseghi, S.V. (1995). 'State duration modelling in hidden Markov models'. In: *Signal Processing* 41.1, pages 31–41. DOI: 10.1016/0165-1684(94)00088-H (cited on page 142).

Vassileva, Iana; Odlare, Monica; Wallin, Fredrik & Dahlquist, Erik (2012). 'The impact of consumers' feedback preferences on domestic electricity consumption'. In: *Applied Energy* 93, pages 575–582. DOI: 10.1016/j.apenergy.2011.12.067 (cited on pages 16, 39).

Victor, Bret (2015). 'What can a technologist do about climate change? (A personal view)'. URL: http://worrydream.com/ClimateChange (cited on page viii).

Vincent, Pascal; Larochelle, Hugo; Bengio, Yoshua & Manzagol, Pierre-Antoine (2008). 'Extracting and Composing Robust Features with Denoising Autoencoders'. In: *25th International Conference on Machine Learning*. ICML. ACM. Helsinki, Finland, pages 1096–1103. DOI: 10.1145/1390156.1390294 (cited on page 167).

Wagner, Liam; Ross, Ian; Foster, John & Hankamer, Ben (2016). 'Trading Off Global Fuel Supply, CO2 Emissions and Sustainable Development'. In: *PLOS ONE* 11.3. Edited by Asim Zia, e0149406. DOI: 10.1371/journal.pone.0149406 (cited on page 12).

Wall, Robert (2012). *Mascot 9 v AC/AC Adaptor*. Technical report. OpenEnergyMonitor (cited on pages 70, 73).

Walton, Robert (2015). 'How the home energy management market is reinventing itself'. In: *Utility DIVE* (cited on page 43).

Werbos, Paul J (1988). 'Generalization of backpropagation with application to a recurrent gas market model'. In: *Neural Networks* 1.4, pages 339–356 (cited on page 157).

Werbos, Paul J (1990). 'Backpropagation through time: what it does and how to do it'. In: *Proceedings of the IEEE* 78.10, pages 1550–1560 (cited on page 165).

Wilhite, Harold; Høivik, Asbjørn & Olsen, Johan-Gjemre (1999). 'Advances in the use of consumption feedback information in energy billing: the experiences of a Norwegian energy utility'. In: *Summer Study on Energy Efficiency and CO2 reduction: the dimensions of the social challenge*. web version of paper. European Council for an Energy-Efficient Economy (ECEEE) (cited on page 16).

Wilhite, Harold & Ling, Rich (1995). 'Measured energy savings from a more informative energy bill'. In: *Energy and Buildings* 22.2, pages 145–155. DOI: 10.1016/0378-7788(94)00912-4 (cited on page 18).

Wilkenfeld, George & Harrington, Lloyd (2015). 'Too smart for our own good: Why intelligent appliances seem as far away as ever'. In: *8th International Conference on Energy Efficiency in Domestic Appliances and Lighting*. EEDAL. Lucerne, Switzerland, pages 937–945 (cited on page 50).

Wilkinson, W. A & Cox, M. D (1996). 'Discrete Wavelet Analysis of Power System Transients'. In: *IEEE Transactions on Power Systems* 11.4, pages 2038–2044. DOI: 10.1109/59.544682 (cited on page 135).

Williams, Ronald J & Zipser, David (1995). 'Gradient-based learning algorithms for recurrent networks and their computational complexity'. In: *Back-propagation: Theory, architectures and applications*, pages 433–486 (cited on page 157).

Winett, Richard A. & Neale, Michael S. (1979). 'Psychological Framework for Energy Conservation in Buildings: Strategies, Outcomes, Directions'. In: *Energy and Buildings* 2.2, pages 101–116. DOI: 10.1016/0378-7788(79)90026-4 (cited on page 17).

Witherden, M.; Rayudu, R.; Tyler, C. & Seah, W.K. (2013). 'Managing peak demand using direct load monitoring and control'. In: *Australasian Universities Power Engineering Conference*. AUPEC. Hobart, TX, pages 1–6. DOI: 10.1109/AUPEC.2013.6725445 (cited on page 24).

Wong, Yung Fei; Şekercioğlu, Y. Ahmet; Drummond, Tom & Wong, Voon Siong (2013). 'Recent Approaches to Non-intrusive Load Monitoring Techniques in Residential Settings'. In: *Symposium on Computational Intelligence Applications In Smart Grid*. CIASG. IEEE. Singapore, pages 73–79. DOI: 10.1109/CIASG.2013.6611501 (cited on page 132).

Wood, Georgina & Newborough, Marcus (2003). 'Dynamic energy-consumption indicators for domestic appliances: environment, behaviour and design'. In: *Energy and Buildings* 35.8, pages 821–841. DOI: 10.1016/S0378-7788(02)00241-4 (cited on pages 37, 38, 43).

Wood, Georgina & Newborough, Marcus (2007). 'Energy-use information transfer for intelligent homes: Enabling energy conservation with central and local displays'. In: *Energy and Buildings* 39.4, pages 495–503. DOI: 10.1016/j.enbuild.2006.06.009 (cited on pages 43, 44).

Xenias, Dimitrios; Axon, Colin J.; Whitmarsh, Lorraine; Connor, Peter M.; Balta-Ozkan, Nazmiye & Spence, Alexa (2015). 'UK smart grid development: An expert assessment of the benefits, pitfalls and functions'. In: *Renewable Energy* 81, pages 89–102. DOI: 10.1016/j.renene.2015.03.016 (cited on page 15).

Yan, Chengchu; Wang, Shengwei & Xiao, Fu (2012). 'A simplified energy performance assessment method for existing buildings based on energy bill disaggregation'. In: *Energy and Buildings* 55, pages 563–574. DOI: 10.1016/j.enbuild.2012.09.043 (cited on page 20).

Yang, Hong-Tzer; Chang, Hsueh-Hsien & Lin, Ching-Lung (2007). 'Design a Neural Network for Features Selection in Non-intrusive Monitoring of Industrial Electrical Loads'. In: *11th International Conference on Computer Supported Cooperative Work in Design*. CSCWD. IEEE, pages 1022–1027. DOI: 10.1109/CSCWD.2007.4281579 (cited on pages 135, 143, 154).

Yilmaz, Selin; Firth, Steven K. & Allinson, David (2015). 'Developing a modelling framework to quantify the demand response potential of domestic appliances in UK homes'. In: *8th International Conference on Energy Efficiency in Domestic Appliances and Lighting*. EEDAL. Lucerne, Switzerland, pages 1339–1351 (cited on page 24).

Ying, JI & Peng, XU (2013). 'Review of Indirect Sub-metering Method for Public Building Energy Consumption'. In: *Building Energy Efficiency* 10, page 021 (cited on page 132).

Zeifman, Michael (2012). 'Disaggregation of Home Energy Display Data Using Probabilistic Approach'. In: *IEEE Transactions on Consumer Electronics* 58.1, pages 23–31. DOI: 10.1109/TCE.2012.6170051 (cited on page 140).

Zeifman, Michael & Roth, Kurt (2011). 'Nonintrusive Appliance Load Monitoring: Review and Outlook'. In: *IEEE Transactions on Consumer Electronics* 57.1, pages 76–84. DOI: 10.1109/TCE.2011.5735484 (cited on pages 1, 132).

Zhang, Guanchen; Wang, Gary; Farhangi, Hassan & Palizban, Ali (2015). 'Residential Electric Load Disaggregation for Low-Frequency Utility Applications'. In: *Power Energy Society General Meeting*. PESGM. IEEE. Denver, CO. DOI: 10.1109/PESGM.2015.7286502 (cited on page 143).

Zhang, J.; Chu, H.; Hong, H.; Virnig, B. A. & Carlin, B. P. (2015). 'Bayesian hierarchical models for network meta-analysis incorporating nonignorable missingness'. In: *Statistical Methods in Medical Research*. DOI: 10.1177/0962280215596185 (cited on page 43).

ZigBee Alliance (2013). *Smart Energy Profile 2 (SEP 2) – IP based Energy Management for the Home*. Technical report (cited on page 57).

ZigBee Alliance & HomePlug Alliance (2013). *Smart Energy Profile 2 Application Protocol Standard*. Technical report (cited on page 58).

Zimmermann, Jean-Paul; Evans, Matt; Griggs, Jonathan; King, Nicola; Harding, Les; Roberts, Penelope & Evans, Chris (2012). *Household Electricity Survey. A Study of Domestic Electrical Product Usage*. Technical report R66141. Defra, Intertek Testing & Certification Ltd and AEA group, page 600 (cited on pages 25, 67, 83, 108).

Zoha, Ahmed; Gluhak, Alexander; Imran, Muhammad Ali & Rajasegarar, Sutharshan (2012). 'Non-Intrusive Load Monitoring Approaches for Disaggregated Energy Sensing: A Survey'. In: *Sensors* 12.12, pages 16838–16866. DOI: 10.3390/s121216838 (cited on pages 1, 132).

Index

AMPds, 54

Dataport, 54

iAWE, 54
IEEE 802.15.4, 54, 56

IHD, 37
IoT, 57, 59

REDD, 54

SEP, 57, 58

Thread, 56
Tracebase, 54

UK-DALE, 54

ZigBee, 57

Glossary

BLE Bluetooth low energy

CAD consumer access device

Datasets for NILM

AMPds The almanac of minutely power dataset recorded by Makonin et al. 2013 and available at http://ampds.org

Dataport The Pecan Street Dataport dataset, available at https://dataport.pecanstreet.org

iAWE The Indian dataset for ambient water and energy recorded by Batra et al. 2013 and available at http://iawe.github.io

REDD The reference energy disaggregation data set recorded by J. Zico Kolter & M. J. Johnson 2011 and available at http://redd.csail.mit.edu

Tracebase A dataset of appliance signatures recorded by Reinhardt et al. 2012 and available at https://www.tracebase.org

UK-DALE The UK domestic appliance-level electricity dataset recorded by Kelly & Knottenbelt 2015b and available at http://www.doc.ic.ac.uk/~dk3810/data/

IAM individual appliance monitor

IEEE 802.15.4 a wireless networking standard which specifies the physical layer (PHY) and media access control (MAC).

IHD in-home display

IoT internet of things

IP Internet Protocol

MAC medium access control

PHY physical networking layer

PLC power-line communication

SEP Smart Energy Profile

SMETS smart metering equipment technical specification

Thread A low-power mesh wireless networking standard built on top of the IEEE 802.15.4 physical networking layer (PHY) and medium access control (MAC).

TRX transceiver

TX transmitter

WSN wireless sensor network

ZigBee A low-power mesh wireless networking standard built on top of the IEEE 802.15.4 PHY and MAC.