

Disaggregation von Haushaltsenergiemessdaten mit tiefen neuronalen Netzen

Zur Erlangung des akademischen Grades eines
Doktors der Ingenieurwissenschaften
(Dr.-Ing.)

von der KIT-Fakultät für Wirtschaftswissenschaften
des Karlsruher Instituts für Technologie (KIT)

genehmigte
DISSERTATION

von
Dipl.-Inform. Kaibin Bao

Datum der mündlichen Prüfung: 27. Januar 2021
Referent: Prof. Dr. Hartmut Schmeck
Korreferent: Prof. Dr. Dominik Engel

Karlsruhe 2021



Dieses Werk ist lizenziert unter einer Creative Commons Namensnennung -
Weitergabe unter gleichen Bedingungen 4.0 International Lizenz (CC BY-SA 4.0):
<https://creativecommons.org/licenses/by-sa/4.0/deed.de>

im Gedenken an
杨一平

Abstract (English Version)

The currently best approaches to disaggregate household energy measurement data, which are recorded by commercially available smart meters, are based on artificial neural networks, which were created with a deep learning method. However, it is difficult to compare the performance of these approaches objectively because the approaches are often evaluated on different data sets, training procedures are not described in detail and test metrics are not standardized. Therefore, the evaluation of known approaches based on a consistent design for disaggregation experiments is necessary. This thesis shows that the practicability of all approaches is limited mainly by the small number of unique device models in the training data set.

Nevertheless, in order to reduce the error in device load estimation for a given dataset, this thesis focuses on the problem that existing approaches often output clearly wrong and unfeasible load profiles that can never be measured by real devices. For this purpose, two different new approaches are being investigated to ensure the plausibility of the estimated load profile. Different parts of a Generative Adversarial Network (GAN) are used to generate feasible device load profiles. A third approach redesigns an existing neural model and combines it with the U-Net-architecture by adding cross connections between layers. This should help to improve the reproduction of detailed information in the load profiles.

In the evaluation of the novel approaches with the same experimental setup, the second approach produces feasible load profiles more frequently whereas the disaggregation accuracy remains at the same level. By another additional exchange of the loss function used in the model training, all considered evaluation metrics can be improved on average over all devices. In addition, for certain device classes, a further improvement of the evaluation metrics can be achieved with the U-Net architecture evaluated in the third approach.

Abstract (German Version)

Die aktuell besten Ansätze zur Disaggregation von Haushaltsenergiemessdaten, die von handelsüblichen Smart Meter erfasst werden, basieren auf künstlichen neuronalen Netzen, die mit einer Deep-Learning-Methodik erstellt sind. Die Leistungsfähigkeit dieser Ansätze objektiv zu vergleichen ist allerdings schwer, da die Ansätze oft auf unterschiedlichen Datensätzen evaluiert werden, Trainingsverfahren nicht ausführlich beschrieben sind und keine einheitlichen Testmetriken verwendet werden. Erst durch die Evaluation bekannter Ansätze basierend auf einem einheitlichen Aufbau für Disaggregationsexperimente wird in dieser Arbeit deutlich, dass die Praxistauglichkeit aller Ansätze insbesondere durch die geringe Anzahl unterschiedlicher Gerätemodelle im Trainingsdatensatz beschränkt ist.

Um für einen festgelegten Trainingsdatensatz den Fehler bei der Gerätelastgangsschätzung dennoch zu verringern, fokussiert sich die vorliegende Arbeit auf das Problem, dass Ansätze oftmals eindeutig falsche und unplausible Gerätelastgänge ausgeben, die von realen Geräten nicht reproduziert werden können. Dazu werden zwei verschiedene neue Ansätze untersucht, die die Plausibilität der geschätzten Lastgänge sicherstellen sollen. Zur Erzeugung von plausiblen Gerätelastgängen werden unterschiedliche Teile eines Generative Adversarial Networks (GAN) verwendet. Ein dritter Ansatz entwirft ein bestehendes Netzmodell neu und kombiniert dieses mit der U-Net-Architektur durch das Hinzufügen von Querverbindungen zwischen Netzschichten. Dies soll helfen, Detailinformationen in den Lastgängen besser zu reproduzieren.

Bei der Evaluation der eigenen Ansätze mit dem gleichen Experimentenaufbau werden bei dem zweiten Ansatz häufiger realisierbare Lastgänge ausgegeben. Dabei bleibt die Disaggregationsgenauigkeit auf dem gleichen Niveau. Durch einen weiteren Austausch der beim Modelltraining verwendeten Verlustfunktion wird erreicht, dass sich alle betrachteten Bewertungsmetriken im Mittel über alle Geräte verbessern. Zudem kann bei bestimmten Geräteklassen mit der im dritten Ansatz evaluierten U-Net-Architektur eine weitere Verbesserung der Bewertungsmetriken erzielt werden.

Inhaltsverzeichnis

Tabellenverzeichnis	xv
Abbildungsverzeichnis	xvii
Abkürzungsverzeichnis	xxi
1 Einleitung	1
1.1 Motivation – Non-Intrusive Load Monitoring	1
1.2 Motivation – Voraussetzungen zur Erhebung von Haushaltsenergiemessdaten	2
1.3 Problemstellung	4
1.3.1 Forschungsfragen	4
1.3.2 Beiträge	6
1.3.3 Abgrenzung	6
1.4 Entstehungskontext	6
1.4.1 Energiedatenanalyse und Privatsphärenschutz	7
1.4.2 Energiedatenanalyse und IT-Sicherheit	8
1.4.3 Smarter Energy	8
1.4.4 Modellierung von Gerätelastgängen mit künstlichen neuronalen Netzen	9
1.5 Gliederung und Vorgehensweise	10
2 Grundlagen	13
2.1 Haushaltsenergiemessdaten	13
2.1.1 Messgrößen in Wechselstromsystemen	14
2.1.2 Haushaltsenergiemessdaten für Mess- und Abrechnungsprozesse des deutschen Energiesystems	16
2.2 Privatsphärenbedrohung durch Energiedatenüberwachung	17
2.2.1 Personenbezogene Informationen in Haushaltsenergiemessdaten . . .	18
2.2.2 Privatsphärenbedrohungen durch Energieverbrauchserfassung	19
2.2.3 Ansätze zum Privatsphärenschutz	20

2.3	Disaggregation von Haushaltsenergiemessdaten	23
2.3.1	Problemstellung	24
2.3.2	Anwendungsfelder	25
2.4	Einordnung klassischer Disaggregationsansätze	27
2.4.1	Geräte Merkmale im Lastgang	28
2.4.2	Maschinelle Lernansätze und Gerätemodelle	31
2.5	Künstliche neuronale Netze	35
2.5.1	Deep Learning	37
2.5.2	Trainings- / Optimierungsverfahren	38
2.5.3	Initialisierung	39
2.5.4	Regularisierung	39
2.5.5	Normalisierung	40
2.5.6	Aktivierungsfunktion	40
2.5.7	Verlustfunktion	41
2.5.8	Netzarchitekturelemente	42
2.5.9	Generative Adversarial Networks (GAN)	45
3	Bestehende Deep-Learning-Ansätze zur Disaggregation	49
3.1	Überblick über den Disaggregations- und Trainingsprozess	51
3.2	Eigenschaften des Haushaltslastgangs	53
3.3	Aufteilung in Sequenzen durch ein gleitendes Zeitfenster	55
3.3.1	Fensterbreite	55
3.3.2	Fensterschrittweite	57
3.4	Generierung der Trainingsbeispiele	60
3.5	Vor- und Nachverarbeitung	62
3.6	Netzarchitektur bestehender Ansätze	64
3.6.1	Darstellung von neuronalen Netzen	64
3.6.2	Lastblock-Regressions-Netzwerk	66
3.6.3	Denoising Autoencoder	67
3.6.4	Rekurrente Netze	69
3.6.5	Faltende neuronale Netze	72
3.6.6	Faltende neuronale Netze mit Querverbindungen	76
3.7	Ausgabeformat	78
3.8	Zusammenführungsstrategie	79
3.9	Trainingsverfahren und -parameter	80
3.10	Datensätze	82
3.11	Fazit und Auswahl der zu untersuchenden Verfahren als Stand der Technik . .	83
4	Szenario und Aufbau der Experimente	87
4.1	Szenario	88
4.2	Betrachtete Geräteklassen	89
4.2.1	Auswahl der Geräteklassen	89

4.2.2	Beschreibung der ausgewählten Haushaltsgeräteklassen	94
4.3	Verwendete Datensätze und spezifische Vorverarbeitungsschritte	101
4.3.1	Anforderungen zur Auswahl der Datensätze	103
4.3.2	Auswahl der Datensätze	104
4.3.3	DRED: Dutch Residential Energy Dataset	104
4.3.4	ECO: Electricity Consumption and Occupancy Dataset	106
4.3.5	ESHL: Energiemessdaten des KIT Energy Smart Home Lab	106
4.3.6	UK-DALE: UK Domestic Appliance-Level Electricity	116
4.3.7	REFIT: Personalised Retrofit Decision Support Tools for UK Homes using Smart Home Technology	117
4.4	Gemeinsame Vorverarbeitungsschritte	119
4.4.1	Identifikation und Umgang mit fehlenden Messwerten	119
4.4.2	Segmentierungsheuristik für Aktivierungen	122
4.5	Aufteilung in Trainings- und Testdatenmenge	124
4.6	Generatoren für Trainingspaare	125
4.6.1	Vorhandene Generatoren für Trainingspaare	125
4.6.2	Zufallssegmentgenerator mit ausbalancierter Gebäude- und Aktivitäts- häufigkeit	126
4.6.3	Haushaltslastgangsynthese durch Austausch von Gerätelastprofilen . .	126
4.6.4	Definition der Trainingsepoche und Vollständigkeit	127
4.6.5	Verwendete Generatoren für Trainingspaare	128
4.7	Betrachtete Auflösung	128
4.8	Evaluationsmetriken	129
4.8.1	Klassifikationsmetriken	129
4.8.2	Regressionsmetriken	131
4.8.3	Energiebasierte Metriken	132
4.9	Implementierung	133
4.9.1	Software-Werkzeuge und Frameworks	134
4.9.2	Rechenbeschleuniger	136
5	Analyse bestehender Deep-Learning-Ansätze	137
5.1	Evaluierte Ansätze	137
5.1.1	DAE: Denoising Autoencoder	137
5.1.2	S2P / S2S: Sequence-to-Point und Sequence-to-Sequence	139
5.1.3	SGN: Subtask Gated Networks	140
5.1.4	GLU: Gated Linear Units	141
5.1.5	DFCN: Dilated Fully Convolutional Network	141
5.2	Bewertung der Disaggregation	142
5.2.1	Waschmaschinen und Wäschetrockner	143
5.2.2	Kühlgeräte	148
5.2.3	Trockner	152
5.2.4	TV und Unterhaltungselektronik	155

5.2.5	Mikrowellen	157
5.2.6	Geschirrspüler	158
5.2.7	Wasserkocher	161
5.3	Zusammenfassung	162
5.3.1	Schlussfolgerungen aus der Analyse	162
5.3.2	Forschungsbedarf	164
6	Ansätze zur Optimierung der Disaggregationsqualität	167
6.1	Ansatz A: Unüberwacht trainierter Generator für Gerätelastgänge	167
6.1.1	Vorgehensweise	169
6.1.2	Einschränkung auf Waschmaschinenlastgänge	170
6.1.3	Erstellung des Lastsequenzgenerators	170
6.1.4	Netzarchitektur des Diskriminators und des Generators	172
6.1.5	Trainingskonvergenz bei alternativen Netzmodellen	174
6.1.6	Wasserstein GAN	176
6.1.7	Validierung des Gerätelastprofilgenerators	177
6.1.8	Disaggregation mit Gerätelastsequenzgeneratoren	184
6.1.9	Schlussfolgerungen	187
6.2	Ansatz B: Diskriminator für Gerätelastgänge	188
6.2.1	Vorgehensweise	188
6.2.2	Architektur der Diskriminatoren	190
6.2.3	Trainingsparameter	193
6.2.4	Ergebnisse	194
6.2.5	Fazit	199
6.3	Ansatz C: U-Net mit Dilertation	201
6.3.1	Trainingsparameter	203
6.3.2	Ergebnisse	203
6.3.3	Fazit	204
6.4	Zusammenfassung	207
7	Ergebnisse	209
7.1	Reproduktion des Gerätelastgangs	210
7.2	Energieverbrauchsbilanz eines Gerätes	217
7.3	Detektion von Geräteaktivität	222
7.4	Analyse	228
7.4.1	Übertragbarkeit auf unbekannte Haushalte	228
7.4.2	Reproduktion realistischer Lastgänge	229
7.4.3	Effekt der L_1 -Verlustfunktion auf das Training	232
7.4.4	Rechenaufwand	232
7.5	Zusammenfassung	236

8 Zusammenfassung und Ausblick	239
8.1 Vorgehensweise	240
8.1.1 Ermittlung Stand der Technik bestehender Deep-Learning-Ansätze . .	240
8.1.2 Szenario und Aufbau der Experimente	241
8.1.3 Analyse der Deep-Learning-Ansätze	243
8.2 Adressierung der Forschungsfragen	243
8.3 Beiträge	244
8.3.1 Evaluation bestehender Ansätze	244
8.3.2 Kombination von Datensätzen	245
8.3.3 Ansätze zur Vermeidung unrealistischer Gerätelastgänge	245
8.3.4 Vergleich der Disaggregationsansätze	248
8.3.5 Weitere Beiträge	248
8.4 Ausblick	250
A Einzelne Ergebnisse bei der Evaluation der Ansätze zur Optimierung der Disaggregationsqualität	253
A.1 Ergebnisse cGAN (Anhang)	253
A.2 Ergebnisse U-Net (Anhang)	257
Literatur	261

Tabellenverzeichnis

3.1	Länge der Haushaltslastsequenzen vorhandener Ansätze für UK-DALE	56
3.2	Länge der Haushaltslastsequenzen vorhandener Ansätze für REDD	56
4.1	Anzahl Lastgänge ähnlicher Gerätetypen und Kombinationsgeräten	91
4.2	Öffentliche Datensätze von Haushaltsenergiemessdaten	102
4.3	Anforderungserfüllung der Datensätze	105
4.4	Gesamtzahl der Aktivierungen im Datensatz DRED	105
4.5	Gesamtzahl der Aktivierungen im Datensatz ECO	107
4.6	Gesamtzahl der Aktivierungen im Testdatensatz aus dem KIT ESHL	112
4.7	Gesamtzahl der Aktivierungen in Datensatz UK-DALE 2017	117
4.8	Gesamtzahl der Aktivierungen im REFIT-Datensatz	120
4.9	Grenzwerte zur Aktivierungssegmentierung	123
4.10	Aufteilung in Trainings- und Testdatenmenge	125
4.11	Konfusionsmatrix bei einer binären Klassifikation	129
4.12	Definition der Konfusionsmatrix für energiebasierte Metriken	133
5.1	Hyperparameter des Verfahrens Denoising Autoencoder	138
5.2	Hyperparameter der Verfahren Sequence-to-Point/-Sequence	139
5.3	Hyperparameter des Verfahrens Dilated Fully Convolutional Network	142
7.1	Rechenaufwand beim Training und der Disaggregation	234
7.2	Rechenaufwand für das Training und die Nutzung von GAN	236
8.1	Bester Disaggregationsansatz pro Geräteklasse und Anwendungsszenario	249

Abbildungsverzeichnis

1.1	Einordnung des Aufbaus dieser Arbeit in den Disaggregationsprozess	12
2.1	NILM aus nachrichtentechnischer Perspektive	23
2.2	Schematischer Ausschnitt eines niedrig aufgelösten Haushaltlastgangs . . .	29
2.3	Ein vollverbundenes neuronales Netz	42
2.4	Ein faltendes neuronales Netz	43
2.5	Ein rekurrentes neuronales Netz	44
2.6	Training eines GAN	46
3.1	Disaggregationsprozess bei Deep-Learning-basierten Ansätzen	51
3.2	Trainingsablauf für ein neuronales Netz zur Disaggregation	52
3.3	Aufteilungsmöglichkeiten in Lastsequenzen mit einem gleitenden Zeitfenster	58
3.4	Blockdarstellung vs. diskrete Darstellung einer vollverbundenen Schicht . . .	64
3.5	Faltungsschicht für 2 Merkmale	65
3.6	Gestapelte Faltungsschichten	65
3.7	Dilatierte Faltungsschichten	65
3.8	Bidirektionales rekurrentes neuronales Netz	66
3.9	Das Lastblock-Regressions-Netzwerk von Kelly und Knottenbelt	67
3.10	Denoising Autoencoder von Kelly und Knottenbelt	68
3.11	RNN mit LSTM von Kelly und Knottenbelt	70
3.12	Faltendes neuronales Netz von Paulo Marques do Nascimento	73
3.13	S2S und S2P von Zhang et al.	74
3.14	CEDN von Garcia und Macabebe	75
3.15	DFCN von Brewitt und Goddard	75
4.1	Anzahl Haushalte mit Lastgängen der jeweiligen Geräteklasse	93
4.2	Ausstattungsgrad von Haushaltsgeräten 2018	95
4.3	Ausstattungsgrad von Haushaltsgeräten 2016	95
4.4	Elektrogeräteenergieverbrauchsanteil in privaten Haushalten 2011	96

4.5	Beispielhafte Lastgänge von Kühl- und Gefrierschränken	96
4.6	Beispielhafte Lastgänge von Waschmaschinen und Waschtrockner	97
4.7	Beispielhafte Lastgänge von Wäschetrocknern	98
4.8	Aktivierungen von Geschirrspülmaschinen	100
4.9	Aktivierungen von Wasserkochern	100
4.10	Aktivierungen von Mikrowellen	100
4.11	Aktivierungen von Fernsehern und Unterhaltungselektronik	101
4.12	Verfügbarkeit der Messdaten im Datensatz DRED	106
4.13	Verfügbarkeit der Messdaten im Gebäude 2 des Datensatzes ECO	107
4.14	Verfügbarkeit der Messdaten (KIT Energy Smart Home Lab (ESHL) 2011-2016)	110
4.15	Verfügbarkeit der Messdaten (ESHL seit 2016)	111
4.16	Anteil am Energieverbrauch (ESHL Mai 2012)	114
4.17	Anteil am Energieverbrauch (ESHL Juli 2018)	114
4.18	Anteiliger Energieverbrauch betrachteter Geräte (ESHL Mai 2012)	115
4.19	Anteiliger Energieverbrauch betrachteter Geräte (ESHL Juli 2018)	115
4.20	Messdatenverfügbarkeit UK-DALE Gebäude 5	118
4.21	Messdatenverfügbarkeit REFIT Gebäude 2	121
4.22	Elemente energiebasierter Konfusionsmatrix (grafische Entsprechung)	134
5.1	Fehlermetriken bekannter Disaggregationsansätze bei Waschmaschinen . . .	143
5.2	Disaggregierter Waschmaschinenlastgangsausschnitt aus REFIT 14	145
5.3	Disaggregierter Waschmaschinenlastgangsausschnitt aus REFIT 9	146
5.4	Weiterer Lastgangsausschnitt aus REFIT 9 mit hoher Gebäudelast	146
5.5	Disaggregierter Waschmaschinenlastgangsausschnitt aus ESHL 1	147
5.6	Disaggregierter Waschmaschinenlastgangsausschnitt aus ESHL 2	148
5.7	Fehlermetriken bekannter Disaggregationsansätze bei Kühlgeräten	149
5.8	Disaggregierter Kühlgerätelastgangsausschnitt aus REFIT 9	150
5.9	Disaggregierter Kühlgerätelastgangsausschnitt aus UK-DALE 5	150
5.10	Disaggregierter Kühlgerätelastgangsausschnitt aus REFIT 19	151
5.11	Disaggregierter Kühlgerätelastgangsausschnitt aus ESHL 2	152
5.12	Fehlermetriken bekannter Disaggregationsansätze bei Trocknern	153
5.13	Disaggregierter Trocknerlastgangsausschnitt aus REFIT 14	153
5.14	Disaggregierter Trocknerlastgangsausschnitt aus ESHL 2	154
5.15	Fehlermetriken bekannter Disaggregationsansätze bei Trocknern	155
5.16	Disaggregierter Lastgangsausschnitt für TV & Unterhaltung in REFIT 9	156
5.17	Disaggregierter Lastgang von TV & Unterhaltung in REFIT 19	157
5.18	Fehlermetriken bekannter Disaggregationsansätze bei Trocknern	158
5.19	Disaggregierter Mikrowellenlastgangsausschnitt aus REFIT 14	159
5.20	Disaggregierter Mikrowellenlastgangsausschnitt aus REFIT 9	159
5.21	Fehlermetriken bekannter Disaggregationsansätze bei Trocknern	160
5.22	Disaggregierter Geschirrspülerlastgangsausschnitt aus REFIT 14	160
5.23	Fehlermetriken bekannter Disaggregationsansätze bei Trocknern	161

5.24	Disaggregierter Wasserkocherlastgangsausschnitt aus ESHL 2	162
6.1	Integration Gerätelastsequenzgenerator in Disaggregationsprozess	168
6.2	Training eines Generative Adversarial Network	170
6.3	Diskriminatornetzarchitektur basierend auf DC-GAN	172
6.4	Generatornetzarchitektur basierend auf DC-GAN	174
6.5	Instabile Trainingskonvergenz bei DC-GAN	175
6.6	Mit DC-GAN generierte Beispiellastgänge der Waschmaschine	176
6.7	Methode zur Validierung des Gerätelastsequenzgenerators	178
6.8	Lastsequenzkodierernetzarchitektur	179
6.9	Fehler bei Generierung von Waschmaschinenlastgängen des Testdatensatzes	180
6.10	Reproduktion von Waschmaschinenlastgängen in ESHL 1	181
6.11	Reproduktion von Waschmaschinenlastgängen in UK-DALE 5	182
6.12	Reproduktion von Waschmaschinenlastgängen in REFIT 9	183
6.13	DAE-Netzarchitektur mit integriertem Lastsequenzgenerator	184
6.14	S2S-Netzarchitektur mit integriertem Lastsequenzgenerator	185
6.15	Fehler bei der Disaggregation mit Gerätelastsequenzgeneratoren	186
6.16	Disaggregation von Waschmaschinenlastgängen mit DC-GAN in ESHL 1	187
6.17	Disaggregatortraining mit GAN-basierter strukturierter Verlustfunktion	189
6.18	Diskriminatornetzarchitektur für DAE-Ansatz	191
6.19	Diskriminatornetzarchitektur für S2S-Ansatz	192
6.20	Diskriminatornetzarchitektur für DFCN	193
6.21	Unterschiedliche Verlustfunktionen bei DAE	195
6.22	Unterschiedliche Verlustfunktionen bei S2S	197
6.23	Unterschiedliche Verlustfunktionen bei DFCN	198
6.24	Architektur des U-Net	202
6.25	Diskriminatorarchitektur für das U-Net	204
6.26	DFCN vs. Dilated U-Net	205
7.1	eMCC und MAE von allen Ansätzen	211
7.2	Regressionsmetriken bei der Disaggregation von Waschmaschinen	213
7.3	Regressionsmetriken bei der Disaggregation von Kühlgeräten	213
7.4	Regressionsmetriken bei der Disaggregation von Trocknern	214
7.5	Regressionsmetriken bei der Disaggregation von TV & Unterhaltung	215
7.6	Regressionsmetriken bei der Disaggregation von Mikrowellen	215
7.7	Regressionsmetriken bei der Disaggregation von Geschirrspülern	216
7.8	Regressionsmetriken bei der Disaggregation von Wasserkochern	217
7.9	Absoluter RETE von allen Ansätzen	218
7.10	Fehlgeschätzter Gesamtenergieverbrauchsanteil Waschmaschine	219
7.11	Fehlgeschätzter Gesamtenergieverbrauchsanteil Kühlgeräte	220
7.12	Fehlgeschätzter Gesamtenergieverbrauchsanteil Trockner	220
7.13	Fehlgeschätzter Gesamtenergieverbrauchsanteil TV & Unterhaltung	221

7.14 Fehlgeschätzter Gesamtenergieverbrauchsanteil Mikrowelle	221
7.15 Fehlgeschätzter Gesamtenergieverbrauchsanteil Geschirrspüler	222
7.16 Fehlgeschätzter Gesamtenergieverbrauchsanteil Wasserkocher	222
7.17 MCC und F1 von allen Ansätzen	223
7.18 Klassifikationsleistungsbewertung Waschmaschine	224
7.19 Klassifikationsleistungsbewertung Kühlgeräte	225
7.20 Klassifikationsleistungsbewertung Trockner	225
7.21 Klassifikationsleistungsbewertung TV & Unterhaltung	226
7.22 Klassifikationsleistungsbewertung Mikrowelle	226
7.23 Klassifikationsleistungsbewertung Geschirrspüler	227
7.24 Klassifikationsleistungsbewertung Wasserkocher	227
7.25 Neu disaggregierter Kühlgerätestrangsausschnitt aus ESHL 2	230
7.26 Neu disaggregierter Trocknerlastgangsausschnitt aus REFIT 19	231
7.27 Neu disaggregierter Trocknerlastgangsausschnitt aus ESHL 2	231
7.28 Theoretische Anzahl Rechenoperationen für die Disaggregation	235
7.29 Theoretische Anzahl Rechenoperationen für das Training	237
A.1 Disaggregation von Waschmaschinen mit verschiedenen Verlustfunktionen .	253
A.2 Disaggregation von Geschirrspülern mit verschiedenen Verlustfunktionen . .	254
A.3 Disaggregation von Trocknern mit verschiedenen Verlustfunktionen	254
A.4 Disaggregation von TV/Unterhaltungselektronik mit verschiedenen Verlust- funktionen	255
A.5 Disaggregation von Mikrowellen mit verschiedenen Verlustfunktionen	255
A.6 Disaggregation von Wasserkochern mit verschiedenen Verlustfunktionen . .	256
A.7 Disaggregation von Kühlgeräten mit verschiedenen Verlustfunktionen	256
A.8 Disaggregation von Waschmaschinen mit der U-Net-Architektur	257
A.9 Disaggregation von Geschirrspülern mit der U-Net-Architektur	257
A.10 Disaggregation von Trocknern mit der U-Net-Architektur	258
A.11 Disaggregation von TV/Unterhaltungselektronik mit der U-Net-Architektur .	258
A.12 Disaggregation von Mikrowellen mit der U-Net-Architektur	259
A.13 Disaggregation von Wasserkochern mit der U-Net-Architektur	259
A.14 Disaggregation von Kühlgeräten mit der U-Net-Architektur	260

Abkürzungsverzeichnis

ADAM A Method for Stochastic Optimization

BSI Bundesamt für Sicherheit in der Informationstechnik

CUDA Compute Unified Device Architecture

CuDNN CUDA Deep Neural Network library

DAE Denoising Autoencoder

DC-GAN Deep Convolutional Generative Adversarial Network

DFCN Dilated Fully Convolutional Network

ESHL KIT Energy Smart Home Lab

FHMM Factorial Hidden Markov Model

GAN Generative Adversarial Network

GRU Gated Recurrent Unit

HMM Hidden Markov Model

LSTM Long Short-Term Memory

MsbG Messstellenbetriebsgesetz

NAG Nesterov's Accelerated Gradient

NILM Non-Intrusive Appliance Load Monitoring

NILMTK Non-Intrusive Load Monitoring Toolkit

ReLU Rectified Linear Unit

SGD Stochastic Gradient Descent

Einleitung

1.1 Motivation – Non-Intrusive Load Monitoring

Ein Baustein für die Energiewende ist die Erhöhung der Energieeffizienz sowie das Energiemanagement, auch im Bereich der Privathaushalte. Dabei bezeichnet Energiemanagement den optimierten Einsatz von energietechnischen Erzeugungs- und Verbrauchseinheiten mit dem Ziel, insbesondere volatile Energieerzeugung durch erneuerbare Energien effizient zu nutzen (Allerding und Schmeck, 2011; Allerding, 2013). Um sowohl die Kostenlast zu reduzieren und die Energieautarkie eines Haushalts zu stärken, sollten Energiemanagementsysteme in der Lage sein, Energieströme im Haushalt über alle Energieträger hinweg sowie die Interdependenzen zwischen den Geräten zu handhaben und gleichzeitig die Optimierung des Gerätebetriebs zu unterstützen (Mauser et al., 2016).

Ein erster Schritt in Richtung Energieverbrauchsoptimierung besteht in der Kenntnis des eigenen Energieverhaltens. Jedoch fehlt im Allgemeinen das Bewusstsein, wieviel elektrischer Energieverbrauch durch welches Gerät im eigenen Haushalt verursacht wird. Dabei helfen statistische Angaben, wie sie auf Energieverbrauchskennzeichnungen zu finden sind, wenig, da diese das individuelle Nutzungsverhalten innerhalb eines Haushalts nicht berücksichtigen können. Dieses kann nur durch eine individuelle Verbrauchsmessung ermittelt werden. Hierbei reicht auch das jährliche oder monatliche Ablesen des Stromverbrauchs nicht aus, da die Bewohner mit diesen Werten nur schwer einen kausalen Zusammenhang zwischen Verhalten und Verbrauch herstellen können.

Gemäß dem Prinzip „If you can not measure it, you can not improve it.“ von Lord Kelvin (Batra, Singh et al., 2015), ist es daher ein wichtiges Anliegen, mehr Transparenz über das eigene Energieverhaltensverhalten zu gewinnen, indem die unterschiedlichen Beiträge der verwendeten Verbrauchseinrichtungen zeitnah aufgeschlüsselt werden.

Um bei der Ermittlung des individuellen Energieverbrauchs die aufwendige Installation und Einrichtung von einzelnen Messgeräten zu vermeiden, ist es durch das Non-Intrusive Appliance Load Monitoring (NILM) möglich, aus einer Gebäudeenergieverbrauchsmessung die einzelnen Geräteverbräuche zu schätzen. Dazu wird bei NILM der Gebäudelastgang analysiert und aufgeteilt (engl. disaggregation) bzw. einzelnen Geräten (engl. appliances) zugeordnet. Neben Gerätemodellen wird dazu lediglich die Messung des Gesamtenergieverbrauchs am Hausanschlusspunkt benötigt und die Installation von einzelnen Messgeräten an den elektrischen Geräten erübrigt sich, daher das Adjektiv „non-intrusive“.

Es bestehen jedoch auch Zweifel, inwiefern die Ergebnisse einzelner Studien, die Energieeinsparpotentiale von bis zu 50 % in Aussicht stellen, in der Realität umsetzbar sind, insbesondere in der langfristigen Betrachtung. So weisen Kelly und Knottenbelt (2016) in ihrer Metastudie zur Wirksamkeit von Energieverbrauchsfeedback darauf hin, dass über die Hälfte der von ihnen untersuchten Studien die Ergebnisse nicht mit Ergebnissen einer Kontrollgruppe gegenübergestellt haben, um den Hawthorne-Effekt (Schwartz et al., 2013) zu messen. Der Hawthorne-Effekt beschreibt die Verhaltensänderung von Studienteilnehmern, welche lediglich aufgrund der Studienteilnahme bzw. dem Gefühl, beobachtet zu werden, entsteht. Im Kontext von Studien zu Energieverbrauchsfeedback muss dieser Effekt berücksichtigt werden, um die Wirkung von Feedback nicht zu überschätzen. Hierbei kann beispielsweise einer Kontrollgruppe zufälliges oder generisches Feedback gegeben werden, welches mit ihrem Energieverbrauch nicht kausal zusammenhängt. Mit dem Hawthorne-Effekt wird erklärt, dass dabei dennoch eine Energieverbrauchsreduktion beobachtet wird (vgl. Kelly und Knottenbelt, 2016; Schwartz et al., 2013). Um die Effekte der Energieverbrauchsdatenaufschlüsselung von Effekten durch Beobachtungsdruck abzugrenzen, sollten weitere Studien durchgeführt werden. Hierbei bieten NILM-Verfahren ein großes Potenzial für größer angelegte Studienformate. Denn NILM kann mit sehr geringen Kosten ausgerollt werden, da es ohne Zusatzkosten für Hardware auskommt, wenn in Zukunft in jedem Haushalt ohnehin intelligente Messeinrichtungen verbaut sind.

1.2 Motivation – Voraussetzungen zur Erhebung von Haushaltsenergiemessdaten

Smart Metering (auch intelligentes Messsystem oder Advanced Metering genannt) steht für die automatische Erhebung und Übermittlung von Energiebezug und Energieeinspeisung eines Netzanschlussnehmers aus oder in ein Energienetz. Im Gegensatz zur klassischen Energieverbrauchsmessung ermöglicht Smart Metering die Darstellung des Momentanverbrauchs und damit auch die Ermittlung des zeitlichen Verlaufs des Energieverbrauchs. Bei dem Energieverbraucher soll durch diese Darstellungsform ein besseres Bewusstsein über

dessen genutzte Energiemengen geschaffen werden. Energieversorger und Energiesystembetreiber können diese Daten für eine genauere Schätzung des Energiebedarfs nutzen, um das Stromnetz effektiver zu betreiben. Jedoch erregt eine solche Art der Energieverbrauchsüberwachung auch Bedenken, da befürchtet wird, dass Energieanbieter oder Dritte die aufgezeichneten Daten missbräuchlich verwenden und die Privatsphäre verletzen könnten. Dies liegt darin begründet, dass sich bei einer zeitlich hochaufgelösten Überwachung und Analyse des Haushaltsenergieverbrauchs theoretisch Rückschlüsse auf Aktivitäten im Haushalt wie Aufstehen, Schlafengehen, Hausputz, Fernsehschauen oder Wäschewaschen ziehen lassen. Damit kann es zu einer erheblichen Gefährdung der Privatsphäre kommen. (vgl. Lisovich et al., 2010; Molina-Markham et al., 2010)

Zum Schutz der Privatsphäre stellt daher das Bundesamt für Sicherheit in der Informationstechnik (BSI) mit dem Schutzprofil für Smart-Meter-Gateways (*BSI-CC-PP-0073*) und den dazu gehörigen Technischen Richtlinien (*TR-03109*) vergleichsweise hohe Anforderungen an die Datensicherheit und den Datenschutz für Energieverbrauchsdaten. Durch Tarifprofile werden Energiemessdaten auf dem Gateway zeitlich aggregiert, bevor die Daten an Energiemarktteilnehmer übermittelt werden. Die zeitlich höchstmögliche Auflösung für jene Lastgänge, die an externe Marktteilnehmer übermittelt werden, entspricht einem Messintervall von 15 Minuten. Der Erfolg von NILM-Verfahren ist dabei durchaus auch ein Indikator für den empirischen Privatsphärenverlust, der durch die Veröffentlichung der Energiemessdaten einhergehen würde.

Begrifflich ist ein Smart Meter bzw. ein intelligentes Messsystem definiert als eine über ein Smart-Meter-Gateway in ein Kommunikationsnetz eingebundene moderne Messeinrichtung zur Erfassung elektrischer Energie, die den tatsächlichen Energieverbrauch und die tatsächliche Nutzungszeit widerspiegelt und den besonderen Anforderungen nach den §§ 21 und 22 des Messstellenbetriebsgesetzes (*MsbG*) genügt, die zur Gewährleistung des Datenschutzes, der Datensicherheit und Interoperabilität in Schutzprofilen und Technischen Richtlinien festgelegt werden können.

In dieser Arbeit werden Messdaten aus dem Haushaltsinterface von Stromzählern verwendet, die mit einer höheren Auflösung als 15 Minuten vorliegen. Die für diese Arbeit genutzten handelsüblichen Haushaltsstromzähler können dem Begriff der modernen Messeinrichtung zugeordnet werden. Hierbei handelt es sich um eine Messeinrichtung, die den tatsächlichen Elektrizitätsverbrauch und die tatsächliche Nutzungszeit widerspiegelt und über ein Smart-Meter-Gateway sicher in ein Kommunikationsnetz eingebunden werden kann (§ 2 Nr. 15 *MsbG*).

Im Gegensatz zum intelligenten Messsystem bedarf die moderne Messeinrichtung keiner Zertifizierung durch das BSI und dient als Zwischenstufe zwischen elektromechanischem Zähler und intelligentem Messsystem (Bundestags-Drucksache 18/7555, S. 74). Moderne Messeinrichtungen sollen bereits technische Möglichkeiten der Verbrauchsvisualisierung bieten, ohne allerdings als „reiner Zähler“ über den Haushalt hinaus sicher kommunizieren zu können. Da diese Differenzierung für die vorliegende Arbeit von untergeordneter

Relevanz ist, werden die adressierten Messeinrichtungen als Smart Meter (intelligente Messeinrichtung) im Sinne eines Oberbegriffs bezeichnet.

1.3 Problemstellung

Zunächst gilt es die Randbedingungen des betrachteten Problems zu veranschaulichen:

Disaggregation ist die Schätzung von einzelnen Gerätelastgängen

Ziel der Ansätze ist die Schätzung der Lastgänge einzelner elektrischer Geräte, ohne hierfür auf weitere, kostenintensive Messsysteme am Gerät angewiesen zu sein. Stattdessen soll die Lastgangmessung des gesamten Gebäudes oder Haushalts ausreichen. Die Qualität der disaggregierten Lastgänge sollte dabei der einer Messung durch günstige am Gerät installierte Energiemessgeräte nahe kommen. Neben der Genauigkeit der geschätzten Lastgänge ist die Bestimmbarkeit der Nutzungszeiten der elektrischen Geräte ein Bewertungskriterium.

Verwendung von Daten eines handelsüblichen Haushaltsstromzählers

Die Schätzung der Gerätelastgänge soll auch mit Energiemessdaten erfolgen, die von einem handelsüblichen Haushaltsstromzähler (moderne Messeinrichtung) erfasst werden können. Handelsübliche Zähler geben dabei Leistungswerte in einem Intervall von einer Sekunde oder länger aus. Dadurch sind NILM-Verfahren, die eine höheraufgelöste Messung elektrischer Merkmale erfordern, nicht anwendbar.

Übertragbarkeit der Disaggregation auf unbekannte Haushalte

Die NILM-Ansätze müssen auch auf Haushalte anwendbar sein, von denen nicht bekannt ist, über welche Geräteausstattung sie verfügen. Insbesondere muss das Verfahren ohne die Verfügbarkeit von Gerätelastgängen des zu disaggregierenden Zielhaushalts anwendbar sein, auf die das Verfahren trainiert werden kann.

1.3.1 Forschungsfragen

Folgende Forschungsfragen liegen im Fokus dieser Arbeit:

1. In welchem Maße generalisieren Disaggregationsverfahren auf unbekannte Haushalte?

Durch eine Analyse und Evaluation bestehender Deep-Learning-Ansätze im Rahmen von Disaggregationsexperimenten sollen Fähigkeiten und Limitierungen bekannter Verfahren ermittelt sowie qualitativ begutachtet werden, um darauf aufbauend Ansatzpunkte für Verbesserungsmöglichkeiten zu identifizieren. Ein Schwerpunkt liegt dabei auf der Fragestellung, welche Qualität die Verfahren anhand ausgewählter Geräteklassen bei unbekanntem Haushalten bieten.

2. Wie können bekannte Disaggregationsverfahren weiterentwickelt werden, um die Disaggregationsqualität auch auf unbekanntem Haushalten zu verbessern?

Auf der Basis der ersten Forschungsfrage werden Ansätze zur Optimierung der Disaggregationsqualität entwickelt und evaluiert. Dabei wird auf Deep-Learning-Ansätze zurückgegriffen, die sich in anderen Anwendungsdomänen wie Bild- oder Sprachverarbeitung bewährt haben.

2a. Wie kann sichergestellt werden, dass Ergebnisse bei der Disaggregation ausschließlich realisierbare Lastgänge enthalten und welchen Einfluss hat diese Strategie auf die Qualität der Disaggregation?

Ein identifiziertes Problem bei der Disaggregation betrifft die Plausibilität von Zwischenergebnissen. Oft wird ein Lastgang geschätzt, der eindeutig nicht von einem realen Gerät gemessen worden sein konnte. Es werden Strategien erarbeitet, diese Plausibilität zu erreichen und evaluiert, ob dies ausreicht, um die Disaggregationsqualität zu erhöhen.

2b. Kann das U-Net-Konzept die Disaggregationsqualität verbessern?

Die U-Net-Architektur ist ein vielversprechendes Muster für den Entwurf von neuronalen Netzen. Während bei Feed-Forward-Netzen alle Informationen über alle Schichten hinweg transportiert werden müssen, teilt sich beim U-Net der Informationspfad auf in niedrig aufgelöste Detailinformationen und abstraktere und großräumigere Merkmale. Genau dieses Muster erscheint auch für die Disaggregation sinnvoll, da eine Unterscheidung zwischen diesen beiden Kategorien vorstellbar ist.

1.3.2 Beiträge

Im Rahmen dieser Arbeit werden drei neue Ansätze zur Disaggregation von Haushaltsenergiedaten entwickelt und evaluiert. Dem liegt zunächst eine Analyse der dem Stand der Technik entsprechenden Disaggregationsverfahren zu Grunde. Über die Durchführung von Vergleichsexperimenten werden Disaggregationsprobleme offengelegt und weiterer Forschungsbedarf aufgezeigt. Um die gefundenen Herausforderungen zu adressieren, werden für diese Anwendung Deep-Learning-Ansätze adaptiert, weiterentwickelt sowie deren Optimierungspotential evaluiert und Limitierungen aufgezeigt. Hieraus ergeben sich zudem relevante Erkenntnisse insbesondere in Bezug auf Datenerfassung und Datenaufbereitung, die bei Durchführung und Design künftiger Forschungsprojekte berücksichtigt werden sollten.

1.3.3 Abgrenzung

Diese Arbeit fokussiert zunächst auf Haushalte als reine Konsumenten von elektrischer Energie. Dabei werden gegebenenfalls vorhandene Erzeugungssysteme als außerhalb des Haushaltsstromkreises und getrennt gemessen angenommen. Ohne eine solche Annahme könnte die erzeugte Last die Lastsignatur einiger Geräte signifikant verändern bis hin zur völligen Auslöschung.

Die in dieser Arbeit untersuchten Disaggregationsverfahren beschränken sich auf solche, die auf Deep Learning basieren. Deep Learning bezieht sich auf Methoden des maschinellen Lernens, welche neuronale Netze mit vielen Schichten einsetzen. Diese Informationsverarbeitungsmethoden entsprechen dem Stand der Technik.

1.4 Entstehungskontext

Diese Dissertation wurde im Rahmen von Projekten des Kompetenzzentrums für Angewandte Sicherheitstechnologien (KASTEL) erstellt. KASTEL ist eines von drei IT-Sicherheitskompetenzzentren, die von dem Bundesministerium für Bildung und Forschung seit 2011 gefördert werden.

Das in diesem Zusammenhang bearbeitete Themenspektrum lässt sich an der Schnittstelle zwischen den Herausforderungen der Transformation des Energiesystems durch Digitalisierung, Integration erneuerbarer Energien und Elektromobilität sowie der Erforschung angewandter Sicherheitstechnologien im Hinblick auf Security und Privacy lokalisieren. In diesem Zusammenhang entstanden inhaltlich begleitende Arbeiten, die allerdings von der thematischen Bandbreite den Rahmen einer Dissertation sprengen würden. Daher soll

an dieser Stelle nur kurz angerissen werden, welche Forschungsfragen neben der im Detail untersuchten Optimierung von NILM-Verfahren in diesem Themenkontext betrachtet wurden.

1.4.1 Energiedatenanalyse und Privatsphärenschutz

Wird die Disaggregationsqualität von NILM-Ansätzen weiter verbessert, können diese im Rahmen der Energieverbrauchsvisualisierung einerseits die Energieoptimierung der Haushaltsbewohner unterstützen, andererseits aber auch missbraucht werden, um Einblicke in des Privatleben der Bewohner zu erlangen wie beispielsweise den Tagesablauf der Bewohner oder die von ihnen genutzten Gerätetypen abzuschätzen. Ebenfalls können die Smart-Meter-Daten zur Verbesserung der Energielastprognosen genutzt werden und somit zur Energiewende beitragen und gleichzeitig durch die Verfügbarkeit aussagekräftiger Daten Privatsphärenrisiken begründen. Daher sollte untersucht werden, welche Risiken für die Privatsphäre mit der Bereitstellung von Haushaltsenergiemessdaten gegenüber Energieversorgern oder Dienstleistern verbunden sein können und wie diese minimiert werden können.

Der Einfluss von Differential-Privacy-Garantien auf Energielastvorhersagen

Das Ausmaß der Verletzung der Privatsphäre variiert je nach der zeitlichen Auflösung und der Menge an offengelegten Messdaten (Eibl und Engel, 2014). Beim Einsatz von Differential Privacy (Dwork, 2006) können sowohl die zeitliche Granularität der Messdaten als auch ein individuelles Schutzlevel zur Berechnung der Privatsphärenbeeinträchtigung verwendet werden. In der Forschungskoooperation von Eibl, Bao et al. (2018) wurde gezeigt, wie die Umsetzung einer privatsphärenschonenden Lastprognose mit Differential-Privacy-Garantien aussehen kann und wie dabei die Schutzgarantie mit dem Nutzen der offengelegten privaten Daten abgewogen werden kann. In dieser Arbeit wurden folgende Aspekte beleuchtet:

- Es wurden unterschiedliche Lastvorhersagemodelle untersucht, mit dem Ergebnis, dass Smart-Meter-Daten nicht in jedem Fall die Energievorhersage verbessern.
- Sofern die Smart-Meter-Daten die Vorhersage verbessern, ermöglicht Differential Privacy ein Abwägen zwischen Nutzen der Datenanalyse und Privatsphärenverlust.
- Es wurde gezeigt, dass für einen Prognoseansatz zur Lastvorhersage ein vernünftiger Nutzen und gleichzeitig eine starke Datenschutzgarantie erreicht werden kann.

Die Arbeit war für die Best-Paper-Auszeichnung der Konferenz Energieinformatik 2018 nominiert.

Privatsphärengerechte Energiedatenvisualisierung

Eine Herausforderung der Energievisualisierung in Smart-Home-Szenarios besteht insbesondere bei gemeinsam genutzten Gebäuden, wobei Mitbewohnende bzw. Mitnutzende keine Rückschlüsse auf das Verhalten anderer durch Analyse der Energiedaten ziehen können sollen. Lösungsoptionen werden in der gemeinsamen Kurzarbeit von Bao, Bräuchle et al. (2015) beschrieben. Die Arbeit wurde mit dem Best Poster Award ausgezeichnet.

1.4.2 Energiedatenanalyse und IT-Sicherheit

Softwaresicherheit von Energiesystemen

Ziel der von Landwehr und Valdes (2017) geleiteten Initiative internationaler Expert*innen ist es, Grundlagen für die Minimierung des Risikos böswilliger Angriffe auf das Energieversorgungssystem zu bieten, die den Systembetrieb behindern oder die Integrität oder Vertraulichkeit der vom System verwendeten oder erzeugten Daten beeinträchtigen könnten. Im Rahmen dieser Zusammenarbeit hat sich der Autor u. a. für die Verwendung sichererer Programmiersprachen wie Ada, Spark und Rust eingesetzt. Der Bericht fungiert als Exempel, wie eine Art Bauvorschriften für die Konstruktion von Software für Energiesysteme entwickelt werden sollten und hält dabei den Konsens einer Gruppe erfahrener Personen aus Industrie, Wissenschaft und staatlichen Stellen fest, die sich mit der Sicherheit zukünftiger Energiesysteme befassen.

Bedrohungsanalyse des ISO 15118-Kommunikationsprotokolls

Die internationale Norm ISO 15118 beschreibt die Schnittstelle zwischen Elektromobil und Ladepunkt. In der gemeinsamen Arbeit von Bao, Valev et al. (2017) wurde eine Sicherheitsanalyse durchgeführt und verschiedene Szenarien vorgestellt, in denen die Verfügbarkeit der Ladeleistung sowie die Integrität, Authentizität oder Vertraulichkeit der Kommunikation auf Protokollebene beeinträchtigt werden können.

1.4.3 Smarter Energy

Intelligente Infrastruktur für Energiemanagementsysteme im Smart Building

In den Arbeiten von Bao, Fiebig et al. (2016) und Bao, Mauser et al. (2016) wird beschrieben, wie die Kombination eines Gebäude-Energiemanagementsystems mit dem Internet der Dinge prototypisch in einem realen Gebäude umgesetzt werden kann, um komplexe Dienste wie privatsphärengerechte Energiedatenvisualisierung und automatisiertes Energiemanagement umzusetzen.

Im Rahmen der Bereinigung der Forschungsdaten für diese Arbeit wurde die Notwendigkeit der Entwicklung einer stabilen Messinfrastruktur offenbar, welche sodann durch die Kooperation mit weiteren Forschungsprojekten sowie Kollegen und Studierenden umgesetzt wurde.

1.4.4 Modellierung von Gerätelastgängen mit künstlichen neuronalen Netzen

Abbildung der Flexibilität von regenerativen Energieerzeugungssystemen

In der gemeinsamen Arbeit von Förderer et al. (2018) wurde die Flexibilität verteilter Energieressourcen mit Hilfe künstlicher neuronaler Netze zur Erstellung realisierbarer Lastprofile modelliert. Hierfür wurden verschiedene Typen und Topologien von künstlichen neuronalen Netzen für entsprechende Realisierungsmuster und unterschiedliche Gerätekonfigurationen untersucht. Im Ergebnis konnte sich keine der untersuchten Technologien als beste Lösung durchsetzen. Stattdessen muss für jedes Muster und jede Gerätekonfiguration eine geeignete individuelle Topologie gefunden werden. Dies wurde am Exempel der Auswertung eines klassifikationsbasierten Musters unter Verwendung von Daten einer realen Kraft-Wärme-Kopplungsanlage in einem intelligenten Gebäude veranschaulicht.

Der Autor half bei der Entwicklung der Idee, entwickelte die Implementierung der neuronalen Netzmodelle und führte die dazugehörigen Experimente durch. Insofern wurde die in dieser Arbeit fortgeführte Idee aufgegriffen, Lastgänge mithilfe von künstlichen neuronalen Netzen zu modellieren.

Generator für Gerätelastgänge auf Basis von Generative Adversarial Networks

Im Rahmen des ersten Ansatzes zur Optimierung der Disaggregationsqualität (siehe Abschnitt 6.1) wird das Konzept weiter ausgearbeitet, den Generator eines Generative Adversarial Network (GAN) zur Gerätelastgangserzeugung zu nutzen, um ungenaue Disaggregations-schätzungen zu reparieren. Diese Idee wird bereits in einer gemeinen Arbeit von Bao, Ibrahimov et al. (2018) beschrieben und in ersten Experimenten erprobt. In der Veröffentlichung wurde gezeigt, dass unter speziellen Trainingsbedingungen Lastgänge der exemplarisch gewählten Geräteklasse Waschmaschine reproduziert werden können. Jedoch gelingt es nicht, mit einem Lastgangsgenerator einen Disaggregationsansatz zu erstellen, der in allen Testgebäuden genauere Lastgängeschätzungen durchführen kann.

Die offene Frage, ob die unzureichende Genauigkeit möglicherweise vom Unvermögen des Generators stammt, genau die Lastgänge aus dem Testdatensatz zu reproduzieren, wird erst in dieser Arbeit in Abschnitt 6.1.7 beantwortet. Hierbei wird die Vermutung bekräftigt,

dass die ungenauen Schätzungen von einer nicht klar erkannten Geräteaktivität stammen.

1.5 Gliederung und Vorgehensweise

Zur Darstellung des Stands der Technik werden in Kapitel 2 zunächst die Grundlagen der Haushaltsenergiemessung, der Disaggregation von Haushaltsenergiemessdaten, eine Einordnung klassischer Disaggregationsansätze sowie die Rolle der künstlichen neuronalen Netze erläutert. Das Kapitel wird um einen Exkurs über die Privatsphärenbedrohung durch Energiedatenüberwachung ergänzt, um den gesellschaftlichen Kontext zu beleuchten, in welchem NILM zu verorten ist. So sollen ambivalente Aspekte dieses technologischen Fortschritts aufgezeigt werden, indem Chancen und Risiken der NILM-Optimierung in dieser Arbeit gleichermaßen Beachtung finden.

In Kapitel 3 werden zunächst bekannte NILM-Verfahren vorgestellt, um diese im Anschluss zu verbessern. Da auf Deep Learning basierende Verfahren derzeit die erfolgversprechendsten Ansätze zur Disaggregation niedrig aufgelöster Haushaltsenergiemessdaten darstellen, konzentriert sich die Arbeit auf diese. Das Kapitel gibt einen Überblick über den Disaggregations- und Trainingsprozess unter Berücksichtigung der besonderen Eigenschaften eines Haushaltslastgangs und widmet sich den einzelnen Disaggregationsschritten wie der Aufteilung des Haushaltslastgangs in einzelne Sequenzen durch gleitende Zeitfenster, der Generierung von Trainingsbeispielen, der Vor- und Nachbearbeitung, der Wahl der Netzarchitektur bei bestehenden Ansätzen, der Datenausgabe, der Zusammenführungsstrategie überlappender Teilergebnisse sowie der Trainingsverfahren und -parameter. Zum Abschluss erfolgt die Bestimmung des aktuellen Stands der Technik, um die im Rahmen dieser Arbeit evaluierten Ansätze auszuwählen.

Bevor in Kapitel 5 die Analyse der ausgewählten Ansätze erfolgen kann, muss in Kapitel 4 zunächst das Szenario und der Aufbau der Experimente beschrieben werden. Daneben werden die am häufigsten verwendeten Datensätze, die Forschungsprojekten zur Verfügung stehen und auch in dieser Arbeit verwendet wurden, beschrieben. Im Rahmen der Evaluation werden dann die Schätzungen bekannter Verfahren mithilfe des zuvor beschriebenen Experimentaufbaus qualitativ begutachtet, um Probleme und mögliche Ansatzpunkte für die Verbesserungen der Disaggregationsqualität zu identifizieren. Die Disaggregation wird dabei aufgeschlüsselt nach den ausgewählten Geräteklassen untersucht. Im Anschluss werden die dabei gewonnenen Erkenntnisse und der für die weitere Arbeit identifizierte Forschungsbedarf dargelegt.

Aufbauend auf diesen Schlussfolgerungen werden in Kapitel 6 drei Ansätze entwickelt und evaluiert, die die zuvor gezeigten Probleme bei der Disaggregation adressieren sollen. Mit Ansatz A in Abschnitt 6.1 wird versucht die Disaggregationsqualität durch eine Integration von

Lastprofilgeneratoren zu verbessern. Dabei basieren die Lastprofilgeneratoren auf dem Konzept GAN, die in die Disaggregationsphase eingebaut werden. Beim Ansatz B in Abschnitt 6.2 wird der Einsatz eines Diskriminators während des Trainings untersucht, der die Korrektheit der Disaggregation erlernt und dessen Bewertung als zusätzlicher Faktor in der Verlustfunktion des Generators/Disaggregators berücksichtigt wird. Über eine U-Net-Architektur, die als Weiterentwicklung des Dilated Fully Convolutional Network (siehe Abschnitt 3.6.5) konzipiert ist, wird im Ansatz C (Abschnitt 6.3) analysiert, wie der dabei implementierte Informationspfad die Disaggregation verbessern kann.

Die Arbeit schließt mit der Aufschlüsselung und Diskussion der Ergebnisse in Kapitel 7 sowie Zusammenfassung und Ausblick in Kapitel 8.

Ein Überblick über die wesentlichen Abschnitte dieser Arbeit und zugleich ein Ablaufdiagramm eines Disaggregationsexperiments ist in Abbildung 1.1 dargestellt.

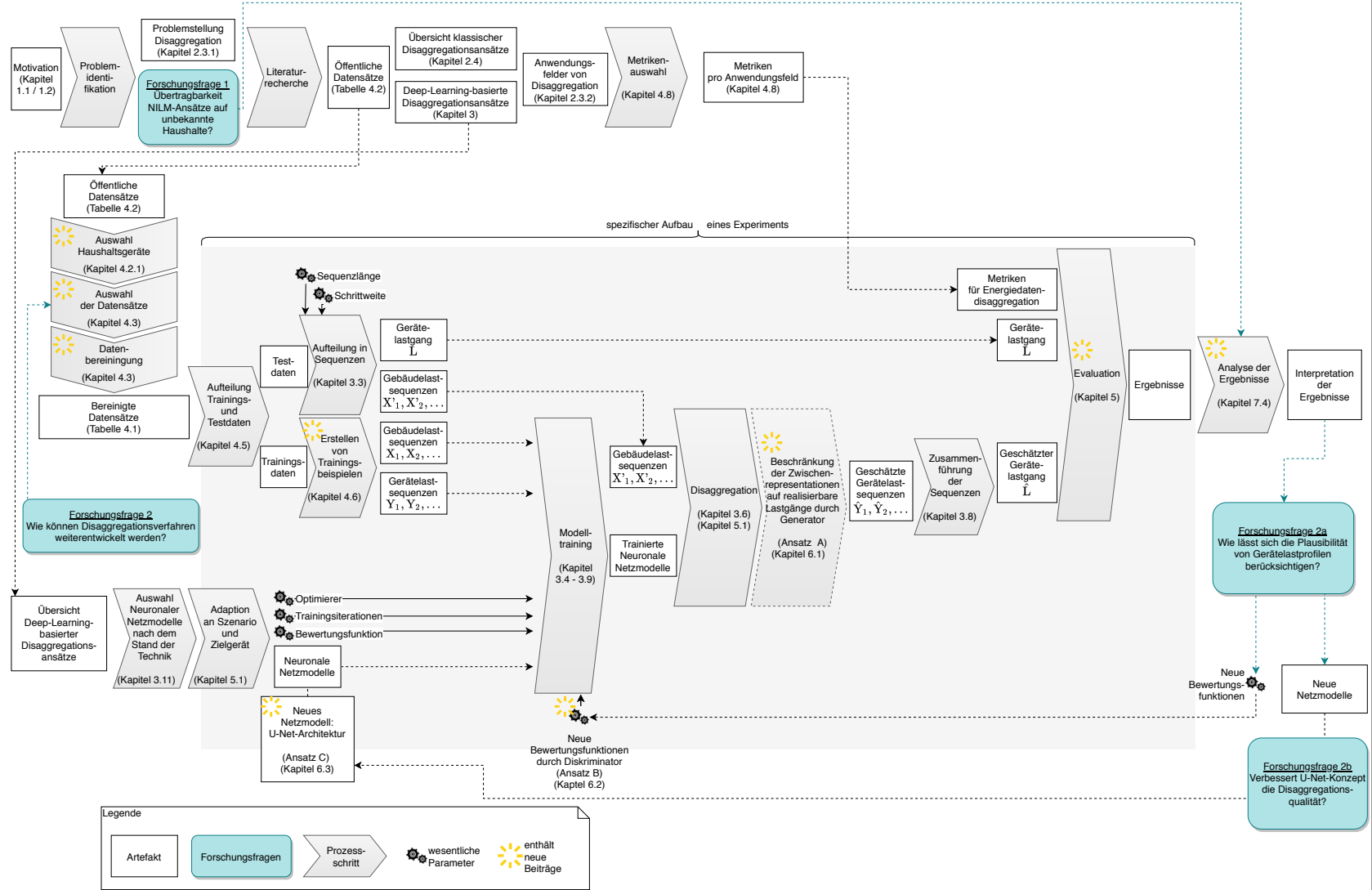


Abbildung 1.1: Einordnung des Aufbaus dieser Arbeit in den Disaggregationsprozess

Kapitel 2

Grundlagen

In diesem Kapitel werden die Grundbegriffe sowie der Kontext dieser Arbeit vorgestellt. Dies umfasst zunächst die Definition von Haushaltsenergiemessdaten und deren Verwendungszweck im Elektroenergiesystem (siehe Abschnitt 2.1), die Grundlagen der Energiedatendisaggregation (siehe Abschnitt 2.3) und einen Überblick über klassische Ansätze zur Disaggregation (siehe Abschnitt 2.4). Als Exkurs werden zusätzlich die Implikationen einer Energieverbrauchsüberwachung durch Messeinrichtungen für die Privatsphäre beschrieben (siehe Abschnitt 2.2), um auch die mit dem technologischen Potenzial verbundenen ethischen Dilemmata zu zeigen.

Das Kapitel wird mit einer kurzen Einführung in tiefe neuronale Netze und Deep Learning abgeschlossen, welche die Grundlage moderner Disaggregationsansätze darstellen (siehe Abschnitt 2.5).

2.1 Haushaltsenergiemessdaten

Weltweit werden zur Verteilung von elektrischer Energie an Privathaushalte Wechselstromnetze eingesetzt. Hierbei ist jeder Haushalt über einen Hausanschlusspunkt (engl. *mains connection*) an ein bis drei Phasen eines Drei-Phasen-Wechselstromnetzes mit dem Verteilnetz verbunden. Zur Rechnungsstellung der bezogenen und erzeugten Energie eines Anschlussnehmers dienen Messeinrichtungen (Stromzähler) am Netzanschlusspunkt.

Im Folgenden werden zunächst die wichtigsten Messgrößen am Netzanschlusspunkt genannt. Anschließend werden die aktuellen Verwendungszwecke dieser Messdaten vorgestellt.

2.1.1 Messgrößen in Wechselstromsystemen

Zur Abrechnung des Energieverbrauchs in Privathaushalten ist aktuell lediglich das Integral der Wirkleistung relevant. Für die Erkennung und Unterscheidung von Haushaltsgeräten bei der Disaggregation können jedoch weitere elektrische Messgrößen am Hausanschlusspunkt herangezogen werden. Die im Folgenden vorgestellten elektrischen Messgrößen bilden die Grundlage für die später in Abschnitt 2.4 beschriebenen Merkmale für Disaggregationsverfahren.

Momentan- bzw. Augenblickswerte von Spannung und Strom

An jeder Phase des Wechselstromnetzes liegt eine zeitlich veränderliche Spannung an und zwischen Haushalt und Stromnetz fließt ein zeitlich veränderlicher Strom. Zu einem Zeitpunkt t lassen sich dabei die Momentan- bzw. Augenblickswerte der Spannung $u(t)$ bzw. des Stroms $i(t)$ direkt messen. (*DIN 40110-1* 1994)

Eine digitale Messung von Spannung und Strom erfordert hohe Abtastraten, da insbesondere elektronische Geräte einen nichtlinearen Energieverbrauch mit hohen Frequenzanteilen im Stromverlauf verursachen können. (Lerch, 2016, Abschnitt 11.10)

Wirkleistung

Mit der Wirkleistung wird die mittelfristige (über eine oder mehrere Schwingungsperioden hinweg) übertragene elektrische Energie quantifiziert. Dazu muss berücksichtigt werden, dass Energie zwischen Netz und Haushalt kurzfristig hin- und herpendeln kann, ohne im Haushalt genutzt zu werden. Dies wird durch herkömmliche Haushaltsgeräte aufgrund ihrer induktiven oder kapazitiven Bauteile verursacht. (vgl. Lerch, 2016, Abschnitt 8.2)

Zur Bestimmung der Wirkleistung werden zunächst die Augenblickswerte der Leistung betrachtet. Die Augenblickswerte der Leistung werden wiederum über die Augenblickswerte von Spannung und Strom ($p(t) = u(t) \cdot i(t)$) berechnet oder über eine Multiplikationsschaltung direkt gemessen. Nach Festlegung des Bezugssystems ist dabei die augenblickliche Fließrichtung der Energie durch das Vorzeichen von $p(t)$ bestimmt.

Die Wirkleistung ist somit die mittlere Leistung über eine Periodendauer T_p (= 20 ms bei einer Netzfrequenz von 50 Hz) (*DIN 40110-1* 1994):

$$P(\tau) = \frac{1}{T_p} \int_{\tau-T_p}^{\tau} p(t) dt = \frac{1}{T_p} \int_{\tau-T_p}^{\tau} u(t) \cdot i(t) dt \quad . \quad (2.1)$$

Effektivwerte von Strom und Spannung

Für einen reinohmschen Verbraucher entsprechen die effektive Spannung U und effektive Stromstärke I in einem Wechselstromsystem genau der Spannung und Stromstärke in einem Gleichstromsystem, bei dem der Verbraucher die gleiche Wirkleistung abrufen (Mühl, 2014, Abschnitt 5.2). Die Effektivwerte können über eine Periodendauer T_p wie folgt bestimmt werden (Lerch, 2016, Abschnitt 13.4):

$$U(\tau) = \sqrt{\frac{1}{T_p} \int_{\tau-T_p}^{\tau} u^2(t) dt} \quad , \quad (2.2)$$

$$I(\tau) = \sqrt{\frac{1}{T_p} \int_{\tau-T_p}^{\tau} i^2(t) dt} \quad . \quad (2.3)$$

Schein- und Blindleistung

Das Produkt der Effektivwerte von Strom und Spannung ergibt die Scheinleistung S (DIN 40110-1 1994):

$$S(\tau) = U(\tau) \cdot I(\tau) \quad . \quad (2.4)$$

Zusammen mit der Wirkleistung P (siehe Gleichung 2.1) berechnet sich daraus die Gesamtblindleistung Q_{tot} (DIN 40110-1 1994):

$$Q_{\text{tot}}(\tau) = \sqrt{S^2(\tau) - P^2(\tau)} \quad . \quad (2.5)$$

Leistungsfaktor und $\cos \phi$

Der Leistungsfaktor λ beschreibt das Verhältnis des Wirkleistungsbetrags zur Scheinleistung (DIN 40110-1 1994):

$$\lambda = \frac{|P|}{S} \quad . \quad (2.6)$$

Bei einem idealen sinusförmigen Spannungs- und Stromverlauf ist der Phasenverschiebungswinkel ϕ zwischen Strom und Spannung definiert. In diesem Fall gilt für Wirk- und Blindleistung (DIN 40110-1 1994):

$$P = S \cdot \cos \phi \quad \text{und} \quad Q = S \cdot \sin \phi \quad . \quad (2.7)$$

Somit gilt im Fall eines sinusförmigen Spannungs- und Stromverlaufs, dass der Leistungsfaktor λ dem Betrag von $\cos \phi$ entspricht:

$$\lambda = \frac{|P|}{S} = \frac{|S \cdot \cos \phi|}{S} = |\cos \phi| \quad \text{für} \quad S > 0 \quad . \quad (2.8)$$

Die Größen ϕ und $\cos \phi$ sind im Allgemeinen für nichtsinusförmige Spannungs- und Stromverläufe nicht definiert. Messsysteme müssen für diese Messgrößen annehmen, dass zumindest eine sinusförmige Spannung vorliegt. Im Gegensatz dazu kann der Leistungsfaktor λ im Allgemeinen bestimmt werden (vgl. *DIN 40110-1* 1994).

Oberschwingungen in der Stromstärke

Insbesondere bei elektronischen Geräten ist der Augenblickswerteverlauf der Stromstärke $i(t)$ nicht sinusförmig. Durch eine Zerlegung der Stromkurve in Oberschwingungsanteile mittels Fouriertransformation ergeben sich Merkmale, die sich zur Differenzierung von Haushaltsgeräten eignen. Eine andere Möglichkeit zur Charakterisierung nichtlinearen Energieverbrauchs ist die Beschreibung der Strom/Spannungskennlinie $i(u)$. (Lam et al., 2007)

2.1.2 Haushaltsenergiemessdaten für Mess- und Abrechnungsprozesse des deutschen Energiesystems

Zur Erläuterung der Nutzungszwecke für Energiemessdaten im Rahmen des elektrischen Energienetzes sind in diesem Abschnitt die von Eibl, Bao et al. (2018) beschriebenen Zusammenhänge zusammengefasst.

Die wichtigsten Nutzungszwecke von Energiemessdaten eines Energiekunden, der Energie für den eigenen Verbrauch bezieht,¹ fallen zum einen in den Bereich Abrechnung von Energiekosten und zum anderen in den Bereich Bilanzkreismanagement (vgl. § 50 *MsbG* 2016).

Abrechnung von Energiekosten

Zur Abrechnung des Energieverbrauchs genügt bei den in Deutschland aktuell verbreiteten zeitunabhängigen Stromtarifen eine akkumulierte Energieverbrauchssumme bzw. ein Energiezählerstand pro Abrechnungsperiode (in der Regel ein Jahr) (§ 55 *MsbG* 2016). Bei dynamischen zeitabhängigen Tarifen mit mehreren Preisstufen wird in den Technischen Richtlinien für Smart Meter Gateways (*TR-03109* 2019) als datensparsame Variante die Führung mehrerer Energiemengenregister vorgesehen. Für den allgemeinen Fall und im Falle eines außergewöhnlich hohen Energieverbrauchs können auch Lastgänge mit einem Messintervall von 15 Minuten erfasst werden (§ 60 *MsbG* 2016).

¹Letztverbraucher im Sinne des § 2 Nr. 8 *MsbG* (2016)

Bilanzkreismanagement

Das Bilanzkreismanagement ist ein marktbasierter Mechanismus, welcher zum Leistungsgleichgewicht zwischen Energieproduktion und -verbrauch im Elektroenergiesystem beiträgt. Dieses Gleichgewicht von Ein- und Ausspeisung ist für die Stabilität und somit zum Betrieb des Energiesystems notwendig. Ein Ungleichgewicht im Wechselstromnetz drückt sich in einer Frequenzabweichung von der Nennfrequenz aus. (FERC, 2015)

Mit dem Konzept des Bilanzkreismanagements wird jeder Energiemarktteilnehmer zu einer ausgeglichenen Ein- und Ausspeisung (mit Ausnahme geplanter Austauschenergiemengen) in seinem Bilanzkreis verpflichtet. Dies geschieht im Vorfeld, bevor Energieflüsse physikalisch realisiert werden. In einem Bilanzkreis wird für jedes Messintervall (in Deutschland aktuell 15 Minuten) über Stromproduktion und -verbrauch Buch geführt. Die Energieflüsse werden prognostiziert und ausgeglichen. Im Falle perfekter Prognosen und ausgeglichenen Bilanzkreisen ist das Stromnetz auch physikalisch ausgeglichen. Prognosefehler und schwankende Energieflüsse innerhalb eines Messintervalls sorgen jedoch für Abweichungen, die mit Ausgleichsenergie (auch Regelenergie) korrigiert werden.

Da Bilanzkreise eine virtuelle Zusammenlegung von Netzanschlusspunkten darstellen, welche sich nicht zwangsläufig mit einer physikalischen Netzstruktur deckt, kann ohne flächendeckendes Smart Metering der tatsächliche Summenlastgang des Bilanzkreises lediglich geschätzt werden (siehe § 1.2 *MaBiS* 2011). Allerdings ist für ein effizientes Energiesystem, in dem die vollständigen Betriebskosten der Energieversorgung bepreist sind, eine genaue Erfassung des Summenlastgangs notwendig. Andernfalls werden aufgrund fehlender Information Betriebskosten unspezifisch aufgeteilt, welches ggf. keinen ausreichenden Kostendruck auf spezifisches effizientes Verhalten erzeugt.

Wie im nächsten Abschnitt detaillierter erläutert wird, bedroht die Energieverbrauchserfassung die individuelle Privatsphäre, weshalb das Ausmaß der Energieverbrauchserfassung mit dem Schutz der Privatsphäre abgewogen werden muss. Das deutsche Messstellenbetriebsgesetz (MsbG) sieht daher die Erfassung und Übermittlung von Haushaltslastgängen erst ab einem Jahresverbrauch von über 10.000 kWh vor (§§ 55,60 *MsbG* 2016).

2.2 Privatsphärenbedrohung durch Energiedatenüberwachung

Die Verfügbarkeit von kostengünstigen Smart Meter (intelligente Messeinrichtungen) ermöglicht technisch eine massenhafte Überwachung von Haushaltsenergieverbräuchen, was prinzipiell weitreichende Folgen für die Privatsphäre haben kann. Zur Beurteilung der Privatsphärenbedrohung ist einerseits zu klären, welche personenbezogenen Informationen sich im Energieverbrauch widerspiegeln und mit welchen Methoden, welchem Aufwand

und welcher Genauigkeit diese Informationen aus den Energieverbrauchsdaten ermittelbar sind. Andererseits ist auch aus rechtlicher Perspektive zu bewerten, zu welchen Zwecken und in welchem Interesse die gesammelten Energiedaten verwendet werden dürfen. (vgl. McDaniel und McLaughlin, 2009)

In dieser Arbeit werden in den folgenden Unterabschnitten nur die zuerst genannten technischen Aspekte umrissen. Bezüglich des rechtlichen Kontextes in Deutschland sei lediglich darauf hingewiesen, dass im internationalen Vergleich in Deutschland ein strenges Datenschutzregime für Energiemessdaten implementiert wird. Dies äußert sich u. a. in dem Schutzprofil (*BSI-CC-PP-0073* 2014) und den Technischen Richtlinien für Smart Meter Gateways (*TR-03109* 2019) sowie dem MsbG, welche die technische und rechtliche Nutzung von Energieverbrauchsdaten einschränken.

2.2.1 Personenbezogene Informationen in Haushaltsenergiemessdaten

In der einschlägigen Literatur ist mit Experimenten gezeigt worden, dass sich insbesondere folgende personenbezogenen Informationen im Energieverbrauch widerspiegeln.

Anwesenheit Bei Versuchen von Lisovich et al. (2010), Molina-Markham et al. (2010) sowie Kleiminger et al. (2013) konnten die Autoren einen Zusammenhang zwischen der Anwesenheit von Bewohnern und dem Haushaltsenergieverbrauch zeigen.

Schlafzyklus Lisovich et al. (2010) demonstrieren zusätzlich die Möglichkeit, aus dem Verlauf des Haushaltsenergieverbrauchs zu schätzen, ob Bewohner schlafen oder wach sind.

Gerätenutzung Basierend auf Ansätzen für NILM lässt sich aus dem Haushaltsenergieverbrauch bestimmen, wann, wie lange und wie oft Haushaltsgeräte genutzt werden (siehe Abschnitt 2.3).

(Pseudo-)Identität Buchmann et al. (2013) zeigen in ihren Versuchen, dass einzelne Haushalte in einer Gruppe von 180 Haushalten anhand ihres Energieverbrauchs mit einer Genauigkeit von 68 % wiederidentifizierbar sind.

Haushaltscharakteristika Beckel, Sadamori et al. (2014) beschreiben die Möglichkeit, mithilfe von Merkmalen u. a. aus dem niedrig aufgelösten Haushaltsverbrauch personenbezogene Informationen wie z. B. Familienstand, Beschäftigungsstatus und Altersgruppe zu schätzen.

Konsum multimedialer Inhalte Greveler et al. (2012) zeigen in einem Laborversuch, wie anhand einer Lastgangmessung mit einem Messintervall von 2 Sekunden der auf einem Fernseher dargestellte Inhalt ermittelt werden kann.

2.2.2 Privatsphärenbedrohungen durch Energieverbrauchserfassung

Aus der Möglichkeit, die vorher genannten personenbezogenen Informationen aus dem Energieverbrauch schätzen zu können, entsteht nach Quinn (2009) eine Reihe von Privatsphärenbedrohungen:

Sozialer Druck In Zeiten des Klimawandels weist Quinn (2009) auf einen möglichen Wandel in der gesellschaftlichen Wahrnehmung von Energieverbrauch, Konsumverhalten und des dadurch verursachten CO₂-Ausstoßes hin. Damit können erhöhte Energieverbrauchswerte an sich zu einem Datum werden, das ein Betroffener geheim halten möchte.

Diskriminierung bei Finanz- und Versicherungsdienstleistungen Versicherungstarife werden im Rahmen des Risikomanagements an bestimmte Kundengruppen angepasst. Quinn (2009) weist darauf hin, dass personenbezogene Informationen über Schlafmangel, pünktliche Abfahrt zur Arbeit oder unbeaufsichtigte Haushaltsgerätenutzung potenziell aus dem Energieverbrauch abgelesen und zur Einstufung für Versicherungstarife genutzt werden könnten. Auch die Nutzung von elektrischen Sportgeräten könnte Krankenversicherungen interessieren. Dies führt zu einer Diskriminierung von Kunden, die solche Informationen nicht preisgeben wollen.

Weiterreichende Diskriminierung Über das Nutzungsverhalten oder das Konsumverhalten im Haushalt können auch Rückschlüsse auf eine kulturelle, religiöse oder sexuelle Zugehörigkeit gezogen werden. Diese Informationen können für eine Diskriminierung von Kunden- oder Bevölkerungsgruppen missbraucht werden. (vgl. Beckel, Sadamori et al., 2014; Quinn, 2009)

Zielgerichtete Vermarktung Der Bedarf an bestimmten Produkten kann möglicherweise anhand des Energieverbrauchs abgelesen werden, beispielsweise durch Erkennung von defekten oder veralteten (Haushalts-)geräten, Gewohnheiten bei der Essenszubereitung, oder der Nutzungshäufigkeit von Unterhaltungselektronik. Dies ermöglicht einen neuen Markt für zielgerichtete Werbemaßnahmen. (Lisovich et al., 2010; Quinn, 2009)

Strafverfolgung Für die Strafverfolgung können beispielsweise aus dem Energieverbrauch geschätzte Anwesenheitsdaten weitere Indizien für die Ermittlung von Straftaten liefern. (Quinn, 2009)

Erleichtern von Einbrüchen Eine häufig genannte Privatsphärenbedrohung ist die Bestimmung der Anwesenheit von Bewohnern basierend auf dem Energieverbrauch. Damit besteht die Bedrohung, dass Kriminelle längere Abwesenheiten für Einbrüche nutzen könnten. (Quinn, 2009)

2.2.3 Ansätze zum Privatsphärenschutz

Zum Schutz der Privatsphäre wird in der einschlägigen Literatur auf die folgenden drei Prinzipien verwiesen, die jeweils ein spezifisches Anwendungsfeld besitzen. Für einen effektiven Schutz müssen unter Umständen mehrere Prinzipien angewandt werden.

Verstecken in der Menge

Bei dem Prinzip des Versteckens in der Menge werden Daten so erfasst oder modifiziert, dass diese möglichst wenig Informationen über einzelne Individuen enthalten. Gleichzeitig soll die Möglichkeit nicht eingeschränkt werden, mit diesen Daten Aussagen über die gesamte repräsentierte Population zu treffen.

Für die Umsetzung dieses Prinzips wurden die Eigenschaften k -Anonymität und deren Weiterentwicklungen l -Diversität und t -Closeness für Datensätze eingeführt. Diesen Eigenschaften liegt jeweils das Modell zugrunde, dass Individuen in einer Datenbank durch Einträge repräsentiert werden. Dabei bestehen Einträge aus identifizierenden Attributen (*identifier* bzw. *pseudo identifier*) und heiklen Attributen (*sensitive attributes*).

k -Anonymität soll vor der Identifizierbarkeit von Einträgen in Datensätzen schützen, indem von den identifizierenden Merkmalen jedes Eintrags gefordert wird, dass mindestens $k - 1$ weitere Einträge mit denselben identifizierenden Merkmalen im Datensatz existieren. (Sweeney, 2002)

l -Diversität verfolgt das Ziel, die Offenlegung von heiklen Informationen über Individuen trotz Erfüllung der k -Anonymitätseigenschaft zu verhindern. Eine Offenlegung geschieht beispielsweise dann, wenn alle k Einträge die gleichen heiklen Daten enthalten. Die Offenlegung kann erschwert werden, indem von den heiklen Daten gefordert wird, dass diese in mindestens l unterschiedlichen Ausprägungen vorkommen. (Machanavajjhala et al., 2006)

t -Closeness verfolgt ebenfalls das Ziel, die Offenlegung von heiklen Informationen trotz Einhaltung der k -Anonymitätseigenschaft zu verhindern. Dies wird mit der Forderung erreicht, dass sich die Verteilung heikler Daten im gesamten Datensatz nicht stark (über den Grenzwert t quantifiziert) von der Datenverteilung innerhalb der k -anonymen Teilgruppen unterscheiden darf. (Li, Li et al., 2007)

Alle drei Begriffe beschreiben hierbei statistische Eigenschaften eines Datensatzes, aus denen jedoch keine unmittelbare Privatsphärengarantie für ein Individuum folgt.

Eine individuelle Privatsphärengarantie bietet erst **Differential Privacy**, welches als „Gold-Standard unter den Privatsphärenbegriffen“ angesehen wird (vgl. Bambauer et al., 2014). Diese Garantie wird erreicht, indem bei Differential Privacy Eigenschaften an Abfrageergebnisse eines Datensatzes geknüpft werden. Dieses Vorgehen unterscheidet sich von den

vorherigen Privatsphärenbegriffen, die Forderungen an die Beschaffenheit eines Datensatzes stellen. Für Differential Privacy müssen die Abfrageergebnisse so randomisiert sein, dass für jedes Individuum eine Unsicherheit besteht, ob das Abfrageergebnis von einem Datensatz stammt, in dem das Individuum vertreten ist oder aus einem Datensatz stammt, in dem die Daten des Individuums nicht enthalten sind. Der Grad der Unsicherheit wird über den Parameter ϵ festgelegt. Gegeben ϵ hängt der Grad der Randomisierung schließlich davon ab, wie stark ein Individuum das Abfrageergebnis beeinflussen kann (Sensitivität). (Dwork, 2006)

Theoretische Untersuchungen zu Differential Privacy machen deutlich, dass der Privatsphärenschutz und die Nützlichkeit der gesammelten Daten diametral entgegengesetzte Eigenschaften sind, die nicht gleichzeitig vollumfänglich erreichbar sind. Mit dem Parameter ϵ kann dabei zwischen Schutz und Nützlichkeit quantitativ abgewogen werden. (Dwork, 2008; Dwork und Roth, 2013)

Das Prinzip des Versteckens in der Menge wurde im Kontext von Energiemessdaten mit Lastdatenaggregation und mit Lastdatenvermischung umgesetzt. Lösungen mithilfe von Aggregation von Lastdaten über mehrere Haushalte existieren von Garcia und Jacobs (2011), Kursawe et al. (2011) und Li, Luo et al. (2010). Mischverfahren werden in Arbeiten von Finster (2013) und Finster und Baumgart (2014) verwendet. Bei den Verfahren hilft eine additive homomorphe Verschlüsselung (Paillier, 1999) bei dem Schutz vor einer Offenlegung von heiklen Informationen während der Aggregation.

Diese Ansätze garantieren zwar k -Anonymität, für eine Differential-Privacy-Privatsphärengarantie ist jedoch zusätzlich die Verfälschung der Summenlast mit einem randomisierten Differential-Privacy-Mechanismus notwendig. Ács und Castelluccia (2011) stellen ein derartiges Verfahren vor, bei dem mit einem Laplace-Mechanismus (siehe Dwork und Roth, 2013) Rauschen auf die Summenlast addiert wird. Basierend darauf wird in der Arbeit von Eibl, Bao et al. (2018) untersucht, ob durch ein solches Verfahren erfasste Lastdaten noch nützlich genug für den Zweck der Lastvorhersage in einem Bilanzkreis bleiben. Diese Studie betrachtet zunächst einen Bilanzkreis als eine Zusammensetzung von geographisch benachbarten Zonen, wobei die Zonenlastgänge über eine Differential-Privacy-konforme Aggregation von Smart-Meter-Daten ermittelt werden. Daraufhin wird die Vorhersagegenauigkeit für den gesamten Bilanzkreis mit der Genauigkeit der kombinierten Vorhersage einzelner Zonen verglichen. Dabei wurde empirisch gezeigt, dass die Genauigkeit bestimmter Vorhersageansätze trotz starker Schutzgarantie durch Zonenlastgänge verbessert werden kann.

Reduktion der zeitlichen Auflösung

Verstecken in der Menge ist keine Option, wenn eine Identitätszuweisung der erfassten Daten erforderlich ist, beispielsweise bei der Stromkostenabrechnung. Das Prinzip der

Datenminimierung entspricht der Reduktion der zeitlichen Auflösung, welches auch vom Bundesamt für Sicherheit in der Informationstechnik (BSI) mit den Technischen Richtlinien für Smart Meter Gateways (TR-03109 2019) verfolgt wird. Dabei wird mit der Wahl von Tarifprofilen vorgesehen, dass je nach Tarif der Verbrauch unterschiedlich stark zeitlich aggregiert werden kann. Das Tarifprofil mit der maximalen Auflösung sieht hierbei die Übermittlung von Lastgängen mit einem Messintervall von 15 Minuten vor. Unabhängig von der Datenübertragung an externe (Energie-)Marktteilnehmer steht es dem Smart Meter Gateway frei, den Haushaltsbewohnern Lastgänge mit höheren Auflösungen über das Haushaltsnetzwerk (engl. *home area network*) anzubieten.

Um eine Übertragung des Lastgangs zu vermeiden, schlagen Molina-Markham et al. (2010), Jawurek et al. (2011) sowie Rial und Danezis (2011) vor, die Stromkostenabrechnung auf Seite des Kunden durchzuführen. Dabei weist der Kunde durch Zero-Knowledge-Beweise nach, dass die Abrechnung korrekt ist.

Engel (2013) sieht ein Konzept vor, bei dem der Lastgang in einer Wavelet-basierten Darstellung verwendet wird. Hierdurch werden Lastgänge mit geringer Redundanz in unterschiedlichen Auflösungen selektiv für unterschiedliche Adressaten freigegeben. Alternativ besteht damit die Möglichkeit, Lastgänge in hohen Auflösungen im Streitfall offenzulegen.

Ein häufig in traditionellen Disaggregationsansätzen verwendeter Vorverarbeitungsschritt ist die Erkennung von Schaltereignissen. Dabei wird die Annahme getroffen, dass während eines Messintervalls nicht mehrere Schaltereignisse gleichzeitig geschehen. Mit einer Auflösungsreduktion werden jedoch gleichzeitige Schaltereignisse häufiger. Eibl und Engel (2014) quantifizieren in ihrer Arbeit, wie die Erkennung von Schaltereignissen mit zunehmender Auflösungsreduktion beeinträchtigt wird.

In der Arbeit von Rigoll (2017) ist die Auflösung ein wesentlicher Bestandteil bei der Quantifizierung eines Privatsphärenverlusts. Das Konzept sieht dabei eine aushandelbare Lastgangsauflösung vor, die ein externer Marktteilnehmer bei dem Kunden abfragt.

Verschleierung heikler Informationen im Lastgang

Alle zuvor dargelegten Privatsphärenschutzansätze erfordern Datenerfassungssysteme, über deren Einsatz ein Letztverbraucher nicht direkt entscheiden kann. Wenn ein Letztverbraucher den Schutz seiner Privatsphäre selbst in die Hand nehmen möchte, ist dies mit der Verschleierung von heiklen Informationen im Lastgang realisierbar.

So schlagen Kalogridis et al. (2010), McLaughlin et al. (2011), Varodayan und Khisti (2011) sowie Yang et al. (2012) vor, den Haushaltslastgang mit Hilfe von Batteriespeichern so zu modifizieren, dass der Energieverbrauch bestimmter Geräte nicht mehr erkennbar ist. Chen,

Irwin et al. (2014) schlagen die Steuerung eines Blockheizkraftwerks vor, um die Anwesenheitsinformation im Lastgang zu verschleiern. Egarter et al. (2014) nutzen zur Verschleierung von Lastgängen eine intelligente Steuerung von Haushaltsgeräten, welche über ein Gebäudeenergiemanagementsystem realisierbar ist.

2.3 Disaggregation von Haushaltsenergiemessdaten

Mit Non-Intrusive Appliance Load Monitoring (NILM) oder Energiedatendisaggregation werden Verfahren zur Überwachung des Energieverbrauchs von Geräten oder Anlagen bezeichnet, wozu keine Instrumentierung der zu überwachenden Geräte mit Messgeräten erforderlich ist. Stattdessen werden für die Informationsgewinnung Messungen an zentralen Verteilungsknoten wie dem Hausanschlusspunkt durchgeführt und algorithmisch ausgewertet. (vgl. Faustine et al., 2017; Pereira und Nunes, 2018)

Erste Veröffentlichungen zu Disaggregationsverfahren für Haushaltsenergiemessdaten stammen von G. W. Hart (1989, 1992), der mit diesen Arbeiten auch den Begriff *Non-Intrusive Load Monitoring* geprägt hat. Dabei bezieht sich *Non-Intrusive* auf die entfallende Notwendigkeit, für die Überwachung des Energieverbrauchs Messgeräte innerhalb des Haushalts installieren zu müssen. Durch die Verbreitung von Smart Metering wird die Erfassung von Haushaltslastgängen erleichtert. Dies ermöglicht die breite Anwendbarkeit von Disaggregationsverfahren.

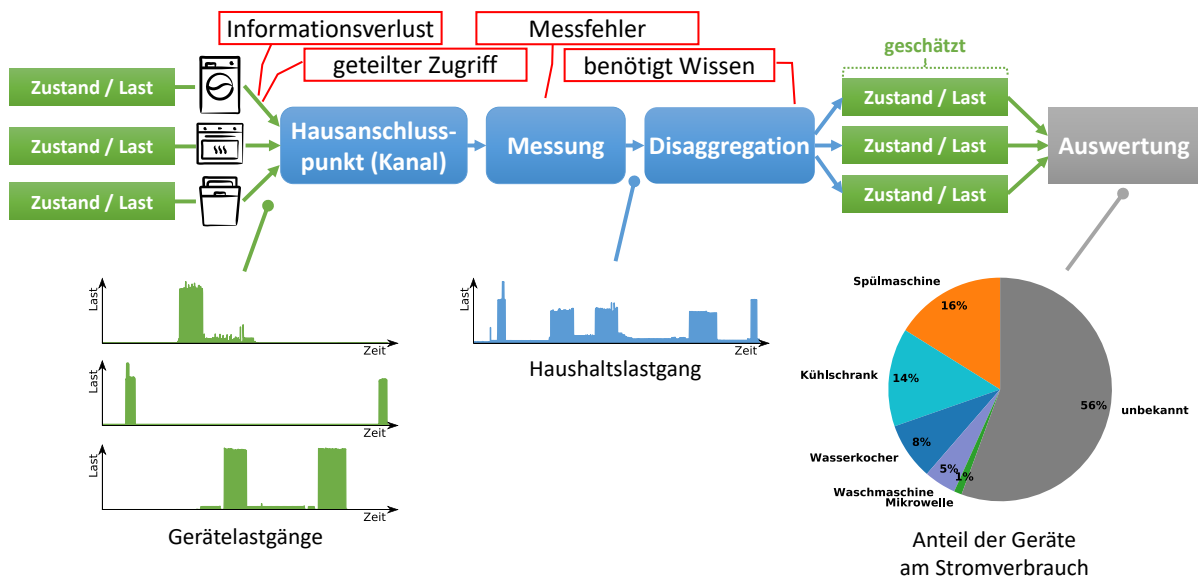


Abbildung 2.1: Non-Intrusive Load Monitoring aus nachrichtentechnischer Perspektive, basierend auf dem *communication model* von Hart (1992).

Wird, wie in Abbildung 2.1 dargestellt, der Aufbau von NILM aus einer nachrichtentechnischen Perspektive betrachtet, so sendet jedes Haushaltsgerät seinen Lastgang oder seine Zustandsfolge als Signal auf einen gemeinsamen Kanal, der hausinternen elektrischen Stromverteilung (Hart, 1992). Die Signale werden durch den gemeinsamen Zugriff aggregiert, sodass die Summenlast aller Geräte am Hausanschlusspunkt empfangen wird. Falls das Haus mit mehreren elektrischen Phasen an das Stromnetz verbunden ist, existieren mehrere Kanäle². Das Disaggregationsverfahren hat die Aufgabe, die einzelnen Signale aus der Messung des Hausanschlusses aufzutrennen und die Signale ihren ursprünglichen Quellen zuzuordnen.

Da die Lastgänge/Signale der Geräte nicht gezielt entworfen sind, besitzen sie in der Regel keine Eigenschaften wie beispielsweise die Einschränkung auf definierte Frequenz-, Zeit- oder Codebereiche, die eine Disaggregation erleichtern. Damit ist die Aufgabe der Disaggregation als unterbestimmtes Quellenseparationsproblem (engl. *source separation / signal separation*) zu sehen, bei dem viele Signale (Lastgänge von vielen elektrischen Geräten) über wenige Kanäle (Messgrößen multipliziert mit Phasenanzahl) übertragen werden (vgl. Parson, 2014, Abschnitt 2.2.3.2).

Ein bekanntes Quellenseparationsproblem ist die Behandlung des *Cocktail-Party-Effekts*, bei dem das Sprachsignal des Gesprächspartners aus der Geräuschkulisse der Party zu extrahieren gilt, oder die Auftrennung von Musikstücken in einzelne Instrumente. Hierzu werden in jüngeren Arbeiten vermehrt neuronale Netze eingesetzt (Chandna et al., 2017; Ephrat et al., 2018).

In diesem Abschnitt werden im Folgenden das NILM-Problem definiert und dessen Anwendungsfelder vorgestellt. Darauf folgt in Abschnitt 2.4 eine Vorstellung klassischer NILM-Ansätze, die nicht auf Deep-Learning-Prinzipien beruhen. In Kapitel 3 werden Deep-Learning-basierte NILM-Ansätze im Detail besprochen und eingeordnet.

2.3.1 Problemstellung

Das Ziel von NILM ist die Schätzung von Lastgängen $L_i(t)$ einzelner Geräte $i \in D$, wozu ausschließlich der am Gebäudeanschluss gemessene Summenlastgang $L(t)$ zur Verfügung steht. Dabei ist der Zusammenhang, wie in Gleichung 2.9 dargestellt, dass sich der gemessene Summenlastgang $L(t)$ im Messintervall $t \in T$ aus den Einzellasten $L_i(t)$ der zu disaggregierenden Geräte D , den Einzellasten $L_j(t)$ von weiteren unbekanntem Geräten U , sowie einem Messfehler $E(t)$ zusammensetzt:

$$L(t) = \sum_{i \in D} L_i(t) + \sum_{j \in U} L_j(t) + E(t) \quad \text{für } t \in T \quad . \quad (2.9)$$

²In der Regel sind dabei die Kanäle unabhängig, wenn keine mehrphasig angeschlossenen Großgeräte benutzt werden. Je nach Messgerät wird an der Schnittstelle u. U. nur die Summenlast zur Verfügung gestellt.

Mit der Aufteilung der Geräte in die Mengen D und U wird verdeutlicht, dass sich praktische Disaggregationsansätze auf eine Untermenge D aller Geräte $D \cup U$ beschränken. D. h. bei der Anwendung von Disaggregationsansätzen ist stets mit einer Menge von unbekanntem Geräten U zu rechnen.

Für Privathaushalte ist die Zielsetzung von NILM hauptsächlich wirtschaftlich motiviert und da nur der über Wirkleistung gemessene Energieverbrauch abgerechnet wird, ist in der Problemformulierung zunächst nur die Wirkleistung der Haushaltsgeräte als Zielgröße relevant.

Neben der Wirkleistung weitere Messgrößen wie Blindleistung oder harmonische Oberschwingungen als Eingabegrößen zu verwenden, ist für eine Disaggregation vorteilhaft, da diese zusätzliche Informationsquellen darstellen, die für die Unterscheidung von Haushaltsgeräten und damit für eine genauere Schätzung nützlich sind (vgl. Hart, 1992; Valenti et al., 2018). Für eine praktische Anwendbarkeit des Ansatzes in der Breite ist jedoch die Verfügbarkeit der Messgrößen zu betrachten, die von einem Smart Meter erfasst und übermittelt werden (Parson, 2014). Hierbei besteht der größte gemeinsame Nenner in der Wirkleistung. Dies äußert sich auch in der Verfügbarkeit der Messgrößen in den öffentlichen Datensätzen von Haushaltsenergiemessdaten (siehe Abschnitt 4.3). Auch wird der überwiegende Anteil an Deep-Learning-basierten Ansätzen (siehe Kapitel 3) auf Wirkleistungsdaten evaluiert. Die im Rahmen dieser Arbeit evaluierten Disaggregationsverfahren beschränken sich daher auf die Wirkleistung als Eingabegröße.

Da NILM ein breites Anwendungsfeld hat (siehe Abschnitt 2.3.2), existieren alternative Möglichkeiten, das Problem der Energiedatendisaggregation zu definieren. So ist beispielsweise die Schätzung der Gerätelastgänge kein notwendiger Zwischenschritt, wenn der Geräteaktivitätszustand bestimmt wird. Die hier verwendete Problemdefinition bietet jedoch durch die disaggregierten Gerätelastgänge eine Grundlage für sämtliche NILM-Anwendungsfälle und wird auch in der einschlägigen Literatur am häufigsten verwendet (vgl. Faustine et al., 2017; Nalmpantis und Vrakas, 2018; Zoha et al., 2012).

2.3.2 Anwendungsfelder

Sobald ein NILM-Verfahren aus dem Haushaltlastgang die einzelnen Gerätelastgänge disaggregieren kann, eröffnen sich folgende aus der Literatur bekannte Anwendungsfälle, die sich im Hinblick auf den Zweck und des erwarteten Ergebnisses der Disaggregation unterscheiden.

Von den in diesem Abschnitt im Folgenden aufgezählten Anwendungsfällen werden in dieser Studie ausschließlich die ersten drei näher betrachtet, da sich diese mit Metriken (siehe Abschnitt 4.8) direkt evaluieren lassen.

Schätzung des Gerätelastgangs Dieser Anwendungsfall entspricht der allgemeinen Problemstellung von NILM (siehe Abschnitt 2.3.1): Aus dem zeitlichen Verlauf des Haushaltslastgangs wird der Lastgang eines oder mehrerer Haushaltsgeräte geschätzt. Die Abweichung der geschätzten Leistungsaufnahme von der tatsächlichen bzw. gemessenen Leistung zählt als Fehler (vgl. Gleichung 2.9).

Auf Basis einer perfekten Lastgangschätzung sind die folgenden zwei Anwendungsfälle trivial bzw. mit einfachen Heuristiken umzusetzen. Da praktische Schätzungen stets fehlerbehaftet sind, führen direkte Ansätze ohne den Umweg über eine Lastgangschätzung in der Regel zu besseren Ergebnissen (vgl. Barsim und Yang, 2018).

Die Disaggregationsansätze in der Literatur zielen in der Regel auf diesen Anwendungsfall ab (siehe u. a. Bonfigli, Felicetti et al., 2018; Kelly und Knottenbelt, 2015a; Mauch und Yang, 2015).

Bestimmung der Geräteaktivität Ziel hierbei ist es, für jedes Messintervall zu bestimmen, ob das betrachtete Haushaltsgerät aktiv ist oder sich im Ruhezustand befindet. Dieser Anwendungsfall kann somit auch als Aneinanderreihung von binären Klassifikationsproblemen gesehen werden.

In der Regel korreliert die Geräteaktivität mit der Gerätenutzung und damit auch mit den Bewohneraktivitäten im Haushalt. Dadurch ist auch die Geräteaktivität als ein sensibles, personenbeziehbares Datum zu bewerten (Lisovich et al., 2010).

Ein Beispiel für die direkte Umsetzung dieses Anwendungsfalls findet sich in dem Disaggregationsansatz von Barsim und Yang (2018). Hierbei wird die Geräteaktivität direkt durch eine logistische Regression mit Hilfe eines neuronalen Netzes bestimmt, ohne eine Gerätelastgangsschätzung als Zwischenschritt explizit durchzuführen.

Identifikation von Energieverbrauchern Der Energiekunde profitiert von der Information, zu welchen Anteilen die jeweiligen elektrischen Haushaltsgeräte und -anlagen an dem Energieverbrauch und damit für die Energiekosten verantwortlich sind. Im Gegensatz zur aktuellen Praxis der Energieberatung, die lediglich auf statistischen Zusammenhängen beruht, werden datengetriebene, auf Energiedatendisaggregation basierende Verfahren eine wesentlich genauere Identifikation der Energieverbraucher in einem Haushalt ermöglichen. (Carrie Armel et al., 2013; Drenker und Kader, 1999)

In diesem Anwendungsfall ist die Bestimmung des anteiligen Energieverbrauchs je Gerät innerhalb einer gesamten Abrechnungsperiode ausreichend. Im Gegensatz zur Schätzung des Gerätelastgangs gleichen sich zeitweiliges Unter- und Überschätzen der Leistungsaufnahme aus.

Neben diesen drei Anwendungsfällen, die in dieser Arbeit evaluiert sind, werden in der Literatur folgende, weitergehende Anwendungsfelder für NILM identifiziert.

Diagnose von Geräteverschleiß und Gerätedefekt Ein Gerätestromgang ermöglicht Rückschlüsse über den Verschleiß- und Funktionszustand des Haushaltsgeräts. Da zunehmender Verschleiß oder ein Komponentendefekt von dem Benutzer unbemerkt zu erhöhtem Energieverbrauch führen können, bietet eine automatische Erkennung die Möglichkeit, überhöhten Energieverbrauch zu vermeiden. (vgl. Carrie Armel et al., 2013; Drenker und Kader, 1999)

Die Verfahren zur Energiedatendisaggregation beruhen jedoch meist darauf, dass Charakteristika der Geräte im Energieverbrauchsverhalten wiedererkannt werden. Da die Verfahren meist mit Daten von einwandfreien Geräten erstellt werden, sind diese Verfahren in der Regel nicht in der Lage, defekte Geräte basierend auf disaggregierten Daten zu erkennen (siehe Batra, Singh et al., 2015; Rashid, Singh et al., 2019; Rashid, Stankovic et al., 2019).

Gerätespezifische Stromtarife Drenker und Kader (1999) schlagen vor, NILM als Wegbereiter für gerätespezifische Stromtarife zu verwenden, indem der Energieverbrauch von Geräte(typen) unterschiedlich bepreist wird, möglicherweise in Verbindung mit bestimmten Nutzungszeiten.

Nachvollziehen ungewöhnlicher Energieverbrauchskosten Im Falle von Unstimmigkeiten bei der Stromkostenabrechnung erwähnen Drenker und Kader (1999), dass disaggregierte Energieverbrauchsdaten zur Schlichtung beitragen. Disaggregation hilft damit bei der Plausibilitätsprüfung des Energieverbrauchs.

Assistenzfunktionen auf Basis von Aktivität, Anwesenheit oder Schlafzyklus

NILM-Methoden eignen sich auch zur Schätzung der Aktivität, Anwesenheit oder Schlafzyklus von Haushaltsbewohnern (siehe Kleiminger et al., 2013; Lisovich et al., 2010). Solche Informationen bilden die Grundlage für Assistenzfunktionen in intelligenten Gebäuden, wie etwa intelligentes Heizen (Becker, Kleiminger et al., 2018) oder Altenpflege (Alcalá et al., 2017).

2.4 Einordnung klassischer Disaggregationsansätze

NILM-Verfahren unterscheiden sich im Wesentlichen in zwei Aspekten: In der Auswahl der Merkmale, die aus einem Haushaltslastgang extrahiert werden, und in den hierzu eingesetzten Maschinenlernverfahren. Im Folgenden werden jeweils Beispiele für die beiden Kategorien Merkmale und Verfahren anhand bekannter Arbeiten vorgestellt.

2.4.1 Gerätemerkmale im Lastgang

Im Rahmen von Disaggregationsansätzen werden Merkmale im Haushaltslastgang für die Erkennung von Geräteaktivität oder die Regression der Leistungsaufnahme eines Gerätes entworfen, anhand derer die Unterscheidung von Haushaltsgeräten durchführbar ist. Da sich die Verfügbarkeit der Merkmale nach dem verwendeten Messsystem richtet, wird in der Literatur unterschieden, ob Merkmale aus zeitlich niedrigaufgelösten oder hochaufgelösten Energiemessdaten stammen. (vgl. Parson, 2014; Zeifman und Roth, 2011)

Zeitlich niedrigaufgelöste Lastgänge besitzen eine Abtastrate ≤ 1 Hz und stellen Effektiv- oder Mittelwerte von elektrischen Messgrößen dar. Hochaufgelöste Lastgänge haben eine Abtastrate ≥ 1 kHz und bestehen aus gemessenen Augenblickswerten elektrischer Messgrößen. (Zeifman und Roth, 2011; Zoha et al., 2012)

Herkömmliche Smart Meter verfügen in der Regel über unzureichende Kommunikationsschnittstellen, um hochaufgelöste Lastgänge zu übermitteln. Für die breite Anwendbarkeit von NILM sind daher nur Ansätze für niedrigaufgelöste Lastgänge relevant. (Parson, 2014)

Eine ausführliche Zusammenstellungen der in klassischen NILM-Verfahren verwendeten Merkmale sind in den Überblicksartikeln von Zeifman und Roth (2011) sowie Zoha et al. (2012) verfügbar. Im Folgenden werden die in vorhandenen Arbeiten verwendeten Merkmale kurz rezipiert.

Merkmale aus niedrigaufgelösten Zeitreihen

Beim Ein- oder Ausschalten eines Gerätes (Schaltereignis) ändert sich dessen Leistungsaufnahme sprunghaft und diese Leistungsänderung ist somit auch im Haushaltslastgang als **Lastgangsflanke** erkennbar. In Abbildung 2.2 ist zur Verdeutlichung ein niedrigaufgelöster Wirkleistungslastgang aus dem KIT Energy Smart Home Lab (ESHL) (siehe Abschnitt 4.3.5) dargestellt. Es sind Abschnitte erkennbar, in denen die Last relativ konstant bleibt. Diese werden durch Lastgangsflanken unterbrochen, die auf ein Schaltereignis eines Haushaltsgeräts hindeuten.

Bereits bei den ersten Ansätzen für NILM von Hart (1992) wurden diese Lastgangsflanken als Merkmal verwendet, da diese reproduzierbar und charakteristisch für eine Haushaltsgeräteklasse sind. Zudem sind diese Flanken als relative Änderungen im Lastgang unabhängig von der Anzahl der zu dem Zeitpunkt aktiven Geräte. Neben dem Disaggregationsverfahren von Hart (1992) wird dieser Merkmalstyp beispielsweise in Ansätzen von Baranski und Voss (2003) und Drenker und Kader (1999) verwendet.

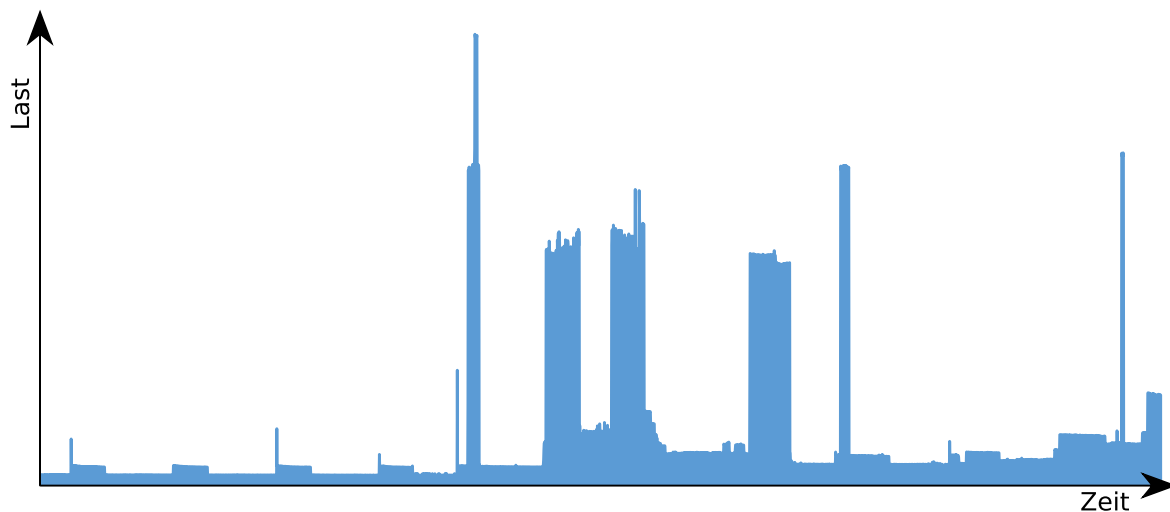


Abbildung 2.2: Schematischer Ausschnitt eines niedrig aufgelösten Haushaltslastgangs. Zwischen Abschnitten mit relativ konstanter Last treten Sprünge auf, die auf das Ein- oder Ausschalten eines Gerätes deuten.

Nachteile bestehen darin, dass (a) mehrere Schaltereignisse innerhalb eines Messintervalls fallen können und dass (b) kleine Geräte mit geringem Energieverbrauch schwer voneinander unterscheidbar sind, da sie Flanken mit geringer Höhe erzeugen. Zur Behandlung gleichzeitiger Schaltereignisse versucht Hart (1992), die einzelnen Schaltereignisse heuristisch zu dekomponieren. Und gegen die schwere Unterscheidbarkeit von kleinen Geräten verwendet Hart (1992) weitere Merkmale wie die Blindleistungsflanken.

Neben Wirk- und Blindleistung werden auch die Amplituden der **harmonischen Oberschwingungen** im Stromfluss als zusätzliche Merkmale verwendet, um Geräte unterscheiden zu können. Diese Merkmale sind zur Differenzierung von nichtlinearen elektronischen Geräten hilfreich, die sich im Wirk- und Blindleistungsverbrauch kaum unterscheiden. (Laughman et al., 2003)

Eine Reihe von NILM-Ansätzen verwendet **quasistationäre Wirk- und/oder Blindleistungswerte** als Merkmal. Dabei wird zwischen Lastgangsflanken angenommen, dass die Wirk- und Blindleistungswerte quasistationär sind, d. h. abschnittsweise nahezu konstant sind. Dabei setzen sich die Leistungswerte aus einer Summe von typischen Geräteleistungsaufnahmen zusammen. Beispiele für Ansätze, die dieses Merkmal verwenden sind Combinatorial Optimization von Hart (1989) und die auf einem Factorial Hidden Markov Model (FHMM) basierenden Ansätze von Johnson und Willsky (2013), Kim et al. (2011) und Kolter und Jaakkola (2012).

Barker, Kalra et al. (2013, 2014a) stellen klar, dass nicht alle Gerätelastgänge abschnittsweise konstant sind und modellieren Lastgänge stattdessen als **Sequenz parametrierbarer Last-**

muster. Zu den Mustern gehören beispielsweise konstante, oszillierende und exponentiell ansteigende/abfallende Lasten. Auf dieser Modellierung beruht schließlich ein Ansatz zur Nachverfolgung von Geräteenergieverbrauch von Barker, Kalra et al. (2014b), der auch zur Energiedatendisaggregation eingesetzt wird. Buneeva und Reinhardt (2017) nutzen diese Form der Lastgangsmodellierung, um synthetische Lastgänge zu erzeugen.

Norford und Leeb (1996) und Laughman et al. (2003) nutzen Merkmale in der Lastkurve, die ein Gerät während **Einschaltvorgängen** erzeugt. Beispielsweise sind bei Einschaltvorgängen mancher Geräte im Wirk- und Blindleistungslastgang Aufwärm- oder Einschwingvorgänge in Form von linearen oder exponentiellen Rampen nachvollziehbar. Diese Merkmale werden zur Differenzierung unterschiedlicher Geräte genutzt.

Neben Merkmalen, die sich aus einer elektrischen Energieverbrauchsmessung ergeben, lassen sich weitere Messdaten von Sensoren oder Zustandsdaten von intelligenten Geräten nutzen, um eine Disaggregation zu unterstützen. Beispielsweise schlagen Azzini et al. (2014) vor, mit einfachen Sensoren den Aktivitätszustand von elektrischen Geräten zu bestimmen, um darauf basierend die Disaggregation der gerätespezifischen Lastgänge durchzuführen.

Merkmale aus hochaufgelösten Zeitreihen

Die Erfassung von zeitlich hochaufgelösten Lastgängen erfordert Messsysteme, die über schnelle Kommunikationsschnittstellen oder entsprechende Speicherkapazitäten verfügen. Dafür sind in hochaufgelösten Messungen feinere Gerätesignaturen enthalten, anhand derer Geräte differenzierbar sind, die sonst mit niedrigaufgelösten Lastgängen schwer unterscheidbar sind.

Im hochaufgelösten Lastgang sind beispielsweise charakteristische **Transienten** zu beobachten, die während des Einschaltens eines Gerätetyps entstehen. Diese kurzfristigen Anlaufströme entstehen beispielsweise durch das Auf- und Entladen von Kondensatoren, Spulen oder durch interne Schaltvorgänge im Gerät. Hingegen entstehen beim Abschalten des Gerätes – bis auf einen schnellen Leistungsabfall – in der Regel keine charakteristischen Stromverläufe. Laughman et al. (2003), Leeb et al. (1995) und Norford und Leeb (1996) verwenden diese Einschalttransienten für NILM.

Weiterhin lassen sich aus dem zeitlich hochaufgelösten Strom- und Spannungsverlauf Merkmale wie Effektiv- und Spitzenwerte, Leistungsfaktor, $\cos\phi$ oder harmonische Oberschwingungen bestimmen. Solche **Merkmale aus der Zeit- oder Frequenzdomäne** werden beispielsweise von Liang et al. (2010) ergänzend zu Merkmalen aus dem Wirk- und Blindleistungsverlauf zur Einordnung von Gerätearten verwendet.

Die **Strom-Spannungs-Kennlinie** beschreibt, wie sich die Stromstärke in Abhängigkeit von der Spannung über eine Wechselstromperiode verhält. Diese Kennlinie charakterisiert

kurzfristige Eigenschaften des Stromflusses von Geräten mit elektronischen Komponenten. Lam et al. (2007) nutzen Merkmale, die an der Strom-Spannungs-Kennlinie abgelesen werden, zur Differenzierung von Gerätetypen im Rahmen eines NILM-Ansatzes. Zur Diskriminierung unterschiedlicher Geräte basierend auf der Strom-Spannungs-Kennlinie werden kürzlich auch Deep-Learning-basierten Ansätzen (De Baets et al., 2018; Ji et al., 2019) verwendet.

Gupta et al. (2010) nutzen hochfrequente **Elektromagnetische Einstrahlungen** in Stromleitungen, die insbesondere von Schaltnetzteilen stammen, um die Aktivität von Haushaltsgeräten zu erfassen. Dazu wird die Spannung mit einer sehr hohen Abtastrate gemessen, wobei die Messung nicht notwendigerweise am Hausanschlusspunkt erfolgt. Mit dieser Methode werden zwar auch Geräte mit kleiner Leistungsaufnahme wie Leuchten und Netzteile unterschieden, jedoch besitzen diese Signaturen Abhängigkeiten vom Gerätemodell und relativen Standort.

2.4.2 Maschinelle Lernansätze und Gerätemodelle

Die nachfolgend kurz vorgestellten Lernansätze wurden in bestehenden Disaggregationsverfahren insbesondere für niedrigaufgelöste Lastgänge verwendet. Der wesentliche Unterschied zwischen den hier aufgeführten Ansätzen und den in Kapitel 3 beschriebenen Verfahren liegt in der Art der Auswahl der Geräte Merkmale. Bei den hier beschriebenen klassischen Ansätzen ist die maschinelle Auswahl bzw. das Erlernen der Merkmale nicht Teil des Modelltrainings.

Clustering-Ansätze

Bei den Clustering-Ansätzen wird nach Häufungspunkten im Merkmalsraum gesucht, die sich einem Gerätetyp eindeutig zuordnen lassen. So identifiziert der NILM-Prototyp von Hart (1992) gerätespezifische Häufungspunkte in den Distanzen der Wirk- und Blindleistungssprünge. Dazu wird eine Heuristik zur Erkennung von Schaltereignissen sowie eine Heuristik zur Trennung von gleichzeitigen Schaltereignissen vorgeschlagen. Neben Hart (1992) basieren beispielsweise die Ansätze von Baranski und Voss (2003), Buneeva und Reinhardt (2017) und Drenker und Kader (1999) auf der Clusteranalyse von Lastgangsmerkmalen.

Support Vector Machines (SVM)

Ähnlich wie in Clustering-Ansätzen werden nach einer Erkennung von Schaltereignissen Support Vector Machines zur Klassifikation dieser Schaltereignisse von unterschiedlichen Geräten eingesetzt. Dabei wird allen Geräten je eine Klasse für das Einschaltereignis und

eine Klasse für das Ausschaltereignis des Gerätes zugeordnet. Für jede Schaltereignisklasse wird ein Zwei-Klassen-SVM-Klassifikator erstellt, der Beispiele dieses Schaltereignistyps gegenüber Beispielen aller anderen Schaltereignisse mit größtem Abstand abgrenzt. Während der Disaggregation werden bei jedem erkannten Schaltereignis alle Klassifikatoren evaluiert. Das Schaltereignis wird der Klasse zugeordnet, deren SVM den größten Abstand zu den Negativbeispielen besitzt. Solche *Multi Class Support Vector Machines* werden beispielsweise in den Arbeiten von Altrabalsi et al. (2014), Kramer et al. (2012) und Mittelsdorf et al. (2013) genutzt.

Einfache kombinatorische Optimierung

Ein früherer Ansatz von Hart (1989) entspricht einer einfachen kombinatorischen Optimierung, bei dem zunächst die Summenlast aller Kombinationen möglicher Gerätezustände während einer Einrichtungsphase hinterlegt wird. Dazu müssen die Leistungsaufnahme(n) aller Geräte während der Einrichtung bekannt sein bzw. bestimmt werden.

Zur Disaggregation wird für jedes Messintervall unabhängig bestimmt, welche Summenlastkombination der gemessenen Gebäudelast am ähnlichsten ist. Die Optimierungsaufgabe besteht somit in der Minimierung der Differenz zwischen der gemessenen Gebäudelast und der geschätzten Summenlast aller möglichen Kombinationen von Gerätezuständen mittels Brute Force.

Der Ansatz ist praktikabel, solange nur eine geringe Anzahl an Geräten im Gebäude betrachtet wird. Der Optimierungsaufwand wird zudem dadurch reduziert, indem jedes Messintervall unabhängig betrachtet wird. Dadurch wird jedoch nicht berücksichtigt, dass Haushaltsgeräte in der Regel nicht oft den Zustand wechseln.

Factorial Hidden (Semi-)Markov Models

Eine Familie von Ansätzen basiert auf der Modellierung des elektrischen Haushaltsenergieverbrauchs als Factorial Hidden Markov Model (FHMM). Dabei wird zunächst jedes Haushaltsgerät unabhängig als Hidden Markov Model (HMM) modelliert. Jedes Gerätemodell bzw. HMM enthält dabei beispielsweise die internen Zustände A_n und A_{us} , die nicht direkt beobachtbar sind. Im Anschluss werden die unabhängigen HMM-Einzelmodelle zu einem faktoriellen Modell zusammengesetzt, in dem die einzelnen Modelle voneinander unabhängige Zustandsübergänge vollziehen können. Die Zustände aller Modelle in einem Zeitschritt emittieren gemeinsam den Haushaltsenergieverbrauch als beobachtbare Ausgabegröße. (siehe Ghahramani und Jordan, 1996; Kim et al., 2011; Kolter und Jaakkola, 2012; Kolter und Johnson, 2011)

Die Parameter des FHMM können überwacht erlernt werden, wenn beispielsweise die Ausgabeverteilung der einzelnen HMM auf die jeweiligen Gerätelastgänge trainiert werden

(wie etwa in dem Referenzansatz von Kolter und Johnson, 2011). Alternativ kann das gesamte FHMM unüberwacht erlernt werden, indem die Gesamtausgabe auf den Haushaltslastgang trainiert wird (Johnson und Willsky, 2013; Kim et al., 2011; Kolter und Jaakkola, 2012; Parson, Ghosh et al., 2012).

Die Herausforderung bei der Verwendung des FHMM als Haushaltsmodell liegt in der mit der Anzahl der Geräte und Zustände rasch steigenden Komplexität für eine exakte Inferenz der internen Zustandsfolge. Der Zustandsraum des kombinierten Modells besteht dabei aus dem Kreuzprodukt der Zustandsräume der einzelnen HMM-Modelle. Da zu einem Zeitpunkt die Ausgabe des FHMM von allen Zuständen abhängt, lässt sich die exakte Inferenz nicht in Teilprobleme zerlegen. Für die praktische Anwendbarkeit verwendet beispielsweise Kim et al. (2011) approximative Inferenz basierend auf Gibbs Sampling. Alternativ lässt sich auch das Verfahren Structured Variational Mean Field (SMF) von Ghahramani und Jordan (1996) verwenden. (Kolter und Jaakkola, 2012)

Im Folgenden werden FHMM-basierte Disaggregationsansätze und deren Ableger vorgestellt.

(Conditional) Factorial Hidden (Semi-)Markov Model (CFHSMM)

Der erste veröffentlichte Ansatz zur Energiedatendisaggregation mit FHMM-basiertem Haushaltsmodell stammt von Kim et al. (2011). Mit den Modellvarianten wurden insbesondere Abhängigkeiten („Conditional“) zwischen den Zuständen verwandter Geräte (z. B. Fernseher und Stereoanlage) und eine beliebige Verteilung der Verweildauern in den Zuständen („Semi-Markov“) in das Modell integriert.

Factorial Hidden Dirichlet Process Hidden Semi-Markov Modell (F-HDP-HSMM)

Johnson und Willsky (2013) demonstrieren die Anwendbarkeit von faktoriellen Hidden-Semi-Markov-Modellen zur Modellierung von hierarchischen Dirichlet Prozessen (F-HDP-HSMM) für NILM. Dabei werden zunächst die Haushaltsgeräte durch A-priori-Verteilungen manuell charakterisiert, welche auf die erwartete Leistungsaufnahme und Verweildauern in den Zuständen hinweisen. Basierend auf dem Haushaltslastgang schätzt das Lernverfahren die konkrete Anzahl und Parameter der Zustände je Gerät sowie die Übergangswahrscheinlichkeiten und Verteilung der Verweildauern. Der Ansatz F-HDP-HSMM ist somit als bayesianisch und unüberwacht einzustufen. Johnson und Willsky (2013) beschleunigen das Verfahren, indem die Zeitpunkte für potenzielle Zustandswechsel durch eine einfache Heuristik vorselektiert werden, wie sie auch Hart (1992) für die Erkennung von Schaltereignissen verwendet.

Additive Factorial Approximate Maximum A-Posteriori estimation (AFAMAP)

Kolter und Jaakkola (2012) formulieren die Inferenz eines faktoriellen Hidden-Markov-Modells als gemischt ganzzahliges quadratisches Optimierungsproblem. Die Autoren wollen damit das Festfahren in lokalen Optima umgehen, wie es beim Gibbs Sampling auftreten kann. Dabei kombinieren die Autoren bei ihrem Additive Factorial Approximate Maximum A-Posteriori estimation (AFAMAP) getauften Ansatz zwei FHMM: Mit

einem FHMM werden die Lastwerte im Haushaltslastgang modelliert, mit dem anderen werden Lastsprünge bzw. die erste numerische Ableitung des Haushaltslastgangs abgebildet. Zusätzlich enthalten die jeweiligen Modelle eine Mixturkomponente, welche die Leistungsaufnahme von unbekanntem Geräten modelliert. Weiterhin schränken die Autoren das Optimierungsproblem ein, indem sie je Messintervall höchstens einen Zustandswechsel eines Geräts zulassen. (Kolter und Jaakkola, 2012)

Die zuvor beschriebenen unüberwachten Verfahren haben den Nachteil, dass entweder nach der Disaggregation die erlernten Einzelmodelle konkreten Gerätetypen zugeordnet werden müssen (wie bei Kims FHMM oder AFAMAP) oder zuvor Anhaltspunkte für das Einschalten des Gerätes in Form der A-priori-Verteilung angegeben werden müssen (wie bei dem Ansatz F-HDP-HSMM). (Parson, Ghosh et al., 2014). Damit solche Disaggregationsansätze auf vielen Haushalten auch ohne Interaktion mit dem Bewohner durchgeführt werden können, schlagen Parson, Ghosh et al. (2014) vor, zunächst mit einem überwachten Lernverfahren generische Gerätemodelle zu erstellen, die mit Hilfe des Haushaltslastgangs des Zielhaushalts unüberwacht verfeinert werden.

Bonfigli, Severini et al. (2016) erweitern den AFAMAP-Ansatz um die Nutzung der Differenz der aktuellen Leistungsaufnahme zur Leistungsaufnahme im zukünftigen Messintervall als weiteres Merkmal. Kombiniert wird dies mit zusätzlichen Einschränkungen für die Inferenz, in Form von Lastgrenzen für Gerätemodelle sowie Konsistenzigenschaften für die Blindleistung.

Neuronale Netze mit manuell entworfenen Merkmalen

In traditionellen NILM-Ansätzen kommen bereits flache neuronale Netze (mit wenigen Schichten) zum Einsatz. Beispielsweise führen Chan et al. (2000) zunächst eine diskrete Wavelettransformation eines hochaufgelösten Lastgangs durch, um darauf basierend den Zustand von drei Geräten mit einem dreischichtigen neuronalen Netz (also nur eine versteckte Schicht) zu klassifizieren. Srinivasan et al. (2006) nutzen ebenfalls eine hochaufgelöste Lastgangsmessung, ermitteln durch eine Fouriertransformation die ersten 16 harmonischen Oberschwingungen und verwenden dreischichtige neuronale Netze zur Gerätezustandsklassifikation. Chang et al. (2007) klassifizieren unterschiedliche Motorentypen basierend auf Merkmalen von kurzzeitigen, hochaufgelösten Transienten in der Lastkurve mithilfe von nicht näher spezifizierten neuronalen Netzen.

Graph Signal Processing

Die Disaggregationsansätze von Stankovic et al. (2014) sowie He, Stankovic et al. (2018) basieren auf graphenbasierter Signalverarbeitung, bei der die Ähnlichkeit zwischen Last-

gangsfanken mit Geräteschaltereignissen in Verbindung gebracht wird. Die Disaggregation erfolgt mittels einer Optimierung, die (a) Schaltereignisse mit ähnlichen Lastgangsfanken präferiert, (b) die Differenz zwischen der Summe der Lastgangsfanken der Geräte und der gemessenen Lastgangsfanke minimiert und (c) seltene Zustandswechsel bevorzugt. In den Arbeiten wird allerdings nicht evaluiert, ob trainierte Modelle auf unbekannte Gebäude übertragbar sind.

Discriminative Disaggregation Sparse Coding

Kolter, Batra et al. (2010) formulieren ihren Disaggregationsansatz für niedrig aufgelöste Lastgangabschnitte, indem für jedes Haushaltsgerät zunächst ein Wörterbuch bestehend aus Gerätelastgangbasen erstellt wird. Daraufhin wird für ein *Sparse Coding* des Gesamtlastgangs der Lastgang jedes Geräts aus einer linearen Kombination möglichst weniger Gerätelastgangbasen zusammengesetzt. In der Arbeit von Kolter, Batra et al. (2010) wird beschrieben, wie sich die Gerätelastgangbasen optimieren lassen, damit sich bei der Dekodierung des Haushaltlastgangs eine möglichst genaue Rekonstruktion der einzelnen Gerätelastgänge erstellen lässt.

Fazit

Von den traditionellen NILM-Ansätzen, die auf von handelsüblichen Smart Meter erfassten niedrig aufgelösten Lastgängen anwendbar sind, stellen die FHMM-basierten Disaggregationsverfahren den Stand der Technik dar. Bonfigli, Felicetti et al. (2018) vergleichen daher AFAMAP, als einen Vertreter FHMM-basierter Ansätze, mit einem der ersten veröffentlichten Deep-Learning-basierten Disaggregationsansätze von Kelly und Knottenbelt (2015a) (siehe Kapitel 3), dem Denoising Autoencoder (siehe Abschnitt 3.6.3). Die Autoren stellen dabei fest, dass der Denoising Autoencoder die Disaggregationsaufgabe in fast allen Fällen genauer bewältigt. Da seit 2015 wesentliche Fortschritte in der Deep-Learning-Methodik erzielt worden sind, verfolgt diese Arbeit die Weiterentwicklung von NILM-Ansätzen mit diesen neuen Methoden.

2.5 Künstliche neuronale Netze

Die grundlegenden Begriffe dieser Arbeit im Bereich künstlicher neuronaler Netze und Deep Learning werden in diesem Abschnitt kurz vorgestellt. Sofern nicht anders gekennzeichnet, basieren die Erläuterungen in diesem Abschnitt auf dem Buch von Goodfellow, Bengio et al. (2016). Für einen Überblick ist auch auf die Vorlesungsaufzeichnung von „CS231n:

Convolutional Neural Networks for Visual Recognition“³ der Stanford University zu verweisen.

Ein künstliches neuronales Netz besteht aus Neuronen bzw. **Einheiten** (engl. *units*), die einen gerichteten azyklischen Graphen bilden. Bei der Anwendung des neuronalen Netzes führt jede Einheit eine numerische Berechnung durch, wobei es die Ergebnisse anderer Einheiten nutzt, zu denen es mit einer eingehenden Kante verbunden ist. Das Berechnungsergebnis wird wiederum von allen Einheiten genutzt, die über ausgehende Kanten verbunden sind.

Durch spezielle Eingabe- und Ausgabeeinheiten wird die Eingabe bzw. Ausgabe des gesamten neuronalen Netzes repräsentiert. Die Struktur des gerichteten azyklischen Graphen wird auch als **Netzarchitektur** oder **Netztopologie** bezeichnet.

In Netzarchitekturen, die in ihrer graphischen Darstellung Zyklen enthalten, wie beispielsweise bei den rekurrenten neuronalen Netzen (RNN), erfolgt die Rückkopplung der Zwischenergebnisse in einem späteren Berechnungsschritt. Daher lösen sich diese Zyklen auf, wenn die Berechnung im zeitlichen Verlauf **ausgerollt** wird. Jedes neuronale Netz ist somit als azyklischer Berechnungsgraph darstellbar. (vgl. Goodfellow, Bengio et al., 2016, Abschnitt 10.1)

Durch die Abhängigkeiten der Einheiten untereinander sind die Einheiten in **Schichten** gruppiert. Dabei haben alle Einheiten einer Schicht gemeinsam, dass deren Berechnungen ausschließlich von den Ergebnissen aus vorhergehenden Schichten abhängen. Die Schichten sind somit untereinander partiell geordnet. Daraus folgt, dass sich zwar die Berechnungen innerhalb einer Schicht parallelisieren lassen, jedoch müssen die Einheiten einer Schicht auf die Berechnung in den vorangegangenen Schichten warten. Die Anzahl der Schichten wird als **Netztiefe** bezeichnet und stellt eine untere Schranke für die Mindestanzahl an Berechnungsschritten dar.

Die Berechnung, die jede Einheit ausführt, besteht aus der Bildung der **gewichteten Summe** der Eingaben und einem Versatzwert (engl. *bias*) sowie der anschließenden Anwendung einer **Aktivierungsfunktion**, die den der nächsten Schicht weiterzugebenden Wert bestimmt. Eine alternative Formulierung für die Berechnung, die in einer Schicht durchgeführt wird, ist die Multiplikation des **Eingangsvektors** mit der **Gewichtsmatrix** und anschließender punktweiser Anwendung der Aktivierungsfunktion.

Je Schicht gehören die Gewichtsmatrix und der Versatzvektor zu den **lernbaren Parametern** des neuronalen Netzes. Je nach Art der Einheit oder Schicht existieren weitere lernbare Parameter, beispielsweise für eine affine Transformation einer Normalisierungsschicht.

Durch eine geeignete Wahl der Gewichte approximiert das gesamte neuronale Netz eine beliebige Funktion. Dabei wird der Raum von Funktionen (Funktionsfamilie), innerhalb

³<http://cs231n.stanford.edu>

derer das neuronale Netz eine Funktion realisieren kann, durch die Netzarchitektur und Aktivierungsfunktion beschränkt.

Das Problem, welches das neuronale Netz lösen soll, wird implizit durch eine große Menge an **Trainingsbeispielen** und eine **Verlustfunktion** beschrieben. Dabei bestehen die Trainingsbeispiele jeweils aus einer Eingabe und einem Referenzergebnis. Die Verlustfunktion (engl. *loss function*) berechnet den Fehler zwischen der Schätzung des neuronalen Netzes und dem Referenzergebnis.

2.5.1 Deep Learning

Deep Learning bezeichnet Methoden für die Anwendung von künstlichen neuronalen Netzen, bei denen die für die gestellte Aufgabe notwendigen Merkmalsextraktoren aus den Daten erlernt werden. Es kommt insbesondere bei denjenigen Aufgaben zum Einsatz, die zwar für einen Menschen intuitiv lösbar sind, es jedoch einem Menschen schwerfällt, dafür einen expliziten Lösungsweg zu entwerfen.

Die Deep-Learning-Methodik löst solche Probleme mit Hilfe von tiefen neuronalen Netzen, die lernen, aus den Daten zunächst einfache Merkmale zu extrahieren und diese hierarchisch zu immer komplexeren Merkmalen zusammensetzen. Dabei sind die ersten Schichten für die einfachen Merkmale zuständig. Schicht für Schicht werden aus einfachen Merkmalen immer komplexere Merkmale erstellt.

Für Klassifikationsaufgaben erstellt beispielsweise das neuronale Netz mit diesen komplexen Merkmalen eine Repräsentation der Eingabe, in der die hervorstechenden Merkmale entflechtet und irrelevante Varianzen oder Rauschen innerhalb einer Klasse unterdrückt sind. Idealerweise sind in dieser **entflechteten Repräsentation** der Daten unterschiedliche Klassen linear trennbar. Die in jüngster Zeit entwickelten Methoden des Deep Learning ermöglichen die Erstellung solcher tiefen neuronalen Netze basierend auf großen Datenmengen.

Eine exakte Abgrenzung von Deep Learning existiert in der wissenschaftlichen Literatur nicht, da sich die Deep-Learning-Methoden aus einer Vielzahl von Verbesserungen an Trainingsverfahren und Architekturen von neuronalen Netzen entwickelt haben und kein eigenständiges Verfahren darstellen. In dieser Arbeit wird Deep Learning daher als Oberbegriff für die in der jüngeren Zeit (wieder-)entdeckten Methoden, Ansätze und Erkenntnisse für neuronale Netze verwendet, deren wesentliches Funktionsprinzip auf der Verwendung von selbsterlernten Merkmalsextraktoren und Repräsentationen beruht.

2.5.2 Trainings-/ Optimierungsverfahren

Mithilfe der Verlustfunktion optimiert das Trainings- bzw. Optimierungsverfahren die Gewichte des neuronalen Netzes, um den Fehler für die Trainingsbeispiele zu reduzieren. In der Praxis haben sich stochastische Gradientenabstiegsverfahren aufgrund ihrer Einfachheit und Effizienz bewährt (LeCun et al., 2012).

Bei dem **Mini-Batch Stochastic Gradient Descent (SGD)** wird zunächst der Fehler des neuronalen Netzes für eine kleine Anzahl an Trainingsbeispielen (**Batch**) aus einem zufällig durchmischten Trainingsdatensatz ausgewertet. Anschließend werden die partiellen Ableitungen (**Gradient**) des Fehlers nach jedem Gewicht des Netzes gebildet. Der durchschnittliche Gradient dieses Batches wird dann mit der **Lernrate** skaliert und von dem jeweiligen Gewicht abgezogen. Dadurch reduziert sich in der Regel der Fehler des gesamten Netzes. Die Anpassung der Gewichte wird iterativ für jeden Batch durchgeführt. In der Regel wird die Zahl der Iterationen, die nötig ist, um alle Trainingsbeispiele zu verarbeiten, als **Trainingsepoche** bezeichnet. (LeCun et al., 2012; Ruder, 2016)

Die Konvergenz stochastischer Gradientenabstiegsverfahren kann durch ein **Momentum** beschleunigt werden, welches häufige Richtungswechsel des Gradienten während des Trainings dämpft. Als Momentum wird dabei das exponentiell gewichtete Mittel des Gradienten über Iterationen hinweg bezeichnet. Anstatt die Gewichte direkt in (entgegengesetzter) Richtung des Gradienten zu verschieben, wird das Momentum zum Aktualisieren der Gewichte verwendet. Das Momentum reduziert Richtungswechsel und führt in der Regel zu einer stabileren Konvergenz. Gleichzeitig beschleunigt das Momentum die Konvergenz in den Situationen, in denen der Gradient über mehrere Iterationen hinweg in eine ähnliche Richtung zeigt. (Ruder, 2016)

Eine weitere Beschleunigung der Konvergenz bietet der **Nesterov's Accelerated Gradient (NAG)**. Wie bei dem Momentum-Ansatz werden die Gewichte um einen Anteil aktualisiert, der auf dem aktuellen Gradienten basiert, sowie um einen Momentum-basierten Anteil, der auf dem exponentiell gewichteten Mittel vorheriger Gradienten basiert. Für das Nesterov-Momentum wird nun die Aktualisierung der Gewichte vorweggenommen, die auf dem Momentum-basierten Anteil beruhen. Damit kann der Gradient an der bereits aktualisierten Stelle im Lösungsraum berechnet werden. Da der Momentum-basierte Anteil an der Gewichteaktualisierung bereits aus der letzten Iteration bekannt ist, sind für das Nesterov-Momentum keine zusätzlichen Berechnungen nötig. (vgl. Ruder, 2016; Sutskever et al., 2013)

Neben SGD mit NAG gehört **A Method for Stochastic Optimization (ADAM)** (Kingma und Ba, 2014) zu den häufigsten in der Praxis verwendeten Optimierungsverfahren für tiefe neuronale Netze. ADAM ist ein stochastisches Gradientenabstiegsverfahren, bei dem die Lernrate für jedes einzelne Gewicht adaptiv geregelt wird. Skalenerunterschiede im Wertebe-

reich der Gewichte können damit ausgeglichen werden. Bei der Nutzung von ADAM erübrigt sich somit eine Feinabstimmung der globalen Lernrate. (Ruder, 2016)

2.5.3 Initialisierung

Die Gewichte der neuronalen Netze werden mit randomisierten Verfahren initialisiert. Dabei ist zu beachten, dass in jeder Schicht eine Matrixmultiplikation durchgeführt und die Aktivierungsfunktion angewandt wird. Bei einer falsch skalierten Initialisierung können die Zwischenergebnisse nach mehreren Schichten insbesondere zu Beginn des Trainings leicht in den Sättigungsbereich der Aktivierungsfunktion fallen, in denen der Gradient extrem klein oder extrem groß (engl. *vanishing / exploding gradients problem*) wird. Diese Probleme führen zu numerischer Instabilität und dies kann wiederum zum Scheitern des Trainings führen.

Für die geeignete Initialisierung der Gewichte werden daher in der Praxis der Ein- und/oder Ausgabegrad einer Schicht betrachtet und die zufällige Initialisierung darauf basierend skaliert. Zu den in der Praxis beliebten Initialisierungsmethoden gehören die nach den Autoren benannten Ansätze von Glorot und Bengio (2010) sowie He, Zhang et al. (2015b).

2.5.4 Regularisierung

Da tiefe neuronale Netze in der Regel eine hohe Anzahl an lernbaren Parametern und damit eine hohe Kapazität besitzen, welche leicht zu einer Überanpassung an die Trainingsdaten (engl. *overfitting*) führt, ist Regularisierung notwendig, um eine für das Problem geeignete Modellpräferenz auszudrücken. Somit sind Regularisierungsmaßnahmen gezielte Modifikationen des Trainingsprozesses, die dem neuronalen Netz für das gestellte Problem eine höhere Generalisierungsfähigkeit verleihen (Goodfellow, Bengio et al., 2016, Abs. 5.2.2).

Eine einfache Regularisierungsmaßnahme ist beispielsweise die Einführung eines Strafterms für die Gewichte des Netzes in die Verlustfunktion. Dies drückt eine Präferenz für „einfache“ Modelle aus.

Dropout (Hinton, Srivastava et al., 2012) ist eine beliebte Regularisierungsmaßnahme, die einzelne Zwischenergebnisse von Einheiten zufällig im Training ausblendet (auf 0 setzt). Dies fördert die Nutzung unterschiedlicher Merkmale und damit die Robustheit der neuronalen Netze.

Noise Injection (An, 1996) speist weißes Rauschen in die Zwischenergebnisse des neuronalen Netzes während des Trainings ein. Vom Prinzip ähnlich zu Dropout fördert es die Robustheit des Netzes.

Eine größere Auflistung gebräuchlicher Regularisierungsmaßnahmen ist in Kapitel 7 des Buchs von Goodfellow, Bengio et al. (2016) zu finden.

Weiterhin haben die im folgenden Abschnitt beschriebenen Normalisierungsansätze ebenfalls einen regularisierenden Effekt auf das Modelltraining.

2.5.5 Normalisierung

Das Ziel einer Normalisierung des Datensatzes ist die Verbesserung der Generalisierungsfähigkeit des erlernten Modells, indem der Wertebereich so verschoben und skaliert wird, dass die Verteilung der Daten in der Anwendung der Trainingsdatenverteilung ähnlich ist. Insbesondere bei der Bildverarbeitung werden so Verschiebungen und Verzerrungen des Wertebereichs ausgeglichen, die aufgrund unterschiedlicher Beleuchtungssituationen leicht auftreten.

Im Gegensatz zu traditionellen maschinellen Lernverfahren, bei dem nur eine einzige Normalisierung des Merkmalsraums durchgeführt wird, erfolgt in tiefen neuronalen Netzen die Berechnung aufgrund der Schichtstruktur in mehreren Schritten. Dies ermöglicht eine Normalisierung der Zwischenergebnisse nach jeder Schicht bzw. jedem Block.

Die in der Bildklassifizierung erfolgreich eingesetzten Normalisierungsansätze **Batch-Normalisierung** (Ioffe und Szegedy, 2015) und **Instanz-Normalisierung** (Ulyanov et al., 2016) werden in dieser Arbeit näher betrachtet. Bei der Batch-Normalisierung wird innerhalb eines Batches⁴ jeder Merkmalsraum unabhängig normalisiert. Die Instanz-Normalisierung führt die Normalisierung jedes Merkmals für jedes Trainingsbeispiel einzeln durch.

Normalisierungsschichten innerhalb eines Netzes haben einen regularisierenden Effekt (siehe Abschnitt 2.5.4), da eine gleichzeitige Aktivität mehrerer Einheiten durch die Normalisierung relativiert wird. Dies fördert spärlich besetzte Zwischenergebnisse und somit eine Spezialisierung von Einheiten auf unterschiedliche Merkmale. (vgl. Ruder, 2016)

2.5.6 Aktivierungsfunktion

Die Besonderheit von neuronalen Netzen gründet auf der Verwendung von nichtlinearen Aktivierungsfunktionen. Ohne Nichtlinearitäten bestünde die Berechnung des neuronalen Netzes nur aus linearen Matrixmultiplikationen, die sich sonst zu einer einzigen Matrixmultiplikation zusammenfassen ließen. Das neuronale Netz wäre somit nicht leistungsfähiger als eine lineare Regression. (vgl. Goodfellow, Bengio et al., 2016, Abschnitt 6.1)

⁴Menge gleichzeitig berechneter Trainingsbeispiele

Als Aktivierungsfunktion der inneren Schichten wurde historisch häufig eine sigmoide Funktion wie der Tangens hyperbolicus verwendet. Im Rahmen von Deep-Learning-Ansätzen wird jedoch häufig das *Rectified Linear Unit* (ReLU) mit der Gleichung $ReLU(z) := \max(0, z)$ verwendet, welche leicht zu berechnen ist. Aufgrund des nicht vorhandenen Gradienten für negative Werte werden auch Varianten wie das *LeakyReLU* verwendet mit $LeakyReLU_\alpha(z) := \max(0, z) + \alpha \cdot \min(0, z)$. (vgl. Goodfellow, Bengio et al., 2016, Abschnitt 6.3)

Die Aktivierungsfunktion der Einheiten in der Ausgangsschicht richtet sich nach der Problemstellung: Beispielsweise genügt für eine Regression eine lineare Aktivierungsfunktion. Für eine logistische Regression, also bei einer binären Klassifikationsaufgabe, wird die logistische Sigmoidfunktion $\sigma(z) := \frac{1}{1+e^{-z}}$ verwendet. Bei einer multinominalen Klassifikation wird die multinominale Verallgemeinerung der logistischen Sigmoidfunktion verwendet, die Softmax-Funktion $Softmax(\mathbf{z}) := \left(\frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \right)_{j=1..K}$. Diese Softmax-Funktion konvertiert den Wertebereich eines reelwertigen K -dimensionalen Vektors \mathbf{z} in das Intervall $[0; 1]^K$, wobei die Vektorelemente sich zu $1 = \sum_{j=1}^K z_j$ summieren. Das Ergebnis der Softmax-Funktion ist damit als Wahrscheinlichkeitsverteilung interpretierbar. (vgl. Goodfellow, Bengio et al., 2016, Abschnitt 6.2.2)

2.5.7 Verlustfunktion

Mit der Verlustfunktion (engl. *loss*) wird ein skalarer Fehlerwert berechnet, der quantifiziert, wie groß der Fehler des neuronalen Netzes bei der Inferenz von Trainingsbeispielen ist. Die Verlustfunktion wird verwendet, um während des Trainings den mittleren Fehlerwert aller Trainingsbeispiele zu minimieren. Auf die folgenden Verlustfunktionen wird in dieser Arbeit Bezug genommen. (vgl. Goodfellow, Bengio et al., 2016, Abschnitt 6.2.1)

NLL-Loss Mit einer Verwendung der *Negative Log-Likelihood* (NLL) als Verlustfunktion wird zunächst die Ausgabe des neuronalen Netzes als Wahrscheinlichkeitsverteilung interpretiert. Der Negative Log-Likelihood entspricht damit der Wahrscheinlichkeit, dass das Modell ein gegebenes Trainingsbeispiel nicht erwartet. Ein bereits trainiertes Modell weist Trainingsbeispielen hohe Wahrscheinlichkeiten zu und dies führt zu einem niedrigen Negative Log-Likelihood. (vgl. Goodfellow, Bengio et al., 2016, Abschnitt 5.5)

L_2 -Loss Der L_2 -Loss ist die durchschnittliche quadrierte Differenz zwischen Zielwert und Schätzergebnis eines Modells und wird häufig bei Regressionsaufgaben eingesetzt. Durch das Quadrieren wird dieser Fehlerwert mit dem eigenen Absolutwert gewichtet, wodurch große Zielwertabweichungen – also Ausreißer – hervorgehoben werden. Das Modelltraining ist mit dieser Verlustfunktion empfindlich gegenüber Ausreißern im Trainingsdatensatz.

L_1 -Loss Der L_1 -Loss ist die durchschnittliche absolute Differenz zwischen Zielwert und Schätzergebnis. Das Modelltraining mit dem L_1 -Loss ist verglichen zur Nutzung des L_2 -Loss unempfindlicher gegenüber Ausreißern, da jede Fehlerdifferenz gleich gewichtet wird.

2.5.8 Netzarchitekturelemente

Viele tiefe neuronale Netze setzen sich aus etablierten Strukturen und Komponenten zusammen, die jeweils bestimmte Modelleigenschaften beim Training bevorzugen bzw. erzwingen. Folgende Strukturen und Komponenten werden in der Architektur von tiefen neuronalen Netzen häufig eingesetzt.

Vollverbundene Schichten

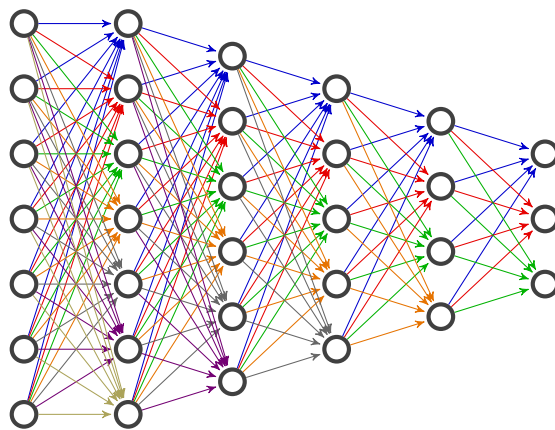


Abbildung 2.3: Verbindungsstruktur eines vollverbundenen neuronalen Netzes. Jede Kante entspricht einem individuellen erlernten Gewicht.

Historisch werden künstliche neuronale Netze mit vorwärtsgerichteten Netzen assoziiert, bestehend aus vollverbundenen Schichten. Das Netz ist dabei in eine Eingabeschicht, einige verdeckte Schichten und eine Ausgabeschicht aufgeteilt. Die Einheiten einer Schicht sind mit allen Einheiten der vorangegangenen Schicht sowie allen Einheiten der nachfolgenden Schicht verbunden (siehe exemplarisch Abbildung 2.3).

Durch die vollverbundene Struktur können Zusammenhänge zwischen allen Merkmalen eines Trainingsbeispiels erlernt werden. Allerdings weisen vollverbundene Schichten eine hohe Anzahl an erlernbaren Parametern auf⁵. Daraus folgt ein entsprechend hoher Be-

⁵Beispielsweise besitzt eine vollverbundene Schicht mit n Einheiten, die einer Schicht mit m Einheiten folgt, $mn + n$ lernbare Parameter

darf an Trainingsdaten, um eine Überanpassung des Modells auf die Trainingsdaten zu vermeiden.

Faltende neuronale Netzschichten

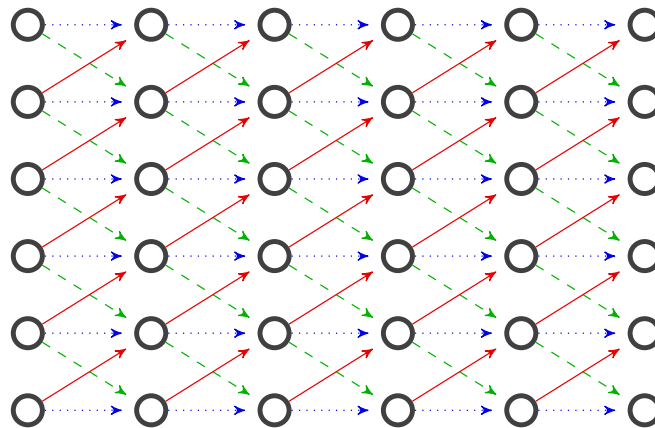


Abbildung 2.4: Verbindungsstruktur eines faltenden neuronalen Netzes. Gleichgefärbte Kanten innerhalb einer Schicht teilen sich den gleichen Gewichtsparameter. Verbindungen zwischen Einheiten sind lokal beschränkt.

In faltenden neuronalen Netzen (engl. *convolutional neural networks*, CNN) besitzen Einheiten einer Schicht eine **Lokalität**. Beginnend mit der Eingabeschicht wird jede Einheit beispielweise mit einer Bildposition, einem Pixel oder einem Zeitschritt assoziiert. Die Einheiten der nachfolgenden Schichten sind ausschließlich mit benachbarten Einheiten der vorangegangenen Schicht verbunden (siehe exemplarisch Abbildung 2.4).

Zudem teilen sich Einheiten einer Schicht dieselben Parameter, um die gleiche(n) Operation(en) an unterschiedlichen Positionen durchzuführen. Die Funktionsweise dieser Einheiten gleicht einer diskreten Faltung, wobei die Gewichte der Einheit als **Filter** bzw. **Kernel** bezeichnet werden.

Innerhalb einer Schicht können mehrere unterschiedliche Faltungsoperationen gleichzeitig durchgeführt werden, um unterschiedliche Merkmale zu extrahieren. Daher wird in einer Faltungsnetzschicht nicht nur ein Merkmalswert berechnet, sondern eine vollständige Merkmalskarte (engl. *feature map*).

Durch die eingeschränkte Verbindungsstruktur fließt nur ein beschränkter, lokaler Bereich der Eingabedaten in die Berechnung jeder Einheit ein. Dieser Bereich wird als **Wahrnehmungsbereich** (engl. *receptive field*) bezeichnet und ist abhängig von den Wahrnehmungsbereichen der vorangegangenen Einheiten. (siehe Goodfellow, Bengio et al., 2016, Abbildung 9.4)

Bei dem Entwurf von faltenden neuronalen Netzen ist daher auf den Wahrnehmungsbereich und damit auf die Größe des Kontextes zu achten, die eine Einheit berücksichtigen kann. Dabei vergrößert sich der Wahrnehmungsbereich in der Regel mit zunehmender Schichttiefe. Wenn also das gesamte neuronale Netz eine Aussage über die vollständige Eingabe treffen soll, muss sich der Wahrnehmungsbereich der Einheiten in der Ausgangsschicht über die vollständige Eingabe erstrecken.

Zusammengefasst modelliert die Struktur des faltenden neuronalen Netzes Muster innerhalb der Eingabe, die sich an unterschiedlichen Positionen oder Zeitpunkten befinden können. Verglichen zu einem vollverbundenen neuronalen Netz reduziert diese Vorgehensweise die Anzahl der Parameter erheblich. Besitzen die Daten keine solche Lokalitätseigenschaft, sind faltende neuronale Netze ungeeignet. (vgl. Goodfellow, Bengio et al., 2016, Kapitel 9)

Rekurrente neuronale Netze

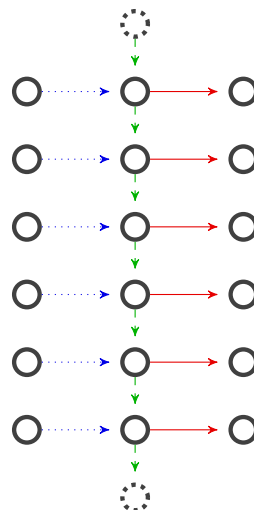


Abbildung 2.5: Verbindungsstruktur eines ausgerollten rekurrenten neuronalen Netzes. In der mittleren Spalte ist eine ausgerollte Einheit dargestellt, die die Eingabe nacheinander einliest und jeweils eine Ausgabe produziert sowie einen Kontextvektor an sich selbst weitergibt. Am Anfang wird ein Initialwert (z.B. 0) als Kontextvektor vorgegeben.

In rekurrenten neuronalen Netzen (RNN) werden Datensequenzen schrittweise durch dieselben verdeckten Einheiten verarbeitet, wobei das Ergebnis der verdeckten Einheiten als Kontextinformation beibehalten wird, indem es aus dem vorigen Zeitschritt zurückgekoppelt wird. Wie in Abbildung 2.5 dargestellt, erhält jede Einheit somit die Daten für den aktuellen Zeitschritt sowie das eigene Ergebnis – oder das Ergebnis der nachgelagerten

Ausgabeschicht – aus dem vorherigen Zeitschritt als Eingabe. (vgl. Goodfellow, Bengio et al., 2016, Kapitel 10)

Eine solche Struktur kann Sequenzen beliebiger Länge verarbeiten. Die Ausgabe des rekurrenten neuronalen Netzes wird durch eine Ausgabeschicht aus dem Ergebnis der verdeckten Schicht berechnet. Alternativ kann auch die vollständige Sequenz bestehend aus den Ausgaben nach jedem Zeitschritt als Gesamtausgabe betrachtet werden.

Da rekurrente Netze aufgrund der schrittweisen Verarbeitung eine besonders große Tiefe vorweisen, tritt das Problem der verschwindenden oder explodierenden Gradienten (*vanishing / exploding gradients problem*, siehe Abschnitt 2.5.3) besonders leicht auf, welches ein effektives Training verhindert kann. Während besonders große Gradienten durch eine Begrenzung des Vektorbetrags (*gradient clipping*) behandelt werden können, werden für die Behebung der verschwindenden Gradienten Einheiten mit Persistenzverhalten verwendet.

Zu den Einheiten mit Persistenzverhalten gehören die Einheiten mit Long-Short Term Memory (LSTM) von Hochreiter und Schmidhuber (1997) sowie die einfacheren Gated Recurrent Unit (GRU) von Cho et al. (2014). Während eine einfache rekurrente Einheit das Weiterreichen von Kontextinformation aktiv erlernen muss, ist ein gesteuerter Weiterreichungsmechanismus bereits Teil von LSTM und GRU.

2.5.9 Generative Adversarial Networks (GAN)

Klassifikations- und Regressionsaufgaben werden durch überwachte Lernansätze für neuronale Netze gelöst. Bei generativen Aufgaben sollen jedoch basierend auf erlernten Trainingsbeispielen neue Beispiele erzeugt werden. Hierbei ist es unklar, wie die Güte eines erzeugten Beispiels bestimmt werden kann. Die Güte ist jedoch essenziell für die Optimierung des neuronalen Netzes.

Goodfellow, Pouget-Abadie et al. (2014) stellen mit dem GAN ein Konzept vor, bei dem die Bewertung eines generierten Beispiels durch ein anderes tiefes neuronales Netz realisiert wird, welches parallel zum generativen neuronalen Netz trainiert wird. Wie in Abbildung 2.6 dargestellt, besteht das GAN somit aus zwei neuronalen Netzen: Der Generator G erzeugt basierend auf einem zufällig gezogenen Vektor Z ein Beispiel X . Der Diskriminator D bewertet, ob das Beispiel X aus dem echten Trainingsdatensatz stammt oder künstlich generiert wurde.

Das Ziel eines GAN ist es, die Verteilung der Daten im Trainingsdatensatz zu erlernen, um somit neue Beispiele gemäß dieser Verteilung erzeugen zu können. Dazu werden der Generator und der Diskriminator abwechselnd trainiert, wobei das

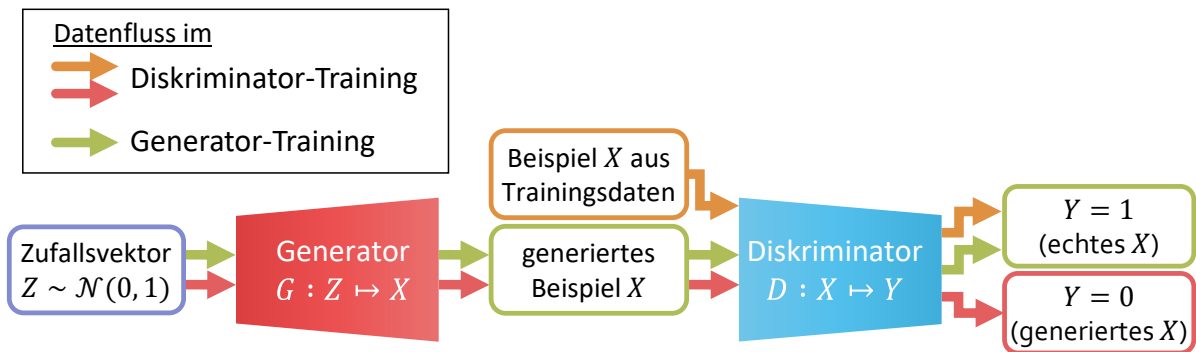


Abbildung 2.6: Training eines Generative Adversarial Networks (GAN). Abbildung nach Le-Cun (2017).

Generator-Training die Fähigkeit des Generators verbessert, mit generierten Beispielen den Diskriminator zu täuschen, sodass der Diskriminator generierte von echten Beispielen nicht unterscheiden kann; und das

Diskriminator-Training dem Diskriminator beibringt, echte Beispiele aus dem Trainingsdatensatz von generierten Beispielen zu unterscheiden.

Nach einem erfolgreichen Training haben sowohl Diskriminator D als auch Generator G die Verteilung der Beispiele aus dem Trainingsdatensatz implizit erlernt. Dabei stellt der Eingabevektor Z für den Generator eine Zufallsquelle dar, um die Variationen im Datensatz darzustellen. Somit kann der Zufallsvektor Z auch als eine Repräsentation des generierten Beispiels X interpretiert werden.

Goodfellow, Pouget-Abadie et al. (2014) adressieren bereits in ihrer Arbeit folgende besondere Aspekte, die bei einem überwachten Lernverfahren nicht auftreten und das Training eines GAN erschweren:

- Für reale Anwendungen ist die Güte des generativen Modells schwer bestimmbar, da sonst eine solche Bewertung für ein überwachtes Training verwendet werden könnte. Stattdessen erfolgt die Bewertung augenscheinlich oder indirekt über die Verwendung der generierten Beispiele für weiterführende Anwendungszwecke wie beispielsweise Objekterkennung.
- Für ein effektives Training sind die Wechselwirkungen zwischen Generator und Diskriminator bei dem zeitgleichen Optimieren der beiden neuronalen Netze zu beachten: Beispielsweise kann ein starker Diskriminator dazu führen, dass geringe Änderungen am generierten Beispiel kaum das Bewertungsergebnis beeinflussen. Daraus ergibt sich ein kleiner Gradient, mit dem der Generator nicht effizient trainiert wird.

Aufgrund solcher Probleme beschäftigt sich eine Vielzahl von GAN-Ansätzen mit der

Effektivität des Trainings⁶. Zu den theoretisch fundierten Ansätzen zur Aufrechterhaltung der Effektivität während des Trainings gehören der Least-Squares-GAN (Mao et al., 2017) und der Wasserstein-GAN (Arjovsky et al., 2017).

- Ein anderes häufig genanntes Problem ist der *mode collapse*, bei dem der Generator lediglich geringfügige Variationen eines oder weniger Beispiele generiert. Tatsächlich fehlt in der GAN-Architektur ein Mechanismus, welcher garantiert, dass der Generator den Raum aller Trainingsbeispiele vollständig abdeckt.

Zusammengefasst ermöglicht das Generative Adversarial Network (GAN) die Erstellung von generativen Modellen, indem eine nichttriviale Bewertung von generierten Beispielen erlernt wird. Von einem erfolgreich trainierten GAN sind sowohl der Generator als generatives Modell als auch die durch den Diskriminator dargestellte Bewertungsfunktion für weitere Einsatzzwecke verwendbar.

⁶einige Beispiele sind Unrolled GAN (Metz et al., 2016), f-GAN (Nowozin et al., 2016), DC-GAN (Radford et al., 2015), AC-GAN (Odena et al., 2016), und InfoGAN (Chen, Duan et al., 2016).

Bestehende Deep-Learning-Ansätze zur Disaggregation von Haushaltsenergiedaten

Auf Deep Learning basierende Verfahren stellen derzeit die erfolgversprechendsten Ansätze zur Disaggregation niedrig aufgelöster, also von Smart Meter erfassbarer, Haushaltsenergiedaten dar. Dies wird durch Bonfigli, Felicetti et al. (2018) in ihrer Studie demonstriert, indem eine Variante des Denoising Autoencoder von Kelly und Knottenbelt (2015a) mit AFAMAP von Kolter und Jaakkola (2012) verglichen wird, welches einer der modernsten traditionellen NILM-Ansätze ist (siehe Abschnitt 2.4.2). Der Deep-Learning-Ansatz führt dabei in fast allen untersuchten Szenarien eine genauere Disaggregation durch. Mittlerweile existieren mehrere Ansätze, die über den Stand des Ansatzes des Denoising Autoencoder hinaus gehen.

Ein wesentlicher Unterschied zwischen traditionellen Disaggregationsverfahren und auf Deep Learning basierenden Verfahren ist die Art und Weise, wie der Lastgang eines Gerätes modelliert wird. Während für traditionelle NILM-Ansätze zunächst Lastgangsmerkmale manuell ausgewählt und daraus handhabbare Gerätemodelle konstruiert werden, werden bei Deep-Learning-basierten Ansätzen generischere Modelle eingesetzt, die Lastgangsmerkmale datengetrieben erlernen. Durch diese Automatisierung wird in der Regel die Modellierung komplexer Merkmale zwar deutlich erleichtert, es erfordert jedoch größere Trainingsdatensammlungen und einen wesentlich höheren Rechenaufwand. (vgl. Goodfellow, Bengio et al., 2016)

Auf diesen Vorteil der automatischen Modellierung von Geräten weisen auch Kelly und Knottenbelt (2015a) hin: Während bei traditionellen Ansätzen kleine Leistungsaufnahmeschwankungen im Gerätelastgang in der Regel als weißes Rauschen modelliert werden,

nutzen Deep-Learning-Modelle versteckte Strukturen im Rauschen zur Unterscheidung unterschiedlicher Geräte. Auch die Arbeiten von Barker, Kalra et al. (2014a) zeigen, dass Lastgänge – verglichen zu deren gängiger Modellierung für Non-Intrusive Appliance Load Monitoring (NILM) als abschnittsweise stationäre Zeitreihe (siehe Abschnitt 2.4.2) – Strukturen wie Anlaufströme oder oszillierenden Stromverbrauch aufweisen. Gegebenenfalls bestehen weitere, nichtoffensichtliche Strukturen im Lastgangsverlauf, die aus menschlicher Sicht mühselig zu entdecken und zu modellieren sind. Mit Deep Learning ist es nun möglich, solche versteckten Strukturen automatisch zu finden und nutzbar zu machen.

In anderen Disziplinen wie der Bild- und Sprachverarbeitung besteht das vergleichbare Problem, dass für natürliche Signale, aufgrund ihrer Komplexität, handhabbare Modelle schwer zu konstruieren sind. Die Arbeit von Krizhevsky et al. (2012) demonstriert erstmalig die praktische Anwendbarkeit von Methoden des Deep Learning für solche Probleme mithilfe von großen Datenmengen. Anstatt Gerätemodelle manuell zu konstruieren oder zu erweitern, liegt es daher auch für NILM nahe, stattdessen Modelle datengetrieben durch tiefe neuronale Netze automatisch erlernen zu lassen. (vgl. Goodfellow, Bengio et al., 2016)

Die Wahl des Netzmodells und des Parametersuchraums für tiefe künstliche neuronale Netze ist auf anwendungsspezifische Eigenschaften anzupassen, da kein einzelnes Modell für alle Anwendungsfelder gute Ergebnisse liefert.¹ Dabei werden der Lösungsraum und darin verortete Lösungspräferenzen durch die Wahl des Netzmodells sowie des Trainingsverfahrens implizit definiert. Beispielsweise wird durch die Wahl von faltenden Schichten mit beschränkter Filtergröße konstruktionsbedingt festgelegt, dass in den Daten wiederkehrende, positionsunabhängige (äquivariante) Muster erkannt werden sollen. Faltende neuronale Netze können dadurch Zusammenhänge zwischen zeitlich weiter entfernten Werten erst durch das Kaskadieren mehrerer Schichten abbilden (siehe Abschnitt 2.5.8). Eine vollverbundene Schicht kann Schlussfolgerungen aus globalen Zusammenhängen der Eingabedaten ziehen, müsste dafür eine ggf. vorhandene Positionsunabhängigkeitseigenschaft durch einen höheren Daten- und Rechenaufwand erlernen.

In Abschnitt 3.1 wird ein Überblick über die einzelnen Schritte gegeben, die während der Disaggregation mit einem tiefen neuronalen Netzmodell und dem Training des Netzmodells durchgeführt werden. Die darauf folgenden Abschnitte diskutieren die Einzelschritte und ordnen bestehende Arbeiten darin ein.

Für die in dieser Studie durchgeführte Evaluation wird später die konkrete Konfiguration bestehender Ansätze in Abschnitt 5.1 zusammengefasst.

¹No Free Lunch Theorem, vgl. Goodfellow, Bengio et al. (2016, Abschnitt 5.2.1)

3.1 Überblick über den Disaggregations- und Trainingsprozess

Bei Deep-Learning-basierten Ansätzen zur Disaggregation hat sich bei vorhandenen Arbeiten etabliert, für jede Haushaltsgeräteklasse ein dediziertes neuronales Netz zu trainieren. Zur Disaggregation erhält das neuronale Netz den vorverarbeiteten Haushaltslastgang als Eingabe, schätzt daraus den Gerätelastgang des modellierten Haushaltsgerätetyps und ignoriert dabei die Leistungsaufnahme aller anderen Geräte. Damit kann, im Unterschied zu traditionellen NILM-Ansätzen, der Disaggregationsprozess modular für jedes Haushaltsgerät einzeln und unabhängig voneinander durchgeführt werden.

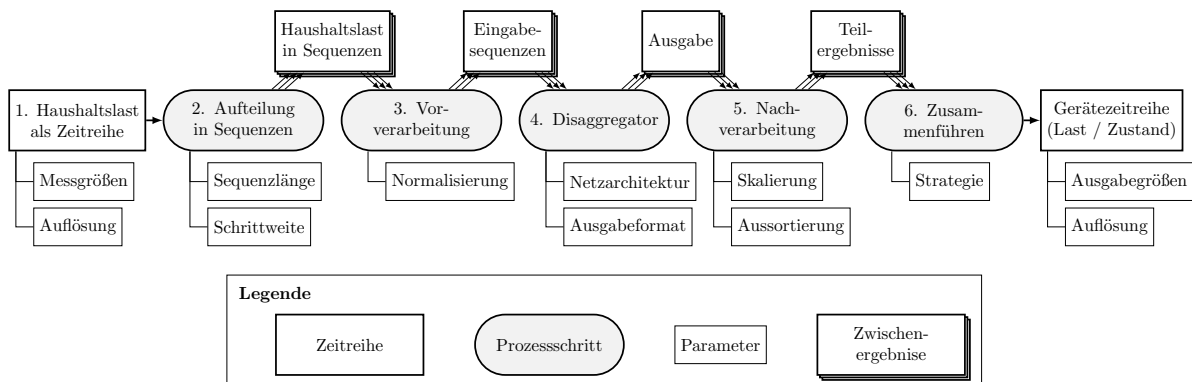


Abbildung 3.1: Einzelschritte, Entwurfsparameter und Zwischenergebnisse im Disaggregationsprozess von Deep-Learning-basierten Ansätzen.

In Abbildung 3.1 sind die einzelnen Schritte während einer Disaggregation mit Hilfe von tiefen neuronalen Netzmodellen dargestellt:

1. Der zu disaggregierende Haushaltslastgang liegt zunächst als Zeitreihe elektrischer Messgrößen mit einer festen zeitlichen Auflösung und variabler Länge vor. Welche elektrischen Messgrößen und Auflösungen in vorhandenen Ansätzen verwendet werden, ist in Abschnitt 3.2 beschrieben.
2. Um beliebig lange Haushaltslastgänge verarbeiten zu können, wird der Lastgang mit einem gleitenden Zeitfenster in einzelne Lastgangssequenzen mit fester Länge aufgeteilt (siehe Abschnitt 3.3).
3. Der Wertebereich der Lastgangssequenzen wird daraufhin normalisiert, damit die Lastgangssequenzen von dem trainierten neuronalen Netz verarbeitet werden können (siehe Abschnitt 3.5).

4. Das neuronale Netzmodell erstellt pro Sequenz eine Schätzung für den disaggregierten Gerätelastgang als Teilergebnis. In Abschnitt 3.6 sind die in der Literatur vorgestellten neuronalen Netzmodelle beschrieben. Von dem gewählten Ansatz ist die Form des Teilergebnisses (siehe Abschnitt 3.7) abhängig.
5. Nachdem das neuronale Netz die Disaggregation durchgeführt hat, erfolgt in der Nachverarbeitung ggf. eine Rücktransformation des Wertebereichs (siehe Abschnitt 3.5).
6. Schließlich werden die einzelnen Teilergebnisse zusammengeführt, um die zum gesamten Haushaltslastgang synchrone Schätzung des gesamten Gerätelastgangs zu erstellen (siehe Abschnitt 3.8).

Der Ansatz für den Disaggregationsprozess bildet die Grundlage für das Training des neuronalen Netzmodells, indem die zum Training verwendeten Beispielpaare möglichst repräsentativ für die während der Disaggregation zu erwartenden Ein- und Ausgabesequenzen sein sollten. Technisch erfolgt das Training der neuronalen Netzmodelle durch eine Variante des Gradientenabstiegsverfahrens, indem Trainingsbeispiele mit dem Netz evaluiert und Fehler zwischen geschätztem disaggregierten Lastgang und dem Referenzlastgang minimiert werden.

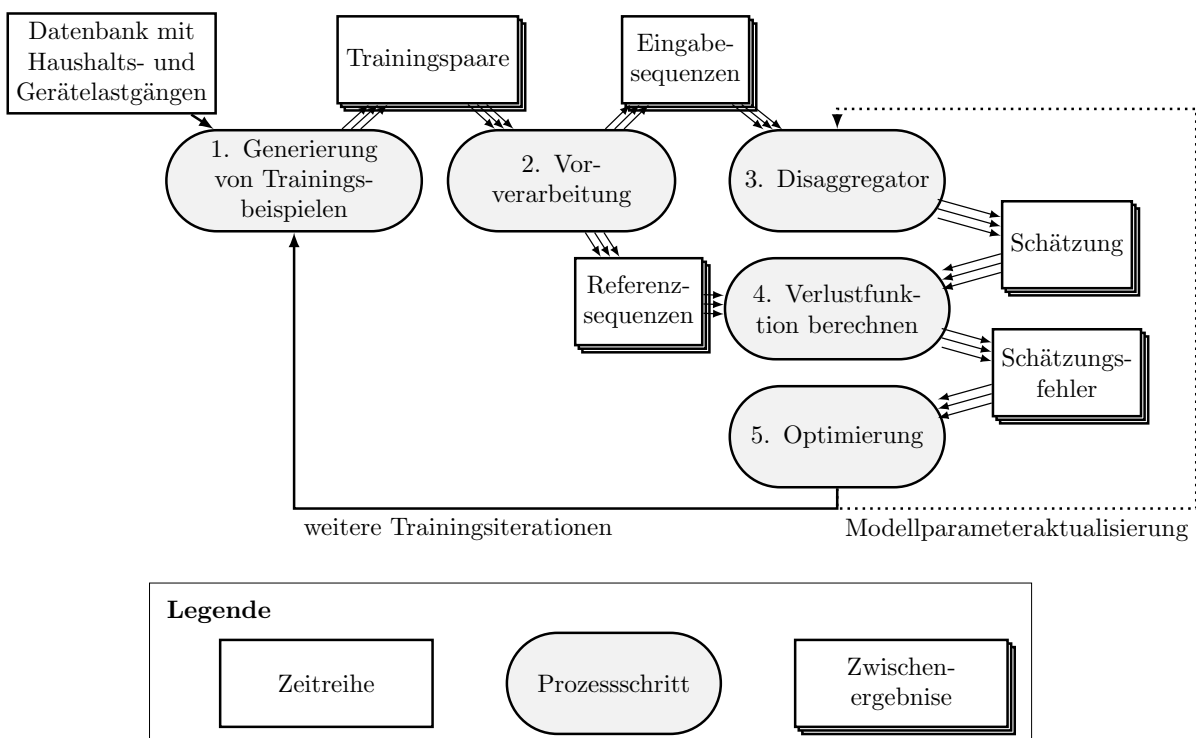


Abbildung 3.2: Trainingsablauf für ein neuronales Netz zur Disaggregation.

In Abbildung 3.2 sind die einzelnen Schritte während des Trainings des tiefen neuronalen Netzmodells dargestellt:

1. Basierend auf dem Trainingsdatensatz (siehe Abschnitt 3.10) werden Trainingspaare bestehend aus je einer Sequenz des Haushaltslastgangs und des dazugehörigen Gerätelastgangs erzeugt. Im einfachsten Fall besteht das Trainingspaar aus einem Ausschnitt einer zeitsynchronen Lastgangsmessung. Die in vorhandenen Ansätzen verwendeten Methoden zur Trainingspaargenerierung sind in Abschnitt 3.4 beschrieben.
2. Die Vorverarbeitung der Haushaltslastgangssequenz im Trainingspaar ist identisch zur Vorverarbeitung, die während des Disaggregationsprozesses durchgeführt wird. Für die Gerätelastgangssequenz werden die Nachverarbeitungsschritte invertiert, die während der Disaggregation durchgeführt werden. (siehe Abschnitt 3.5)
3. Der Disaggregator erstellt eine Schätzung des disaggregierten Lastgangs basierend auf dem aktuellen Modellzustand. Die in der Literatur verwendeten neuronalen Netze sind in Abschnitt 3.6 beschrieben.
4. Daraufhin wird der Fehler zwischen Schätzung und der tatsächlichen Gerätelastsequenz mit der Verlustfunktion bestimmt.
5. Basierend auf den Gradienten der Verlustfunktion werden die Parameter des Netzmodells optimiert, um den Fehler zu minimieren. Der Prozess wird daraufhin mit neugenerierten Trainingspaaren wiederholt. In Abschnitt 3.9 werden die in der Literatur verwendeten Optimierungsverfahren und Trainingsparameter beschrieben.

In den folgenden Abschnitten werden die einzelnen Komponenten während des Trainings- und Disaggregationsprozesses im Detail beschrieben und analysiert.

3.2 Eigenschaften des Haushaltslastgangs

Als Ausgangspunkt für die Disaggregation wird eine am Hausanschluss erfasste Zeitreihe elektrischer Messgrößen verwendet, die in einer bestimmten zeitlichen Auflösung vorliegt. Bei der Wahl der Messgrößen und der Auflösung beschränken sich die jüngeren Disaggregationsansätze mit tiefen neuronalen Netzen auf solche, die von handelsüblichen Smart Meter bereitgestellt werden, damit die Ansätze auch in realistischen Szenarien anwendbar sind.

Messgrößen

Handelsübliche Smart Meter für Privathaushalte sind selten fähig bzw. freigeschaltet, über Wirkleistung und direkt davon abgeleitete Werte hinaus gehende Messgrößen auszugeben. Die Wirkleistungswerte werden über einem Messintervall im Sekundenbereich gemittelt und können periodisch abgerufen werden. Auch wenn eine akkurate digitale Messung der Wirkleistung eine hohe Abtastrate von Strom und Spannung erfordert, aus denen sich eine Vielzahl weiterer Messgrößen ableiten ließe (siehe Abschnitt 2.4.1), werden solche Messgrößen und höhere Auflösungen selten an der Schnittstelle zur Verfügung gestellt.

Vor diesem Hintergrund wurden bisherige Disaggregationsverfahren in der Regel ausschließlich auf Wirkleistungslastgängen evaluiert (Barsim und Yang, 2018; Bonfigli, Felicetti et al., 2018; Kelly und Knottenbelt, 2015a; Zhang et al., 2016).

Bisher werden lediglich in der Arbeit von Valenti et al. (2018) die Wirk- und Blindleistungslastgänge des Haushalts als Ausgangspunkt für die Disaggregation verwendet. Die Autoren bestätigen dabei, dass das Hinzuziehen der Blindleistungsinformation in allen Szenarien zu einer Erhöhung der Disaggregationsgenauigkeit führt.

Ein anderer Aspekt ist die Verfügbarkeit von Trainingsdaten mit der geforderten Messgröße und Auflösung. Öffentliche Datensätze (siehe Abschnitt 4.3) enthalten jedoch mindestens Wirkleistungslastgänge. Die Verfügbarkeit weiterer Messgrößen ist jedoch nicht einheitlich.

Zeitliche Auflösung

Auch die Wahl der zeitlichen Auflösung wird bei bestehenden Ansätzen vorwiegend durch die verfügbaren Datensätze bestimmt.

In den Datensätzen ist die Auflösung der Gerätelastgänge geringer als die der Haushaltslastgänge. Dennoch verwenden alle bekannten Ansätze eine einheitliche Auflösung für den Haushaltslastgang und den Gerätelastgang und reduzieren somit die Auflösung für den Haushaltslastgang.

Die Ansätze, die den Datensatz UK-DALE (siehe Abschnitt 4.3) verwenden, verarbeiten Lastgänge mit einem Messintervall von 6 Sekunden (Bonfigli, Felicetti et al., 2018; Kelly und Knottenbelt, 2015a; Krystalakos et al., 2018; Zhang et al., 2016).

Für Disaggregationsexperimente mit dem Datensatz REDD werden in der Literatur Lastgänge auf ein Messintervall von 3 Sekunden (Bonfigli, Felicetti et al., 2018; Chen, Wang et al., 2018; Garcia und Macabebe, 2017; Mauch und Yang, 2015) bzw. 4 Sekunden (Paulo Marques do Nascimento, 2016) umgerechnet.

Eine Ausnahme stellt die Arbeit von Barsim und Yang (2018) dar, in der die Lastgänge zunächst auf ein Messintervall von einer Sekunde hochskaliert werden.

3.3 Aufteilung in Sequenzen durch ein gleitendes Zeitfenster

Der nächste Schritt bei der Disaggregation ist das Aufteilen des Haushaltslastgangs in einzelne Lastgangssequenzen mit fester Länge, da beliebig lange Haushaltslastgänge disaggregiert werden sollen und die Eingabe für die neuronalen Netze praktisch beschränkt ist. Die einzelnen Lastgangssequenzen werden unabhängig voneinander durch das neuronale Netz verarbeitet.

Für die Aufteilung wird ein gleitendes Zeitfenster (engl. *sliding window*) über den Lastgang geschoben und in jedem Schritt eine Sequenz ausgeschnitten. Die Art der Aufteilung wird durch die Parameter Fensterbreite und Schrittweite definiert.

Ein Orientierungspunkt für die Wahl der Fensterbreite und der Schrittweite liegt in der Dauer von typischen *Geräteaktivierungen*. Eine Geräteaktivierung beschreibt dabei den zusammenhängenden Zeitbereich, in dem ein Gerät aktiv ist und eine zusammenhängende Arbeit verrichtet bzw. Programm ausführt. Während einer Geräteaktivierung können auch kurze Inaktivitätspausen vorkommen. Vor und nach einer Geräteaktivierung ist das Geräte ein längere Zeit inaktiv. In dieser Arbeit ist die Geräteaktivierung auch ein Synonym für den Gerätelastgang, der während der Aktivierung gemessen wurde.

3.3.1 Fensterbreite

Die Fensterbreite bzw. Sequenzlänge begrenzt den Wahrnehmungsbereich des neuronalen Netzes, den es für die Schätzung der Gerätelastwerte verarbeiten kann. Hierbei ist zu beachten, dass zeitlich weiter auseinander liegende Zusammenhänge im Lastgang, wie die Relation zwischen Ein- und Ausschaltereignis, bei zu kleinen Sequenzlängen nicht erkannt werden können (Paulo Marques do Nascimento, 2016).

Andererseits kann die Sequenzlänge jedoch nicht beliebig vergrößert werden, da für höherdimensionale Ein- und Ausgaben in der Regel größere Mengen an Trainingsdaten nötig sind. Beispielsweise beschreiben Kelly und Knottenbelt (2015a), dass sich in ihren Experimenten die Disaggregationsergebnisse verschlechtern, sobald eine größere Sequenzlänge gewählt wird als die in der Arbeit angegebene.

Die Tabellen 3.1 und 3.2 zeigen eine Übersicht der in bestehenden Verfahren verwendeten Sequenzlängen für die am häufigsten verwendeten Datensätze UK-DALE und REDD (siehe

Tabelle 3.1: Länge der Haushaltlastsequenzen vorhandener Ansätze bei dem Datensatz UK-DALE.

Haushaltsgerät / Verfahren	Geschirr- spülma- schine	Kühl- schrank	Mikro- welle	Wasch- ma- schine	Wasser- kocher
Kelly und Knottenbelt, 2015a	153,6 min	51,2 min	28,8 min	102,4 min	12,8 min
He und Chai, 2016	107,5 min	46,5 min	7,2 min	124,6 min	4,0 min
Zhang et al., 2016	59,9 min	59,9 min	59,9 min	59,9 min	59,9 min
Barsim und Yang, 2018	180,0 min	180,0 min	180,0 min	180,0 min	180,0 min
Krystalakos et al., 2018	5,0 min	5,0 min	5,0 min	10,0 min	5,0 min
Shin et al., 2018	43,2 min	43,2 min	43,2 min	43,2 min	43,2 min

Tabelle 3.2: Länge der Haushaltlastsequenzen vorhandener Ansätze bei dem Datensatz REDD.

Haushaltsgerät / Verfahren	Geschirr- spülma- schine	Kühl- schrank	Ofen	Mikro- welle	Wasch- ma- schine	Wäsche- trockner
Mauch und Yang, 2015	250,0 Min	250,0 Min	–	250,0 Min	–	–
Paulo Marques do Nascimento, 2016	39,2 Min	160,1 Min	–	12,7 Min	–	–
Zhang et al., 2016	59,9 Min	59,9 Min	–	59,9 Min	59,9 Min	–
Chen, Wang et al., 2018	40,0 Min	40,0 Min	–	40,0 Min	–	–
Garcia und Maca- bebe, 2017	25,0 Min	25,0 Min	25,0 Min	25,0 Min	25,0 Min	25,0 Min
Bonfigli, Felicetti et al., 2018	115,2 Min	24,8 Min	–	4,8 Min	–	76,8 Min
Shin et al., 2018	43,2 Min	43,2 Min	–	43,2 Min	43,2 Min	–

Abschnitt 4.3). Einige Ansätze verwenden so kurze Sequenzlängen, dass das neuronale Netzmodell nicht gleichzeitig den Ein- und Ausschaltvorgang wahrnehmen kann. Die Disaggregation muss in diesen Fällen anhand anderer Lastgangsmerkmale erfolgen. In der Literatur wird die Wahl der Sequenzlängen meist nicht näher motiviert.

3.3.2 Fensterschrittweite

Die Schrittweite (engl. *stride*) beschreibt, um wieviele Messwerte das gleitende Fenster weitergeschoben wird, bevor eine neue Sequenz erstellt wird. Dieser Parameter beeinflusst, wie stark sich die Haushaltlastsequenzen und damit die Teilergebnisse überlappen. Es ergeben sich die folgenden drei, in Abbildung 3.3 dargestellten, sinnvollen Kombinationen bezüglich der Ein- und Ausgabelastsequenzen.

Überlappende Haushalts- und Gerätelastsequenzen

Ist die Schrittweite deutlich kleiner als die Länge der Lastsequenzen, überlappen sich sowohl die Haushaltlastsequenzen als auch die Gerätelastsequenzen. Überlappende Haushaltlastsequenzen bieten den Vorteil, dass eine Geräteaktivierung an unterschiedlichen Positionen des Wahrnehmungsbereichs des neuronalen Netzes platziert wird, sodass mehrere Chancen zur Detektion der Geräteaktivierung entstehen. Dadurch ist die Wahrscheinlichkeit höher, dass eine Geräteaktivierung durch mindestens einen Wahrnehmungsbereich vollständig überdeckt wird, falls die Eingangssequenzlänge groß genug ist.

Eine Überlappung der Gerätelastsequenzen führt jedoch dazu, dass für ein Messintervall mehrere Schätzungen durch unterschiedlich positionierte neuronale Netze vorhanden sind. Dies erfordert die Festlegung einer geeigneten Zusammenführungsstrategie.

Bestehende Ansätze verwenden in der Regel kleine Schrittweiten, sodass sich sowohl Haushalts- als auch Gerätelastsequenzen überlappen. Beispielsweise verwenden Kelly und Knottenbelt (2015a) beim Disaggregieren eine Schrittweite von 16 Messintervallen bzw. 1,6 Minuten, während Paulo Marques do Nascimento (2016) das Zeitfenster jeweils nur um ein Messintervall weiterschiebt.

Überlappende Eingabesequenzen, überlappungsfreie Ausgabesequenzen

Die ausgegebenen Gerätelastsequenzen der Disaggregation sind überlappungsfrei, wenn das neuronale Netz einen einzigen Lastwert ausgibt oder die Schrittweite und die Länge der Gerätelastsequenz identisch sind – eine größere Schrittweite ist nicht sinnvoll, da Lücken im geschätzten Gerätelastgang entstehen. Bei überlappungsfreien Gerätelastsequenzen ist die Zusammenführung durch eine einfache Aneinanderreihung der Teilergebnisse möglich.

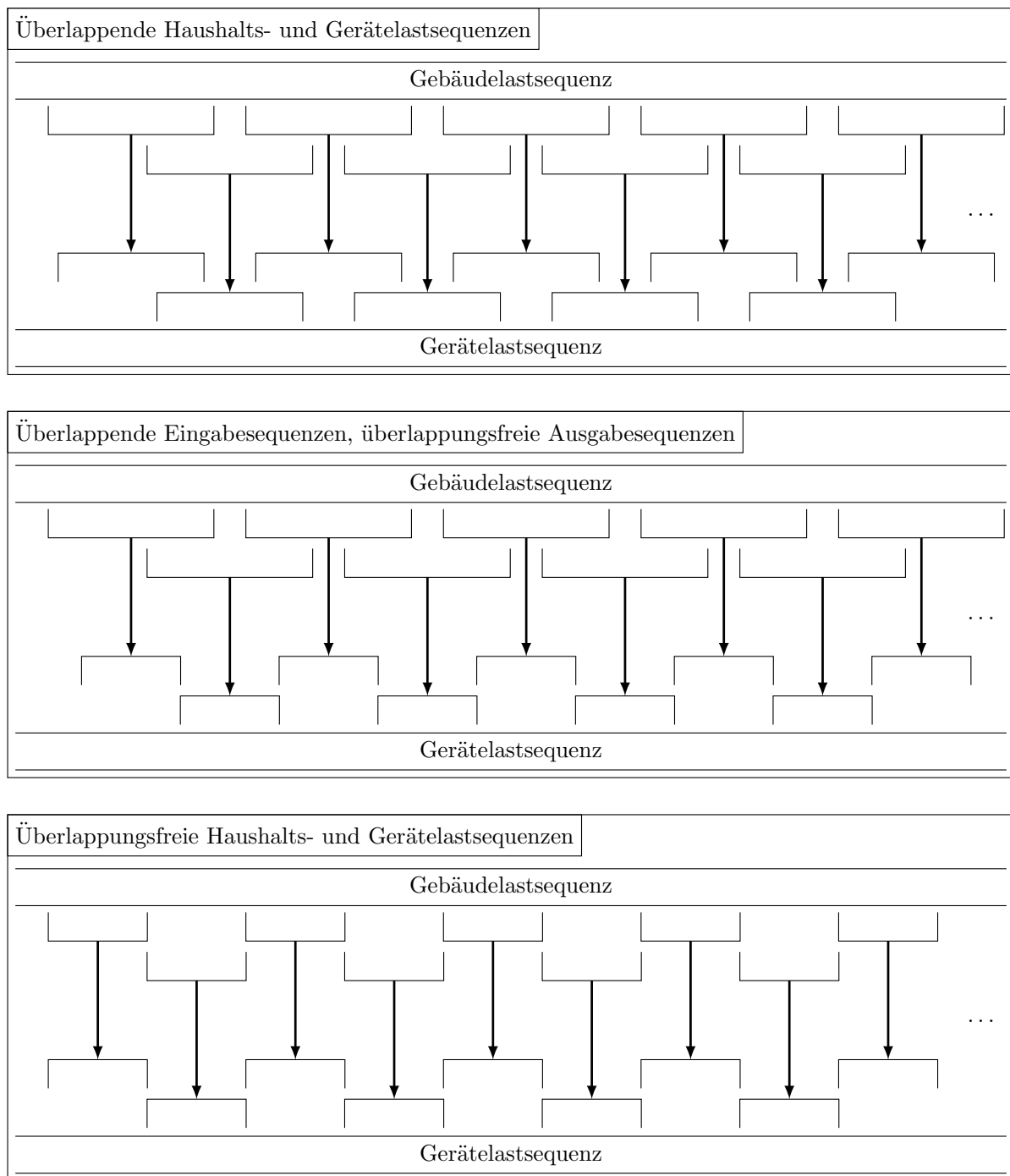


Abbildung 3.3: Schematische Darstellung der für Disaggregationsverfahren sinnvollen Aufteilungsmöglichkeiten in Lastsequenzen mit einem gleitenden Zeitfenster (*sliding window*).

Beispielsweise schätzt der Sequence-to-Point-Ansatz von Zhang et al. (2016) in jedem Schritt nur einen Lastwert, verwendet dazu jedoch die Haushaltslastgangswerte vor und nach diesem Zeitpunkt.

Bei dem Ansatz von Brewitt und Goddard (2018) gibt das neuronale Netz zwar eine Gerätelastsequenz aus, die Verbindungsstruktur der Einheiten ist jedoch so entworfen, dass das Ergebnis in einem Messintervall eindeutig ist und nicht von der relativen Positionierung des Netzes abhängt. Die Werte in den Teilergebnissen können somit direkt in das Endergebnis übertragen werden. Praktisch wird dies realisiert, indem die Schrittweite mit der Gerätesequenzlänge gleichgesetzt wird.

Überlappungsfreie Haushalts- und Gerätelastsequenzen

Die Haushaltslastsequenzen überlappen sich nicht, wenn das Zeitfenster jeweils um die volle Sequenzlänge weitergeschoben wird. In diesem Fall müssen die Länge der Gerätelastsequenz und die Länge der Haushaltslastsequenz identisch sein – bei kürzeren Gerätelastsequenzen ergeben sich Lücken im Endergebnis, für längere Gerätelastsequenzen fehlen dem Disaggregator Informationen über die Haushaltslast in den Randbereichen.

Dieses Vorgehen erzeugt zwar eindeutige Ergebnisse, jedoch werden nicht alle Zeitpunkte gleichbehandelt. So kann ein Geräteaktivitätszyklus auf zwei Sequenzen aufgeteilt werden und das neuronale Netz verarbeitet zu keinem Zeitpunkt den vollständigen Zyklus. In der Literatur wird diese Lastgangaufteilung bisher nur von Mauch und Yang (2015) angewendet.

Abschließend betrachtet bietet der Ansatz, aus überlappenden Haushaltslastsequenzen überlappungsfreie Gerätelastsequenzen zu erzeugen, klare Vorteile gegenüber den Alternativen. Bei diesem Ansatz treten keine besonderen Effekte am Rand von Sequenzen auf und es existieren keine Nebeneffekte durch die Wahl der Schrittweite. Weiterhin gibt es keine Nebeneffekte bei Zusammenführung der Ergebnisse, da die Ausgabewerte positionsunabhängig und eindeutig sind. Es kann auch so interpretiert werden, dass durch diesen Ansatz die Zusammenführung Bestandteil des neuronalen Netzes ist, welche somit auch erlernt wird.

Zhang et al. (2016) führen in ihrer Veröffentlichung für das angenommene Gerätemodell einen formalen Beweis, dass eine punktweise Lastschätzung wahrscheinlicher ist als ein durch Mittelwertbildung zusammengeführtes Endergebnis. Bei dem Beweis wird jedoch ein einfaches Gerätemodell vorausgesetzt, dessen Gültigkeit angezweifelt werden kann.

3.4 Generierung der Trainingsbeispiele

Ein neuronales Netz wird in der Regel mit einer der Varianten von Stochastic Gradient Descent (SGD) trainiert, indem Fehler bei der Inferenz eines Trainingsbeispiels (X, Y) , welches aus der Eingabesequenz X und dem Referenzergebnis Y besteht, minimiert werden (siehe Abschnitt 2.5.2). Zuvor muss jedoch definiert sein, wie die Trainingspaare (X, Y) aus dem Datensatz generiert sowie in welcher Reihenfolge die Trainingspaare verwendet werden. Da die Datengrundlage für NILM aus Zeitreihen unbestimmter Länge besteht, ist die Erzeugung von Trainingsbeispielen nicht offensichtlich.

Ein Trainingsbeispiel kann aus einer real gemessenen Situation erzeugt werden, indem Haushalts- und Gerätelastgang aus dem gleichen Zeitraum ausgeschnitten werden. Alternativ kann ein Trainingsbeispiel synthetisch erzeugt werden, beispielsweise durch das Aufsummieren mehrerer Lastgänge unterschiedlicher Geräte zu einem synthetischen Haushaltslastgang. Letzteres wird als Datenanreicherung (engl. *data augmentation*) bezeichnet, bei der durch algorithmische Verfahren neue realistische Trainingspaare erzeugt werden. In der Literatur wurden bisher folgende Methoden zur Erstellung von Trainingspaaren verwendet.

Chronologische Ausschnitte realer Lastgänge Die Vorgehensweise zur Erstellung von chronologischen Ausschnitten realer Lastgänge ist analog zur Aufteilung des Haushaltslastgangs in Sequenzen im Disaggregationsprozess: Ein Zeitfenster fester Breite gleitet in chronologischer Reihenfolge über den Haushalts- und Gerätelastgang (engl. *sliding window*) und die Trainingsbeispiele ergeben sich aus den Fensterausschnitten. Entwurfsparameter sind Fensterbreite und Schrittweite, mit der das Fenster jeweils weitergeschoben wird.

Der Parameter Fensterbreite stimmt in der Regel für das Training und die Disaggregation überein, da sich trainierte neuronale Netze im Allgemeinen nicht ohne weiteres auf andere Fensterbreiten adaptieren lassen.

Werden die so generierten Trainingsbeispiele auch in chronologischer Reihenfolge für das Training verwendet, sind insbesondere überlappende aufeinanderfolgende Trainingsbeispiele nicht unabhängig verteilt. Viele Haushaltsgeräte sind zudem sehr selten aktiv, sodass das Verhältnis der Trainingsbeispiele für Aktivität und Inaktivität nicht ausgeglichen ist (Kelly und Knottenbelt, 2015a). Diese Aspekte können zu einer verringerten Effizienz des Trainings führen (LeCun et al., 2012).

Ein solcher Ansatz zur Trainingspaarbildung wird daher selten verwendet – bei bestehenden Deep-Learning-Ansätzen für NILM bisher nur zum Training der rekurrenten neuronalen Netze von Mauch und Yang (2015).

Randomisierte Ausschnitte realer Lastgänge Dieser Ansatz zur Trainingsbeispielgenerierung gleicht dem Ansatz „Chronologische Ausschnitte realer Lastgänge“, bei dem

auch zunächst die gleichen Zeitabschnitte aus realen Haushalts- und Gerätelastgängen ausgeschnitten werden.

Beim Training werden jedoch alle so erzeugten Trainingsbeispiele zufällig durchmischt, sodass die Annahme der paarweisen unabhängigen Verteilung zweier Trainingsbeispiele stärker erfüllt ist. Gleich bleibt bei dieser Generierungsmethode, dass das Verhältnis von Geräteaktivität und Inaktivität nicht modifiziert wird.

Randomisierte Ausschnitte realer Lastgänge werden in der Literatur zum Training der Ansätze Sequence-to-Point/Sequence von Zhang et al. (2016) und dem Fully Convolutional Network von Brewitt und Goddard (2018) verwendet.

Balancierte Auswahl gemessener Lastgänge Hierbei sollen Trainingspaare erstellt werden, bei denen Beispiele mit Geräteaktivität und Beispiele ohne Geräteaktivität in einem definierten Wahrscheinlichkeitsverhältnis vorkommen. Dadurch wird bei Haushaltsgeräten, die typischerweise selten aktiv sind, der Trainingskorpus ausbalanciert.

Für diesen Trainingsbeispielgenerator werden in einem Vorverarbeitungsschritt zunächst alle zusammenhängenden Gerätelastgänge in Zeitabschnitte mit Geräteaktivität und Zeitabschnitte ohne Geräteaktivität partitioniert. Wie in Abschnitt 4.4.2 ausführlicher beschrieben, werden Zeitabschnitte mit Geräteaktivität gefunden, wenn eine andauernde Überschreitung der Last über einen Grenzwert vorliegt. Dabei sind Lastgrenzwert und Mindestdauer der Überschreitung empirisch bestimmte Haushaltsgeräte-spezifische Werte. Da während der Aktivität eines Gerätes Pausen vorkommen können, in denen die Last kurzfristig unter den Grenzwert fällt, werden kurzfristige Unterschreitungen des Grenzwerts ignoriert. (Batra, Kelly et al., 2014; Kelly und Knottenbelt, 2015a)

Einzelne Trainingsbeispiele entstehen durch das Ausschneiden des Haushalts- und Gerätelastgangs mit einem Zeitfenster fester Breite. Für die Platzierung des Zeitfensters wird zunächst zufällig entschieden, ob ein aktives oder inaktives Gerät abgebildet werden soll. Die Wahrscheinlichkeit dieser zufälligen Entscheidung ist als Entwurfparameter festgelegt. Innerhalb der entsprechenden Zeitabschnitte wird das Zeitfenster zufällig platziert. Dabei wird jedes mögliche Zeitfenster mit gleicher Wahrscheinlichkeit ausgewählt. (Kelly und Knottenbelt, 2015a)

Bei der Platzierung des aktiven Abschnitts ist zu beachten, dass sich der aktive Abschnitt nicht zwangsweise vollständig innerhalb des Zeitfensters befinden muss. Somit fängt die Geräteaktivität während des Zeitfensters an bzw. hört die Aktivität während des Zeitfensters auf. Dies ist auch bei der Disaggregation der Fall, da die Zeitfenster sukzessive über die aktiven Abschnitte gleiten. Das neuronale Netz muss also auch Teile von aktiven Abschnitten erkennen und somit müssen die Trainingsdaten auch solche Fälle enthalten. (vgl. Kelly und Knottenbelt, 2015a)

Zusammengefasst bestehen die gerätespezifischen Entwurfparameter dieses Trainingsbeispielgenerators aus dem Lastgrenzwert, den Mindestauern für Aktivität und Inaktivität, der Wahrscheinlichkeit für die Generierung eines Trainingsbeispiels mit aktivem Gerät sowie der Entwurfsentscheidung, ob Sequenzen mit teilweise aktivem Gerät im Trainingsdatensatz vorkommen sollen.

Kelly und Knottenbelt (2015a) und Bonfigli, Felicetti et al. (2018) verwenden u. a. diesen Trainingsbeispielgenerator.

Haushaltlastgangsynthese aus Aktivierungen Die zuvor beschriebenen Trainingsbeispielgeneratoren sind auf gemessene Haushaltlastgänge angewiesen. Im Gegensatz dazu basiert dieser Generator nur auf gemessenen Gerätelastgängen, da aus diesen ein vollsynthetischer Haushaltlastgang erzeugt wird.

Vergleichbar mit der Methode „Balancierte Auswahl gemessener Lastgänge“ werden zunächst pro Gerät zunächst die Zeitabschnitte mit Geräteaktivierung bestimmt. Zur Generierung eines Trainingsbeispiels wird für jedes Gerät zufällig entschieden, ob das Gerät aktiv sein soll. Für aktive Geräte wird ein zufällig gewählter Geräteaktivierungszyklus im Trainingsbeispiel positioniert und im Anschluss werden alle positionierten Gerätelastgänge aufsummiert. (Kelly und Knottenbelt, 2015a)

Diese Art, Trainingspaare zu generieren, wird in den Ansätzen von Kelly und Knottenbelt (2015a) und Bonfigli, Felicetti et al. (2018) parallel zu „Balancierte Auswahl gemessener Lastgänge“ angewandt.

Kombination von Trainingsbeispielgeneratoren Die zuvor genannten Trainingsbeispielgeneratoren können auch gleichzeitig verwendet werden. Dazu wird bei der Generierung eines Trainingsbeispiels zunächst per Zufallsgenerator entschieden, welcher der Generatoren verwendet werden soll. Beispielsweise kommt in den Ansätzen von Kelly und Knottenbelt (2015a) mit jeweils 50 % Wahrscheinlichkeit die „Balancierte Auswahl gemessener Lastgänge“ und die „Haushaltlastgangsynthese aus Aktivierungen“ zum Einsatz.

3.5 Vor- und Nachverarbeitung

Die praktische Anwendung neuronaler Netze erfordert die Skalierung der Eingabe- und Ausgabevektoren in einen für den Optimierer des Netzes geeigneten Wertebereich, um einen numerischen Unter- oder Überlauf zu vermeiden und um die Konvergenz günstig zu beeinflussen. Idealerweise sind die Eingabevektoren für das Training mittelwertsbereinigt und besitzen Einheitsvarianz, LeCun et al. (2012). Fehlt die Anpassung des Wertebereichs, konvergiert das Training in der Regel nicht.

Für die Vorverarbeitung der Haushaltslastsequenzen, also der Eingabevektoren, wurden in der Literatur bisher folgende Methoden verwendet:

Haushaltslastvarianznormalisierung mit individueller Mittelwertsbereinigung

Kelly und Knottenbelt (2015a) teilen für ihre Ansätze die Haushaltslastwerte durch die über den gesamten Trainingsdatensatz ermittelte Standardvarianz der Haushaltslastwerte. Paulo Marques do Nascimento (2016) verwendet anstatt der empirisch ermittelten Standardvarianz den Faktor 500 für die Division der Haushaltslastsequenzen. Dabei sei der genaue Wert des Faktors nicht relevant. Der Faktor dient lediglich dazu, die Größenordnung der Zahlenwerte in einen Bereich zu projizieren, die von neuronalen Netzmodellen und dessen Optimierungsverfahren praktisch verarbeitbar sind. In beiden Fällen wird anschließend der individuelle Mittelwert jeder Haushaltslastsequenz von der jeweiligen Sequenz abgezogen.

Skalierung mit der durchschnittlichen Geräteleistungsaufnahme

Zhang et al. (2016) normalisieren den Haushaltslastgang mit dem Mittelwert und der Standardvarianz der Leistungsaufnahme des Zielgerätes während des aktiven Zustands.

Für die Vorverarbeitung der Gerätelastsequenzen, d. h. der Ausgabevektoren, werden in veröffentlichten Ansätzen folgende Vorgehensweisen verwendet:

Skalierung des Gerätelastgangs Kelly und Knottenbelt (2015a) und Krystalakos et al. (2018) teilen die Lastgangswerte durch eine gerätespezifische Maximallast, damit die Werte in das Intervall $[0,1]$ fallen.

Normalisierung des Gerätelastgangs Zhang et al. (2016) verwenden die identische Vorgehensweise und Werte zur Mittelwertsbereinigung und zum Herbeiführen der Einheitsvarianz sowohl für den Gerätelastgang als auch für den Haushaltslastgang: Beide Lastgänge werden mit dem Mittelwert und Standardvarianz der Leistungsaufnahme des Zielgeräts im aktiven Zustand normalisiert.

Extraktion des ersten Lastblocks Für den Lastblockregressionsansatz von Kelly und Knottenbelt (2015a) werden die relative Start- und Endposition des Lastblocks innerhalb der Sequenz sowie die Durchschnittslast als Zielvektor für das neuronale Netz kodiert. Die Positionen sind als relative Positionsangaben innerhalb des Eingabezeitfensters mit einem Wert innerhalb des Intervalls $[0,1]$ ausgedrückt. Die Durchschnittslast ist ebenfalls relativ zu einer gerätespezifischen Maximallast als Wert zwischen 0 und 1 angegeben. Falls es in der Lastsequenz mehrere Lastblöcke gibt, werden die Daten des ersten Blocks als Zielvektor verwendet.

Während der Disaggregation werden diese Vorverarbeitungsschritte invers auf die Ausgabe des neuronalen Netzes angewandt. Die Nachverarbeitung der Lastblöcke betrifft nur den Durchschnittslastwert.

3.6 Netzarchitektur bestehender Ansätze

In den folgenden Abschnitten werden die neuronalen Netzarchitekturen vorhandener Disaggregationsansätze aufgezählt und nach Möglichkeit in Kategorien eingeteilt. Die ältesten Ansätze werden dabei zuerst genannt.

3.6.1 Darstellung von neuronalen Netzen

Die Architekturen künstlicher neuronaler Netze bekannter Ansätze werden in den folgenden Abschnitten mit Hilfe von PlotNeuralNet von Iqbal (2018) graphisch dargestellt.

Für eine kompaktere und abstraktere Darstellung der Architekturen werden neuronale Netzschichten als Blöcke dargestellt. Verschiedene Arten von Schichten werden mit unterschiedlichen Farben dargestellt. In Abbildung 3.4 ist beispielsweise eine Eingabeschicht und eine vollverbundene Schicht sowohl in Blockdarstellung als auch als Netz mit diskreten Einzelneuronen dargestellt. In der Blockdarstellung ist mit gestrichelten Linien der Eingabebereich eines Neurons in der vollverbundenen Schicht verdeutlicht. Dass in der vollverbundenen Schicht die Aktivierungsfunktion Rectified Linear Unit (ReLU) verwendet wird, ist in der Blockdarstellung durch die Zweifärbung des Blocks dargestellt. In der diskreten Darstellung ist die Anwendung der Aktivierungsfunktion explizit im Graphen gezeigt.

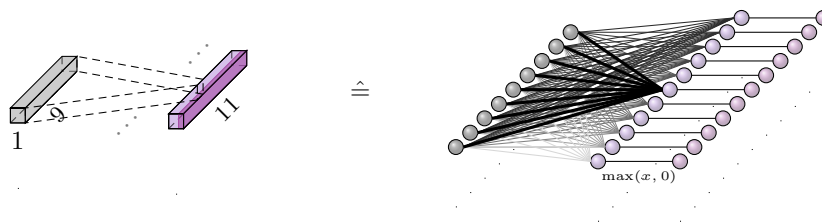


Abbildung 3.4: Blockdarstellung vs. diskrete Darstellung einer vollverbundenen Schicht mit ReLU-Aktivierung.

Einige Varianten von Faltungsschichten werden in Abb. 3.5 bis 3.7 in Blockdarstellung mit der Darstellung als diskretes Netz gegenübergestellt.

In Abbildung 3.5 berechnet eine Faltungsschicht zwei unterschiedliche Filter für die Eingabe. Die Filtergröße ist 3, sodass ein Faltungsneuron nur mit jeweils 3 Eingabeneuronen verbunden ist.

In Abbildung 3.6 sind gestapelte Faltungsschichten dargestellt, die direkt hintereinander angewandt werden. Im Randbereich wird mit Nullwerten aufgefüllt (padding 1), damit

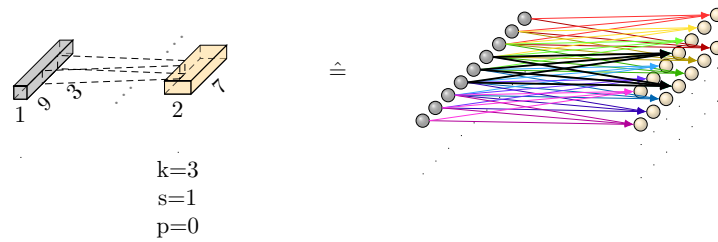


Abbildung 3.5: Faltungsschicht mit Filtergröße 3, parallel für 2 Merkmale / Kanäle.

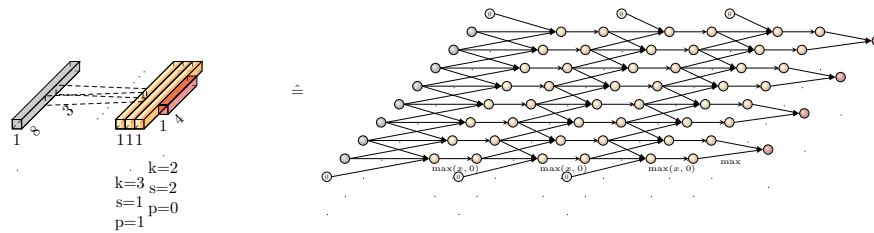


Abbildung 3.6: Gestapelte Faltungsschichten mit ReLU-Aktivierungsfunktion, gefolgt von einer Max-Pooling-Schicht.

die Ausgabelänge nach jeder Faltung gleich bleibt. Nach jeder Faltung wird die ReLU-Aktivierungsfunktion angewandt. Am Ende wird ein Max-Pooling durchgeführt, bei dem das Maximum der Eingabewerte gebildet wird. In der dargestellten Konfiguration reduziert die Max-Pooling-Schicht die zeitliche Auflösung um die Hälfte.

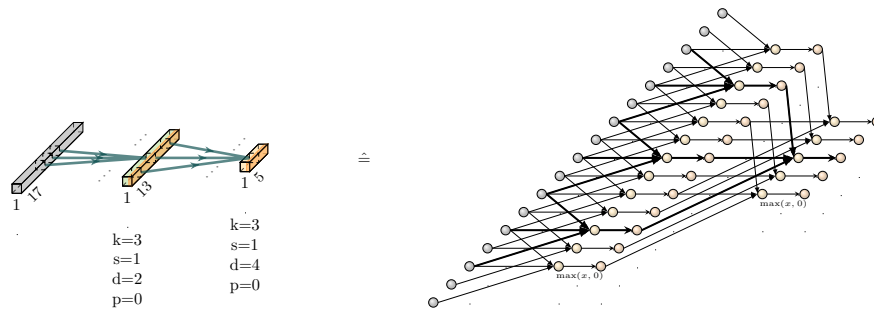


Abbildung 3.7: Dilatierte Faltungsschichten mit unterschiedlichen Dilationsfaktoren und ReLU-Aktivierungsfunktion.

In Abbildung 3.7 sind dilatierte Faltungsschichten dargestellt. Bei dilatierten Faltungsoperationen werden Eingabewerte übersprungen, sodass der Filter einen größeren Bereich auf Kosten der Auflösung abdeckt. In der Netzdarstellung ist gezeigt, welche Eingabewerte für ein Neuron in der letzten Schicht berücksichtigt werden.

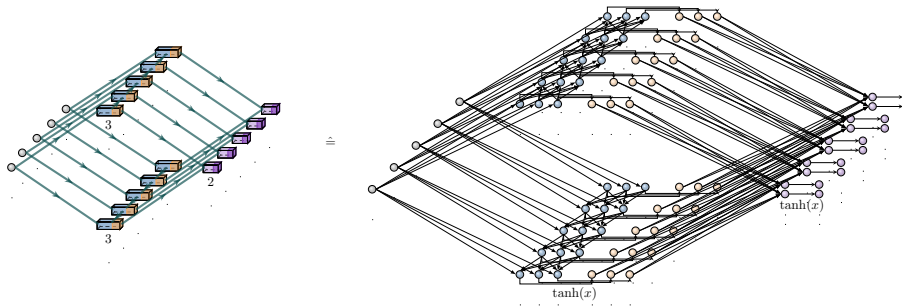


Abbildung 3.8: Bidirektionales rekurrentes neuronales Netz mit \tanh -Aktivierungsfunktion, gefolgt von einer punktwisen vollverbundenen Schicht mit \tanh -Aktivierung.

Ein Beispiel für ein bidirektionales rekurrentes neuronales Netz ist in Abbildung 3.8 dargestellt. Die Neuronen in der oberen Reihe sind von links nach rechts miteinander verbunden, während die Neuronen in der unteren Reihe von rechts nach links verbunden sind. Alle Neuronen sind mit einem bestimmten Zeitpunkt assoziiert. So wird auch die vollverbundene Schicht im Anschluss an die bidirektionale RNN-Schicht für jeden Zeitpunkt einzeln auf die Ergebnisse eines Zeitpunkts angewandt. Die Gewichte der vollverbundenen Schicht sind dabei für jeden Zeitpunkt gleich.

3.6.2 Lastblock-Regressions-Netzwerk

Einer der drei von Kelly und Knottenbelt (2015a) beschriebenen Ansätze ist ein neuronales Netz zur Regression von rechteckförmigen Lastblöcken. Das Netz soll für eine gegebene Lastgangssequenz erkennen, in welchem Zeitabschnitt das Zielgerät aktiv ist und wieviel das Gerät in diesem Abschnitt durchschnittlich verbraucht.

Falls das Zielgerät innerhalb des Eingabezeitfensters mehrmals aktiv ist, werden die Parameter der zeitlich frühesten Aktivierung geschätzt. Sollte das Zielgerät innerhalb des Eingabezeitfensters nicht aktiv sein, wird ein durchgängiger Lastblock mit einer Durchschnittslast von 0 W geschätzt. Die Ausgabe des Netzes besteht somit aus 3 Werten: dem Anfangszeitpunkt, dem Endzeitpunkt und der durchschnittlichen Last der ersten Aktivierung.

Die Architektur des neuronalen Netzes ist in Abbildung 3.9 dargestellt. Zwei Faltungsschichten berechnen 16 Merkmale für den Gesamtlastgang. Das Ergebnis wird über fünf vollverbundene Schichten schrittweise auf die drei Parameter der frühesten Aktivierung innerhalb der Eingangssequenz reduziert.

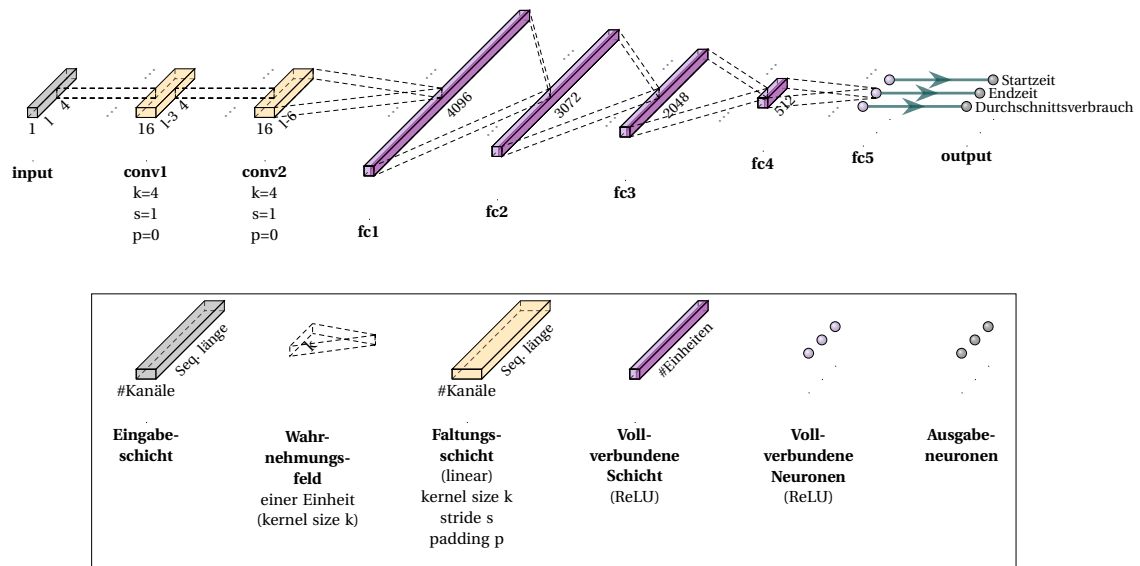


Abbildung 3.9: Das Lastblock-Regressions-Netzwerk von Kelly und Knottenbelt (2015a) (eigene Darstellung).

Für dieses Ausgabeformat wird das in Abschnitt 3.8 beschriebene „Zusammenführen von rechteckförmigen Lastblöcken“ als Rücktransformation in einen Zielgerätestrang verwendet, bei der alle geschätzten Lastblöcke übereinandergelagert werden und für jedes Messintervall der Median der Lastblöcke gebildet wird.

3.6.3 Denoising Autoencoder

Mit dem Denoising Autoencoder (DAE) interpretieren Kelly und Knottenbelt (2015a) die Disaggregation als Entrauschungsproblem, bei dem der Lastgang des Zielgeräts durch die Lastgänge aller anderen elektrischen Geräte verrauscht wird, bevor die Messung stattfindet. Ein DAE (Vincent et al., 2008) soll aus dem gemessenen Summenlastgang den ursprünglichen Lastgang des Zielgeräts extrahieren.

In Abbildung 3.10 ist der von Kelly und Knottenbelt (2015a) vorgeschlagene DAE zur Disaggregation dargestellt. Der Aufbau des DAE folgt dabei dem Prinzip des Autoencoders (Hinton und Salakhutdinov, 2006), bei dem die Eingabe schrittweise auf eine niedrigdimensionale Zwischenrepräsentation reduziert wird. Aus der Zwischenrepräsentation wird durch einen meist spiegelsymmetrischen Netzaufbau die Aufgabe schrittweise aufgebaut. Die als Flaschenhals (engl. *bottleneck*) bezeichnete erzwungene Reduktion auf eine niedrigdimensionale Zwischenrepräsentation soll bezwecken, dass nur die Merkmale aus der Eingabe durch das Netz extrahiert werden, die für die Generierung der Ausgabe wesentlich sind. Bei dem ursprünglichen Autoencoder-Konzept von Hinton und Salakhutdinov (2006) soll

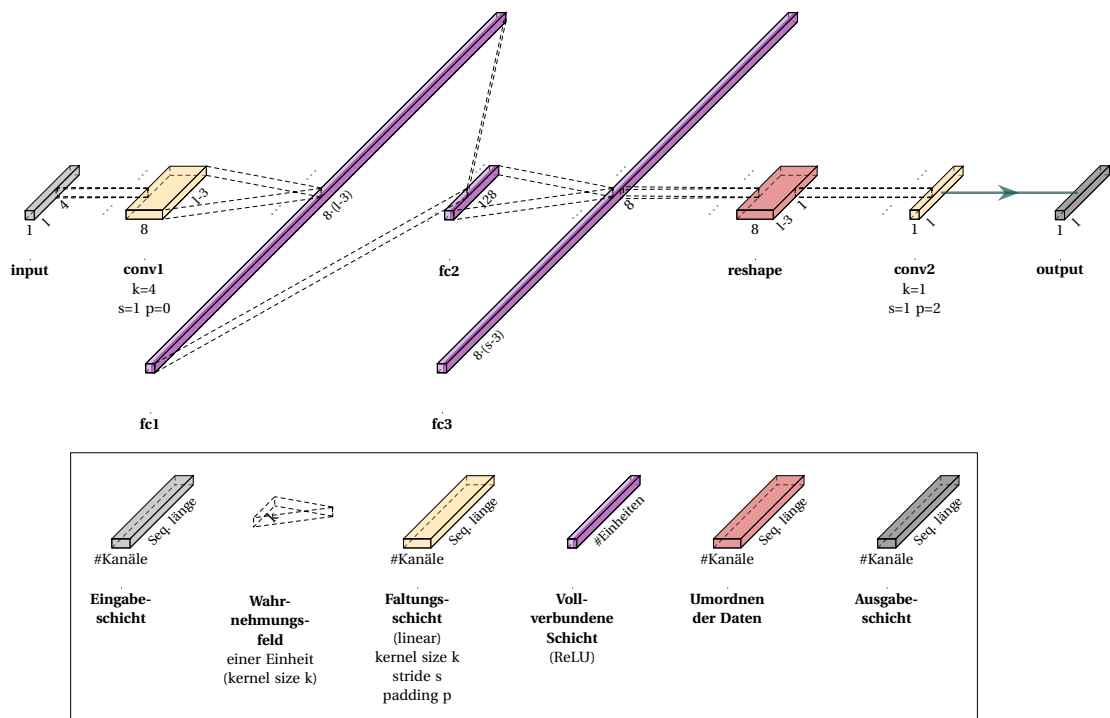


Abbildung 3.10: Denoising Autoencoder von Kelly und Knottenbelt (2015a) (eigene Darstellung).

das neuronale Netzwerk die Eingabe als Ausgabe reproduzieren. Der DAE soll hingegen die Eingabe entauschen.

Weiterhin wird die DAE-Architektur von Bonfigli, Felicetti et al. (2018) verwendet. Dort wird statt einer Mittelwertbildung der Median verwendet, um die Teilergebnisse zusammenzuführen.

3.6.4 Rekurrente Netze

Rekurrente Neuronale Netze (RNN) eignen sich insbesondere für die Verarbeitung von sequenziellen Daten, da eine RNN-Schicht die einzelnen Werte der Sequenz nacheinander einliest und sich damit ohne Änderung der lernbaren Parameter auf eine beliebige Sequenzlänge anpassen kann (vgl. Goodfellow, Bengio et al., 2016, Kapitel 10).

In allen bekannten RNN-basierten NILM-Ansätzen wird die vollständige Ausgabesequenz an die nächste Schicht weitergegeben. Die Alternative, nur die letzte Ausgabe aus dem letzten Berechnungsschritt als Ergebnis der Schicht weiterzugeben, wird nicht verwendet.

Eine schrittweise Verarbeitung bedingt jedoch, dass während des Einlesens Daten bis zum Ende der Sequenz nicht berücksichtigt werden. Um dem zu entgegen, werden bidirektionale RNN-Schichten verwendet, in denen parallel zu den vorwärtsgerichteten RNN-Einheiten zusätzlich rückwärtsgerichtete RNN-Einheiten die Daten vom Sequenzende beginnend einlesen. Die Ausgabe dieser bidirektionalen RNN-Schichten besteht aus der Aneinanderreihung der Ergebnisse der vorwärts- und rückwärtsgerichteten RNN-Einheiten.

Ansatz von Mauch und Yang (2015)

Im Ansatz von Mauch und Yang (2015) werden zwei bidirektionale RNN-Schichten mit jeweils 140 Long Short-Term Memory (LSTM)-Einheiten (Hochreiter und Schmidhuber, 1997) in jeder Richtung verwendet. Ergänzt werden die bidirektionalen RNN-Schichten mit einer Ein- und Ausgabeschicht. In der Eingabeschicht wird der Eingabewert an einer jeweiligen Position mit den vorherigen 9 Werten zu einem Vektor ergänzt. Die Ausgabe der letzten RNN-Schicht wird punktweise durch eine erlernte lineare Projektion auf den Ausgabewert transformiert, welches einer für jeden Zeitpunkt unabhängig angewandten vollverbundenen linearen Schicht entspricht. Die Struktur des Netzes ist vergleichbar mit dem rekurrenten Netz von Kelly und Knottenbelt (2015a) in Abbildung 3.11.

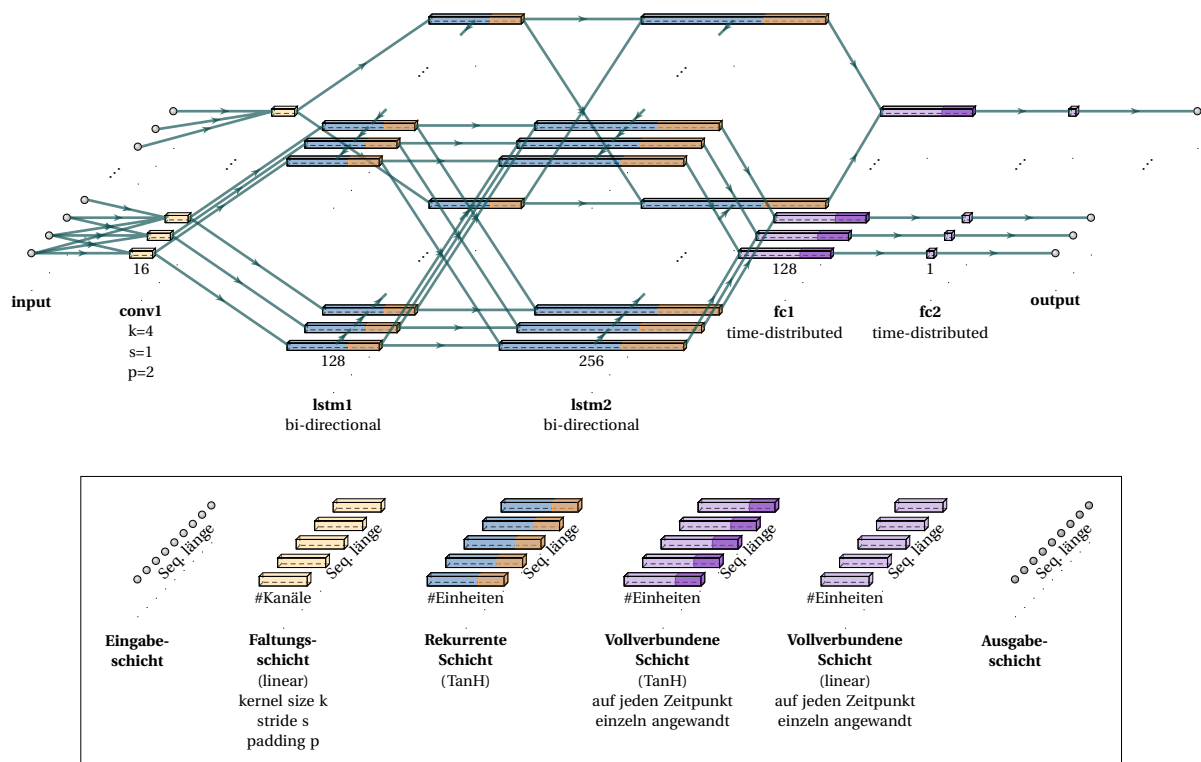


Abbildung 3.11: Das Rekurrente Neuronale Netz mit Long Short-Term Memory von Kelly und Knottenbelt (2015a) (eigene Darstellung).

Ansatz von Kelly und Knottenbelt (2015a)

Der RNN-Ansatz von Kelly und Knottenbelt (2015a), wie in Abbildung 3.11 dargestellt, verwendet ebenfalls zwei bidirektionale RNN-Schichten, eine Schicht mit 128 LSTM-Einheiten gefolgt von einer mit 256 LSTM-Einheiten (jeweils in beiden Richtungen). Für die Vorverarbeitung wird jedoch eine Faltungsschicht mit 16 linearen Einheiten und einer Filtergröße von 4 verwendet. Zur Ausgabe werden für jeden Zeitpunkt unabhängig zwei vollverbundene Schichten auf die Ausgabe der zweiten RNN-Schicht angewandt: Eine vollverbundene Schicht mit 128 Einheiten und Tangens hyperbolicus als Aktivierungsfunktion, gefolgt von einer linearen Projektion auf einen Ausgabekanal (geschätzte Wirkleistung des Zielgeräts).

Ansatz von He und Chai (2016)

He und Chai (2016) stellen ein rekurrentes neuronales Netz vor, welches wie die vorigen Ansätze im Wesentlichen aus zwei bidirektionalen rekurrenten Schichten mit LSTM-Einheiten besteht. Die Eingabesequenz wird jedoch parallel von mehreren unterschiedlich großen Faltungen vorverarbeitet. Dabei werden je 32 Faltungen mit den Filtergrößen 3, 5 und 7 berechnet und die Ergebnisse aneinandergelagert. Die rekurrente Schicht besteht aus 128 Einheiten in jeder Richtung. Nach den rekurrenten Schichten verarbeitet eine vollverbundene Schicht punktweise jeden einzelnen Zeitpunkt zum Endergebnis.

Ansätze von Paulo Marques do Nascimento (2016)

Paulo Marques do Nascimento (2016) untersucht u. a. zwei Ansätze mit bidirektionalen RNN-Schichten. Bei einem Ansatz werden die RNN-Schichten aus LSTM-Einheiten gebildet, beim zweiten Ansatz bestehen die RNN-Schichten aus einfacheren Gated Recurrent Unit (GRU)-Einheiten (siehe Abschnitt 2.5.8). Die Architektur des Netzes gleicht strukturell dem Netz von Kelly und Knottenbelt (2015a) in Abbildung 3.11. Zur Vorverarbeitung kommt jeweils eine Faltungsschicht mit 8 Filtern und einer Filtergröße von 5 Werten zum Einsatz. In beiden Ansätzen werden zwei bidirektionale RNN-Schichten verwendet. Die erste RNN-Schicht besteht aus 64 Einheiten und die zweite Schicht aus 128 Einheiten (in jeder Richtung). Vergleichbar zu den vorherigen Ansätzen findet eine Nachverarbeitung durch eine punktweise Anwendung einer vollverbundenen Schicht statt.

Ansätze von Krystalakos et al. (2018)

Die Arbeit von Krystalakos et al. (2018) basiert auf den beiden rekurrenten neuronalen Netzen von Paulo Marques do Nascimento (2016) und evaluiert damit einen rekurrenten

Ansatz mit GRU-Einheiten und einen mit LSTM-Einheiten. Krystalakos et al. (2018) fügen jedoch nach jeder rekurrenten Schicht und nach der ersten vollverbundenen Schicht je eine Dropout-Schicht ein.

3.6.5 Faltende neuronale Netze

Faltende neuronale Netze (engl. *convolutional neural networks (CNN)*) bestehen vorwiegend aus mehreren hintereinandergereihten faltenden Netzschichten (siehe Abschnitt 2.5.8). Innerhalb einer Schicht werden identische Faltungsoperationen entlang der Raum- oder Zeitdimensionen durchgeführt. Die Parameter der Faltungsschicht sind somit positionsunabhängig und werden, im Vergleich zu vollverbundenen Schichten, im Training effizienter erlernt, sofern Merkmale an unterschiedlichen Positionen auftreten können.

Jede Einheit liest ihre Eingabe aus einem begrenzten lokalen Wahrnehmungsbereich aus, so dass Zusammenhänge zwischen räumlich oder zeitlich weiter entfernten Werten sukzessive erst nach mehreren Schichten verarbeitet werden können. Faltende neuronale Netze setzen sich daher in der Regel aus einer hohen Anzahl von faltenden Schichten zusammen. In den vorherigen Ansätzen werden zwar vereinzelt faltende Schichten eingesetzt, jedoch werden diese lediglich als Vor- oder Nachverarbeitungsstufe verwendet.

Faltendes neuronales Netz von Paulo Marques do Nascimento (2016)

Eines der fünf von Paulo Marques do Nascimento (2016) vorgestellten Netzarchitekturen zur Disaggregation ist ein faltendes neuronales Netz, so wie in Abbildung 3.12 dargestellt. Es besteht aus vier Blöcken mit je 3 Faltungsschichten, jeweils gefolgt von einer Bündelschicht (*Max-Pooling*). Nach den Faltungsblöcken berechnet eine vollverbundene Schicht das Endergebnis. Die Faltungsschichten haben eine Filtergröße von 3 mit Ausnahme der ersten Faltungsschicht, die eine Filtergröße von 5 besitzt. Die drei Faltungsschichten im ersten Block besitzen jeweils 8 Filter. Nach jedem Block verdoppelt sich die Anzahl der Filter. Die Bündelschicht berechnet das Maximum aus 4 benachbarten Werten (Max Pooling) von jedem zweiten Zeitpunkt (stride 2), sodass sich die Zeitauflösung nach dem Bündeln halbiert.

Sequence-to-Sequence und Sequence-to-Point

Die zwei von Zhang et al. (2016) verwendeten Netzarchitekturen Sequence-to-Sequence (S2S) und Sequence-to-Point (S2P) benutzen fünf hintereinandergereihte Faltungsschichten, gefolgt von zwei vollverbundenen Schichten. Lediglich in der letzten Schicht unterscheiden sich die Ansätze: Sequence-to-Sequence gibt eine Schätzung der zur Gesamtlastsequenz

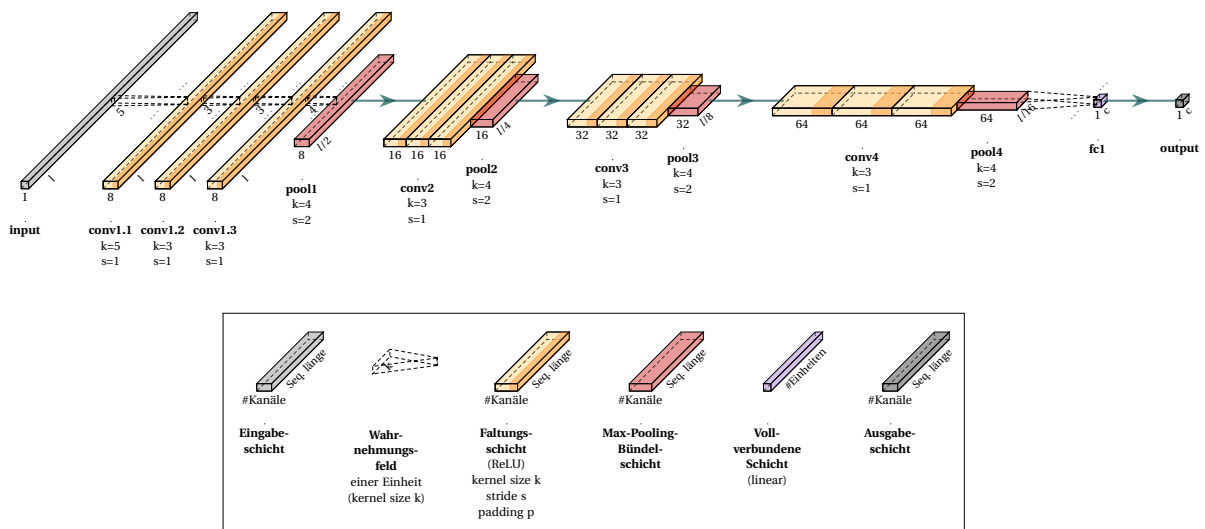


Abbildung 3.12: Faltendes neuronales Netz von Paulo Marques do Nascimento (2016) (eigene Darstellung).

korrespondierenden vollständigen Einzellastsequenz aus, während Sequence-to-Point lediglich die Last in einem Messintervall des Zielgerätes schätzt. Wie in Abbildung 3.13 dargestellt, verringert sich die Filtergröße der faltenden Schichten von anfänglich 10 auf 5, während die Anzahl der Filter von anfänglich 30 auf 50 steigt. Die erste vollverbundene Schicht im Anschluss hat 1024 Einheiten und eine ReLU-Aktivierung. Die Anzahl der Einheiten der zweiten vollverbundenen Schicht hängt von dem Ansatz ab.

Mit Sequence-to-Point vermeiden Zhang et al. (2016) unscharfe Endergebnisse, die durch die Mittelwertbildung entstehen, da bei Sequence-to-Point für jedes Messintervall ein eindeutiges Teilergebnis existiert.

Das Sequence-to-Point-Netzwerk wird auch von Krystalakos et al. (2018) in ihren Disaggregationsexperimenten eingesetzt, allerdings wird die Eingabesequenzlänge geräteabhängig auf zwischen 50, 100 oder 200 Messwerte reduziert. In deren Ansatz werden zusätzliche Dropout-Schichten nach jeder Faltungsschicht und nach der ersten vollverbundenen Schicht eingefügt.

Convolutional Encoder Decoder Network

Die Struktur des Convolutional Encoder Decoder Network (CEDN) von Garcia und Macabebe (2017) ist ähnlich dem des Denoising Autoencoder aufgebaut, jedoch verwendet das CEDN faltende statt vollverbundene Schichten. Die Eingabe wird durch zwei Blöcke mit jeweils einer Faltung und einer Max-Pooling-Bündelung auf insgesamt ein Viertel der ursprünglichen zeitlichen Auflösung reduziert. Anschließend wird das Ergebnis durch zwei Blöcke mit

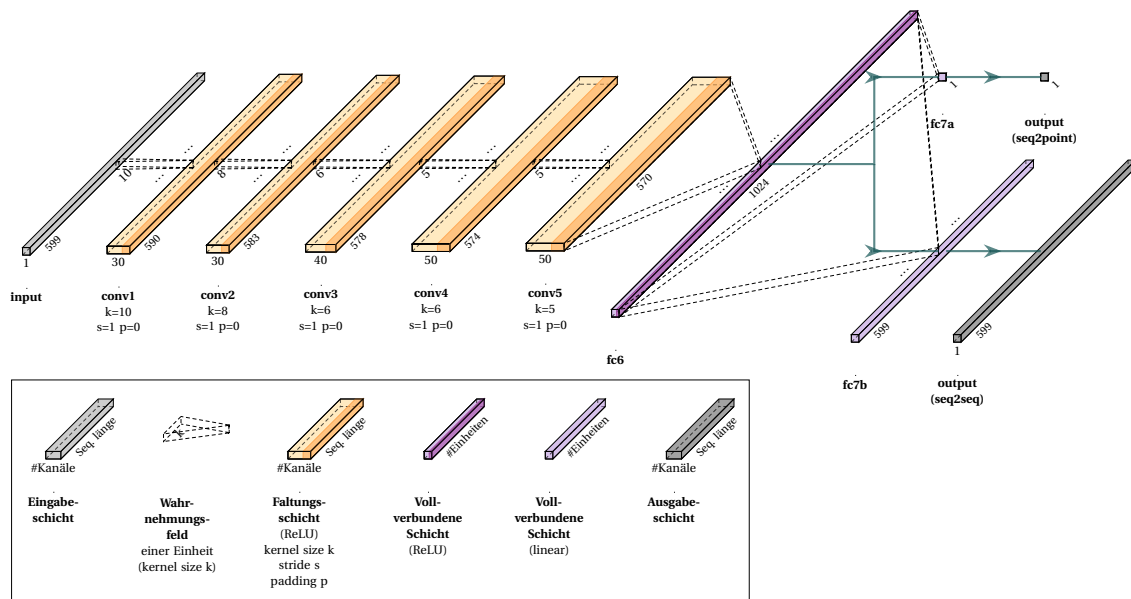


Abbildung 3.13: Sequence-to-Sequence (S2S) und Sequence-to-Point (S2P) von Zhang et al. (2016) (eigene Darstellung).

jeweils einer Faltung und einer Auflösungsverdopplung auf die ursprüngliche zeitliche Auflösung hochskaliert. Schließlich erzeugt eine letzte Faltungsschicht den Zielgerätestrang und reduziert dabei die Anzahl an Kanälen auf eins.

Dilated Fully Convolutional Network

Die Netzarchitektur von Brewitt und Goddard (2018) gehört zum ersten Ansatz, der auf vollverbundene Schichten als nachgelagerte Netzschichten verzichtet und stattdessen vollständig auf faltenden Netzschichten basiert. Zudem ist es der erste Ansatz, welcher in den Faltungsschichten Dilation (Yu und Koltun, 2016) einsetzt. Bei dilatierten Faltungen werden Eingangswerte übersprungen, sodass bei gleichbleibender Filtergröße ein breiterer (Zeit-)raum verarbeitet wird. Beispielsweise wird bei einer Dilation von 2 jeder zweite Wert übersprungen, eine normale Faltung hat eine Dilation von 1.

Das Netzmodell von Brewitt und Goddard (2018) ist in Abbildung 3.15 dargestellt. Es besteht aus einer ersten Faltungsschicht als Vorverarbeitung, gefolgt von 9 hintereinandergereihten Faltungsschichten, deren Dilatation mit jeder Schicht verdoppelt wird. Zum Schluss wird jeder Ausgabewert punktweise (Filtergröße 1) durch zwei Faltungsschichten auf das Endergebnis abgebildet. Mit Ausnahme der letzten Schicht werden alle Faltungen mit 128 individuellen Filtern durchgeführt.

3.6 Netzarchitektur bestehender Ansätze

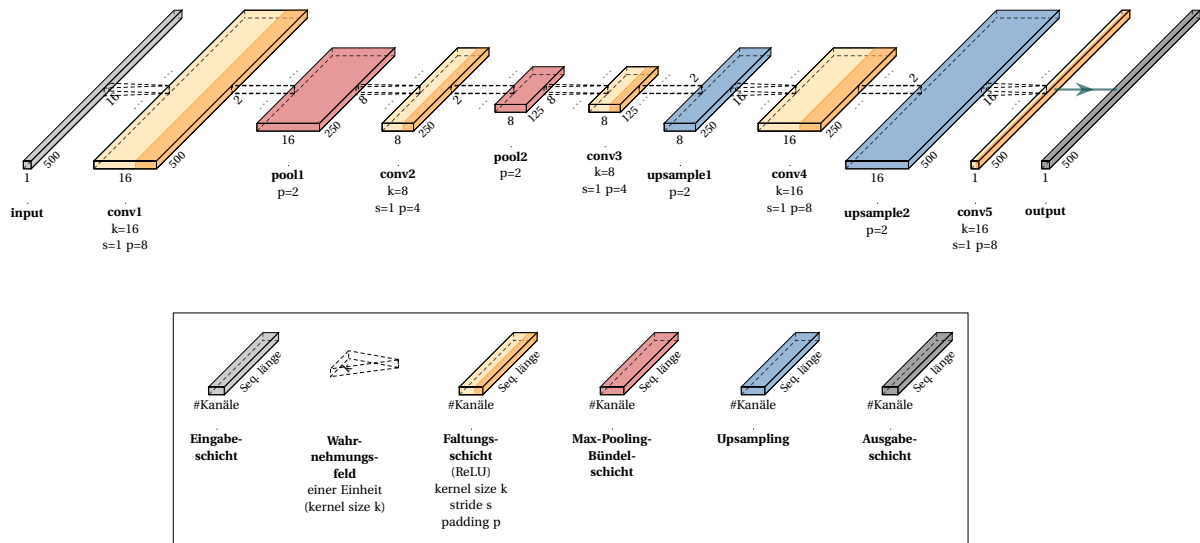


Abbildung 3.14: Convolutional Encoder Decoder Network (CEDN) von Garcia und Macabebe (2017) (eigene Darstellung).

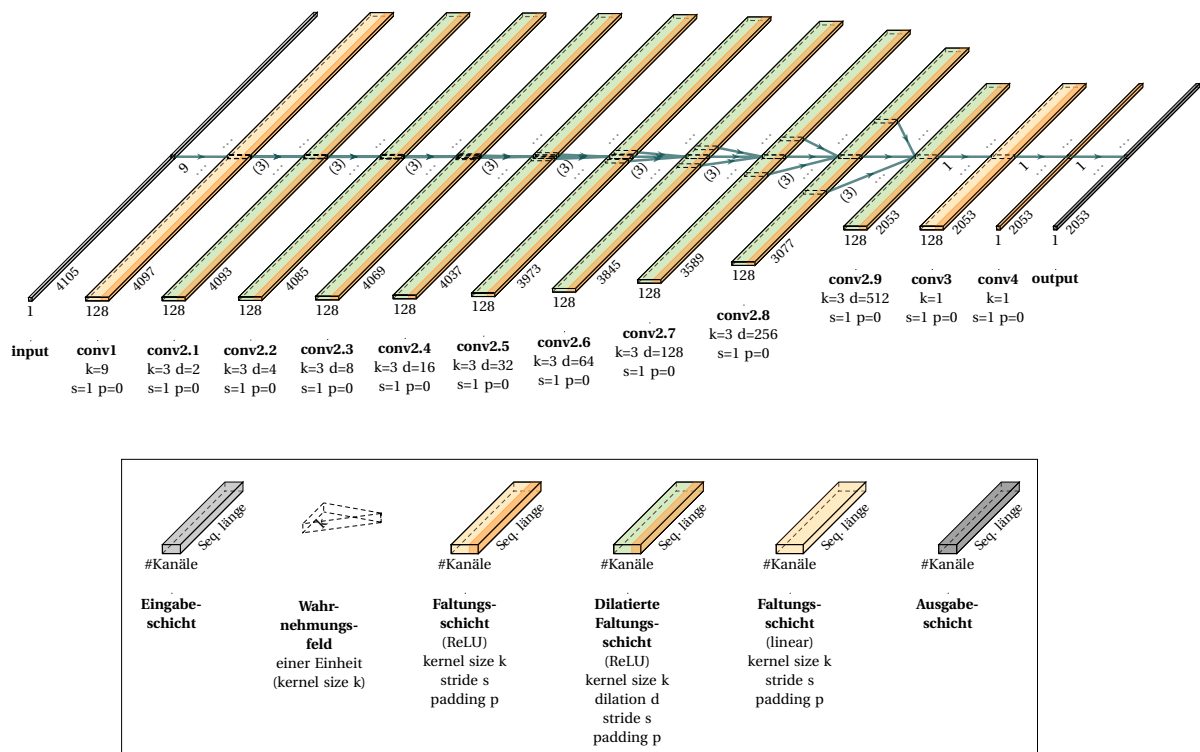


Abbildung 3.15: Dilated Fully Convolutional Network (DFCN) von Brewitt und Goddard (2018) (eigene Darstellung).

Die vollständig auf dilatierten Faltungsschichten basierende Architektur ermöglicht die Skalierung des Netzes auf beliebig lange Eingabesequenzen (ab einer bestimmten Mindestlänge). Bei einer Eingabelänge von $2052 + n$ wird eine Sequenz von n (für $n \geq 1$) Werten ausgegeben. Für größere n werden Zwischenergebnisse der früheren Faltungsschichten wiederverwendet. Dies ist verglichen zur Anwendung auf ein verschobenes Eingabefenster effizienter.

Obwohl der Ansatz in der Regel zur Effizienzsteigerung eine Sequenz ausgibt, ist die Architektur konzeptionell vergleichbar mit Sequence-to-Point, da für jeden Zeitpunkt der Ausgabewert eindeutig ist.

Subtask Gated Network

Das Subtask Gated Network (SGN) von Shin et al. (2018) verwendet zwei Sequence-to-Sequence-Netze von Zhang et al. (2016) parallel. Dabei spezialisiert sich ein Netz auf die Klassifikation des Zielgerätezustands und das zweite Netz auf die Regression der Lastwerte. Dazu bildet das Sigmoid der Ausgabe des Klassifikationsnetzes eine Schranke (Gate), mit der die Ausgabe des Regressionsnetzes multipliziert wird. Beide Netze erhalten die gleiche Eingabesequenz.

Das Klassifikationsnetz wird mit einem Cross-Entropy-Loss trainiert. Dabei wird der Zielgerätezustand mittels Grenzwertverfahren aus dem Einzellastgang berechnet. Das Training des Regressionsnetzwerks erfolgt mittels L_2 -Loss zwischen dem tatsächlichen Lastwert und dem Produkt aus der Regressionsschätzung und der Schätzung des Klassifikationsnetzes. Mit dieser Verlustfunktion wird das Regressionsnetz kaum trainiert, sofern das Klassifikationsnetz den Gerätezustand als Aus erkennt (Shin et al., 2018). Das Training des gesamten Netzes erfolgt mit der Summe beider Verlustfunktionen.

3.6.6 Faltende neuronale Netze mit Querverbindungen

In faltenden neuronalen Netzen werden Querverbindungen (engl. *skip connections*) als Seitenkanäle eingesetzt, um Informationen über mehrere Schichten hinweg zu transportieren. Dazu haben sich Residualblöcke (engl. *residual block*) (He, Zhang et al., 2015a, 2016) und das U-Net (Ronneberger et al., 2015) als Architekturmuster etabliert:

Wenn in neuronalen Netzen Informationen über mehrere Schichten hinweg konserviert werden sollen, muss das Netz dies aktiv erlernen. Dies wird mit zunehmender Anzahl an Netzschichten schwieriger und begrenzt die Anzahl an Schichten in der Architektur. He, Zhang et al. (2015a) schlagen daher vor, das Netzwerk aus Residualblöcken aufzubauen, in denen mittels einer Querverbindung die Eingabe des Blocks auf die Ausgabe aufaddiert

wird. Somit ist die Durchleitung von Information über beliebig viel Schichten hinweg bereits gewährleistet. (He, Zhang et al., 2016)

In einer Autoencoder-Architektur steigt der Abstraktionsgrad zunächst mit zunehmender Netztiefe bei gleichzeitiger Abnahme der räumlichen oder zeitlichen Auflösung. Nach dem Flaschenhals (engl. *bottleneck*) erfüllt das Netz eine Generierungsaufgabe und der Abstraktionsgrad sinkt mit zunehmender Auflösung. Dabei müssen auch die Informationen über höheraufgelöste Details durch den Flaschenhals auf der höchsten Abstraktionsstufe. Das U-Net von Ronneberger et al. (2015) vermeidet dies, indem es Querverbindungen zwischen Schichten der gleichen Abstraktionsstufe einführt (siehe Abbildung 6.24 als Beispiel einer U-Net-Architektur). Damit werden Informationen, die zwar für die detaillierte Generierung des Ergebnisses relevant sind, jedoch nicht zur Abstraktion der Eingabe dienen, über die Querverbindung transportiert.

Faltendes neuronales Netz mit Residualblöcken von Paulo Marques do Nascimento (2016)

Das Faltungsnetz mit Residualblöcken von Paulo Marques do Nascimento (2016) besteht aus einer Faltungsschicht als Eingangsstufe mit anschließenden sechs Residualblöcken. Nach jedem zweiten Residualblock wird die Filteranzahl verdoppelt, von anfänglich 16 Filter auf 64. Am Ende der Verarbeitungskette wird mittels Max Pooling die zeitliche Auflösung halbiert und schließlich das Ergebnis mit einer vollverbundenen Schicht auf die Laststufenklassen abgebildet.

Gated Linear Unit

Chen, Wang et al. (2018) führen in ihrem Disaggregationsansatz Gated Linear Units (GLU) und Residualblöcke ein. Ein Gated Linear Unit ist ein Faltungsblock, der jeweils aus einer Faltung für die Lastregression und einer Faltung für die Zustandsklassifikation besteht. Das Ergebnis der Klassifikation wird mit einer Sigmoid-Funktion auf den Regressionswert multipliziert und bildet ähnlich zum Ansatz von Shin's Subtask Gated Network (SGN) eine Schranke. Dieses Schranke wird jedoch nicht nur einmalig auf die Lastgangsschätzung angewandt. Chen, Wang et al. reihen in der Netzarchitektur mehrere GLU-Blöcke hintereinander und wenden somit mehrfach die Schrankenfunktionalität an.

Die neuronale Netzarchitektur von Chen, Wang et al. (2018) besteht aus drei GLU-Blöcken, jeweils gefolgt von Max-Pooling-Bündelschichten, um die zeitliche Auflösung zu halbieren. In der Mitte der Netzarchitektur befindet sich eine vollverbundene Schicht, deren Ergebnis von drei Residualblöcken und einer finalen vollverbundenen Schicht auf den Ziellastgang abbildet wird.

U-Net von Barsim und Yang (2018)

Im Disaggregationsansatz von Barsim und Yang (2018) wird ein U-Net verwendet, das aus der Eingangslastsequenz die Zustandssequenz des Zielgeräts schätzt.

Die U-Net-Architektur von Barsim und Yang (2018) ist symmetrisch aufgebaut und besteht zunächst aus jeweils einer Faltungsschicht als Vor- und Nachverarbeitung. Auf der Eingangsseite verarbeiten fünf Residualblöcke die Lastgangssequenz zu einer Zwischenrepräsentation, wobei die Auflösung nach jedem Block durch eine Max-Pooling-Bündelung reduziert wird. In der Mitte existiert ein zusätzlicher Residualblock zur Transformation der Zwischenrepräsentation. Auf der Ausgangsseite sind die 5 Residualblöcke gespiegelt, die die Zwischenrepräsentation schrittweise auf die ursprüngliche zeitliche Auflösung erhöhen. Dabei erhalten diese Residualblöcke sowohl das Zwischenergebnis des vorherigen Blocks als auch das Zwischenergebnis des gespiegelten Blocks der Eingangsseite, was charakteristisch für eine U-Net-Architektur ist.

Im Gegensatz zu Brewitt's DFCN verwendet diese Architektur einen konstanten Dilatationsfaktor (Yu und Koltun, 2016) für alle Faltungsschichten.

Weitere Besonderheiten verglichen zu anderen Ansätzen ist die Verwendung von Batch-Normalisierung (siehe Abschnitt 2.5.5) sowie Rauscheinspeisung (siehe Abschnitt 2.5.4) als Regularisierungstechniken.

3.7 Ausgabeformat

Als Endergebnis des Disaggregationsprozesses wird der Lastgang oder die Zustandszeitreihe eines Zielgeräts erwartet. Die Ausgabe des neuronalen Netzes kann jedoch eine beliebige Zwischenrepräsentation annehmen, solange sich daraus der Lastgang bzw. die Zustandszeitreihe in der Nachverarbeitung und Zusammenführung rekonstruieren lässt. In der Literatur wurden bisher folgende Möglichkeiten für die Repräsentation der Teilergebnisse verwendet.

Lastsequenz Zur Schätzung des Gerätelastgangs ist die naheliegendste Zwischenrepräsentation die Sequenz von Lastwerten, analog zur Eingabe des neuronalen Netzes. Bei den betrachteten Anwendungsfällen von NILM genügt meist die Schätzung von Wirkleistungswerten.

Die Ausgabesequenzlänge ist bei den meisten Ansätzen identisch zur Eingabesequenzlänge (Barsim und Yang, 2018; He und Chai, 2016; Kelly und Knottenbelt, 2015a; Krystalakos et al., 2018; Mauch und Yang, 2015; Zhang et al., 2016).

Zwei jüngere Ansätze geben eine verglichen zur Eingangssequenzlänge kürzere Lastsequenz aus. Zur Reproduktion des Gerätelastgangs steht somit verhältnismäßig mehr Kontext zur Verfügung. (Chen, Wang et al., 2018; Shin et al., 2018).

Wie im vorigen Abschnitt diskutiert, überlappen sich die Ausgabelastsequenzen bei kleinen Schrittweiten, sodass eine Zusammenführung der Teilergebnisse erforderlich wird.

Einzellastwert Statt der Schätzung von mehreren Lastwerten gleichzeitig schlagen Zhang et al. (2016) im *Sequence-to-Point* genannten Ansatz vor, in jedem Schritt nur einen Lastwert auszugeben. Der gesuchte Lastwert korrespondiert zum Zeitpunkt in der Mitte der Eingabelastsequenz. Dieses Vorgehen soll die bei der Zusammenführung entstehende Unschärfe vermeiden.

Aktivitätswahrscheinlichkeiten Für die Schätzung der Zustandszeitreihe, bei der nur zwischen den Zuständen An und Aus unterschieden wird, ist die naheliegende Zwischenrepräsentation eine Sequenz von Wahrscheinlichkeitswerten. Die Werte drücken aus, wie sicher das neuronale Netz glaubt, dass das Zielgerät im jeweiligen Messintervall an ist. In der Literatur verwenden Barsim und Yang (2018) diese Form der Zwischenrepräsentation.

Wahrscheinlichkeiten für Leistungsaufnahmeklassen Die binäre Geräteaktivität lässt sich auf mehrere Gerätezustände verallgemeinern, womit sich mehrstufige Lastkurven ausdrücken lassen. Paulo Marques do Nascimento (2016) findet für seine Verfahren zunächst Häufungspunkte der Lastwerte jedes seiner betrachteten Zielgeräte. Die Häufungspunkte dienen als diskrete Gerätezustände, deren Wahrscheinlichkeiten das neuronale Netz in jedem Messintervall schätzen soll. Die Disaggregation wird damit als Klassifikationsproblem aufgefasst.

Lastblöcke Eine der drei von Kelly und Knottenbelt (2015a) vorgeschlagenen Disaggregationsverfahren abstrahiert Geräteaktivitätszyklen als rechteckige Lastblöcke mit uniformer Leistung. In den jeweiligen Haushaltslastsequenzen soll das neuronale Netz den Start- und Endzeitpunkt sowie die Durchschnittslast des ersten Lastblocks des Zielgeräts erkennen.

3.8 Zusammenführungsstrategie

In der Regel wird das neuronale Netz für die Disaggregation auf zeitlich überlappende Sequenzen der Haushaltslast angewandt und liefert überlappende Teilergebnisse zurück. Daher muss der Gerätelastgang aus den einzelnen Teilergebnissen berechnet werden. Dafür werden in der Literatur folgende Optionen verwendet:

Mittelwertbildung Für die Zusammenführung von Lastsequenzen wird bei bestehenden Verfahren am häufigsten eine Mittelwertbildung verwendet. Dazu wird für jeden Zeitpunkt jeweils der Mittelwert der Schätzungen aus allen relevanten Teilergebnissen gebildet.

Kelly und Knottenbelt (2015a) wenden diese Strategie für den Denoising Autoencoder und für das rekurrente Netz an. Zhang et al. (2016) verwenden diese Strategie nur für den Sequence-to-Sequence-Ansatz.

Medianbildung Bonfigli, Felicetti et al. (2018) bilden anstelle des Mittelwerts den Median aus den Teilergebnissen und konnten dadurch in manchen Fällen die Disaggregationsgenauigkeit verbessern.

Aneinanderreihen Falls sich die Teilergebnisse nicht überlappen, so ist für jeden Zeitpunkt das Ergebnis eindeutig und die Teilergebnisse können unverändert aneinandergereiht werden. Beispielsweise werden bei dem auf rekurrente Netze basierten Ansatz von Mauch und Yang (2015) die Lastgänge zeitpunktweise sequenziell eingelesen und ausgegeben, sodass für jeden Zeitpunkt eine eindeutige Schätzung der Gerätelast vorliegt. Ein anderes Beispiel ist der Sequence-to-Point-Ansatz von Zhang et al. (2016), bei dem zwar überlappende Haushaltslastsequenzen eingelesen werden, die Ausgabe jedoch lediglich aus einem Wert besteht, der die geschätzte Gerätelast in der Mitte der Sequenz repräsentiert.

Zusammenführen von rechteckförmigen Lastblöcken Für das Lastblock-Regressions-Netzwerk von Kelly und Knottenbelt (2015a) (siehe Abschnitt 3.6.2) wird der Lastgang des Zielgeräts in Form von rechteckigen Lastblöcken abstrahiert. Da sich bei diesem Ansatz die Eingangslastsequenzen überlappen, überlappen sich in der Regel auch die Schätzungen der Lastblöcke.

Kelly und Knottenbelt (2015a) verwenden daher folgende Heuristik für die Rücktransformation der Lastblockschätzungen in einen Gerätelastgang: Für jedes Messintervall des Zielgerätelastgangs werden jene Lastblöcke betrachtet, die dieses Messintervall überdecken. Der maximale Lastwert, der innerhalb 50 % dieser Lastblöcke liegt, wird als Schätzung für das Messintervall genommen, welches einer Medianbildung entspricht. Falls der dabei gefundene Lastwert einen Mindestwert unterschreitet, wird die Last für diesen Zeitpunkt auf 0 W gesetzt.

3.9 Trainingsverfahren und -parameter

Alle bekannten NILM-Ansätze mit tiefen neuronalen Netzen verwenden eine Variante des stochastischen Gradientenabstieg mit Trainingsbeispielchargen (Mini-Batch Stochastic Gradient Descent, siehe Abschnitt 2.5.2) als Optimierungsverfahren für die erlernbaren

Parameter des Netzes. In der Literatur wird entweder das Optimierungsverfahren Nesterov's Accelerated Gradient (NAG) oder A Method for Stochastic Optimization (ADAM) verwendet (siehe Abschnitt 2.5.2).

Optimierungsverfahren und Lernrate

Nesterov's Accelerated Gradient (NAG) wird in den Ansätzen von Kelly und Knottenbelt (2015a), Bonfigli, Felicetti et al. (2018) und Valenti et al. (2018) eingesetzt. Paulo Marques do Nascimento (2016) nutzt NAG ausschließlich für die faltenden neuronalen Netze. In allen Arbeiten ist der Dämpfungsfaktor auf den in Deep-Learning-Bibliotheken voreingestellten Wert von 0.9 belassen².

A Method for Stochastic Optimization (ADAM) kommt bei einer Vielzahl jüngerer NILM-Ansätze zum Einsatz (Brewitt und Goddard, 2018; Chen, Wang et al., 2018; Krystalakos et al., 2018; Shin et al., 2018) sowie bei den rekurrenten Netzen von Paulo Marques do Nascimento (2016). Barsim und Yang (2018) verwendet ADAM mit zusätzlichem Nesterov-Momentum.

Als initiale Lernrate für NAG ist 10^{-1} (Bonfigli, Felicetti et al., 2018; Paulo Marques do Nascimento, 2016) oder 10^{-2} (Kelly und Knottenbelt, 2015a) eingestellt.

Für ADAM ist die anfängliche Lernrate auf 10^{-3} (Brewitt und Goddard, 2018) oder 10^{-4} (Paulo Marques do Nascimento, 2016; Shin et al., 2018) gesetzt.

Bei den Ansätzen von Brewitt und Goddard (2018), Kelly und Knottenbelt (2015a) und Paulo Marques do Nascimento (2016) wird im Verlauf des Trainings die Lernrate um Faktor 10 reduziert.

Trainingschargengröße

Die Trainingschargengröße (*mini batch size*) beschreibt, wieviele Trainingsbeispiele (also Paare von Haushalts- und Gerätelastsequenzen) gleichzeitig während einer Trainingsiteration evaluiert werden, um den Durchschnittsgradienten für die Aktualisierung der Gewichtparameter zu erhalten (siehe Abschnitt 2.5.2).

Für eine höhere Auslastung der Graphikkarte als Rechenbeschleuniger wird eine möglichst große Charge bevorzugt. Die Chargengröße ist jedoch praktisch durch den Speicher der Graphikkarte beschränkt. Somit hängt die maximal mögliche Chargengröße von den verwendeten Rechenbeschleunigern, der Sequenzlänge und der Anzahl der Parameter des neuronalen Netzes ab.

²Der Ursprung dieses Wertes stammt aus der Veröffentlichung von Sutskever et al. (2013).

In der Literatur liegt die Chargengröße bei 16 (Shin et al., 2018), 32 (Chen, Wang et al., 2018), 64 (Kelly und Knottenbelt, 2015a) oder 256 (Brewitt und Goddard, 2018).

Trainingsdauer

Wie lange ein neuronales Netz trainiert wird, wird als Anzahl an Trainingsiterationen oder als Anzahl an Trainingsepochen angegeben. Dabei beschreibt eine Iteration die Evaluation einer Trainingscharge sowie die Aktualisierung der Netzparameter. Eine Epoche umfasst die Anzahl an Iterationen, die nötig ist, um den Trainingsdatensatz einmal vollständig durchzuarbeiten.

Leider wird in der Literatur die Trainingsdauer häufig nicht angegeben. Brewitt und Goddard (2018) trainieren ihre Modelle bis zu 200 Epochen lang. Bonfigli, Felicetti et al. (2018) geben eine maximale Trainingsdauer von 200'000 Iterationen an. Beide Ansätze brechen das Training ab, sofern sich das Netz mehrere Epochen bzw. Iterationen lang auf einem Validationsdatensatz nicht verbessert.

Verlustfunktion

Als Verlustfunktion (siehe Abschnitt 2.5.7) geben die meisten Ansätze den L_2 -Loss bzw. den Mean Squared Error (MSE) an (Garcia und Macabebe, 2017; Kelly und Knottenbelt, 2015a; Mauch und Yang, 2015).

Neuronale Netze, die Laststufen- bzw. Gerätezustandswahrscheinlichkeiten ausgeben, wie die Ansätze von Barsim und Yang (2018) und Paulo Marques do Nascimento (2016), verwenden einen Cross-Entropy-Loss bzw. NLL-Loss.

Shin et al. (2018) kombinieren im Ansatz Subtask Gated Network den L_2 -Loss und den NLL-Loss für das jeweilige Teilnetz, welches die Lastregressions- bzw. Zustandsklassifikationsaufgabe erfüllt.

3.10 Datensätze

Zur Evaluation der Ansätze werden bei bestehenden Deep-Learning-basierten NILM-Ansätzen am häufigsten die Datensätze REDD (Kolter und Johnson, 2011) und UK-DALE (siehe Abschnitt 4.3) verwendet.

Barsim und Yang (2018), Kelly und Knottenbelt (2015a) und Krystalakos et al. (2018) verwenden dabei UK-DALE.

Chen, Wang et al. (2018), Garcia und Macabebe (2017), Mauch und Yang (2015) und Paulo Marques do Nascimento (2016) verwenden REDD.

Bonfigli, Felicetti et al. (2018), Shin et al. (2018) und Zhang et al. (2016) evaluieren ihre Ansätze unabhängig auf beiden Datensätzen.

Brewitt und Goddard (2018) verwendet einen unveröffentlichten Datensatz. Bonfigli, Felicetti et al. (2018) verwendet AMPds (Makonin et al., 2016) als dritten Datensatz.

Die Datensätze werden in Disaggregationsexperimenten im Rahmen von zwei Szenarien evaluiert:

Bekannte-Gebäude-Szenario Hier wird die Disaggregation auf Daten von Gebäuden und Geräten durchgeführt, die Teil des Trainingsdatensatzes sind und somit dem Netz bekannt sein sollten. Beispielsweise werden 80 % der Lastgangsaufzeichnungen jedes Gebäudes zum Training verwendet und die verbleibenden 20 % werden als Testdaten betrachtet. Dieses Szenario dient lediglich zur Validierung, ob das Konzept und das Training des Ansatzes prinzipiell funktionieren. Da Lastgangsmessungen der verwendeten Haushaltsgeräte in der Regel nicht vorliegen, stellt dieses Szenario keinen typischen Anwendungsfall für die Disaggregation dar.

Unbekannte-Gebäude-Szenario In diesem Szenario wird die Disaggregation auf Daten unbekannter Gebäude ausgeführt, deren Lastgänge zuvor nicht beim Training mitverwendet wurden. Dazu werden zum Training je Geräteklasse ein oder zwei Gebäude als Testgebäude deklariert. Ausschließlich die Daten der verbleibenden Gebäude im Datensatz werden zum Training verwendet. Dies entspricht einem realistischen Anwendungsfall für Disaggregationsansätze, da die Ansätze auf Haushalte angewandt werden sollen, zu denen keine spezifischen Daten vorliegen.

Alle Deep-Learning-Ansätze in der NILM-Literatur untersuchen das Unbekannte-Gebäude-Szenario (Brewitt und Goddard, 2018; Chen, Wang et al., 2018; Garcia und Macabebe, 2017; Krystalakos et al., 2018; Shin et al., 2018; Zhang et al., 2016).

Das Bekannte-Gebäude-Szenario wird in einigen Ansätzen zusätzlich evaluiert (Barsim und Yang, 2018; Bonfigli, Felicetti et al., 2018; Kelly und Knottenbelt, 2015a; Mauch und Yang, 2015; Paulo Marques do Nascimento, 2016).

3.11 Fazit und Auswahl der zu untersuchenden Verfahren als Stand der Technik

Der Stand der Technik ist objektiv schwer zu bestimmen, da sich bisher kein einheitlicher Aufbau für Disaggregationsexperimente etabliert hat. Ein Hindernis für einen solchen

Standard ist die Nichtverfügbarkeit eines ausreichend großen Datensatzes mit vielen unterschiedlichen Geräteklassen und einer Vielzahl unterschiedlicher Gerätemodelle pro Geräteklasse.

Um dennoch den Stand der Technik basierend auf der Literatur abzuschätzen, werden die Ansätze bezüglich ihres mittleren absoluten Fehlers (MAE) im Unbekannte-Gebäude-Szenario verglichen.

In der Literatur dienen am häufigsten zwei Veröffentlichungen, die auch in dieser Arbeit evaluiert werden, als Vergleichsreferenz:

Denoising Autoencoder (DAE) Dieser Ansatz von Kelly und Knottenbelt (2015a) ist eine der ersten Veröffentlichungen zu NILM mit tiefen neuronalen Netzen und wird von vielen Folgearbeiten als Referenz betrachtet.

Sequence-to-Sequence (S2S) und Sequence-to-Point (S2P) Die beiden Ansätze von Zhang et al. (2016) stellen einen Fortschritt gegenüber DAE dar, die jeweils niedrigere MAE-Werte erreichen.

In dieser Arbeit werden nur die Ansätze als Stand der Technik erachtet, die nachweislich einen geringeren Fehler (MAE) bei der Disaggregation aufweisen als einer der Ansätze von Zhang et al. (2016). Somit werden folgende Ansätze nicht zum aktuellen Stand der Technik aufgefasst:

Die Ansätze von Mauch und Yang (2015) und Paulo Marques do Nascimento (2016) führen keine vergleichenden Disaggregationsexperimente mit Ansätzen anderer Autoren durch.

Die Disaggregationsansätze von He und Chai (2016), Garcia und Macabebe (2017) und Bonfigli, Felicetti et al. (2018) zeigen jeweils eine Verbesserung gegenüber DAE, vergleichen sich jedoch nicht mit S2S oder S2P.

Für die Erweiterung des DAE von Bonfigli, Felicetti et al. (2018) wird eine Verbesserung gegenüber DAE nachgewiesen, jedoch fehlt ein Vergleich zu S2S oder S2P.

Die Ansätze von Krystalakos et al. (2018) werden mit einer Variante von S2P verglichen, dessen Wahrnehmungsbereich reduziert wurde. Dennoch konnte S2P in den meisten Experimenten einen niedrigeren MAE erzielen.

Damit verbleiben folgende Ansätze, die zum aktuellen Stand der Technik gehörig betrachtet werden und die im Rahmen dieser Studie evaluiert werden:

Dilated Fully Convolutional Network (DFCN) Brewitt und Goddard (2018) zeigen, dass ihr Ansatz einen niedrigeren MAE erreicht als S2P von Zhang et al. (2016).

Subtask Gated Network (SGN) Shin et al. (2018) vergleichen ihren Ansatz mit S2P und DAE und übertreffen die Disaggregationsqualität beider Ansätze mit einem niedrigeren MAE.

Gated Linear Unit (GLU) Chen, Wang et al. (2018) vergleichen sich auf dem Datensatz REDD. GLU ist dabei besser als S2P von Zhang et al.

Kapitel 4

Szenario und Aufbau der Experimente

Dieses Kapitel legt mit der Beschreibung des Szenarios und des Aufbaus der Experimente die Grundlage für eine systematische und gründliche Untersuchung der Leistungsfähigkeit der entsprechend der Analyse in Abschnitt 3.11 ausgewählten Disaggregationsverfahren. Dazu wird zunächst in Abschnitt 4.1 das TestszENARIO zusammengefasst. Um eine Vergleichsbasis zu schaffen, werden die Disaggregationsexperimente mit dem in den folgenden Abschnitten definierten gemeinsamen Aufbau durchgeführt.

Die Auswahl der in dieser Studie betrachteten Haushaltsgeräte und Datensätze wird in den Abschnitten 4.2 und 4.3 getroffen und diskutiert. Dabei enthält Abschnitt 4.3 eine detaillierte Charakterisierung der Datensätze. Die Bereinigung der Datensätze sowie gemeinsame Vorverarbeitungsschritte sind in Abschnitt 4.4 beschrieben. In Abschnitt 4.5 ist die Aufteilung der Datensätze in Trainings- und Testmenge dargelegt. Die Methoden zur Erstellung von Trainingspaaren und die dabei erzielte Anreicherung der Trainingsdatenmenge werden in Abschnitt 4.6 besprochen. Der Bereich an betrachteten Auflösungen für Haushaltlastgänge ist durch gesetzliche und technische Beschränkungen bei der Messung und Übermittlung der Haushaltlastgänge definiert und wird in Abschnitt 4.7 erläutert. In Abschnitt 4.8 werden die Bewertungsmetriken benannt, die jeweils auf unterschiedliche Qualitätsaspekte der Disaggregation abzielen. Schließlich werden in Abschnitt 4.9 die den Experimenten zugrundeliegenden Software-Werkzeuge und die verwendete Hardware dieser Studie beschrieben.

4.1 Szenario

Das gewählte Szenario fokussiert auf die Energieverbrauchsdatendisaggregation für privat bewohnte Haushalte. Der Vorteil dieser Fokussierung liegt im Vorhandensein von Verbrauchsdaten vieler unterschiedlicher Gerätetypen.

Dabei beschränkt sich das Szenario auf die **Wirkleistungsaufnahme** und den **Aktivitätszustand** der elektrischen Haushaltsgeräte als relevante Information zur Energieverbrauchsoptimierung aber auch schützenswertes privatsphärengefährdendes Datum. Beide Werte hängen mittelbar mit der Gerätenutzung der Bewohner zusammen. Im Gegensatz zu anderen personenbezogenen Daten, wie Bewohneraktivitäten oder Schlaf- und Wachzyklus, sind die Lastgänge¹ einzelner Geräte direkt messbar und in den öffentlichen Datenbanken verfügbar.

Aus dem Lastgang ist der Aktivitätszustand eines Gerätes ableitbar. Dabei beschreibt der Aktivitätszustand, ob das elektrische Haushaltsgerät gegenwärtig Arbeit verrichtet und damit eine erhöhte Leistungsaufnahme aufweist oder ausgeschaltet ist bzw. sich mit einer geringen Leistungsaufnahme im Standbymodus befindet.

Bei den durchgeführten Experimenten wird jeweils untersucht, wie genau sich der zeitliche Verlauf des Aktivitätszustands sowie der Lastgang der Haushaltsgeräte aus unterschiedlich zeitlich aufgelösten Haushaltslastgängen mit Hilfe von Disaggregationsansätzen rekonstruieren lässt. Zur Konstruktion der Disaggregationsansätze werden öffentlich verfügbare Datensätze genutzt. Insbesondere werden mehrere Datensätze kombiniert zum Training der Ansätze verwendet. Für die Evaluation der Ansätze sind einzelne Gebäude aus den Datensätzen entnommen und deren Daten sind nicht Bestandteil des Trainingsdatensatzes.

Die Herausforderung für Disaggregationsverfahren liegt insbesondere in der Erkennung von aktiven Geräten und in der Schätzung der Lastgänge von Geräten im aktiven Zustand, da viele Haushaltsgeräte bezogen auf ihre gesamte Standzeit selten aktiv sind und Geräte im aktiven Zustand in der Regel komplexe Lastgänge aufweisen. Wenn ein Gerät aktiv ist, ist ein zeitlich zusammenhängender Arbeitsvorgang im Lastgang identifizierbar. Ein solcher zeitlich zusammenhängender und abgeschlossener Arbeitsvorgang sowie der damit assoziierte Lastgangsabschnitt wird als **(Geräte-)Aktivierung** bezeichnet.

In diesem Zusammenhang ist darauf hinzuweisen, dass sich durch die Einschränkung auf Haushaltslastgänge auch Einschränkungen für die Gültigkeit der Ergebnisse für andere Gebäudeszenarien ergeben:

- Werden Haushalte zu sogenannten Prosumern, d. h. nicht nur Konsumenten sondern auch Energieerzeugern, und sollte dabei keine gesonderte Messung erfolgen,

¹In dieser Arbeit wird der Lastgang stets als Synonym für den zeitlichen Verlauf der elektrischen Leistungsaufnahme verwendet.

stellen sich zusätzliche Herausforderungen, die in dieser Arbeit nur eingeschränkt betrachtet werden. In dieser Arbeit werden Energieeinspeisungen im Rahmen der Datenbereinigung berücksichtigt. Durch eine Gleichanteilsunterdrückung wird die Energieerzeugung durch große Solaranlagen und Blockheizkraftwerke subtrahiert. Wenn allerdings Energieerzeugungssysteme mit einem falschen bzw. fehlenden Vorzeichen gemessen werden, ist eine Datenbereinigung nicht möglich. Die Daten solcher Haushalte werden in dieser Arbeit entsprechend nicht verwendet.

- Die Auswahl der evaluierten Geräteklassen motiviert sich aus der Verfügbarkeit in privat bewohnten Haushalten. Für die Energiedatenvisualisierung im Smart-Building-Kontext von Industrie- und Bürogebäuden oder Co-Working-Spaces sind die Arbeiten, insbesondere die Methodik, grundsätzlich übertragbar. Es stellen sich aber durchaus Einschränkungen wie eine Vielzahl gleichartiger Geräte im gleichen Stromkreis, welche in dieser Arbeit nicht betrachtet werden sollen. Um die Übertragbarkeit der grundsätzlichen Vorgehensweise und verwendeten Modelle genauer zu untersuchen, bedarf es der Verfügbarkeit entsprechender Datensätze. In dieser Arbeit wird lediglich mitberücksichtigt, dass es mehrere Instanzen einer Geräteklasse (z. B. Kühlgeräte) geben kann.
- Da der Lastgang defekter Geräte zur Trainingszeit nicht bekannt ist, können die Verfahren nicht für die Erkennung defekter Geräte eingesetzt werden.

4.2 Betrachtete Geräteklassen

In diesem Abschnitt werden die zu disaggregierenden Haushaltsgeräteklassen ausgewählt und im Anschluss anhand von exemplarischen Lastgängen bzw. Aktivierungen vorgestellt.

4.2.1 Auswahl der Geräteklassen

Für diese Studie werden sieben Haushaltsgeräteklassen ausgewählt, deren Lastgänge von den Disaggregationsansätzen geschätzt werden sollen. Diese Beschränkung auf sieben Geräteklassen limitiert nicht die Anwendbarkeit auf echte Haushaltslastgangsmessungen, da der Haushaltslastgang stets alle elektrischen Geräte umfasst. Da die Disaggregationsexperimente auf öffentlichen Datensätzen beruhen, ist die Auswahl an Haushaltsgeräten in erster Linie durch die Verfügbarkeit der einzeln gemessenen Lastgänge beschränkt.

Bildung von Geräteklassen

Vor der Auswahl werden zunächst Haushaltsgeräte mit einem ähnlichen Funktionsprinzip (z. B. Kühl- und Gefrierschränke) sowie Kombinationsgeräte (z. B. Waschtrockner und Kühl/Gefrierkombinationen) in Geräteklassen zusammengefasst.

In Tabelle 4.1 ist aufgeführt, wieviele der genannten Kombinationsgeräte sowie Haushaltsgeräte mit einem ähnlichen Funktionsprinzip in den verwendeten Datensätzen (siehe Abschnitt 4.3) vorkommen. Die Bildung von Geräteklassen bezweckt, dass in vielen Gebäuden mindestens eine Geräteinstanz vorhanden ist. In einigen Gebäuden insbesondere des REFIT-Datensatzes sind jedoch mehrere Geräteinstanzen der Klasse Kühl- und Gefriergeräte vorhanden.

Kühl- und Gefriergeräte besitzen das gleiche Funktionsprinzip und somit ist nicht bekannt, ob im Allgemeinen Lastgänge von Kühlschränken und Gefrierschränken unterscheidbar sind. Zwar kann man aus Erfahrung davon ausgehen, dass Kühlschränke öfter geöffnet werden als Gefrierschränke und die Lastgänge deswegen eine höhere Varianz aufweisen sollten. Allerdings ist nicht klar, ob diese Form von Varianz von einem neuronalen Netz ausgenutzt werden kann, wenn man die beschränkte Eingabelänge bedenkt. Zudem dürfte die Öffnungshäufigkeit auch tageszeitabhängig variieren, was vom gewählten Modell nicht berücksichtigt wird. Ebenfalls ist es nicht offensichtlich, wie Lastgänge von Kühl/Gefrierkombinationen einzuordnen sind, wenn zwischen Kühl- und Gefriergeräten unterschieden werden soll. Daher werden im Rahmen dieser Studie Kühl- und Gefriergeräte sowie Kombinationsgeräte unter einer Haushaltsgeräteklasse zusammengefasst.

Waschmaschinen und Wäschetrockner besitzen hingegen unterschiedliche Funktionsprinzipien, was auch anhand deren Lastgängen (siehe Abschnitt 4.2.2) deutlich wird. Waschtrockner sind jedoch Kombinationsgeräte, die im Betrieb zwischen den beiden Funktionsprinzipien wechseln können. Deren Lastgänge gleichen somit nur abschnittsweise denen der dedizierten Geräte. Im Rahmen dieser Studie wird angenommen, dass Waschtrockner primär als Waschmaschinen genutzt werden und somit bilden diese Gerätetypen eine gemeinsame Geräteklasse. Wäschetrockner werden hingegen als getrennte Geräteklasse gehandhabt.

Weitere Geräteklassen ergeben sich durch die verwendeten Datensätze (Abschnitt 4.3), bei denen oftmals mehrere Geräte am gleichen Standort durch ein einziges Messgerät erfasst sind. Dies betrifft TV und Unterhaltungselektronik sowie Computer und Zubehör. So wird beispielsweise in den Metadaten des REFIT-Datensatzes (siehe Abschnitt 4.3.7) beschrieben, dass der Lastgang des Fernsehgeräts u. a. gemeinsam mit DVD-Player, Set-Top-Boxen, PCs und Spielekonsolen gemessen wurde. Zur Leistungsaufnahme des Computers ist u. a. der des Monitors, Druckers, Scanners sowie von Netzwerkgeräten erfasst worden. Daher ist eine weitere Unterteilung der Unterhaltungselektronik in einzelne Geräteklassen aufgrund der Nichtverfügbarkeit separater Messdaten nicht möglich. Die erfassten Gerätetypen variieren

Tabelle 4.1: Anzahl aufgezeichneter Lastgänge von Gerätetypen mit ähnlichem Funktionsprinzip oder Lastgänge von Kombinationsgeräten. Ein • steht für einen gemessenen Lastgang in den jeweiligen Datensätzen.

Datensatz, Gebäude	Kühl- schrank	Tiefkühl- schrank	Kühl/Gefrier- kombination	Wasch- maschine	Wasch- trockner	Wäsche- trockner
UK-DALE	1		•		•	
	2	•			•	
	3					
	4		•		•	
	5			•		•
REFIT	1	•	•		•	•
	2			•	•	
	3		•	•	•	•
	4	•	•	•	•	
	5			•	•	•
	6		•		•	
	7	•	••		•	•
	8	•	•		•	•
	9			•	•	•
	10		•	•	•	
	11	•		•	•	
	12			•		
	13				•	•
	15			•	•	•
	16			••	•	
	17		•	•	•	•
	18	•	•	•	•	•
	19			•	•	
	20	•	•		•	•
	21			•	•	•
	ECO	1	•	•		•
2		•	•			
3		•	•			
4		•	•			
5		•				
6		•				
DRED	•					
ESHL	•	•		•		•
Anzahl Gebäude	16	16	16	24	4	11

dabei je nach Gebäude. Somit bilden TV & Unterhaltungselektronik sowie Computer & Zubehör jeweils eine Geräteklasse. Die verbleibenden nicht genannten Gerätetypen werden jeweils als eigenständige Geräteklasse angesehen.

Umgang mit mehreren Geräteklasseinstanzen

Insbesondere durch die Bildung der Geräteklassen kommt es vor, dass in einem Haushalt mehrere Instanzen einer Geräteklasse vorhanden sind. In diesem Fall wird angenommen, dass die Lastgänge unterschiedlicher Instanzen im Haushaltslastgang nicht unterscheidbar sind. Die Lastgänge aller Geräte werden daher aufsummiert, sodass bei einer Disaggregation die Summenlast aller Geräteinstanzen geschätzt werden soll. Für die Summe besteht kein Bedarf der Unterscheidbarkeit der Geräteinstanzen. Insofern soll keine Unterscheidung der Geräte innerhalb einer Geräteklasse erfolgen.

Auswahl der häufigsten Geräteklassen in den Datensätzen

Für eine möglichst große verwertbare Datenmenge für diese Studie wird unter den häufigst vorkommenden Geräteklassen ausgewählt. In Abbildung 4.1 ist dazu aufgeführt, in wievielen der insgesamt 33 Haushalte der verwendeten Datensätze (Auswahl der Datensätze in Abschnitt 4.3) mindestens ein Lastgang der jeweiligen Geräteklasse zur Verfügung steht.

Unter den neun häufigsten Geräteklassen werden die Geräteklassen Computer & Zubehör sowie Toaster aus folgenden Gründen bei der Auswahl ausgeschlossen.

Bei Computer & Zubehör sowie bei TV & Unterhaltungselektronik repräsentiert der Lastgang eine unbestimmte Zahl an Geräten und Gerätetypen. Somit ist nicht klar, ob die Disaggregation basierend auf diesen Daten eine wohldefinierte Aufgabe ist. Diese Studie befasst sich daher nur mit einer dieser beiden Geräteklassen und schließt Computer & Zubehör aufgrund der geringeren Datenmenge aus.

Weiterhin wird die Geräteklasse Toaster ausgeschlossen, da mit Wasserkocher bereits eine Geräteklasse untersucht wird, deren Funktionsprinzip auf der kurzzeitigen Erzeugung von Hitze durch elektrischen Widerstand beruht. Der Wasserkocher kommt jedoch in den Datensätzen häufiger vor.

Somit besteht schließlich die Liste der Geräteklassen, die im Rahmen dieser Studie evaluiert werden, aus Kühl- und Gefriergeräten, TV & Unterhaltungselektronik, Wasserkocher, Waschmaschinen & Wäschetrockner, Mikrowellen, Geschirrspülmaschinen und Wäschetrockner (in der Abbildung 4.1 fett markiert). Diese Auswahl der Haushaltsgeräte umfasst die fünf Haushaltsgeräteklassen, die auch Kelly und Knottenbelt (2015a) und Bonfigli, Felicetti et al. (2018) für ihre Experimente gewählt haben.

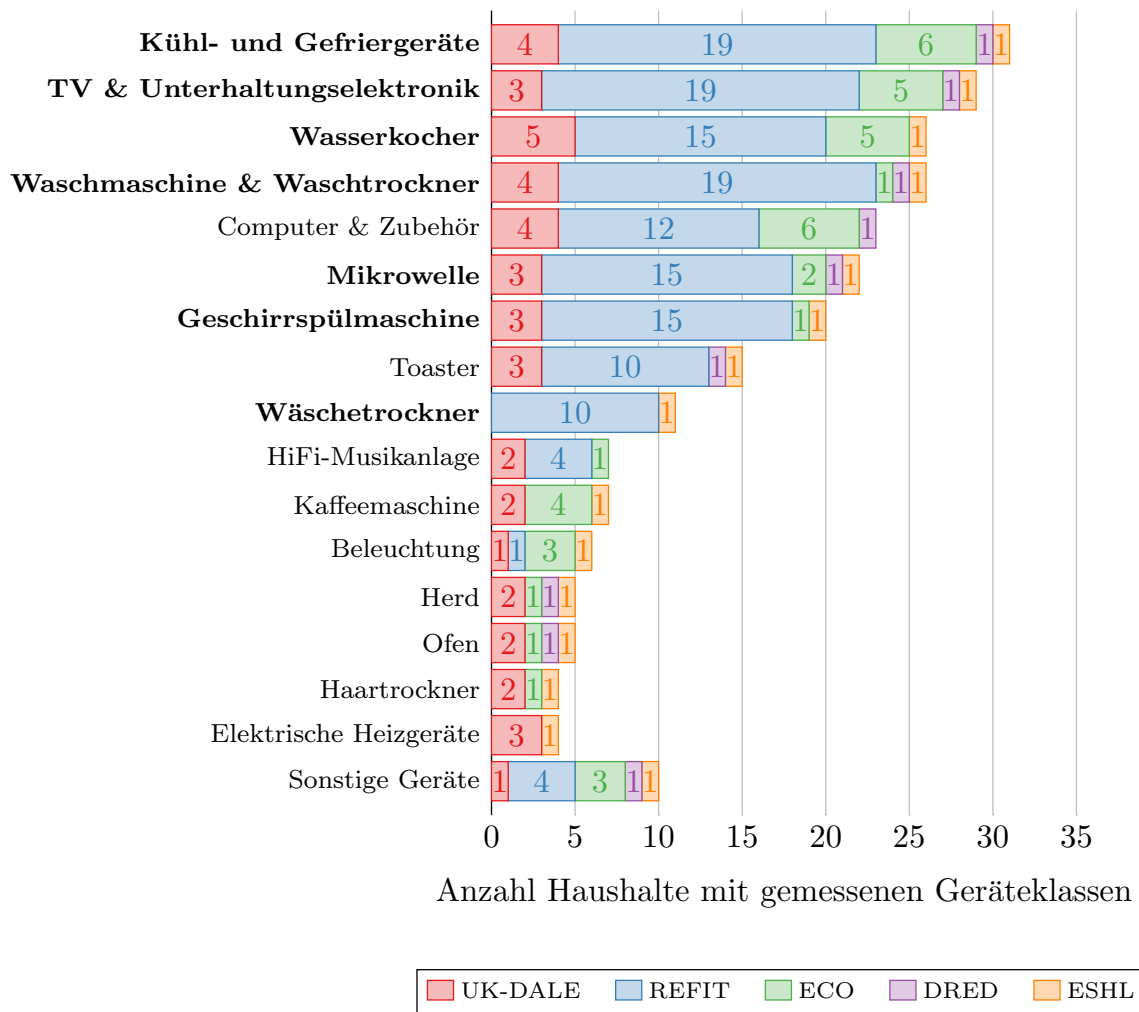


Abbildung 4.1: Anzahl an Haushalten mit vorliegender Lastgangsmessung der jeweiligen Geräteklasse, farblich nach Datensätzen (siehe Abschnitt 4.3) gekennzeichnet.

Diskussion

Die obige Auswahl an betrachteten Haushaltsgeräten basiert letztendlich auf der Vorausswahl, die von den Erstellern der Datensätze durch ihren Messaufbau bereits getroffen wurde. Im Hinblick auf eine breite Anwendbarkeit der Disaggregationsansätze sind die statistisch häufigsten in Haushalten vorkommenden Geräte zu berücksichtigen. Laut Umfragen des Statistischen Bundesamtes (2016; 2018), welche Haushaltsgeräte einen hohen Ausstattungsgrad in deutschen Haushalten aufweisen (siehe Abbildungen 4.2 und 4.3), zählen bereits viele der ausgewählten Geräteklassen (in den Abbildungen fett markiert) zu den Haushaltsgeräten mit den höchsten Ausstattungsgraden: Über 90 % der Haushalte verfügen über einen Kühl-

schränk, eine Waschmaschine und einen Wasserkocher. In knapp über 70 % der Haushalte befinden sich eine Mikrowelle und eine Geschirrspülmaschine. Wäschetrockner sind in etwa 42 % bis 44 % der Haushalte vorhanden. Die Umfragen treffen keine Aussagen bezüglich Fernsehgeräten und Unterhaltungselektronik. Zu den Haushaltsgeräten mit einem Ausstattungsgrad über 80 %, die im Rahmen dieser Studie jedoch nicht betrachtet werden, zählen Herd, Staubsauger, Bügeleisen, Haartrockner sowie Kaffeemaschine.

Unter der alternativen Zielsetzung, durch eine Disaggregation die größten Energieverbraucher in einem Haushalt zu identifizieren, ist die Auswahl der statistisch größten Energieverbraucher im Haushalt sinnvoll. Dazu wurden, wie in der Rangliste in Abbildung 4.4 dargestellt, durch eine Erhebung der Süddeutschen Zeitung (2011), folgende Geräteklassen als die größten Energieverbraucher in Privathaushalten identifiziert: Bürogeräte, Unterhaltungselektronik, Durchlauferhitzer, Kühlgeräte, Beleuchtung und Geräte zum Kochen. Überdurchschnittlich viel Energie verbrauchen sowohl Geräte mit hoher Leistungsaufnahme und kurzen Nutzungszeiten wie ein Durchlauferhitzer, als auch unter Dauernutzung stehende Geräte mit geringen Leistungen wie Kühlgeräte oder Umwälzpumpen.

Mit der getroffenen Geräteauswahl sind statistisch lediglich 43,4 % des Energieverbrauchs abgedeckt. Dieses Missverhältnis spiegeln auch die empirisch im KIT Energy Smart Home Lab (ESHL) gesammelten Energieverbrauchsdaten wieder (siehe Abschnitt 4.3.5). Um die Disaggregation zur Identifikation der größten Energieverbraucher in deutschen Haushalten einzusetzen, fehlt somit die Berücksichtigung der Bürogeräte, Heizungssysteme, Wasserboiler, Beleuchtung sowie Elektroherde. Dies sollte bei der Erstellung zukünftiger Messkampagnen von Haushaltsenergiedaten berücksichtigt werden. Diese Studie ist auf die derzeit verfügbaren Daten beschränkt.

4.2.2 Beschreibung der ausgewählten Haushaltsgeräteklassen

Nach der vorangegangenen Auswahl stellen somit die Lastgänge folgender sieben Haushaltsgeräteklassen die Ziele dar, die durch unterschiedliche Ansätze disaggregiert werden. Im Folgenden werden die Haushaltsgeräte anhand typischer Charakteristika im Geräte- bzw. Haushaltlastgang vorgestellt.

Kühl- und Gefriergeräte Kühl- und Gefriergeräte sind dauerhaft angeschaltet, vollziehen jedoch einen sensorregulierten Arbeitszyklus, der zwischen Ruhezustand und Kompressionsvorgang wechselt. Im Haushaltlastgang ist der regelmäßige Arbeitszyklus von Kühl- und Gefriergeräten während der Nacht gut erkennbar, da sich andere Haushaltsgeräte während der Nacht in der Regel im Ruhezustand befinden. In Abbildung 4.5 sind typische Lastkurven dieser Gerätekategorie dargestellt. Zum Beginn eines Arbeitszyklus kann es hierbei aufgrund des Anlaufens des Kompressors zu kurzfristigen großen Leistungsaufnahmen kommen.

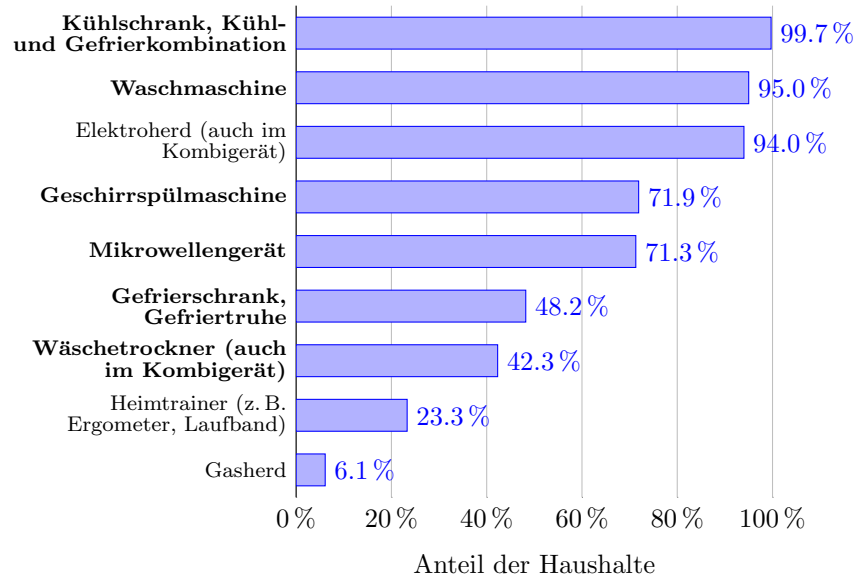


Abbildung 4.2: Ausstattungsgrad privater Haushalte in Deutschland mit Haushalts- und sonstigen Geräten im Jahr 2018.

Quelle: Statistisches Bundesamt, 2018.

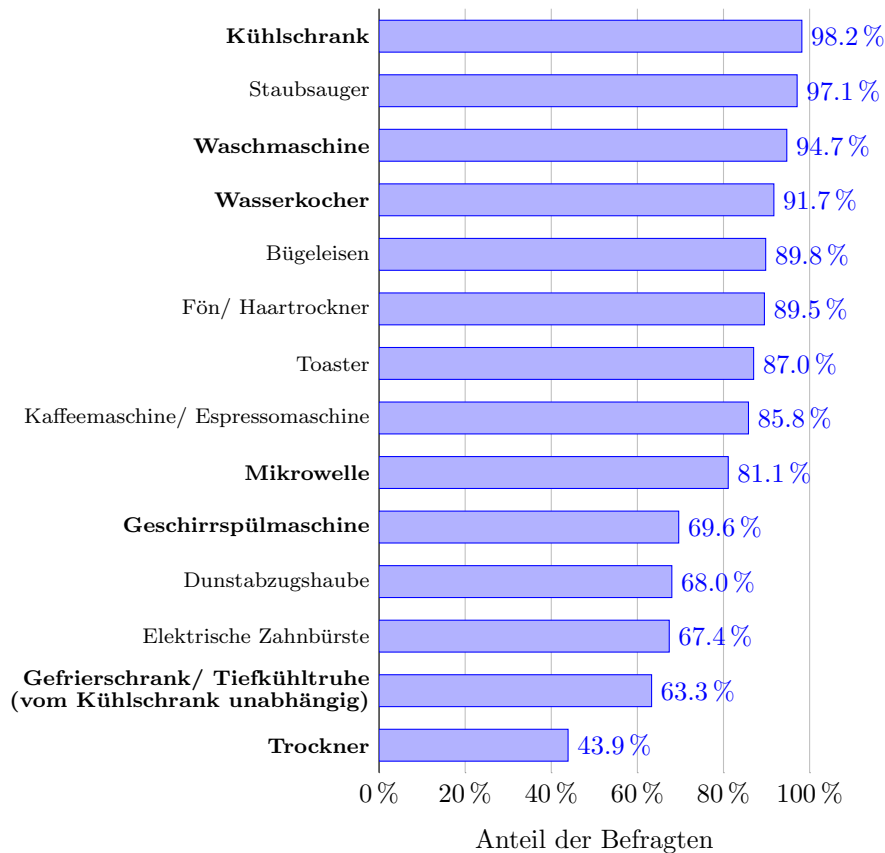


Abbildung 4.3: Umfrage zur Ausstattung mit Haushaltsgeräten in deutschen Haushalten 2016. Markierte Geräte werden in den Disaggregationsexperimenten betrachtet. Quelle: Statista, 2016.

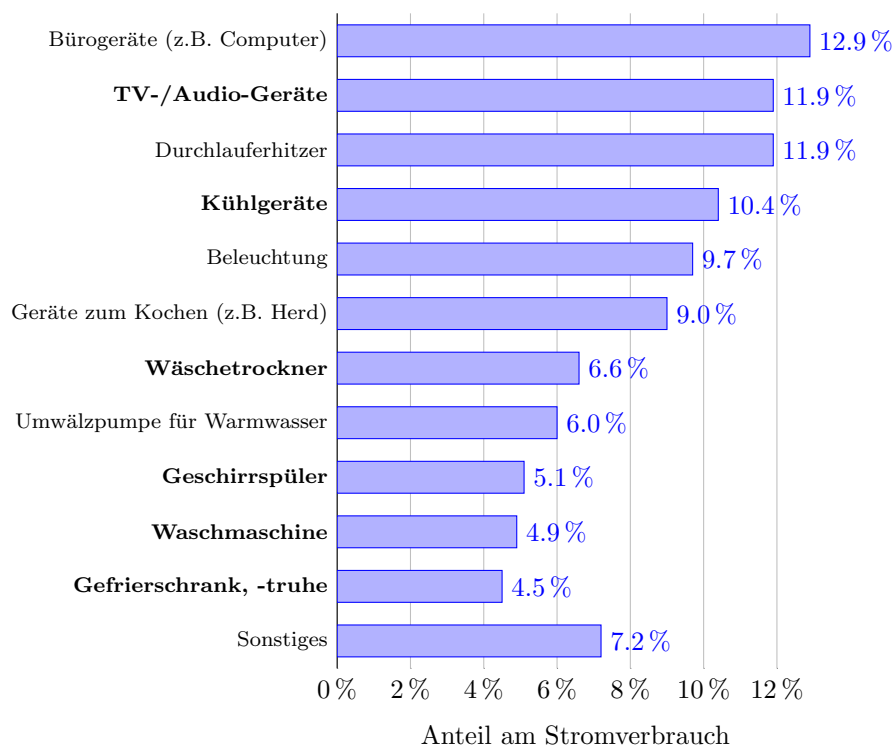


Abbildung 4.4: Anteil am Energieverbrauch von Elektrogeräten in privaten Haushalten im Jahr 2011. Markierte Geräte werden in den Disaggregationsexperimenten betrachtet. Quelle: Süddeutsche Zeitung, 2011.

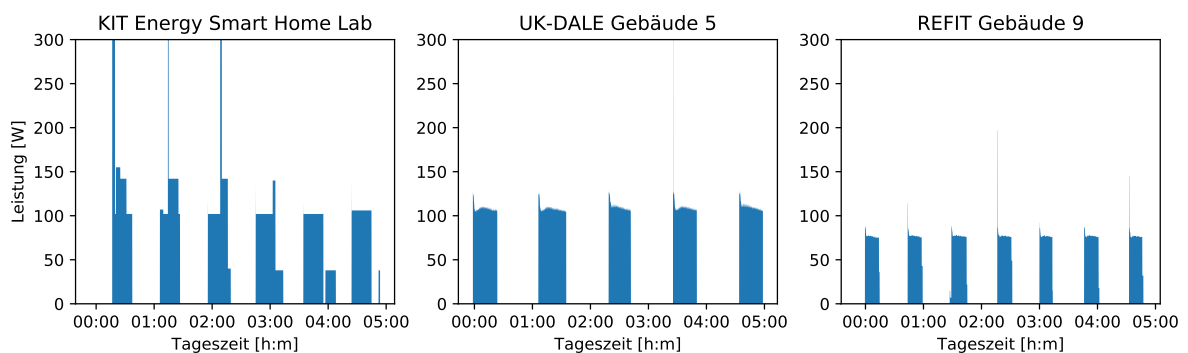


Abbildung 4.5: Beispielhafte Lastgänge von Kühl- und Gefrierschränken aus unterschiedlichen Gebäuden des Testdatensatzes. Dargestellt wird ein Zeitraum von 5,5 Stunden.

In den Datensätzen kommt es vor, dass in einem Haushalt mehrere Kühlschränke und Gefrierschränke oder Kühl-Gefrier-Kombinationen installiert sind. Da über einzelne Modelle dieser Gerätekategorie abstrahiert werden soll, werden die Messungen mehrerer Geräte aggregiert betrachtet. Im KIT Energy Smart Home Lab (ESHL) befindet sich beispielsweise sowohl ein Kühlschrank als auch ein Gefrierschrank. Bei dem ersten dargestellten Lastgang in Abbildung 4.5 ist um 4:00 Uhr ein Arbeitszyklus des Gefrierschranks zu erkennen, dem ein Arbeitszyklus des Kühlschranks folgt. In den vorigen Arbeitszyklen überlagern sich die Lastkurven beider Geräte.

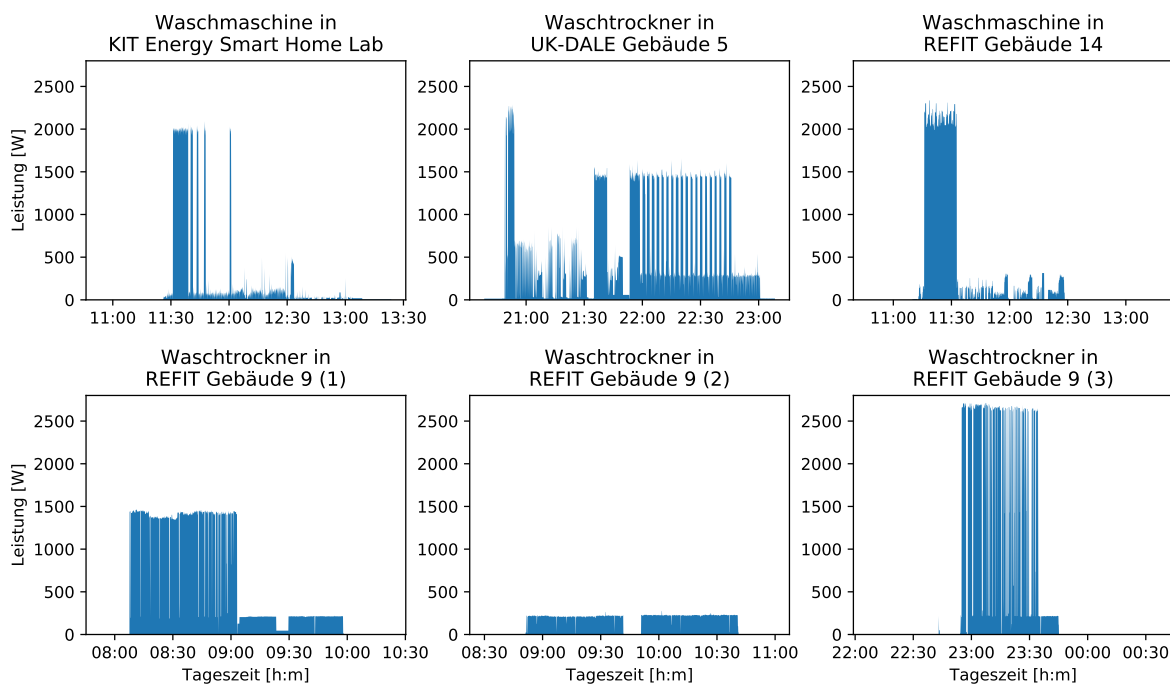


Abbildung 4.6: Beispielhafte Lastgänge von Waschmaschinen und Wäschetrockner aus unterschiedlichen Gebäuden des Testdatensatzes. In Gebäude 5 des Datensatzes UK-DALE befindet sich ein Wäschetrockner, dessen Lastgang sich typischerweise aus einem Waschzyklus gefolgt von einem Trocknenzyklus zusammensetzt. In den Darstellungen wird jeweils ein Zeitraum von 2,5 Stunden gezeigt.

Waschmaschinen und Wäschetrockner In Abbildung 4.6 sind links oben und rechts oben zwei Lastgänge von Waschmaschinen dargestellt. Der Energieverbrauch von Waschmaschinen wird hauptsächlich durch das Aufheizen des Wassers zu Beginn des Waschprogramms und durch das Schleudern der Wäsche am Ende des Programms verursacht.

Wäschetrockner sind Waschmaschinen mit einem zusätzlichen Funktionsprinzip zum Trocknen der Wäsche. Abbildung 4.6 zeigt in der mittleren oberen Darstellung einen Wäschetrock-

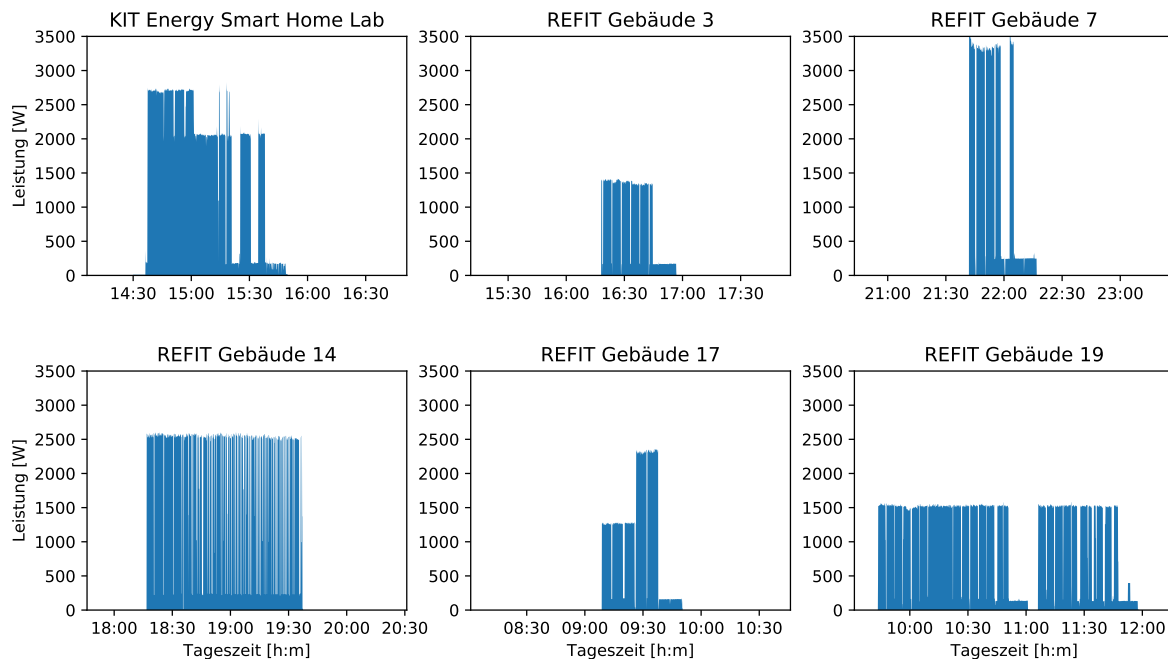


Abbildung 4.7: Beispielhafte Lastgänge von Wäschetrocknern unterschiedlicher Gebäude aus dem gesamten Datensatz.

nerlastgang, welcher sich aus einem Waschzyklus gefolgt von einem Trocknungszyklus zusammensetzt. In den unteren Lastgängen in Abbildung 4.6 sind Lastgänge unterschiedlicher Trocknungsvorgänge von Waschtrocknern dargestellt. Der Trocknungsvorgang verursacht ebenfalls einen hohen Energieverbrauch.

Durch die begrenzte Anzahl an Waschprogrammen ist der Raum möglicher Lastgänge von Waschmaschinen und Wäschetrocknern beschränkt. Variationen im Lastgang entstehen durch unterschiedliche thermische Bedingungen und die Befüllung. Wie viele Haushaltsgeräte sind auch private Waschmaschinen und Wäschetrockner im Verhältnis zur gesamten Standzeit des Gerätes selten aktiv.

Wäschetrockner Ein Wäschetrockner kann auf unterschiedlichen Trocknungsverfahren basieren, insbesondere Ablufttrocknung sowie Kondensationstrocknung mit oder ohne Wärmepumpe. Dadurch ergeben sich unterschiedliche Lastgänge, mit denen das Disaggregationsverfahren arbeiten muss. In Abbildung 4.7 sind exemplarische Lastgänge von Trockneraktivierungen dargestellt. Wie bereits an den wenigen Beispielen zu erkennen ist, unterscheiden sich die Lastgänge stark u. a. in der maximalen Leistungsaufnahme und der Dauer der Aktivierung.

Alle Wäschetrockner werden unter einer Geräteklasse zusammengeführt, sodass die Disag-

gregationsverfahren in der Lage sein sollen, jede Lastgangsvariante und Trocknungsverfahren zu erkennen.

Geschirrspülmaschinen Der Verlauf der Spülmaschinenlastgänge, wie in Abbildung 4.8 exemplarisch dargestellt, lässt sich beschreiben mit einer ersten Phase, in der das Spülwasser erhitzt wird. Danach folgt eine Spülphase mit niedrigem Energieverbrauch. Gegebenenfalls wird das Spülwasser nochmals nachgeheizt. Schließlich wird zum Abspülen und Trocknen frisches Spülwasser erhitzt, weshalb erneut eine erhöhte Leistungsaufnahme zu beobachten ist.

Wasserkocher Abbildung 4.9 zeigt drei exemplarische Aktivierungen unterschiedlicher Wasserkocher. Wasserkocher haben eine hohe Leistungsaufnahme, verbrauchen jedoch durch ihren sehr kurzen Aktivitätszyklus nicht viel Energie. Da ein Wasserkocher in der Regel nur wenige Minuten aktiv ist, ist zur besseren Erkennbarkeit des Lastgangs nur ein kurzer Zeitraum in den Abbildungen dargestellt.

In Abbildung 4.9 ist auch die geringere Leistungsaufnahme des Wasserkochers im ESHL verglichen zu den britischen Modellen im Datensatz UK-DALE oder REFIT zu erkennen. So muss bei Wasserkochern mit unterschiedlichen Leistungsaufnahmen je nach Hersteller und Gerätemodell gerechnet werden.

Mikrowelle Abbildung 4.10 zeigt drei beispielhafte Aktivierungen unterschiedlicher Mikrowellen. Wie Wasserkocher sind Mikrowellen nur kurz durchgängig aktiv. Ebenfalls gibt es viele unterschiedliche Hersteller und Modelltypen für Mikrowellen, wodurch die Identifikation dieser Geräteklasse erschwert wird. Zudem verfügen manche Mikrowellen über eine Grillfunktion.

Fernseher und Unterhaltungselektronik Die Leistungsaufnahme von Fernseher und Unterhaltungselektronik ist in der Regel mit unter 200 W vergleichsweise gering, jedoch sind diese Geräte häufig über mehrere Stunden aktiv. In Abbildung 4.11 sind einige Beispiele für Lastgänge dieser Geräteklasse dargestellt, wobei die Diagramme einen Zeitraum von 12 Stunden zeigen.

Da bei den verwendeten Datensätzen die Leistungsaufnahme des Fernsehers inklusive der Unterhaltungselektronik in der Nähe des Fernsehers gemessen wurde, ist die Anzahl der in der Lastkurve vorkommenden Geräte variabel. Zusätzliche Quellen für unterschiedliche Lastgangseigenschaften sind die verwendeten Modelle des Fernsehers und der Unterhaltungselektronik sowie der wiedergegebene Medieninhalt (siehe Greveler et al. (2012)).

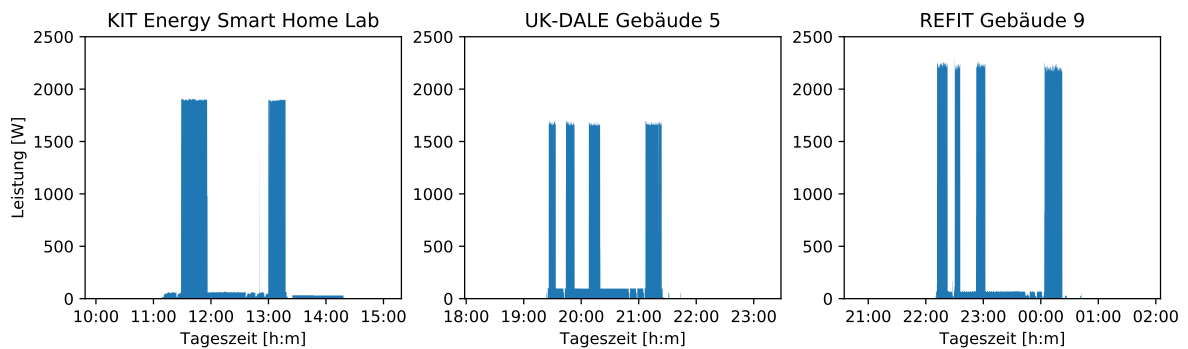


Abbildung 4.8: Aktivierungen von Geschirrpülmaschinen aus unterschiedlichen Gebäuden des Testdatensatzes. Dargestellt wird ein Zeitraum von 5 Stunden.

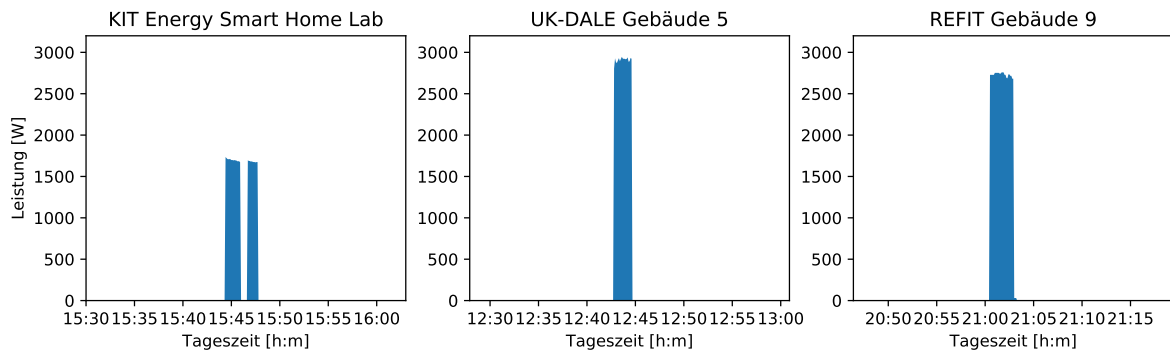


Abbildung 4.9: Aktivierungen von Wasserkochern aus unterschiedlichen Gebäuden des Testdatensatzes. Ein Zeitraum von 30 Minuten ist visualisiert.

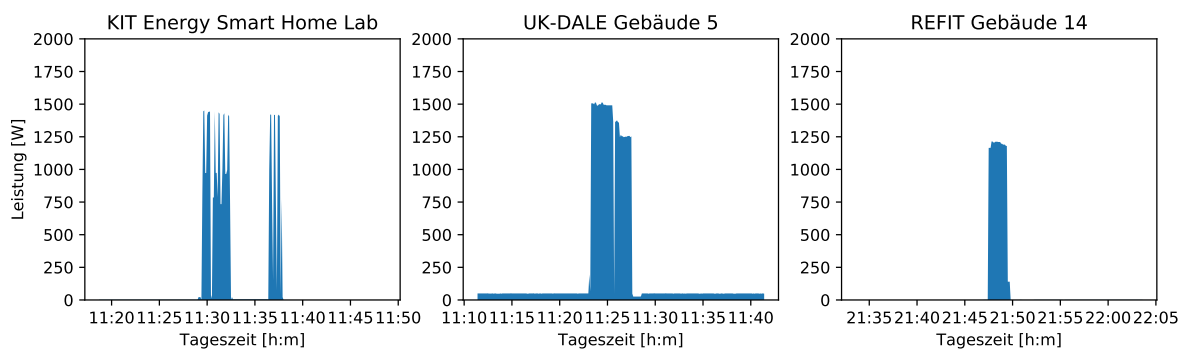


Abbildung 4.10: Aktivierungen von Mikrowellen aus unterschiedlichen Gebäuden des Testdatensatzes. Gezeigt wird ein Zeitraum von 30 Minuten.

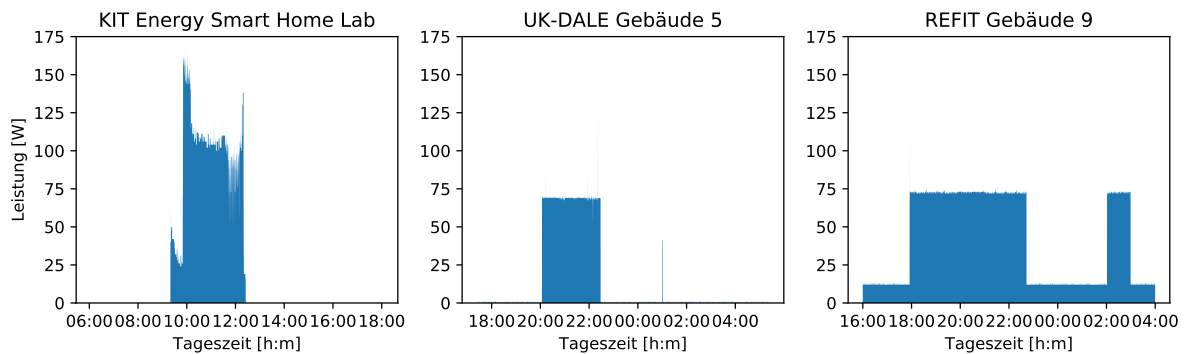


Abbildung 4.11: Aktivierungen von Fernsehern und Unterhaltungselektronik unterschiedlicher Gebäude aus dem Testdatensatz

Um für den Betrachter die Lastprofile deutlicher darzustellen und aufgrund der unterschiedlichen Aktivierungsdauern, werden unterschiedliche Zeiträume in den betrachteten Gebäuden gezeigt.

4.3 Verwendete Datensätze und spezifische Vorverarbeitungsschritte

Es existieren bereits über 19 öffentliche Datensätze mit Haushaltsenergiemessdaten, die aus der einschlägigen Literatur bekannt sind (siehe u. a. die Überblicksarbeiten von Faustine et al., 2017; Hosseini et al., 2017; Pereira und Nunes, 2018).² Diese Datensätze sind mit ihren Eigenschaften in Tabelle 4.2 zusammengefasst. Die Tabelle zeigt zudem die Eigenschaften der in dieser Arbeit verwendeten Energiemessdaten aus dem ESHL (siehe Abschnitt 4.3.5).

Aufgrund der unterschiedlichen Anwendungsbereiche von NILM (siehe Abschnitt 2.3), eignet sich nicht jeder Datensatz für das in dieser Arbeit untersuchte Szenario. Im Folgenden werden daher Anforderungen für eine Auswahl der Datensätze formuliert. Im Anschluss werden die ausgewählten Datensätze hinsichtlich Datenqualität und Datenumfang im Detail charakterisiert.

²Ferner wird eine Liste von Datensätzen für NILM auf der Webseite wiki.nilme.eu/datasets.html gepflegt.

Tabelle 4.2: Öffentliche Datensätze von Haushaltsenergiemessdaten. Die Eigenschaften der Datensätze sind aus den jeweiligen Veröffentlichungen, sowie dem Quellcode von NILMTK (siehe Abschnitt 4.9.1) und Zusammenfassungskartikeln zusammengetragen. Vereinzelt fehlen Eigenschaften, die aus diesen Quellen nicht ermittelt werden konnten.

Datensatzbezeichnung	Ursprungsland (als ISO 3166-1 Ländercode)	Haushalte	Phasen	Einzellastgänge	Datenmenge je Haushalt	Datenmenge Gesamt	Auflösung Einzellastgänge	Auflösung Haushaltslastgänge	Quelle
ACS-F1	CH	–	–	100	1 - 2 h		10 s	–	Gisler et al. (2013)
AMPds(2)	CA	1	1	20	2 y	2 y	60 s	60 s	Makonin et al. (2016)
BERDS	US	1	1	4	1 y	1 y	20 s	20 s	Maasoumy et al. (2013)
BLUED	US	1	2	–	8 d	8 d	(Ereignisse)	12 kHz	Filip (2011)
COMBED	IN	2	1	6 - 13	1 m	1 m	30 s	30 s	Batra, Parson et al. (2014)
Dataport	US	10		1200+	7 d		60 s	60 s	Dataport (dataport.cloud)
DRED	NL	1	1	12	142 d	142 d	1 s	1 s	Uttama Nambi et al. (2015)
ECO	CH	6	3	7 - 12	138 - 245 d	1227 d	1 s	1 s	Beckel, Kleiminger et al. (2014)
GREEND	AT & IT	9		9	1 y		1 s	–	Monacchi et al. (2014)
HES	GB	251			1 m - 1 y		120 - 600 s	120 - 600 s	Zimmermann et al. (2012)
iAWE	IN	1	2	9	73 d	73 d	1 - 6 s	60 s	Batra, Gulati et al. (2013)
IHEPCDS	FR	1		3	47 m	47 m	60 s	60 s	Hébrail und Bérard (2012)
REDD	US	6	2	9 - 24	3 - 19 d	119 d	3 s	1 s & 15 kHz	Kolter und Johnson (2011)
REFIT	GB	20		11	1 - 2 y	< 40 y	8 s	8 s	Murray, David (2015)
Smart*	US	3	2	21 - 26	3 m	9 m	1 s	1 s	Barker, Mishra et al. (2012)
SustData	PT	50		–	51 - 511 d	1144 d	–	(Ereignisse)	Pereira, Quintal et al. (2014)
SustDataED	PT	1		17	10 d	10 d	2 s	50 Hz	Ribeiro et al. (2016)
Tracebase	DE	–	–	122	–	> 1836 d	1 & 8 s	–	Reinhardt, Baumann et al. (2012)
UK-DALE	GB	5	1	5 - 54	36 - 1483 d	1946 d	6 s	1 - 6 s & 16 kHz	Kelly und Knottenbelt (2015b)
ESHL	DE	1	3	33	19 d	19 d	1 s	1 s	siehe Abschnitt 4.3.5

4.3.1 Anforderungen zur Auswahl der Datensätze

Aus dem in Abschnitt 4.1 zusammengefassten Szenario ergeben sich die drei folgenden Anforderungen an die Datensätze:

Verfügbarkeit von gemessenen Haushalts- und Einzellastgängen

Sowohl für das Training als auch für das Testen der Disaggregationsverfahren ist das Vorhandensein von synchron erfassten Haushalts- als auch Einzelgerätestrangängen erforderlich.

Wichtig dabei ist insbesondere, dass statt eines synthetisch erzeugten Summenlastgangs der tatsächlich gemessene Haushaltslastgang mit einer Teilmenge an Haushaltsgeräten vorliegt, da der Haushaltslastgang auch von nicht berücksichtigten Haushaltsgeräten beeinflusst wird. In einem realistischen Anwendungsfall erschwert die hohe Anzahl der elektrischen Geräte in einem Haushalt die Disaggregationsaufgabe.

Die einzelnen Gerätestrangänge sind als Referenzgröße sowohl zur Bewertung des Disaggregationsmodells während des Trainings als auch zur Evaluation des Verfahrens erforderlich.

Auflösung der Einzel- und Haushaltslastgänge

Einzel- und Haushaltslastgänge der Datensätze müssen über ein Messintervall im Bereich von 1 bis 8 Sekunden verfügen.

Mit dieser Anforderung wird ein einheitlicher Bereich für die zeitliche Lastgangauflösung festgelegt, um die Leistung unterschiedlicher Verfahren gegeneinander unter gleichzeitiger Nutzung mehrerer Datensätze evaluieren zu können. Dabei ergibt sich die untere Grenze daraus, dass handelsübliche Smart Meter in der Regel nicht in der Lage sind, Lastgänge mit einem Messintervall unter einer Sekunde zu erfassen. Die obere Grenze ergibt sich aus der Abwägung, einerseits möglichst hochaufgelöste Lastgänge zu fordern, da angenommen wird, dass in zeitlich höheraufgelösten Lastgängen mehr Merkmale für die Disambiguierung von Geräteaktivität vorhanden sind. Andererseits ist die Verfügbarkeit hochaufgelöster Lastgänge in den Datensätzen zu berücksichtigen. Dabei besitzt die Mehrheit der in Tabelle 4.2 aufgeführten Datensätze (10 von 19) Einzellastgänge mit einem Messintervall zwischen 1 und 8 Sekunden.

Obwohl weniger Datensätze über Haushaltslastgänge mit einem Messintervall zwischen 1 und 8 Sekunden verfügen (6 von 15 Datensätzen mit erfassten Haushaltslastgängen in Tabelle 4.2), wird auch für die Haushaltslastgänge dieser Auflösungsbereich gefordert, um den Einfluss der Auflösung auf die Leistungsfähigkeit der Verfahren zu ermitteln.

Auslegung der Haushaltsgeräte auf europäisches Energiesystem

Haushaltsgerätemodelle sind oft an ein spezifisches Energiesystem und einen Kulturkreis ausgelegt. So unterscheiden sich beispielsweise die Lastgänge von amerikanischen Waschmaschinen im REDD-Datensatz (Kolter und Johnson, 2011) zu den Lastgängen europäischer Waschmaschinen im Datensatz UK-DALE (Kelly und Knottenbelt, 2015b). Um Variationen im Lastgang innerhalb einer Geräteklasse durch diesen Faktor zu begrenzen, wird daher in dieser Studie gefordert, dass ausschließlich Datensätze aus dem europäischen Raum berücksichtigt werden. Dabei wird angenommen, dass sich Haushaltsgerätemodelle, die für das europäische Verbundnetz entworfen worden sind, nicht grundlegend unterscheiden.

4.3.2 Auswahl der Datensätze

Das Ergebnis der Anforderungsprüfung ist in Tabelle 4.3 zusammengefasst. Von den öffentlichen Datensätzen sind damit DRED, ECO, REFIT und UK-DALE mit dem gestellten Szenario kompatibel. Zusätzlich werden im Zuge dieser Arbeit auch Energiemessdaten aus dem ESHL verwendet, welche ebenfalls die gestellten Anforderungen erfüllen.

Im Folgenden werden diese ausgewählten fünf Datensätze charakterisiert. Dazu wird die Datenqualität bezüglich Häufigkeit und Dauer von Messausfällen qualitativ beurteilt sowie die Anzahl der Aktivierungen je Gebäude im Datensatz und Haushaltsgeräteklasse aufgeführt. Dabei ist eine Aktivierung ein zeitlich zusammenhängender Lastgangsabschnitt, in dem das elektrische Haushaltsgerät Arbeit verrichtet und Energie verbraucht. Aktivierungen werden dazu mit dem in Abschnitt 4.4.2 beschriebenen Verfahren heuristisch bestimmt.

4.3.3 DRED: Dutch Residential Energy Dataset

Der *Dutch Residential Energy Dataset (DRED)* von Uttama Nambi et al. (2015) aus den Niederlanden besteht lediglich aus einem Gebäude und enthält, wie in Tabelle 4.4 dargestellt, nur vier der sieben relevanten Haushaltsgeräteklassen.

Verwertbare Messdaten sind in Abbildung 4.12 veranschaulicht. In den schwarzen Bereichen sind in regelmäßigen Abständen Messwerte vorhanden. Zeitabschnitte, in denen über 2 Minuten lang keine Messwerte erfasst worden sind, werden als Messlücken identifiziert und sind in der Abbildung weiß markiert. Orange markierte Bereiche sind kurze Zeitabschnitte unter einer Stunde, die von Messlücken unterbrochen sind. In den Experimenten werden ausschließlich die schwarz markierten Bereiche mit in regelmäßigen Abständen vorhandenen Messwerten verwendet.

Tabelle 4.3: Anforderungserfüllung der Datensätze

Datensatz- bezeichnung	Verfügbarkeit Einzel- und Haushalts- lastgängen	Messintervall Einzel- lastgänge zwischen 1 und 8 s	Messintervall Haushalts- lastgänge zwischen 1 und 8 s	Europäischer Datensatz
ACS-F1	×	✓	–	✓
AMPds(2)	✓	×	×	×
BERDS	✓	×	×	×
BLUED	×	–	×	×
COMBED	✓	×	×	×
Dataport	✓	×	×	×
DRED	✓	✓	✓	✓
ECO	✓	✓	✓	✓
GREEND	×	✓	–	✓
HES	✓	×	×	✓
iAWE	✓	✓	×	×
IHEPCDS	✓	×	×	✓
REDD	✓	✓	✓	×
REFIT	✓	✓	✓	✓
Smart*	✓	✓	✓	×
SustData	×	×	×	✓
SustDataED	✓	✓	×	✓
Tracebase	×	✓	×	✓
UK-DALE	✓	✓	✓	✓
ESHL	✓	✓	✓	✓

Tabelle 4.4: Gesamtzahl der Aktivierungen in Datensatz DRED

DRED Gebäude	
	1
Datenmenge in Tagen	142
Anzahl Aktivierungen je Haushaltsgerät	
Waschmaschine	33
Mikrowelle	346
Kühlschrank	3419
TV	151

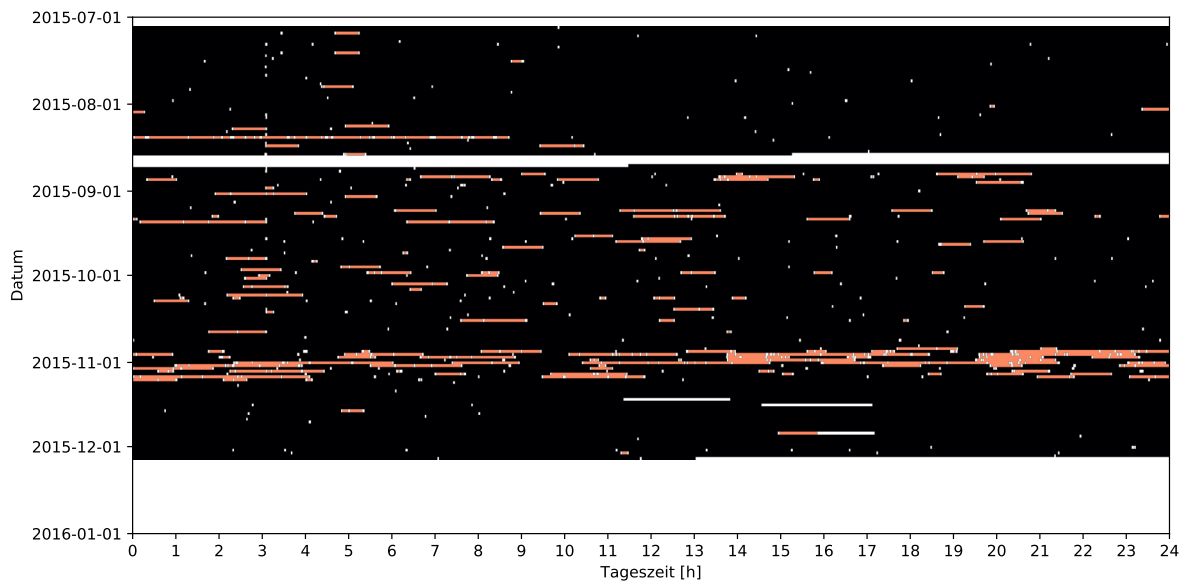


Abbildung 4.12: Verfügbarkeit der Messdaten im Datensatz DRED

4.3.4 ECO: Electricity Consumption and Occupancy Dataset

Der schweizer Datensatz Electricity Consumption and Occupancy Dataset (ECO) von Beckel, Kleiminger et al. (2014) umfasst 6 Haushalte, deren Energieverbrauch zwischen einem halben und einem Jahr aufgezeichnet wurde. Spül- und Waschmaschine sowie Fernsehgerät wurden jeweils nur in einem Gebäude gemessen. Wasserkocher und Kühlgeräte sind in mehreren Gebäuden vorhanden. Im ganzen Datensatz existiert jedoch kein Wäschetrockner. Zur Übersicht ist in Tabelle 4.5 die Anzahl der Aktivierungen nach Geräteklasse und Gebäude aufgeschlüsselt.

Die Anzahl an Messausfällen im ECO-Datensatz ist gering. Jedoch existieren längere Messausfälle von ein bis drei Tagen. Dies ist am Beispiel der zeitlichen Messwerteabdeckung von Gebäude 2 zu erkennen, welches in Abbildung 4.13 dargestellt ist. Aufgrund der seltenen Messausfälle sowie seltenen Abschnitte mit intermittierenden Messausfällen auch in den anderen Gebäuden des ECO-Datensatzes kann fast der vollständige Datensatz verwendet werden.

4.3.5 ESHL: Energiemessdaten des KIT Energy Smart Home Lab

Das *KIT Energy Smart Home Lab (ESHL)* ist ein Forschungslabor für Gebäudeenergiemanagementsysteme, welches ursprünglich im Rahmen des Projekts *MeRegioMobil*³ auf dem Cam-

³meregiomobil.forschung.kit.edu

Tabelle 4.5: Gesamtzahl der Aktivierungen im Datensatz ECO

	ECO Gebäude					
	1	2	3	4	5	6
Datenmenge in Tagen	221	240	136	217	212	165
Anzahl Aktivierungen je Haushaltsgerät						
Geschirrspülmaschine	–	60	–	–	–	–
Waschmaschine	233	–	–	–	–	–
Wasserkocher	381	307	10	–	3	69
Mikrowelle	–	–	–	865	70	–
Kühlschrank	2148	8628	198	3422	4353	10041
Tiefkühlschrank	6774	10405	15	843	–	–
TV	–	333	–	–	–	–

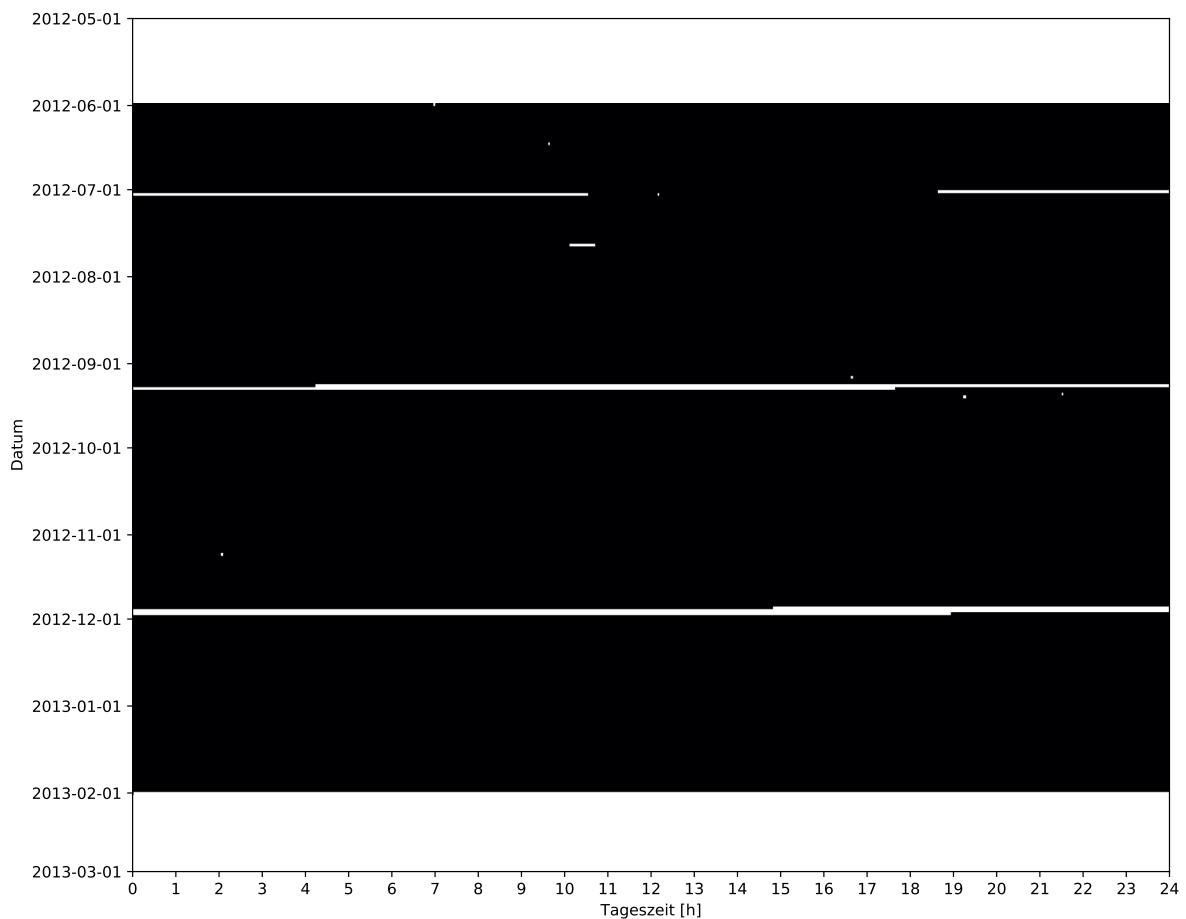


Abbildung 4.13: Verfügbarkeit der Messdaten im Gebäude 2 des Datensatzes ECO

pus des Karlsruher Instituts für Technologie (KIT) errichtet wurde (Allerding und Schreck, 2011). Das Labor wurde in den Folgeprojekten *IZEUS*⁴ und *grid-control*⁵ sukzessiv weiterentwickelt. Forschungsthemen waren unter anderem:

- Integriertes Energiemanagement für intelligente Gebäude und Elektromobile (Mültin et al., 2012),
- Nutzeroberflächen für Gebäudeenergiemanagementsysteme (Becker, Kellerer et al., 2012; Xu et al., 2018),
- Energiemanagement für mehrere Energieträger (Mauser et al., 2016),
- Softwareinfrastruktur in Forschungslaborumgebungen für intelligente Gebäude (Bao, Mauser et al., 2016), sowie
- Co-Simulation von intelligenten Gebäuden in Niederspannungsnetzen (Kochanneck, 2019).

Das Smart-Home-Forschungslabor ist ein freistehendes Gebäude mit einer Wohnfläche von 60 m², welches in einen Hauptraum mit offener Küche und Wohnzimmer, zwei Arbeits- bzw. Schlafzimmer sowie ein Badezimmer aufgeteilt ist. Zusätzlich zum Wohnbereich verfügt das ESHL über einen Raum für elektro- und informationstechnische Infrastruktur.

Aus dem Wohnbereich wird der Energieverbrauch jedes fest installierten Haushaltsgeräts sowie von jeder Steckdosengruppe gemessen. Somit erfasst das Messsystem 31 einzelne Stromkreise im Wohnbereich. Zusätzlich wird der Stromfluss am Hausanschluss, der Photovoltaikanlage, dem Mikroblokheizkraftwerk, dem elektrischen Tauchheizkörper, der Elektromobilladesäule sowie dem Batteriespeicher dreiphasig gemessen.

Detailliertere Beschreibungen des ESHL können den Dissertationen von Allerding (2013), Mauser (2017) und Kochanneck (2019) entnommen werden.

Nutzungsmuster des KIT Energy Smart Home Labs

Das ESHL wurde überwiegend als Forschungslabor zur Entwicklung von Energiemanagementsystemen genutzt. Zeitweise ist das Smart Home als Wohngebäude für Studien genutzt worden. Entsprechend ist das Nutzungsmuster, welches sich im Energieverbrauch wieder spiegelt, vergleichbar mit dem eines gemeinschaftlichen Büros mit unregelmäßiger Nutzung (Co-Working-Space).

Zur Untersuchung des häuslichen Szenarios eignen sich insbesondere die Messdaten, die während der *Wohnphasen* im ESHL gemessen wurden. Als Wohnphasen werden dabei

⁴www.izeus.kit.edu

⁵projekt-grid-control.de

die Zeiträume bezeichnet, in denen das Gebäude Probanden übergeben wird, die für Nutzungsstudien zeitweise das Smart Home beziehen. Mit den Probanden wurden insbesondere ergonomische Aspekte von Gebäudeenergiemanagementsystemen sowie die Nutzung der Flexibilität von Haushaltsgeräten evaluiert. (siehe Paetz et al., 2013; Paetz et al., 2011)

Die dabei erteilte Einverständniserklärung zur Nutzung der Wohnphasendaten lässt jedoch keine Veröffentlichung der detaillierten Energiemessdaten zu. Daher werden in dieser Arbeit ausschließlich die Energiemessdaten verwendet, die während einer Testwohnphase des Autors dieser Arbeit erfasst wurden sowie Daten, die während einer Evaluation eines Co-Simulations-Frameworks durch einen Kollegen des Autors (siehe Kochanneck, 2019) entstanden sind.

Reduktion von Messsystemausfällen

Das Messsystem des ESHL war in der Vergangenheit Bestandteil von Experimenten mit unterschiedlichen thematischen Ausrichtungen, die jeweils exklusiv mit allen Komponenten des Smart Home durchgeführt worden sind. Dadurch fand in der Vergangenheit keine kontinuierliche Messdatenaufzeichnung statt, sodass es häufig zu Messausfällen gekommen ist. Eine zusätzliche Ursache von Messlücken liegt an der Unzuverlässigkeit der eingesetzten speicherprogrammierbaren Steuerungsmodule, die bei erhöhtem Netzwerkverkehr oft abgestürzt sind. Zur Veranschaulichung ist in Abbildung 4.14 die Verfügbarkeit von Messdaten über die Zeit dargestellt, wobei schwarz markierte Bereiche Messdatenverfügbarkeit und weiß markierte Bereiche Messlücken darstellen.

Um die Messausfälle im ESHL zu reduzieren, wurde daher 2016 das Messsystem ausgetauscht. Einerseits wurden hardwareseitig die speicherprogrammierbaren Steuerungsmodule sowie Messeinheiten durch aktuellere Modelle ersetzt (siehe Kochanneck, 2019). Andererseits wurden bei der Softwareinfrastruktur durch die Umsetzung einer Mikroservice-Architektur das Messsystem und die wissenschaftlichen Experimente im ESHL entkoppelt (siehe Bao, Mauser et al., 2016).

Das dadurch neuimplementierte Messsystem erlaubt die zeitgleiche Verwendung des Labors für unterschiedliche Experimente sowie eine unterbrechungsfreie Messdatenaufzeichnung. Die architekturbedingte Entkopplung der Softwarekomponenten erleichtert zudem Neu- und Weiterentwicklungen, da sich Entwickler nur auf die wesentlichen Komponenten für ihre Implementierung konzentrieren können.

In Abbildung 4.15 sind die in der Datenbank verzeichneten Messausfälle mit dem neuen Messsystem dargestellt. Nach einer etwa einjährigen Entwicklungszeit ist im Jahr 2018 nur ein elektrotechnisch bedingter Messausfall aufgetreten. Mit dem Wechsel das alten Messsystems (Abbildung 4.14) auf das neu entwickelte Messsystem (Abbildung 4.15) wurde die Datenverfügbarkeit wesentlich verbessert.

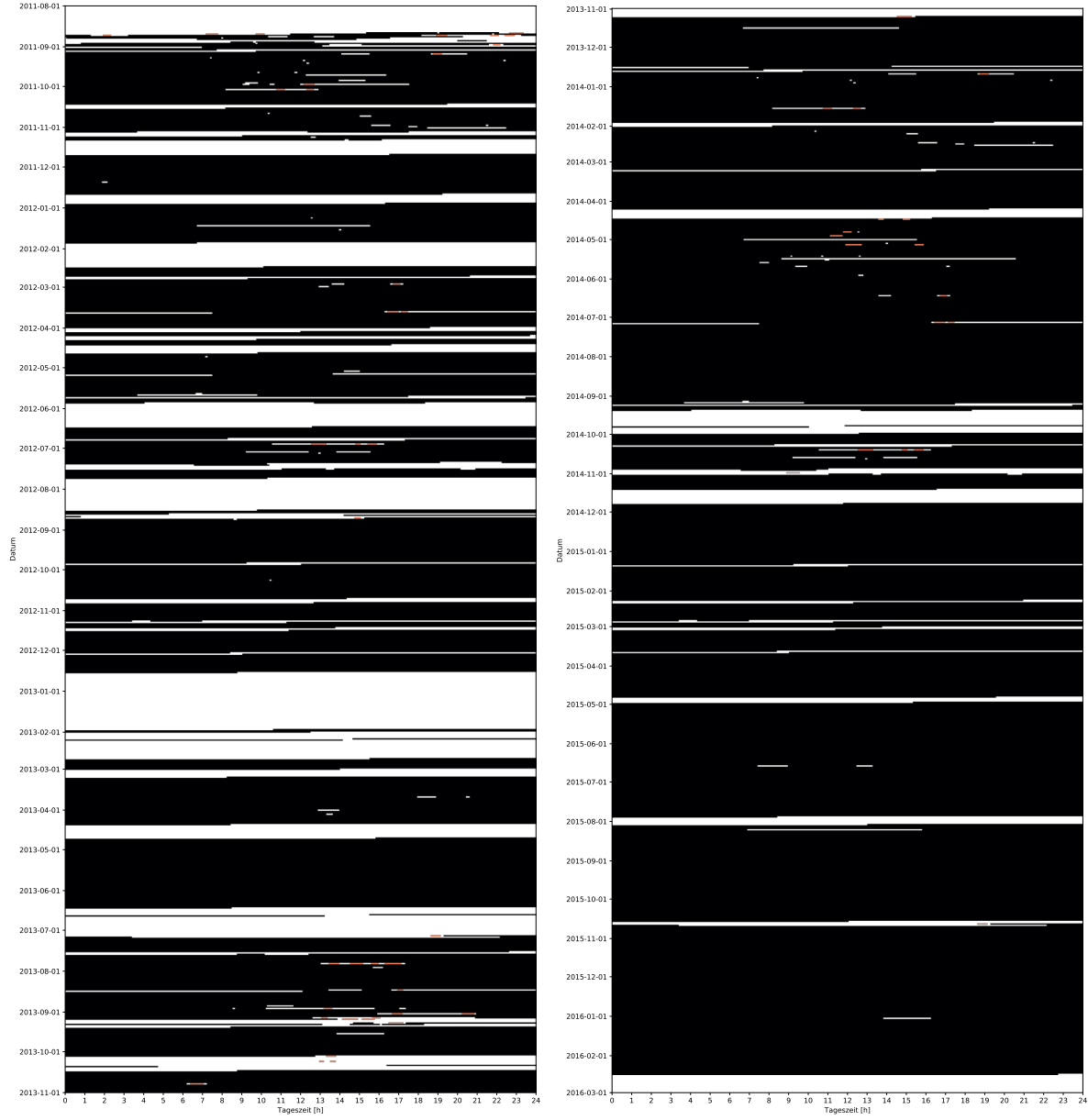


Abbildung 4.14: Verfügbarkeit der Messdaten im ESHL, erfasst durch das von 2011 bis 2016 eingesetzte Messsystem.

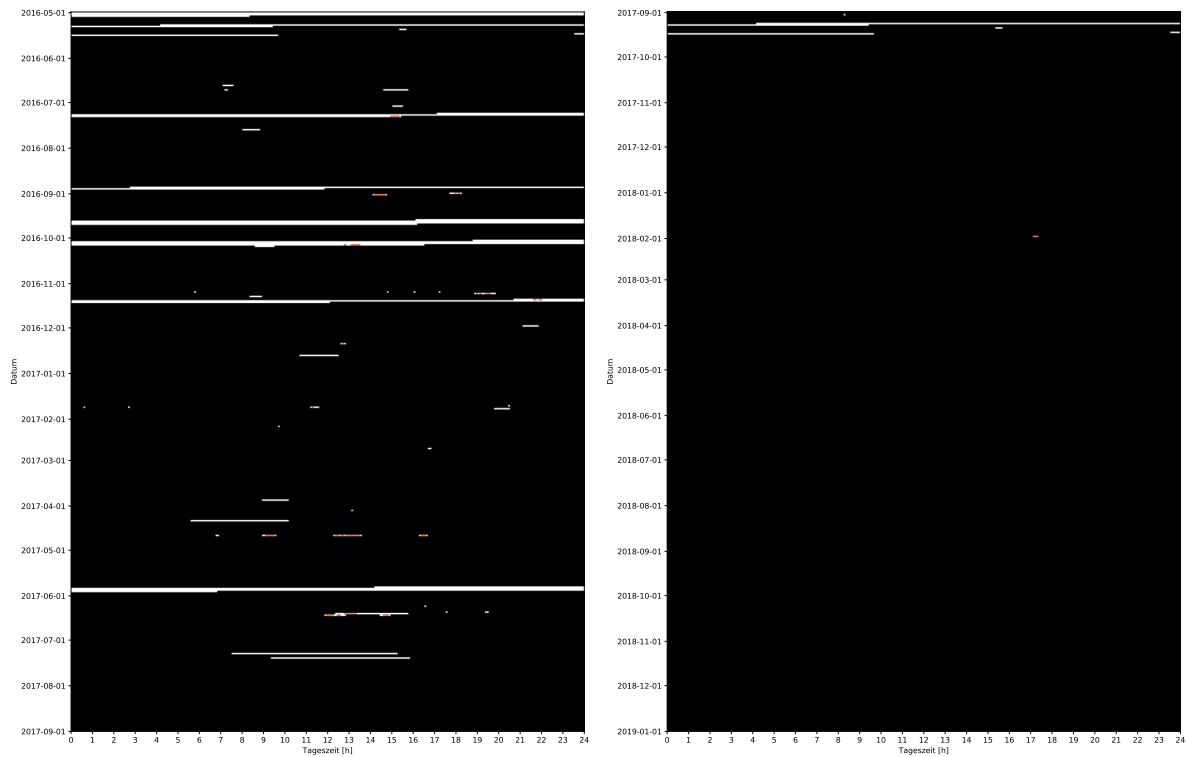


Abbildung 4.15: Verfügbarkeit der Messdaten im ESHL, erfasst durch das seit Mai 2016 eingesetzte Messsystem.

Ausgewählte Abschnitte des Datensatzes

Für die Disaggregationsexperimente werden zwei unterbrechungsfreie Zeitabschnitte des ESHL-Datensatzes verwendet, die jeweils eine Wohnnutzung repräsentieren und im Rahmen dieser Arbeit als Rohdaten mitveröffentlicht werden.

Der erste Abschnitt stammt aus dem Mai 2012 und ist über 9 Tage lang. Während dieser Zeit hat der Autor dieser Arbeit im ESHL zum Testen des Energiemanagementsystems gewohnt und gearbeitet. Dieser Abschnitt wurde so gewählt, dass die untersuchten Haushaltsgeräte möglichst häufig genutzt werden.

Der zweite Abschnitt repräsentiert 10 Tage aus dem Juli 2018, von denen 6 Tage tatsächlich eine Person im ESHL gewohnt hat. In den zusätzlichen 4 Tagen wurde das ESHL zur Vor- und Nachbereitung der Wohnphase als Büroraum genutzt. Da die relevanten Haushaltsgeräte auch während dieser 4 Tage häufig genutzt wurden, werden die zusätzlichen Daten ebenfalls für die Disaggregationsexperimente verwendet. Innerhalb der 6 Tage ereignete sich ein simulierter Stromausfall von etwa 2 Stunden.

Da beide Abschnitte kurz sind und der Datensatz nur ein Gebäude enthält, werden alle Daten des ESHL ausschließlich zur Evaluation als Testdaten verwendet.

Elektrischer Energieverbrauch in den verwendeten Abschnitten

Alle für diese Studie ausgewählten Geräteklassen sind in beiden Abschnitten des Datensatzes vorhanden und wurden in dieser Zeit mehrmals benutzt. In Tabelle 4.6 ist dazu eine Zählung der Aktivierungen in den zwei Abschnitten des ESHL-Datensatzes aufgeführt.

Tabelle 4.6: Gesamtzahl der Aktivierungen im Testdatensatz aus dem ESHL.

	Zeitabschnitte	
	1	2
Datenmenge in Tagen	9	10
Anzahl Aktivierungen je Haushaltsgerät		
Geschirrspülmaschine	15	5
Waschmaschine	3	4
Wasserkocher	22	3
Mikrowelle	4	3
Kühlschrank	251	266
Tiefkühlschrank	227	259
TV	3	37

Unter Betrachtung der Verteilung des tatsächlichen Energieverbrauchs im ESHL während dieser beiden Abschnitte wird deutlich, dass die zuvor ausgewählten Haushaltsgeräteklassen sich nur teilweise mit den Haushaltsgeräten, die sich durch den größten Energieverbrauch auszeichnen, überschneiden. Zur Veranschaulichung sind in den Abbildungen 4.16 und 4.17 jeweils 9 Haushaltsgeräte dargestellt, die während der beiden Zeitabschnitte den größten Anteil am Energieverbrauch hatten. Jahreszeitbedingt unterscheiden sich die anteilmäßig größten Energieverbraucher in beiden Abschnitten.

Im ersten Zeitabschnitt herrschte ein wechselhaftes, überwiegend kaltes Wetter. Der Energieverbrauch wird durch den elektrischen Einschraubheizkörper dominiert, welcher für die Heizung und Bereitstellung von warmem Brauchwasser genutzt wurde. An zwei Tagen dieses Zeitabschnitts stieg die Temperatur stark an, sodass die Klimaanlage genutzt wurde.

Im zweiten Zeitabschnitt wurde für die Heizung und das Warmwasser das Blockheizkraftwerk mit Gas als Primärenergiequelle eingesetzt. Der Energieverbrauch wird durch die Klimaanlage dominiert, da in diesem Zeitraum die Tageshöchsttemperaturen zwischen 24 °C und 36 °C schwankten.

Wie in Abschnitt 4.2 diskutiert, zeigt sich auch am Beispiel der Energieverbrauchsdaten des ESHL, dass die Vorauswahl der Haushaltsgeräte für das Anwendungsszenario der Identifikation der größten Energieverbraucher im Haushalt ungeeignet ist. Dies geben die Abbildungen 4.18 und 4.19 wieder, in denen jeweils der Anteil der ausgewählten Haushaltsgeräte mit dem elektrischen Gesamtenergieverbrauch in Relation gesetzt wird: Über 79 % bzw. 84 % des elektrischen Energieverbrauchs stammen nicht von den ausgewählten Haushaltsgeräten. Für zukünftige NILM-Datensätze sollte daher der Fokus auf Einzellastgänge von insbesondere thermischen Anlagen gelegt werden, um die Anwendbarkeit von Energiedatendisaggregation zu erhöhen.

Anschluss der Energieerzeuger

Die auf dem ESHL installierte Photovoltaikanlage sowie das Mikroblokheizkraftwerk sind hinter dem Stromzähler des Haushalts angeschlossen, sodass deren erzeugte Energie durch den Haushaltsstromzähler mitgemessen wird. Vorhandene Arbeiten betrachten bisher ausschließlich Haushalte, die vollständig aus Energieverbrauchern bestehen. Durch die Aufnahme von Energieerzeugern in der Leistungsbilanz ändert sich die Problemstellung der Disaggregation, die durch Gleichung 2.9 repräsentiert wird, zunächst nicht. Es muss jedoch damit gerechnet werden, dass die Summe auch Komponenten mit negativen Vorzeichen enthält.

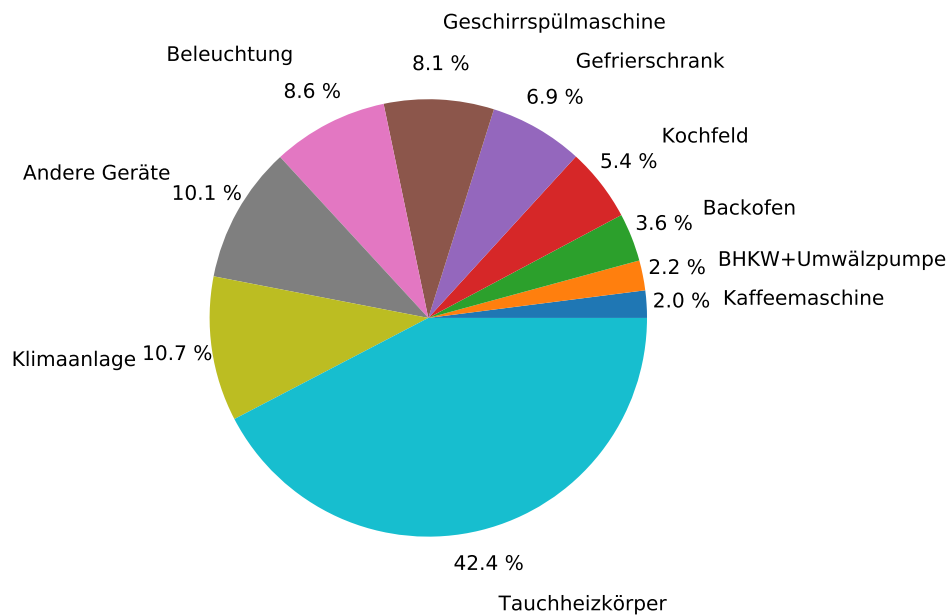


Abbildung 4.16: Haushaltsgeräte mit dem größten Anteil am Energieverbrauch während des ersten Zeitabschnitts (Mai 2012) im ESHL-Datensatz.

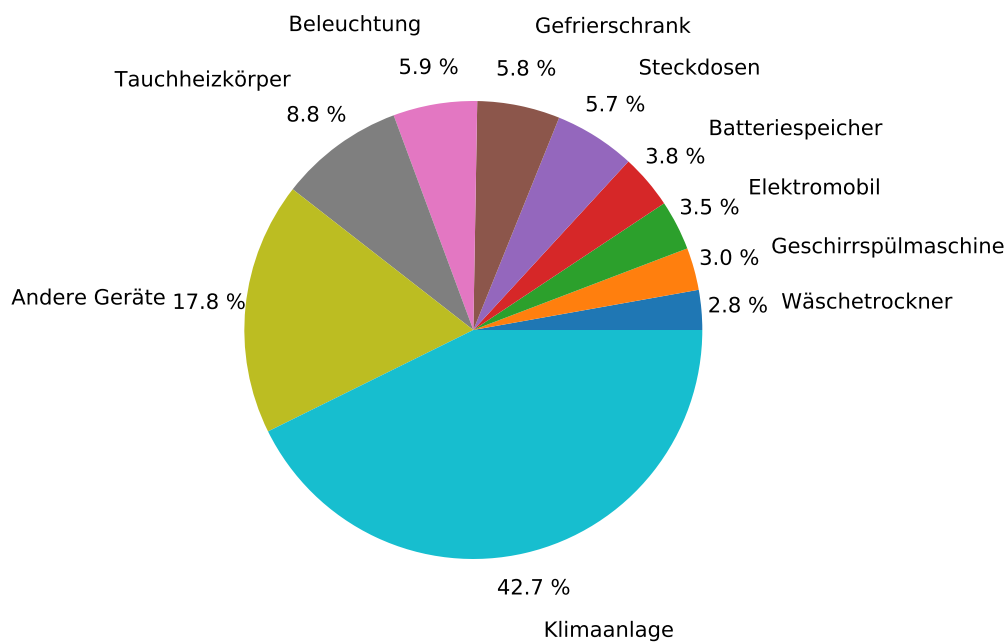


Abbildung 4.17: Haushaltsgeräte mit dem größten Anteil am Energieverbrauch während des zweiten Zeitabschnitts (Juli 2018) im ESHL-Datensatz.

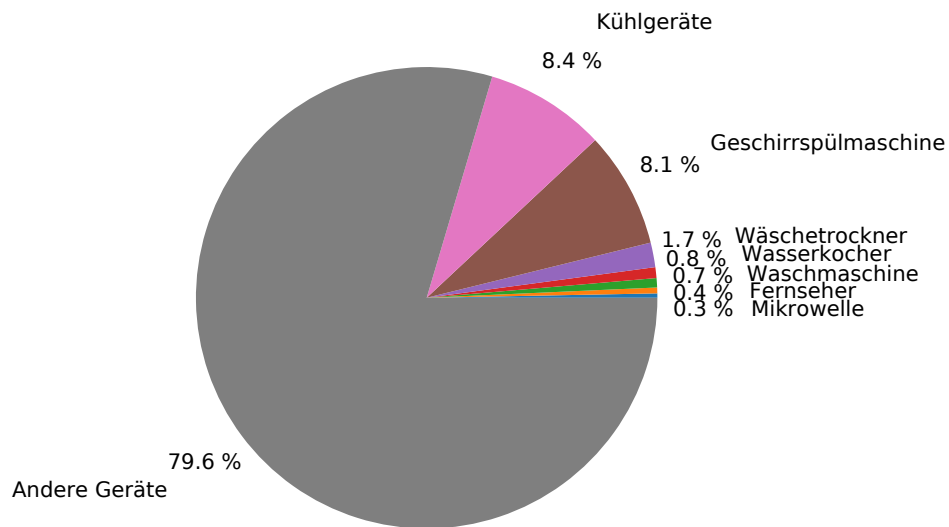


Abbildung 4.18: Anteiliger Energieverbrauch von Haushaltsgeräten, die für eine Disaggregation betrachtet werden. Daten stammen aus dem ersten Zeitabschnitt (Mai 2012) im ESHL-Datensatz.

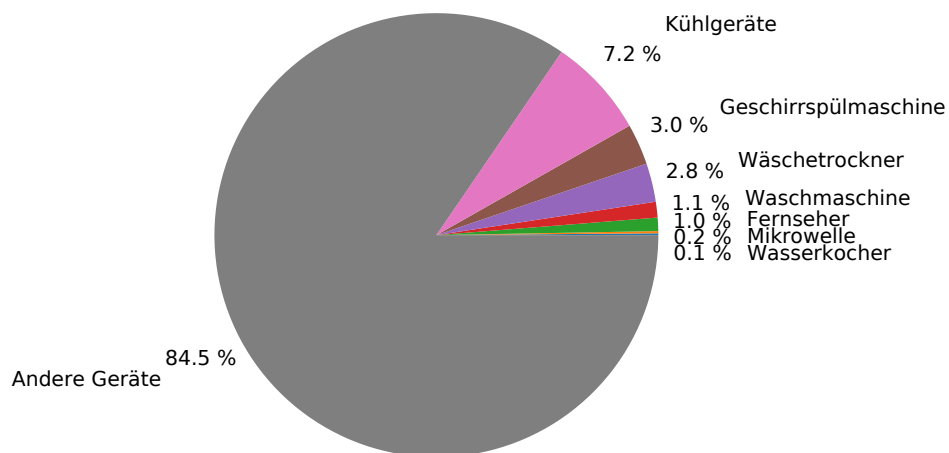


Abbildung 4.19: Anteiliger Energieverbrauch von Haushaltsgeräten, die für eine Disaggregation betrachtet werden. Daten stammen aus dem zweiten Zeitabschnitt (Juli 2018) im ESHL-Datensatz.

Korrektur des Haushaltslastgangs bezüglich des Einschraubheizkörpers

Im ersten Abschnitt war der Einschraubheizkörper irrtümlicherweise nicht hinter dem Haushaltsstromzähler angeschlossen. Wie in Abbildung 4.16 dargestellt, gehört dieses Gerät jedoch zu den größten Energieverbrauchern im Haushalt. Der Haushaltslastgang des ersten Abschnitts wurde daher um die Last des separat gemessenen Einschraubheizkörpers korrigiert.

Im zweiten Zeitabschnitt war der Einschraubheizkörper korrekterweise an den Stromkreis des Wohnbereichs angeschlossen und wurde somit durch den Haushaltsstromzähler erfasst. Für den zweiten Zeitabschnitt werden die Messwerte somit nicht korrigiert.

Nutzung von dreiphasigen Messdaten

Im ESHL sind Blockheizkraftwerk, Solaranlage, Einschraubheizkörper und Elektromobilladebox⁶ mit drei Phasen im Stromnetz der Wohnung angeschlossen, da diese elektrischen Anlagen mit höherer Leistung Strom erzeugen bzw. verbrauchen als gewöhnliche Haushaltsgeräte. Die Anlagen im ESHL belasten die drei Phasen symmetrisch mit etwa der gleichen Leistungsaufnahme pro Phase. Diese Eigenschaft wird in dieser Studie ausgenutzt, um mit einer Gleichtaktunterdrückung (engl. Common Mode Rejection, siehe Siegl (2010), Kapitel 5) Energieverbrauch bzw. -produktion dieser Geräte aus dem Haushaltslastgang herauszufiltern.

Für die Studie werden sowohl unveränderte Haushaltslastgänge des ESHL als auch solche nach Gleichung 4.1 berechnete Haushaltslastgänge mit Gleichtaktunterdrückung evaluiert.

$$P_{cmr} = P_1 + P_2 + P_3 - \max\{P_1, P_2, P_3\} \quad (4.1)$$

4.3.6 UK-DALE: UK Domestic Appliance-Level Electricity

Der Datensatz *UK Domestic Appliance-Level Electricity (UK-DALE)* von Kelly und Knottenbelt (2015b) besteht aus Haushalts- und Gerätelastgangsmessungen von 5 Gebäuden im Vereinigten Königreich. Neben dem Datensatz REDD ist UK-DALE eine der am häufigsten verwendeten Datensätze für Deep-Learning-basierte NILM-Ansätze (siehe Kapitel 3). Der Datensatz wurde erstmals im Jahr 2015 veröffentlicht und ist im Jahr 2017 aktualisiert worden. Die aktuellere Version aus dem Jahr 2017 enthält eine längere Aufzeichnung von Gebäude 1.

⁶Im ersten Abschnitt wurde die dreiphasige Wandladebox verwendet, im zweiten Abschnitt wurde ein Elektroauto einphasig über eine Außensteckdose geladen.

Der Datensatz hat eine vergleichsweise gute Qualität, da selten fehlende Messwerte zu beobachten sind. Beispielhaft sind in Abbildung 4.20 valide Messdaten aus Gebäude 5 als schwarze Balken dargestellt. Es ist zu erkennen, dass der Messaufbau bis auf wenige Ausnahmen unterbrechungsfrei gearbeitet hat.

In Tabelle 4.7 sind Merkmale dieses Datensatzes zusammengefasst. In Gebäude 1 wurde die umfangreichste Messkampagne durchgeführt, während von Gebäude 3 kaum Daten von den zuvor definierten, relevanten Geräten vorliegen.

Tabelle 4.7: Gesamtzahl der Aktivierungen in Datensatz UK-DALE 2017. Die Segmentierungsheuristik ist in Abschnitt 4.4.2 beschrieben.

	UK-DALE Gebäude				
	1	2	3	4	5
Datenmenge in Tagen	1483	140	36	155	132
Anzahl Aktivierungen je Haushaltsgesamt					
Geschirrspülmaschine	638	91	–	–	47
Waschmaschine	–	46	–	79	–
Waschtrockner	1243	–	–	–	94
Wasserkocher	7200	661	70	891	195
Mikrowelle	6573	347	–	992	54
Kühlschrank	–	3444	–	–	–
Tiefkühlschrank	–	–	–	4825	–
Kühl/Tiefkühlschrank	37294	–	–	–	2983
TV	2098	–	–	56	128

4.3.7 REFIT: Personalised Retrofit Decision Support Tools for UK Homes using Smart Home Technology

Der im Projekt *Personalised Retrofit Decision Support Tools for UK Homes using Smart Home Technology (REFIT)* entstandene Datensatz besteht aus Leistungsmessungen von 20 Haushalten im Vereinigten Königreich, die zwischen 2013 und 2015 erfasst wurden (Murray, David, 2015). Neben der Wirkleistung am Hausanschluss wurden jeweils bis zu 9 Haushaltsgesamtgeräte einzeln gemessen. Das Messintervall beträgt zwischen 6 und 8 Sekunden. Diese Studie verwendet die im Jahr 2017 veröffentlichte bereinigte Version des Datensatzes von Murray et al. (2017). In Tabelle 4.8 sind die Merkmale dieses Datensatzes zusammengefasst.

In 13 der 18 in dieser Studie verwendeten Gebäude sind mehrere Instanzen der Geräteklassen Waschmaschine, Kühlgeräte oder Fernseher vorhanden. Da zwischen unterschiedlichen

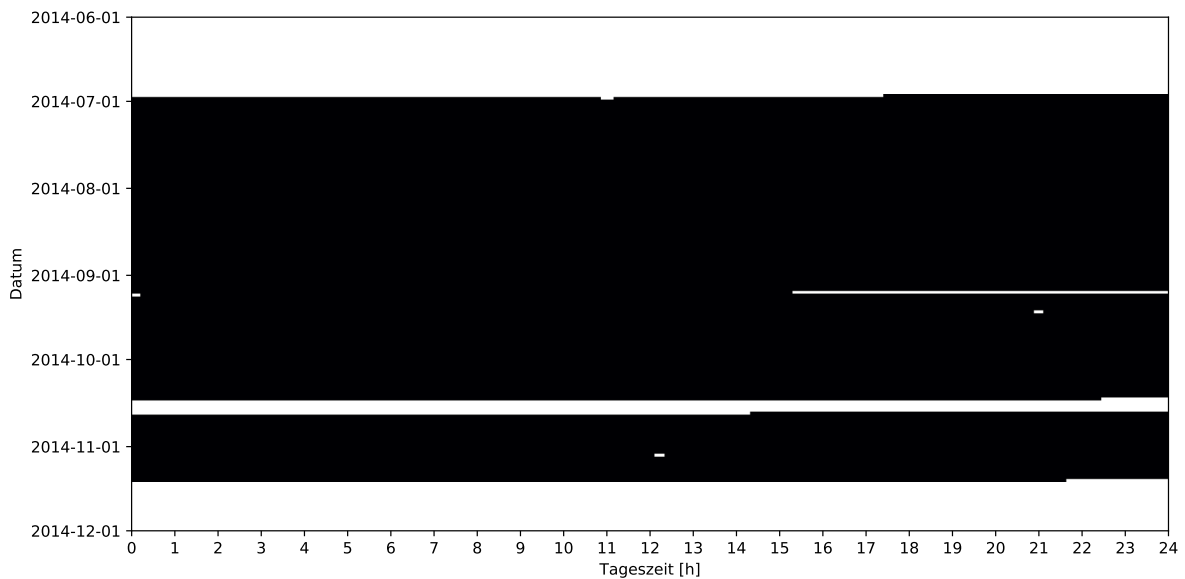


Abbildung 4.20: Verfügbarkeit der Messdaten von UK-DALE Gebäude 5. Unterbrechungen werden als weiße Balken dargestellt. Die Erkennung von Messausfällen ist in Abschnitt 4.4.2 beschrieben.

Instanzen einer Geräteklasse nicht unterschieden wird, ist das Ziel der Disaggregation die jeweiligen Summenlastgänge einer Geräteklasse.

Der Datensatz enthält keinen Haushalt mit der Nummer 14. Weiterhin werden für diese Studie die Daten der Haushalte 11 und 21 nicht verwendet, da die Photovoltaikanlage dieser Haushalte fehlerhafte Messdaten verursacht hat. Die Energieproduktion der Photovoltaikanlagen müsste zu einer Verringerung des Lastgangs führen und zeitweise zu negativen Lastwerten, wenn die Produktion den Eigenverbrauch übersteigt. In den Zeitreihen dieser Gebäude ist die charakteristische Produktionskurve mit positiven Vorzeichen zu erkennen. Auch verringern sich die Lastgangswerte, wenn Haushaltsgeräte aktiv sind während viel Photovoltaikstrom eingespeist wird. Dies legt den Schluss nahe, dass das eingesetzte Messsystem die Stromflussrichtung nicht erfasst hat. Die Daten entsprechen somit nicht der Summenlast in Gleichung 2.9, sondern dem Betrag der Summenlast. Somit ist eine der Grundannahmen von NILM-Verfahren verletzt.

Da die Leistung der Photovoltaikanlage nicht einzeln gemessen wurde, besteht keine Möglichkeit, die Betragsbildung rückgängig zu machen. Daher wird auf die Verwendung der Daten aus den Gebäuden 11 und 21 verzichtet. Auch ist anhand der Lastgänge zu erkennen, dass in Gebäude 1 bis Februar 2014 eine Photovoltaikanlage installiert war. Daher werden von diesem Gebäude ausschließlich die Messdaten ab Februar 2014 genutzt.

Die Zeitreihen im REFIT-Datensatz enthalten vergleichsweise viele Messlücken. In Abbildung 4.21 sind beispielhaft die Messlücken von REFIT Gebäude 2 als weiße Bereiche dar-

gestellt. In den orange markierten Abschnitten sind zwar Messdaten vorhanden, jedoch enthalten diese Abschnitte mit einer Länge geringer als eine Stunde potentiell unvollständige Geräteaktivierungen. Für diese Studie werden daher lediglich die schwarz markierten unterbrechungsfreien Abschnitte verwendet.

4.4 Gemeinsame Vorverarbeitungsschritte

Für die Datensätze wird die in Abschnitt 4.4.1 beschriebene Erkennung und Bereinigung von Messlücken durchgeführt, um eine einheitliche Datenqualität zu erhalten und fehlerhafte Lastgänge während des Trainings und der Evaluation der Disaggregationsansätze zu vermeiden.

Als weiterer gemeinsamer Vorverarbeitungsschritt werden die Aktivierungen der jeweiligen Haushaltsgeräte bzw. Geräteklasse identifiziert. Diese in Abschnitt 4.4.2 beschriebene Unterscheidung des Geräteaktivitätszustands wird benötigt, da Lastgänge von inaktiven Geräten in der Regel lediglich aus einem einfachen konstanten Energieverbrauch bestehen und aktive Geräte in der Regel komplexe Lastgänge aufweisen. Bei der Generierung von Trainingspaaren (siehe Abschnitt 4.6) werden daher häufiger Lastgänge mit aktiven Geräten verwendet, um das Modelltraining zu beschleunigen.

4.4.1 Identifikation und Umgang mit fehlenden Messwerten

Die zuvor in Abschnitt 4.3 beschriebenen Datensätze enthalten Messlücken, die entweder durch das Auffüllen mit Ersatzwerten oder durch das Überspringen der Messintervalle behandelt werden. Beide Ansätze verfälschen Gerätemerkmale in den Lastgängen und verlagern damit das Lernziel für die Verfahren. In der Regel verringert sich dadurch die Leistungsfähigkeit der Disaggregationsansätze.

Eine Alternative zur Behandlung einzelner Messlücken ist die ausschließliche Verwendung von messlückenfreien Abschnitten, sodass Messlücken ausschließlich vor oder nach den verwendeten Abschnitten auftreten. Allerdings treten bei einigen der ausgewählten Datensätze häufig Messlücken auf, sodass bei einer Beschränkung auf messlückenfreie Abschnitte ein Großteil der Daten nicht nutzbar wäre.

Für die Disaggregationsexperimente werden daher kurze Messlückenabschnitte unter 2 Minuten toleriert und mit Ersatzwerten aufgefüllt. Abschnitte mit andauernden Messlücken über 2 Minuten werden nicht verwendet. Die Toleranzgrenze von 2 Minuten richtet sich nach der von (Murray et al., 2017) beschriebenen Bereinigung des REFIT-Datensatzes, um die Fragmentierung der Lastgänge zu beschränken.

Tabelle 4.8: Gesamtzahl der Aktivierungen im REFIT-Datensatz.

	REFIT Gebäude								
	1	2	3	4	5	6	7	8	9
Datenmenge in Tagen	454	461	542	541	583	487	535	496	479
Anzahl Aktivierungen je Haushaltsgerät									
Geschirrspülmaschine	119	436	441	–	728	93	643	–	360
Waschmaschine (1)	285	315	630	56	644	119	829	366	239
Waschmaschine (2)	–	–	–	198	–	–	–	–	–
Waschtrockner	–	–	–	–	–	–	–	–	255
Wäschetrockner	12	–	508	–	529	–	429	96	–
Wasserkocher	–	3436	3889	1916	4248	4487	1541	3464	3137
Mikrowelle	–	1307	912	1763	3965	2313	–	1721	267
Kühlschrank	4654	–	–	6983	–	–	11100	9680	–
Tiefkühlschrank (1)	11320	–	20862	6302	–	18093	22858	11663	–
Tiefkühlschrank (2)	9239	–	–	–	–	–	1038	–	–
Kühl/Tiefkühlschrank (1)	–	7278	4974	14462	25905	–	–	–	15136
Kühl/Tiefkühlschrank (2)	–	–	–	–	–	–	–	–	–
TV (1)	1123	1160	1533	1959	793	3920	1460	1548	821
TV (2)	–	–	–	–	–	–	–	–	–

	REFIT Gebäude								
	10	12	13	15	16	17	18	19	20
Datenmenge in Tagen	505	452	371	486	438	440	416	438	432
Anzahl Aktivierungen je Haushaltsgerät									
Geschirrspülmaschine	464	–	165	23	346	–	151	–	136
Waschmaschine (1)	579	–	398	236	298	279	118	254	185
Waschmaschine (2)	–	–	–	–	–	–	–	–	–
Waschtrockner	–	–	–	–	–	–	12	–	–
Wäschetrockner	–	–	107	63	–	20	–	–	171
Wasserkocher	–	2564	1109	1340	–	4623	–	2544	2035
Mikrowelle	783	964	515	288	–	1046	330	882	996
Kühlschrank	–	–	–	–	–	–	9807	–	9977
Tiefkühlschrank (1)	8643	–	–	–	–	9720	10278	–	7743
Tiefkühlschrank (2)	–	–	–	–	–	–	–	–	–
Kühl/Tiefkühlschrank (1)	11637	5082	–	8853	5348	9092	20898	9670	–
Kühl/Tiefkühlschrank (2)	–	–	–	–	9788	–	–	–	–
TV (1)	11	1583	128	755	1355	1246	38	1210	1281
TV (2)	–	615	480	–	–	110	–	–	–

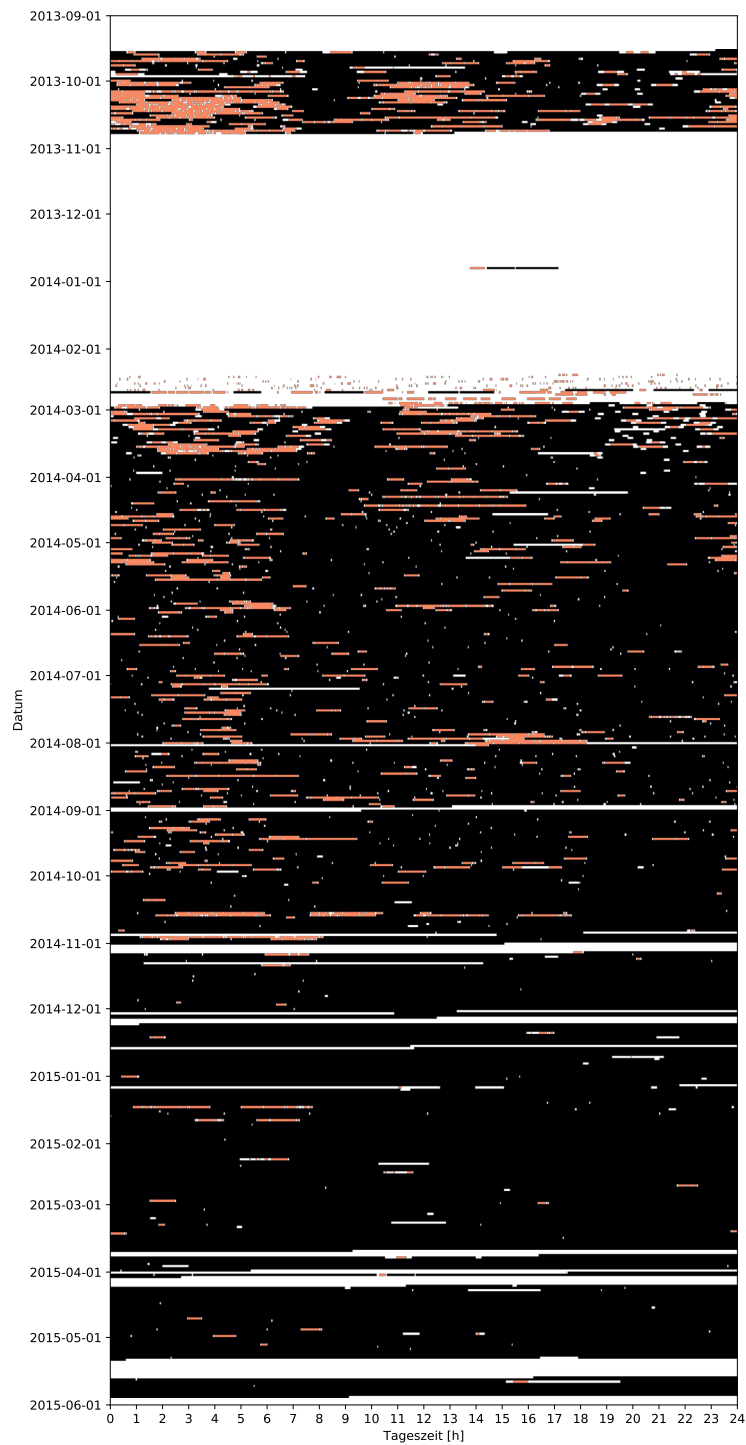


Abbildung 4.21: Verfügbarkeit der Messdaten von REFIT Gebäude 2. Unterbrechungen werden als weiße Balken dargestellt. Zeitabschnitte mit unzuverlässigen Messsystem sind orange dargestellt. Die Erkennung von Messausfällen ist in Abschnitt 4.4.1 beschrieben.

Die Messlücken sind nicht immer durch fehlende Werte erkennbar, da die Messlücken in manchen Datensätzen bereits aufgefüllt sind und Messsystemausfälle nicht durch Fehlwerte gekennzeichnet sind. Für die Datenbereinigung wird daher die Beobachtung genutzt, dass im Haushaltslastgang stets eine Grundlast existiert, deren gemessene Last rauschbehaftet ist. Dadurch unterscheiden sich die Messwerte zwischen benachbarten Messintervallen in der Regel.

Somit werden Messlücken im Haushaltslastgang sowohl anhand fehlender Messwerte als auch durch das Fehlen von Messrauschen erkannt. Die Methodik lässt sich nicht direkt zum Erkennen von Messlücken in Gerätelastgängen übertragen, da die Leistungsaufnahme von inaktiven Geräten so gering ist, dass das Messsystem das Rauschen nicht auflöst. Die Identifikation und das Aussondern von messfehlerbehafteten Lastgängen bei Geräteaktivität wird im Zuge der Segmentierung von Aktivierungen im nächsten Abschnitt erklärt.

4.4.2 Segmentierungsheuristik für Aktivierungen

Das Ziel der Segmentierungsheuristik für Aktivierungen ist die Einteilung von Gerätelastgängen in zusammenhängende Lastgangsabschnitte, in denen das Gerät entweder inaktiv ist oder aktiv mit einem Arbeitsvorgang beschäftigt ist. Da Gerätelastgangsabschnitte mit aktivem Gerät in der Regel komplexer und damit schwieriger zu erlernen sind, ist die Anzahl der Aktivierungen eine wesentliche Kenngröße für einen Datensatz. Die Unterscheidung zwischen Ruhezustand und Aktivität spielt auch eine wesentliche Rolle bei der Verteilung der Trainingspaare.

Geräteaktivität ist grundsätzlich an einem erhöhten Energieverbrauch zu erkennen. Während ein Haushaltsgerät mit einem zusammenhängenden Arbeitsvorgang beschäftigt ist, können jedoch auch Pausen mit niedriger Leistungsaufnahme vorkommen. Um möglichst zusammenhängende und vollständige Abschnitte zu bilden, wird folgende Heuristik mit den in Tabelle 4.9 definierten Schwellwerten verwendet: Innerhalb eines Aktivitätszyklus muss die Leistungsaufnahme den Grenzwert $P_{min}^{(on)}$ mindestens $T_{min}^{(on)}$ (nicht zwingend zusammenhängende) Sekunden lang überschritten sein. Nach einem Aktivitätszyklus ist der Grenzwert $P_{min}^{(on)}$ mindestens $T_{min}^{(off)}$ zusammenhängende Sekunden lang unterschritten. Alle anderen Abschnitte werden als inaktive Lastgangssegmente gewertet.

Die in dieser Arbeit verwendeten Lastgrenzwerte sind in Tabelle 4.9 zusammengefasst. Dabei basieren die Grenzwerte auf der Arbeit von Kelly und Knottenbelt (2015a), wurden für die größere Anzahl der in dieser Studie verwendeten Datensätze so angepasst, dass Aktivierungen weniger wahrscheinlich abgeschnitten werden. Diese Wahl dieser Grenzwerte bewirkt, dass eigentlich unabhängige Geräteaktivierungen, die zeitlich direkt aufeinander folgen, als eine lange Aktivierung angesehen werden.

Da bei manchen Trainingsansätzen Aktivierungen außerhalb ihres Kontextes verwendet werden (siehe Abschnitt 4.6), führt das Zusammenfügen von mehreren Aktivierungen im Gegensatz zum verfrühten Abschneiden von Aktivierungen nicht zu ungültigen Lastgängen, da auch zusammengefügte Aktivierungen vollständig abgeschlossene Arbeitsvorgänge abbilden. Das Zusammenführen beschränkt lediglich die Kombinationsmöglichkeiten für die Trainingspaargeneratoren.

Bei der Evaluation wird diese Segmentierungsheuristik nicht verwendet. Hier gilt ein Gerät lediglich als aktiv, wenn für das Messintervall der Leistungsgrenzwert $P_{min}^{(on)}$ überschritten wird.

Tabelle 4.9: Grenzwerte zur Aktivierungssegmentierung je Haushaltsgerät:

Minimale Last für Aktivität $P_{min}^{(on)}$, Mindestdauer einer Aktivierung $T_{min}^{(on)}$,
Mindestdauer für die Geräteinaktivität $T_{min}^{(off)}$.

Haushaltsgerät	$P_{min}^{(on)} / W$	$T_{min}^{(on)} / s$	$T_{min}^{(off)} / s$
Geschirrspülmaschine	15	1800	1800
Waschmaschine/Waschtrockner	20	1800	900
Wäschetrockner	20	1800	360
Wasserkocher	400	12	0
Mikrowelle	200	12	72
Kühl/Gefriergerät	30	60	12
TV/Unterhaltungselektronik	15	300	300

Verwerfen von Aktivierungen mit Messfehlern

Mit der Aufteilung des Lastgangs in aktive und inaktive Segmente können aktive Lastgangssegmente strenger gefiltert werden. Während das Lastprofil eines inaktiven Gerätes konstant niedrig ist, kann das Lastprofil in den aktiven Phasen sehr variabel sein. Messausfälle führen in diesen Zeiten daher zu großen Fehlern.

Lastgangssegmente von aktiven Geräten werden daher auf folgende empirisch bestimmte Kriterien geprüft:

- mindestens 19 % der Messwerte im Gerätelastgang müssen sich vom jeweiligen Vorwert unterscheiden und
- mindestens 32 % der Messwerte im Haushaltslastgang müssen sich vom jeweiligen Vorwert unterscheiden.

Die Zeitbereiche der Aktivierungen, die diese Kriterien nicht erfüllen, werden von dem Training ausgeschlossen. Diese spezifischen Grenzwerte wurden empirisch ermittelt, um genau jene bei der manuellen Prüfung der Datensätze entdeckten, eindeutig fehlerhaften Aktivierungen auszusortieren.

Eine analoge Heuristik lässt sich für Lastgänge inaktiver Geräte nicht anwenden, da es im Ruhezustand kaum Änderungen in den Gerätelastgangswerten gibt. Damit sind bei einem inaktiven Gerät gültige Messdaten und Messlücken nicht unterscheidbar.

4.5 Aufteilung in Trainings- und Testdatenmenge

Eine Evaluation von Disaggregationsverfahren erfordert eine überschneidungsfreie Trennung des verwendeten Datensatzes bzw. der verwendete Datensätze in Trainings- und Testdatenmenge, um die Leistung der Ansätze mit echten ungesehenen Daten abschätzen zu können (Mitchell, 1997). Da in dieser Arbeit Daten von mehreren Gebäuden verwendet werden, besteht die Möglichkeit, sämtliche Daten eines Gebäudes zur Testdatenmenge zu zählen oder lediglich bestimmte Zeitabschnitte der jeweiligen Gebäude als Testdaten zu verwenden.

Bei einer gemeinsamen Verwendung von Daten eines Gebäudes für Training und Evaluation besteht das Problem, dass sich die Verfahren beim Training auf die Zielhaushalte und die Zielgeräte adaptieren können. Beispiele der spezifischen Gerätelastgänge sind damit bereits aus dem Training bekannt. Im Anwendungsfall erfordert eine solche Aufteilung, dass eine aufwendige Instrumentierung des zu disaggregierenden Gebäudes mit einzelnen Messgeräten durchgeführt wird, um später die Disaggregation ohne Hilfe der einzelnen Messgeräte durchführen zu können. Dies entspricht nicht dem Ziel dieser Arbeit, dass die Ansätze insbesondere auf unbekannte Haushalte anwendbar sein sollen.

Aus diesem Grund wird die Trainings-/Testmengentrennung nach Gebäuden durchgeführt. Bei der Wahl der Testgebäude werden die Gebäude mit einem möglichst vollständigen Satz der zuvor ausgewählten Haushaltsgeräteklassen bevorzugt. Auch werden Testgebäude mit geringerer Datenmenge bevorzugt, um der Trainingsphase wenig Daten zu entziehen. Da die Disaggregationsstudie einen hohen Rechenaufwand für das Training erfordert, ist die Trainings-/Testdatenaufteilung fix und es wird keine Kreuzvalidierung durchgeführt.

Damit erfolgt die Aufteilung in Trainings- und Testdatenmenge so wie in Tabelle 4.10 dargestellt. Der Datensatz DRED enthält nur ein Gebäude und dies wird daher nur als Testdatensatz verwendet. In den Gebäuden des Datensatzes ECO sind die ausgewählten Geräte spärlich verteilt und somit eignet sich die Daten keines Gebäudes als Testdatensatz. Von dem ESHL-Datensatz werden lediglich zwei etwa einwöchige Zeitabschnitte verwendet, die

Tabelle 4.10: Aufteilung in Trainings- und Testdatenmenge nach Gebäuden

Datensatz	Testgebäude	Trainingsgebäude
DRED	1	–
ECO	–	1 - 6
ESHL	1 - 2	–
UK-DALE	5	1 - 4
REFIT	9, 15, 20	1 - 8, 10, 12, 13, 16 - 19

Abschnitte enthalten jedoch jeweils mehrfache Aktivierungen der betrachteten Haushaltsgeräte. Beide Abschnitte werden als Testdaten verwendet. Bei den Datensätzen UK-DALE und REFIT wurden die Gebäude mit möglichst vollständigem Gerätesatz zum Testen ausgewählt, wobei gleichzeitig keine Testgebäude mit dem größten Anteil an der Gesamtdatenmenge ausgewählt sind. Insgesamt zählt somit etwa ein Fünftel der verwendeten Gebäude zum Testdatensatz.

4.6 Generatoren für Trainingspaare

Zum Training der neuronalen Netze werden Datenpaare bestehend aus Eingabe- und Zielvektoren mit einer einheitlichen Vorgehensweise erzeugt, um die Vergleichbarkeit der untersuchten Verfahren untereinander sicherzustellen. Basierend auf den bestehenden Trainingspaargeneratoren, die in Abschnitt 4.6.1 erläutert werden, erfolgten im Rahmen dieser Arbeit eigene Ergänzungen. Dementsprechend werden in den folgenden Unterabschnitten 4.6.2 und 4.6.3 zwei Modifikationen vorgestellt, die für die verwendeten Datensätze geeigneter erscheinen. Im Anschluss wird zusammengefasst, welche Trainingspaargeneratoren bei den jeweiligen Haushaltsgeräteklassen verwendet werden.

4.6.1 Vorhandene Generatoren für Trainingspaare

Die aus der Literatur bekannten Möglichkeiten für die Erstellung von Trainingspaaren sind in Abschnitt 3.4 ausführlich beschrieben worden. Zusammenfassend sind folgende Methoden zur Trainingspaargenerierung bekannt:

Balancierte Auswahl gemessener Lastgänge

Hier erfolgt die Aufteilung des Datensatzes in Lastgangssegmente fester Länge mit unabhängiger zufälliger Wahl von Gebäuden und deren Lastgangssegmenten. Geräteaktivierungen und Lastgangssegmente mit inaktivem Geräte werden mit einer festgelegten Wahrscheinlichkeit ausgewählt. (Kelly und Knottenbelt, 2015a)

Randomisierte Ausschnitte realer Lastgänge

Dies beschreibt die Aufteilung des vollständigen Datensatzes in Lastgangssegmente mit anschließender sequenzieller Auswahl der zufällig durchmischten Lastgangssegmente. Sobald alle Segmente verarbeitet sind, wird der Datensatz erneut zufällig durchmischt. (Zhang et al., 2016)

Haushaltlastgangsynthese aus Aktivierungen

Bei diesem Ansatz erfolgt die Generierung von synthetischen Haushaltlastgängen aus einer zufälligen Auswahl von Geräteaktivierungen der betrachteten Haushaltsgeräte. (Kelly und Knottenbelt, 2015a)

4.6.2 Zufallssegmentgenerator mit ausbalancierter Gebäude- und Aktivitätshäufigkeit

Der „Randomisierte Ausschnitte realer Lastgänge“ ist für den Aufbau dieser Studie ungeeignet, da sich die Größe der Datensätze stark unterscheidet und jedes Lastgangssegment unabhängig der Quelle mit der gleichen Wahrscheinlichkeit ausgewählt wird. Somit würden beim Training jene Geräteinstanzen mit größerer Datenmenge stärker gewichtet, obwohl die Datenmengen nicht die Vorkommenshäufigkeit der Geräte in Haushalten repräsentieren.

Zur Vermeidung dieser unkontrollierten Gewichtung werden mit dem „Zufallssegmentgenerator mit ausbalancierter Gebäude- und Aktivitätshäufigkeit“ die Lastsegmente zufällig so gewählt, dass alle Gebäude der Trainingsdatenmenge mit der gleichen Häufigkeit vertreten sind. Dazu wird nach der Aufteilung des vollständigen Datensatzes in Lastgangssegmente bei der Generierung jedes Trainingspaars unabhängig zufällig entschieden, welches Gebäude der Trainingsdatenmenge ausgewählt wird. Daraufhin wird aus diesem Gebäude zufällig ein Lastgangssegment als Trainingspaar erstellt.

4.6.3 Haushaltlastgangsynthese durch Austausch von Gerätelastprofilen

Der Trainingspaargenerator „Haushaltlastgangsynthese aus Aktivierungen“ verfolgt bereits eine Datenanreicherungsstrategie (engl. data augmentation), bei dem das Ziel die Generierung neuer Trainingsbeispiele ist, die nicht direkt im Datensatz vorkommen. Dennoch müssen die generierten Beispiele für den jeweiligen Anwendungsfall plausibel sein.

Ein aus Aktivierungen generierter synthetischer Haushaltlastgang ist zwar plausibel in dem Sinne, dass die synthetischen Haushaltlastgänge möglich sind und dass der Disaggregationsansatz auch bei der Eingabe der synthetischen Lastgänge die entsprechenden Gerätelastgänge ausgeben soll. Da jedoch mit der Auswahl der Haushaltsgeräte nur ein

Bruchteil der am Gesamtenergieverbrauch beteiligten elektrischen Geräte betrachtet wird, fehlt in den synthetischen Haushaltslastgängen verglichen zu realistischen Lastgängen ein beträchtlicher Anteil des tatsächlichen Stromverbrauchs. Am Beispiel der ESHL-Daten decken die betrachteten Haushaltsgeräte lediglich 20,4 % des Gesamtstromverbrauchs ab (siehe Abbildungen 4.18 und 4.19). Die so generierten Daten sind zwar plausibel, jedoch ist es unwahrscheinlich, dass solche Daten bei einer tatsächlichen Disaggregation beobachtet werden. Aus diesem Grund wird die Haushaltslastgangsynthese aus Aktivierungen im Rahmen dieser Studie nicht verwendet.

Stattdessen wird die alternative Datenanreicherungsstrategie „Haushaltslastgangsynthese durch Austausch von Gerätelastprofilen“ vorgestellt, die lediglich die Zielgerätelast von der Haushaltslast subtrahiert und Geräteaktivierungen zufällig neuplatziert. Damit bleibt die Last aller anderen Geräte – auch jene, die nicht zu den ausgewählten Geräten zählen – im Haushaltslastgang erhalten.

Zusammengefasst werden synthetische Haushaltslastgänge erzeugt, indem zunächst ein Abschnitt des Haushaltslastgangs zufällig ausgewählt wird, während dessen sich das Haushaltsgerät im Ruhezustand befindet. Eine Aktivierung der gleichen Geräteklasse aus einem anderen Haushalt wird nun zufällig ausgewählt und auf den Haushaltslastgang aufsummiert. Die Leistungsaufnahme des ursprünglichen Geräts im Ruhezustand wird von dem Haushaltslastgang subtrahiert.

4.6.4 Definition der Trainingsepoche und Vollständigkeit

Mit Ausnahme des „Segmentgenerators für zufällig durchmischte Datensätze“ besteht ein Nachteil der vorgestellten Trainingspaargeneratoren darin, dass sich nicht definieren lässt, wann die komplette Trainingsdatenmenge vollständig abgearbeitet ist, indem jedes mögliche Trainingspaar genau einmal generiert wurde. Dieser Zeitpunkt bildet jedoch die Grundlage für die klassische Definition einer Trainingsepoche (siehe Abschnitt 2.5.2). Daher wird eine Trainingsepoche stattdessen durch eine feste Anzahl an Trainingsiterationen definiert.

Ein weiterer Nachteil besteht in der Gefahr, dass durch die unabhängige und zufällige Auswahl bestimmte Lastgangssegmente nie zum Training verwendet werden. Für diese Studie wird dies in Kauf genommen, da angenommen wird, dass das Angleichen der Auswahlwahrscheinlichkeit von Lastgängen mit und ohne Geräteaktivität aufgrund der geringen Datenmenge von Aktivierungen und der höheren Komplexität von Aktivierungen sich positiv auf die Trainingsgeschwindigkeit auswirkt.

4.6.5 Verwendete Generatoren für Trainingspaare

In den vorigen Unterabschnitten ist dargelegt, dass die Trainingspaargeneratoren „Zufallssegmentgenerator mit ausbalancierter Gebäude- und Aktivitätshäufigkeit“ und „Haushaltslastgangsynthese durch Austausch von Gerätelastprofilen“ für die verwendeten Datensätze geeigneter sind als die in der Literatur beschriebenen Generatoren „Randomisierte Ausschnitte realer Lastgänge“ und „Haushaltslastgangsynthese aus Aktivierungen“.

Daher wird der „Zufallssegmentgenerator mit ausbalancierter Gebäude- und Aktivitätshäufigkeit“ für Kühl- und Gefriergeräte und eine Kombination des „Balancierte Auswahl gemessener Lastgänge“ und der „Haushaltslastgangsynthese durch Austausch von Gerätelastprofilen“ für alle übrigen Haushaltsgeräteklassen verwendet.

Kühl- und Gefriergeräte sind häufig und regelmäßig aktiv, sodass keine Datenanreicherungsstrategie notwendig ist. Zudem wird es bei größeren Segmentlängen schwierig, Segmente ohne Aktivierung zu finden. Daher wird für diese Geräteklasse ausschließlich der „Zufallssegmentgenerator mit ausbalancierter Gebäude- und Aktivitätshäufigkeit“ verwendet.

Alle anderen Haushaltsgeräteklassen werden selten verwendet und benötigen Trainingspaargeneratoren, die Geräteaktivierungen häufiger vorlegen. Dazu wird während des Trainings für jedes Datenpaar zunächst gleichwahrscheinlich entschieden, ob ein echter Haushaltslastgang mit dem „Balancierte Auswahl gemessener Lastgänge“ oder ein synthetischer Lastgang mit der „Haushaltslastgangsynthese durch Austausch von Gerätelastprofilen“ erzeugt wird. Der gewählte Generator entscheidet anschließend mit je 50 % Wahrscheinlichkeit, ob in dem zu generierenden Lastgang das Zielgerät aktiv sein soll oder sich im Ruhezustand befindet. Ein entsprechendes Lastgangssegment wird basierend auf diesen Spezifikationen als Trainingspaar gewählt oder erzeugt.

4.7 Betrachtete Auflösung

Als Basisauflösung wird ein Messintervall von 6 Sekunden definiert. Dies ist die höchste gemeinsame Auflösung der ausgewählten Datensätze.

Für Sensitivitätsanalysen könnten künftig unterschiedliche Auflösungen bzw. Messintervalllängen betrachtet werden, um z. B. die Wirkung der Auflösungsreduktion auf einen Messintervall von 15 Minuten zu untersuchen. Da in Zukunft möglicherweise die Auflösung der Lastgänge, die zwischen Strommarktakteuren ausgetauscht werden, auf ein Messintervall von 5 Minuten erhöht wird, wäre auch diese Auflösung relevant.

Tabelle 4.11: Konfusionsmatrix bei einer binären Klassifikation (vgl. Ting (2017))

	Tatsächlich Positiv		Tatsächlich Negativ	
Positiv klassifiziert	TP	Echt Positiv (engl. true positive)	FP	Falsch Positiv (engl. false positive)
Negativ klassifiziert	FN	Falsch Negativ (engl. false negative)	TN	Echt Negativ (engl. true negative)

Zunächst gilt es jedoch im Rahmen einer möglichst geringen Auflösung festzustellen, wie die Disaggregationsqualität optimiert werden kann. Um hierfür eine möglichst breite Datenbasis heranziehen zu können, orientiert sich die gewählte Basisauflösung an den vorhandenen Datensätzen.

4.8 Evaluationsmetriken

Evaluationsmetriken für die Energiedatendisaggregation dienen zum quantitativen Vergleich unterschiedlicher NILM-Verfahren. Dabei heben die Metriken spezifische Qualitäten der Verfahren hervor und sind auf einzelne Anwendungsfälle zugeschnitten.

4.8.1 Klassifikationsmetriken

Mit der Familie der Klassifikationsmetriken wird beurteilt, ob ein Verfahren in der Lage ist, aus dem Gesamtlastgang den Aktivitätszustand des Zielgeräts für jedes Messintervall korrekt zu schätzen. Dabei wird in dieser Studie nur zwischen den beiden Zuständen Aktiv und Ruhezustand unterschieden. Mit dem aktiven Zustand werden gegebenenfalls unterschiedliche aktive Zustände zusammengefasst. Der Ruhezustand umfasst Standby-Modi und den vollständig ausgeschalteten Zustand.

Mit Klassifikationsmetriken wird die Disaggregation daher als binäres Klassifikationsproblem betrachtet, wobei die positive Klasse dem An- bzw. Aktiv-Zustand entspricht und die negative Klasse für den Aus- bzw. Ruhezustand des Gerätes steht. Mit einer Gegenüberstellung des tatsächlichen und geschätzten Zustands lässt sich eine Konfusionsmatrix (siehe Tabelle 4.11) erstellen, in dem die Häufigkeit der jeweiligen Kombination ermittelt wird.

Mit den Werten der Konfusionsmatrix (Tabelle 4.11) sind u. a. folgende binäre Klassifikationsmetriken definiert:

Accuracy Die *Accuracy* Acc beschreibt das Verhältnis der richtig klassifizierten Beispiele gegenüber allen Beispielen der Testdatenmenge:

$$Acc := \frac{TP + TN}{TP + TN + FP + FN} \quad . \quad (4.2)$$

Im Kontext von Haushaltsenergiedaten ist die *Accuracy* keine geeignete Metrik für die Bewertung von Disaggregationsverfahren. Da sich viele Haushaltsgeräte größtenteils im Ruhezustand befinden, erreicht eine triviale Schätzung, die stets negativ klassifiziert, bereits eine hohe Anzahl an echt negativen Beispielen (true negatives). Die *Accuracy*-Formel wird durch den Wert TN dominiert und ist nahe 1.

F₁-Maß Um eine Metrik zu erhalten, die Disaggregationsansätze stärker unterscheidet, lässt sich in der *Accuracy*-Formel TN durch TP substituieren, sodass effektiv die Anzahl an TN ignoriert werden. Das Ergebnis der Substitution ist äquivalent zum F_1 -Maß, das sich als harmonisches Mittel zwischen *Recall* R und *Precision* P ausdrücken lässt:

$$R = \frac{TP}{TP + FN} \quad , \quad (4.3)$$

$$P = \frac{TP}{TP + FP} \quad , \quad (4.4)$$

$$F_1 = \frac{2 \cdot R \cdot P}{R + P} \quad , \quad (4.5)$$

$$= \frac{2 \cdot TP \cdot TP}{(TP + FN)(TP + FP)} \cdot \frac{(TP + FN)(TP + FP)}{TP(TP + FN) + TP(TP + FP)} \quad , \quad (4.6)$$

$$= \frac{2 \cdot TP \cdot TP}{TP(TP + FN + TP + FP)} \quad , \quad (4.7)$$

$$= \frac{TP + TP}{TP + TP + FP + FN} \quad . \quad (4.8)$$

Bei Haushaltsgeräten, die sich nicht überwiegend im Ruhezustand befinden, führt die Nutzung des F_1 -Maßes möglicherweise zu falschen Schlüssen über die Güte von Disaggregationsverfahren. Sollten Haushaltsgeräte überwiegend aktiv sein, führt eine triviale Schätzung, die stets positiv klassifiziert, zu einer hohen Anzahl an echt positiven Beispielen. Ein solcher trivialer Schätzer könnte in diesem Fall ein hohes F_1 -Maß erreichen, obwohl das Schätzverfahren nicht als gut bezeichnet werden kann.

Matthews Correlation Coefficient und Balanced Accuracy Die Maße *Matthews Correlation Coefficient* MCC und *Balanced Accuracy* $BAcc$ bleiben im Gegensatz zu *Accuracy* auch bei nicht balancierten Konfusionsmatrizen aussagekräftig und berücksichtigen dabei alle Elemente der Konfusionsmatrix:

$$\text{TNR} = \frac{\text{TN}}{\text{TN} + \text{FP}} \quad , \quad (4.9)$$

$$\text{NPV} = \frac{\text{TN}}{\text{TN} + \text{FN}} \quad , \quad (4.10)$$

$$\text{Informedness} = \text{R} + \text{TNR} - 1 \quad , \quad (4.11)$$

$$\text{Markedness} = \text{P} + \text{NPV} - 1 \quad , \quad (4.12)$$

$$\text{MCC} = \sqrt{\text{Informedness} \cdot \text{Markedness}} \quad , \quad (4.13)$$

$$= \frac{(\text{TP} \cdot \text{TN}) - (\text{FP} \cdot \text{FN})}{\sqrt{(\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN})}} \quad , \quad (4.14)$$

$$\text{BACC} = \frac{1}{2} \left(\frac{\text{TP}}{\text{TP} + \text{FN}} + \frac{\text{TN}}{\text{TN} + \text{FP}} \right) \quad . \quad (4.15)$$

Das MCC-Maß liegt im Intervall $[-1; 1]$ und das BACC-Maß liegt im Intervall $[0; 1]$. Ein trivialer Klassifikator, der stets negativ klassifiziert, würde bei einem überwiegend ausgeschalteten Haushaltsgerät ein MCC-Maß von 0 und ein BACC-Maß von 0,5 erreichen, da $\text{TP} = \text{FP} = 0$ und $\text{TN} \gg 1$.

Anwendung eines Leistungsgrenzwertes Da der Gerätezustand in den meisten Fällen nicht Bestandteil des Datensatzes ist, wird der Aktivitätszustand über den Energieverbrauch geschätzt. Übersteigt der Energieverbrauch einen gerätespezifischen Grenzwert, wird angenommen, dass das Gerät aktiv ist. Der gleiche Grenzwert wird für den geschätzten Energieverbrauch angewandt, sodass am Ende zwei An/Aus-Zeitreihen miteinander verglichen werden. Für die ausgewählten Zielhaushaltsgeräte werden die zuvor in Tabelle 4.9 definierten Grenzwerte $P_{min}^{(on)}$ verwendet. Dabei werden im Gegensatz zur Aktivierungssegmentierung auch kurzzeitige Unterschreitungen des Grenzwertes als Ruhezustand interpretiert.

Der absolute Wert von Klassifikationsmetriken unterschiedlicher Studien ist mit Vorsicht zu genießen, da die Ergebnisse stark von dem Grenzwert abhängen. Mit spezifischen Grenzwerten kann zwischen hohen *Recall* und *Precision*-Werten abgewogen werden.

4.8.2 Regressionsmetriken

Mit der Betrachtung von Disaggregation als Regressionsproblem bietet sich die Verwendung von Regressionsmetriken an, bei denen jede Differenz zwischen geschätzten und gemessenen Lastwert in die Bewertung einfließt. Für die gemessene Gerätelast $L(t)$ im Messintervall $t \in T$ und die entsprechende Schätzung $\hat{L}(t)$ sind der durchschnittliche Vorhersagefehler *Mean Average Error* MAE, der durchschnittliche quadratische Fehler *Mean*

Squared Error MSE, und die Wurzel davon, *Root Mean Squared Error* RMSE, gebräuchliche Fehlermetriken:

$$\text{MAE} := \frac{1}{|T|} \sum_{t \in T} |L(t) - \hat{L}(t)| \quad , \quad (4.16)$$

$$\text{MSE} := \frac{1}{|T|} \sum_{t \in T} (L(t) - \hat{L}(t))^2 \quad , \quad (4.17)$$

$$\text{RMSE} := \sqrt{\frac{1}{|T|} \sum_{t \in T} (L(t) - \hat{L}(t))^2} \quad . \quad (4.18)$$

Wie genau der Gesamtenergieverbrauch über einen längeren Zeitraum geschätzt wird, wird mit den zuvor beschriebenen Metriken nicht abgebildet, da sich der Fehler im ungünstigsten Fall aufsummieren könnte. Für eine möglichst genaue Schätzung des Gesamtenergieverbrauchs über einen längeren Zeitraum sollten sich überschätzte und unterschätzte Energiemengen innerhalb des Zeitraums ausgleichen. Unter Berücksichtigung dieser erwünschten Eigenschaft wird die Metrik *Relative Error in Total Energy* RETE für den beschriebenen Anwendungsfall definiert:

$$\text{RETE} := \frac{\sum_{t \in T} \hat{L}(t) - \sum_{t \in T} L(t)}{\sum_{t \in T} L(t)} \quad . \quad (4.19)$$

Der RETE ist dabei vorzeichenbehaftet: Ein negativer Wert bedeutet, dass zu wenig Energieverbrauch geschätzt wird. Ein positiver Wert entsteht durch eine Überschätzung des bilanziellen Energieverbrauchs.

Wenn ein Durchschnitt der RETE-Metrik über mehrere Gebäude gebildet wird, wird über die absoluten Beträge der Mittelwert gebildet, denn es ist nicht sinnvoll, dass ein unterschätzter Energieverbrauch in einem Gebäude einen überschätzten Energieverbrauch eines anderen Gebäude ausgleicht. Das Ergebnis ist der Anteil des fehlschätzten Energieverbrauchs.

Ein Vorteil von RETE ist, dass es als relative Metrik nicht von dem betrachteten Haushaltsgerät (bzw. dessen Durchschnittsstromverbrauch) abhängt und somit die Metrikergebnisse unterschiedlicher Geräte und Gebäude vergleichbar sind. Dies gilt nicht für die Metriken MAE, MSE und RMSE. Um eine zwischen Geräten und Gebäuden vergleichbare Metrik zu erhalten, die dennoch die genaue Reproduktion des Gerätelastgangs bewertet, werden die im nächsten Abschnitt beschriebenen energiebasierten Metriken vorgestellt.

4.8.3 Energiebasierte Metriken

Eine Mischform von beiden Metrikenfamilien bilden energiebasierte Klassifikationsmetriken. Hierbei wird die Anwendung eines gerätespezifischen Grenzwertes vermieden. Für

energiebasierte Klassifikationsmetriken bleiben die Formeln der Metriken bestehen. Lediglich die Werte der Konfusionsmatrix basieren auf Energiemengen statt auf Zählungen. Dabei ist für die Definition der Echt-Negativ-Menge eine Maximalleistung festzulegen, die weder vom Gerät noch von der Schätzung überschritten wird. Die Maximalleistung L^{max} wird daher basierend auf den Daten berechnet:

$$L^{max} = \max \left\{ \max_{t \in T} L(t), \max_{t \in T} \hat{L}(t) \right\} . \quad (4.20)$$

Zusammengefasst ist bei einer gemessenen Gerätelast $L(t)$ im Messintervall $t \in T$, der entsprechenden Schätzung $\hat{L}(t)$ sowie der Leitungsmaximallast L^{max} die energiebasierte Konfusionsmatrix wie in Tabelle 4.12 definiert.

Tabelle 4.12: Definition der Konfusionsmatrix für energiebasierte Metriken

	Tatsächlich Positiv	Tatsächlich Negativ
Positiv klassifiziert	$TP^E = \sum_{t \in T} \min\{\hat{L}(t), L(t)\}$	$FP^E = \sum_{t \in T} \max\{\hat{L}(t) - L(t), 0\}$
Negativ klassifiziert	$FN^E = \sum_{t \in T} \max\{L(t) - \hat{L}(t), 0\}$	$TN^E = \sum_{t \in T} \min\{L^{max} - \hat{L}(t), L^{max} - L(t)\}$

Die vier Elemente der Matrix lassen sich als Fläche zwischen den gemessenen und geschätzten Lastkurven darstellen. Die Formeln sind durch die eingefärbten Flächen in Abbildung 4.22 widergespiegelt.

4.9 Implementierung

Mit Ausnahme von Kellys Implementierungen⁷⁸ stand für die vorgestellten Ansätze kein Quellcode öffentlich zur Verfügung. Auch Kellys Implementierungen konnten nicht direkt verwendet werden, da die eingesetzten Deep Learning Frameworks Theano⁹ und Lasagne¹⁰ nicht mehr gepflegt werden und als veraltet gelten.

Aus diesem Grund wurden alle Ansätze in PyTorch¹¹ reimplementiert. Für die Datenaufbereitung wurde auf das Non-Intrusive Load Monitoring Toolkit (NILMTK)¹² und Kellys Trainingspaargeneratoren aus NeuralNILM⁸ zurückgegriffen.

⁷https://github.com/JackKelly/neuralnilm_prototype

⁸<https://github.com/JackKelly/neuralnilm>

⁹<http://www.deeplearning.net/software/theano>

¹⁰<https://github.com/Lasagne/Lasagne>

¹¹<https://pytorch.org>

¹²<https://github.com/nilmtk/nilmtk>

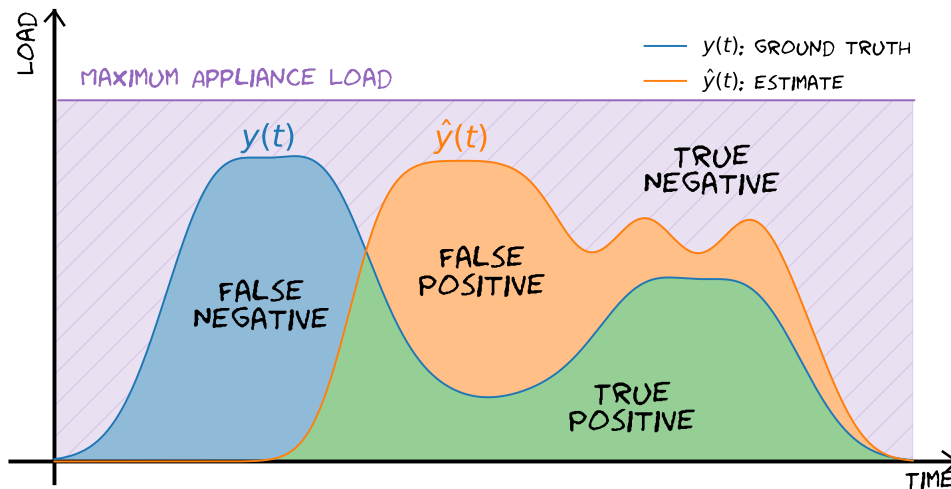


Abbildung 4.22: Grafische Entsprechung der Elemente einer energiebasierten Konfusionsmatrix (eigene Darstellung).

4.9.1 Software-Werkzeuge und Frameworks

Für die Durchführung der Studie kommen die im folgenden beschriebenen Werkzeuge zum Einsatz.

Non-Intrusive Load Monitoring Toolkit (NILMTK)

Das NILMTK¹² ist ein Python-basiertes Framework mit Hilfsfunktionen zum Umgang mit Energiezeitreihen, Datensätzen und Referenzimplementierungen einiger einfacher NILM-Verfahren sowie Evaluationsmetriken. NILMTK verwendet intern die Zeitreihenbibliothek Pandas¹³ für die Repräsentation der Energiedaten. NILMTK wird jedoch unregelmäßig gepflegt und das Projekt hatte wiederholt Schwierigkeiten, mit Änderungen der Zeitreihenbibliothek Pandas Schritt zu halten.

In dieser Studie wurde eine Entwicklungsversion von NILMTK verwendet, die zwischen den Versionen 0.2 und 0.3 liegt. Notwendige Anpassungen werden im folgenden beschrieben.

Datenformat für Haushaltsenergiemessdaten Ein großer Vorteil von NILMTK ist ein standardisiertes Datenformat für Datensätze von Haushaltsenergiemessdaten. Weiterhin stehen bereits einige Konvertierungswerkzeuge für öffentliche NILM-Datensätze zur Verfügung. Das Konvertierungswerkzeug bzw. die Metadaten von REFIT musste

¹³<http://pandas.pydata.org>

jedoch aktualisiert werden, da einige Zeitreihen falschen Geräten zugeordnet waren. Die entsprechenden Änderungen sind in einem Fork auf Github verfügbar¹⁴.

Fehlerbehebung beim Laden der Zeitreihen In der verwendeten Entwicklungsversion von NILMTK wurde beim Laden von Zeitreihen der Parameter für die maximale Anzahl an Messwerten, die bei einem Ausfall des Messsystems aufgefüllt werden sollen, nicht respektiert.

Weiterhin kommt es beim Laden von Zeitreihen, die aus mehreren Segmenten bestehen, vor, dass sich die Segmente überlappen. Das führt zu doppelten Einträgen im Zeitindex und die Weiterverarbeitung dieser Zeitreihen wird von der von NILMTK verwendeten Zeitreihenbibliothek Pandas verweigert. Dies wurde korrigiert, indem die Zeitreihe in dieser Situation gegebenenfalls dedupliziert wird.

Beide Fehler wurden für diese Arbeit behoben und in dem zuvor genannten Fork verfügbar gemacht¹⁴.

NeuralNILM

Die Datenaufbereitung und Trainingspaargenerierung wurde auf Basis von NeuralNILM⁸ von Kelly und Knottenbelt (2015b) entwickelt. Dazu war eine Portierung auf ein aktuelles Deep-Learning-Framework notwendig, da die ursprünglich verwendeten Bibliotheken Theano und Lasagne seit dem September 2017 nicht gepflegt werden.

Bei dem Training von einfachen neuronalen Netzen wurde die Trainingspaargenerierung auf dem Zentralprozessor als limitierender Faktor bezüglich des Durchsatzes identifiziert. Im Rahmen dieser Arbeit wurde daher die Trainingspaargenerierung optimiert und parallelisiert. Dabei wurde die Verwendung der Zeitreihenbibliothek Pandas¹³ möglichst durch NumPy¹⁵ ersetzt. Eine Parallelisierung erfolgte durch die parallele Generierung von Trainingspaaren in mehreren Prozessen¹⁶.

Alle für diese Arbeit durchgeführten Änderungen sind in einem Fork auf Github aufgezeichnet¹⁷. Größere Teile der Bibliothek, die für diese Arbeit nicht verwendet wurden, wurden in diesem Fork (als) entfernt (markiert).

¹⁴<https://github.com/KaibinBao/nilmk>

¹⁵<http://www.numpy.org>

¹⁶Eine Parallelisierung durch mehrere Threads führt bei Python aufgrund des Global Interpreter Locks (GIL) nicht immer zu dem erwarteten Geschwindigkeitsvorteil. Der GIL wird durch die Verteilung der Rechenaufgaben auf mehrere Prozesse umgangen.

¹⁷<https://github.com/KaibinBao/neuralnilm>

Deep-Learning-Framework PyTorch

Die Verwendung eines Deep-Learning-Frameworks abstrahiert das Durchführen von aufwendigen Berechnungen auf Rechenbeschleunigern – in dieser Arbeit Graphikkarten – auf eine nutzerfreundliche Weise. Für diese Arbeit wurden erste Experimente mit dem Framework Keras¹⁸ durchgeführt. Experimente mit Generative Adversarial Networks erforderten jedoch eine höhere Flexibilität des Trainingsprozesses, sodass die in dieser Arbeit beschriebenen Experimente schließlich mit PyTorch¹¹ realisiert worden sind. Die in dieser Studie beschriebenen Experimente wurden mit PyTorch Version 0.4.1 erstellt, welches mit CUDA¹⁹ 9.2 und CuDNN²⁰ 7.1.4 vorkompiliert wurde.

Veröffentlichter Quelltext dieser Arbeit

Auf Basis der zuvor genannten Bibliotheken wurden die Experimente implementiert. Die Aufzeichnung und das Konfigurationsmanagement der Experimente wurde mit Sacred²¹ realisiert. Der finale Quelltext der in dieser Arbeit beschriebenen neuronalen Netzmodelle und Experimente ist ebenfalls auf Github veröffentlicht²².

4.9.2 Rechenbeschleuniger

Für das iterative Training der neuronalen Netzmodelle ist viel Rechenleistung notwendig. Insbesondere Modelle mit vielen Faltungsschichten erfordern bei großer Eingabedaten-dimensionalität Rechenbeschleuniger für Gleitkommaberechnungen, welche heutzutage relativ kostengünstig durch eine Graphikkarte bereitgestellt werden.

Die Verwendung von Graphikprozessoren zur Beschleunigung von Gleitkommaberechnungen wird u. a. durch die Programmierschnittstelle CUDA erleichtert. Für die in dieser Arbeit durchgeführten Experimente wurden insgesamt vier handelsübliche Computer mit je ein bis zwei Nvidia-Graphikkarten mit Graphikprozessoren der Pascal-Architektur und mindestens 8 GB Arbeitsspeicher verwendet. Da für diese Arbeit eine Vielzahl von unabhängigen Experimenten durchgeführt werden musste, verwendet ein Experiment jeweils nur eine Graphikkarte. Zum Verteilen und Verwalten der Experimenteausführung wurde das Schedulingssystem Slurm²³ eingerichtet.

¹⁸<https://keras.io>

¹⁹Compute Unified Device Architecture (CUDA) ist eine Programmierschnittstelle zur Verwendung von Graphikprozessoren für numerische Aufgaben.

²⁰CUDA Deep Neural Network library (CuDNN) ist eine optimierte Bibliothek für häufig genutzte Rechenoperationen, die im Rahmen von tiefen neuronalen Netzen verwendet werden.

²¹<https://github.com/IDSIA/sacred>

²²<https://github.com/KaibinBao/neuralnilm-pytorch>

²³<https://slurm.schedmd.com>

Kapitel 5

Analyse bestehender Deep-Learning-Ansätze

In diesem Kapitel werden die Schätzungen bekannter Verfahren qualitativ begutachtet, um Probleme und mögliche Ansatzpunkte für die Verbesserungen der Disaggregationsqualität zu identifizieren.

Dazu werden die in Abschnitt 3.11 identifizierten Stand-der-Technik-Ansätze mit demselben in Kapitel 4 beschriebenen Experimentenaufbau trainiert. Die spezifische Parametrierung der Ansätze ist in Abschnitt 5.1 aufgeführt. In Abschnitt 5.2 werden die Ergebnisse der Ansätze diskutiert und mögliche Problemursachen anhand der disaggregierten Lastgänge identifiziert. Schließlich werden in Abschnitt 5.3 die Erkenntnisse zusammengefasst und umsetzbare Forschungsansätze formuliert.

5.1 Evaluierte Ansätze

Für die Disaggregationsexperimente werden die bestehenden Ansätze zum Training wie folgt parametrisiert.

5.1.1 DAE: Denoising Autoencoder

Bei dem Denoising Autoencoder (DAE) von Kelly und Knottenbelt (2015a) (siehe Abschnitt 3.6.3) orientieren sich die Hyperparameter an dem Originalansatz:

Tabelle 5.1: Die für das Verfahren Denoising Autoencoder verwendeten Hyperparameter. Zu den Parametern zählen die Sequenzlänge T_S und die Maximale Geräteleistung P_{max} je betrachtetes Haushaltsgerät.

Haushaltsgerät	T_S / Min	P_{max} / W
Waschmaschine	102,4	2500
Wasserkocher	12,8	3100
Mikrowelle	28,8	3000
Geschirrspüler	153,6	2500
Kühlgeräte	51,2	300
Wäschetrockner	102,4	2500
TV/ Unterhaltungselektronik	28,8	1200

Ein- und Ausgabeformat Für den DAE werden die Lastgänge in die in Tabelle 5.1 aufgeführte geräteabhängige Sequenzlänge aufgeteilt. Das Fenster zum Ausschneiden der Sequenzen wird nach jedem Schritt um 16 Messintervalle, also 1,6 Minuten weitergeschoben.

Teilergebniszusammenführung Die sich überlappenden Sequenzen werden voneinander unabhängig disaggregiert. Für ein Messintervall ergeben sich dadurch mehrere Schätzungen, die durch Mittelwertbildung zum Endergebnis zusammengeführt werden.

Normalisierung des Gerätelastgangs Die Gerätelastgänge werden wie bei dem Originalansatz durch einen gerätespezifischen P_{max} (siehe Tabelle 5.1) dividiert, bevor sie als Zielvektoren verwendet werden.

Normalisierung des Haushaltlastgangs Die Haushaltlastgänge werden durch die Standardabweichung aller Haushaltlastgangszeitreihen des Trainingsdatensatzes dividiert. Im Anschluss wird von jedem Haushaltlastgangsegment dessen individueller Mittelwert abgezogen. Als Standardabweichung wurde ein Wert von 700 W verwendet¹.

Modelloptimierer Exakt wie bei dem Originalverfahren werden die Modellparameter mit SGD und Nesterov-Momentum optimiert. Die Mini-Batch-Größe entspricht 64 Trainingspaare. Insgesamt besteht das Training aus 100'000 Iterationen. Die Lernrate ist initial 10^{-1} , wird nach 50'000 Iterationen zunächst auf 10^{-2} reduziert und schließlich nach weiteren 25'000 Iterationen auf 10^{-3} verringert. Experimente mit dem Optimierer ADAM führten zu schlechteren Ergebnissen.

¹Bei der Normalisierung spielt der exakte verwendete Wert keine Rolle. Vielmehr stellt die Normalisierung sicher, dass die verwendeten Werte größenordnungsmäßig in einem Bereich liegen, der zu keinem Unter- oder Überlauf während der Berechnungen führt.

5.1.2 S2P / S2S: Sequence-to-Point und Sequence-to-Sequence

Die Parametrierung von Sequence-to-Point (S2P) und Sequence-to-Sequence (S2S) von Zhang et al. (2016) (siehe Abschnitt 3.6.5) richtet sich nach dem von den Autoren bereitgestellten Quellcode², da in der Veröffentlichung von Zhang et al. (2016) keine Details bezüglich des Modelltrainings beschrieben wurden.

Tabelle 5.2: Die für die Verfahren Sequence-to-Point bzw. Sequence-to-Sequence verwendeten Hyperparameter. Unterschiede zu den Parametern von Zhang et al. (2016) sind fett gedruckt. Aufgeführt ist die Sequenzlänge T_S , die Mindestlast bei Geräteaktivität $P_{min}^{(on)}$, der ungefähre Mittelwert der Leistungsaufnahme bei Geräteaktivität P_μ sowie die Standardabweichung der Leistungsaufnahme bei Geräteaktivität P_σ je betrachtetes Haushaltsgerät.

Haushaltsgerät	T_S / Min	$P_{min}^{(on)}$ / W	P_μ / W	P_σ / W
Waschmaschine	59,9	20	400	700
Wasserkocher	59,9	400	700	1000
Mikrowelle	59,9	200	500	800
Geschirrspüler	59,9	15	200	1000
Kühlschrank	59,9	30	200	400
Wäschetrockner	59,9	20	700	1000
TV/ Unterhaltungselektronik	59,9	15	200	400

Ein- und Ausgabeformat Die Ein- und Ausgabevektoren haben in den durchgeführten Experimenten das gleiche Format: Für eine Lastsequenz mit 599 Messwerten (unabhängig von dem betrachteten Gerät) werden je nach Verfahren 599 Werte (S2S) bzw. ein Wert (S2P) ausgegeben. Bei beiden Ansätzen wird das Fenster zum Ausschneiden der Sequenzen um ein Messintervall, also 6 Sekunden, weitergeschoben.

Teilergebniszusammenführung Die sich überlappenden Sequenzen werden voneinander unabhängig disaggregiert. Für S2S ergeben sich je Messintervall mehrere Schätzungen, die durch Mittelwertbildung zum Endergebnis zusammengeführt werden. Bei S2P ist das Ergebnis für jedes Messintervall eindeutig, sodass die einzelnen Teilergebnisse zum Endergebnis aneinandergereiht werden.

Normalisierung der Lastgänge Sowohl der Haushaltslastgang als auch der Gerätelastgang werden bei den Verfahren von Zhang et al. mit dem größenordnungsmäßig genauen Wert der Standardabweichung P_σ und des Mittelwerts P_μ des Zielgerätes normalisiert. In Tabelle 5.2 sind die verwendeten Werte für diese Studie gezeigt. Änderungen bzw. Ergänzungen zum Originalverfahren sind fett gedruckt.

²<https://github.com/MingjunZhong/NeuralNetNilm>

Modelloptimierer Im Quellcode des Ansatzes wird eine Mini-Batch-Größe von 1000 Trainingsbeispielen und eine Trainingsdauer von 50 Epochen verwendet. Als Modellparameteroptimierer wird ADAM mit einer Lernrate von 10^{-3} verwendet. In Rahmen dieser Studie wird für S2S und S2P ebenfalls der ADAM-Optimierer mit einer initialen Lernrate von 10^{-3} verwendet. Die Mini-Batch-Größe umfasst jedoch nur 64 Trainingspaare. Da es bei den verwendeten Trainingspaargeneratoren keine Definition von Epoche gibt, werden feste 100'000 Trainingsiterationen durchgeführt. Die Lernrate wird nach 50'000 Iterationen zunächst auf 10^{-4} reduziert und schließlich nach weiteren 25'000 Iterationen auf 10^{-5} verringert.

5.1.3 SGN: Subtask Gated Networks

Die Parametrisierung des Verfahrens Subtask Gated Networks (SGN) orientiert sich an der Veröffentlichung von Shin et al. (2018) (siehe Abschnitt 3.6.5).

In der Veröffentlichung werden mehrere Varianten von Subtask Gated Networks vorgestellt. Insbesondere wird bei zwei Varianten die Leistungsaufnahme von Geräten im Ruhezustand gesondert betrachtet, indem das Gesamtnetzwerk bei geschlossener Schranke eine selbsterlernte Standby-Leistung ausgibt, anstatt 0 W. Da diese „SGN-sp“ genannte Variante in der Veröffentlichung die niedrigsten durchschnittlichen Fehlerwerte erzielte, wird in dieser Studie diese Variante evaluiert.

Ein- und Ausgabeformat Shin et al. (2018) verwenden ein 43,2 Minuten langes Eingabefenster und ein 3,2 Minuten langes Ausgabefenster bei den Experimenten mit dem UK-DALE-Datensatz. Diese Vorgaben werden für diese Studie übernommen und für alle Datensätze angewandt.

Teilergebniszusammenführung In der Veröffentlichung wird die Schrittweite für das Ausschneiden der Lastsequenzen während der Disaggregation nicht beschrieben. Es wird angenommen, dass sich das Verfahren an (Zhang et al., 2016) orientiert und die Schrittweite von einem Messwert verwendet. Teilergebnisse werden durch Mittelwertbildung zum Endergebnis zusammengesetzt.

Normalisierung der Lastgänge Dieser Aspekt wird in dem Beitrag nicht ausgeführt. Es wird angenommen, dass die Normalisierung sich an Zhang et al. orientiert. Diese Studie verwendet zur Normalisierung des Haushalts- und Gerätelastgangs die in Tabelle 5.2 definierten, zielgerätspezifischen Werte.

Modelloptimierer Die Experimente von Shin et al. (2018) trainieren ihre Modelle mit dem Optimierer ADAM bei einer Lernrate von 10^{-4} und einer Mini-Batch-Größe von 16 Trainingspaaren. Die Anzahl an Epochen bzw. Iterationen ist nicht angegeben. Im Rahmen dieser Studie wird für das SGN ebenfalls mit dem ADAM-Optimierer bei einer

initialen Lernrate von 10^{-4} trainiert. Die Mini-Batch-Größe umfasst jedoch 64 Trainingspaare. Es werden 100'000 Trainingsiterationen durchgeführt. Die Lernrate wird nach 50'000 Iterationen zunächst auf 10^{-5} reduziert und schließlich nach weiteren 25'000 Iterationen auf 10^{-6} verringert.

5.1.4 GLU: Gated Linear Units

Der Ansatz von Chen, Wang et al., 2018 mit Gated Linear Units (GLU) (siehe Abschnitt 3.6.6) wird mit folgenden Parametern trainiert:

Ein- und Ausgabeformat Für GLU wird, wie bei dem Originalansatz, ein 80 Minuten langes Eingabefenster und ein 10 Minuten langes Ausgabefenster verwendet. Das Fenster zum Ausschneiden der Sequenzen wird nach jedem Schritt um 5 Messintervalle, also 30 Sekunden, weitergeschoben.

Teilergebniszusammenführung Die Teilergebnisse werden durch Mittelwertbildung zum Endergebnis zusammengeführt.

Normalisierung der Lastgänge Zur Normalisierung der Lastgänge dividieren Chen, Wang et al., 2018 die Gerätelastgänge durch gerätespezifische Werte. In dieser Studie werden die Gerätelastgänge durch die Standardabweichung P_σ in Tabelle 5.2 dividiert. Der Haushaltslastgang wird, wie bei Chen, Wang et al., 2018, durch 1000W dividiert.

Modelloptimierer Die Arbeit von Chen, Wang et al. (2018) erwähnt lediglich die Verwendung des ADAM-Optimierers. Im Rahmen dieser Studie wird für das GLU mit dem ADAM-Optimierer bei einer initialen Lernrate von 10^{-3} trainiert. Die Mini-Batch-Größe umfasst jedoch 64 Trainingspaare. Es werden 100'000 Trainingsiterationen durchgeführt. Die Lernrate wird nach 50'000 Iterationen zunächst auf 10^{-4} reduziert und schließlich nach weiteren 25'000 Iterationen auf 10^{-5} verringert.

5.1.5 DFCN: Dilated Fully Convolutional Network

Der Ansatz Dilated Fully Convolutional Network (DFCN) von Brewitt und Goddard (2018) (siehe Abschnitt 3.6.5) wird mit folgenden Parametern trainiert:

Ein- und Ausgabeformat Nach dem Vorbild von Brewitt und Goddard (2018) wird für das DFCN ein 410,5 Minuten langes Eingabefenster und ein 205,3 Minuten langes Ausgabefenster verwendet. Wie bei dem Originalansatz hängt jeder Ausgangswert von genau 2053 Eingangswerten ab. Das Fenster zum Ausschneiden der Sequenzen wird nach jedem Schritt um 2053 Messintervalle weitergeschoben.

Tabelle 5.3: Die für das Verfahren Dilated Fully Convolutional Network verwendeten mittleren Lastwerte von Geräteaktivierungen.

Haushaltsgerät	$P_{\mu}^{(on)}$ / W
Waschmaschine	576,84
Wasserkocher	2127,95
Mikrowelle	738,03
Geschirrspüler	1059,89
Kühlschrank	109,11
Wäschetrockner	1211,02
TV/	
Unterhaltungselektronik	85,94

Teilergebniszusammenführung Die Teilergebnisse dieses Ansatzes überlappen sich nicht und werden lediglich aneinandergereiht.

Normalisierung der Lastgänge Zur Normalisierung der Gerätelastgänge werden die Werte durch die in Tabelle 5.3 aufgeführten gerätespezifischen Lastwerte geteilt. Die Werte entsprechen den mittleren Lasten während Geräteaktivierungen und wurden aus den Trainingsdaten berechnet. Die Normierung des Haushaltslastgangs erfolgt durch eine Division durch die 700 W.

Modelloptimierer Die Arbeit von Brewitt und Goddard (2018) verwendet den ADAM-Optimierer mit einer initialen Lernrate von 10^{-3} und einer Mini-Batch-Größe von 256 Trainingspaaren. Da im Rahmen dieser Studie das Training mit der Lernrate 10^{-3} nicht immer konvergierte, ist in den Experimenten die initiale Lernrate auf 10^{-4} reduziert. Auch ist die Mini-Batch-Größe auf 64 Trainingspaare reduziert. Wie bei den anderen Ansätzen werden 100'000 Trainingsiterationen durchgeführt. Die Lernrate wird nach 50'000 Iterationen zunächst auf 10^{-5} reduziert und schließlich nach weiteren 25'000 Iterationen auf 10^{-6} verringert.

5.2 Bewertung der Disaggregation

Für jede Geräteklasse werden im Folgenden die durch die Ansätze durchgeführten Disaggregationen zunächst anhand von drei Fehlermetriken beurteilt und im Anschluss Probleme bei der Disaggregation mit Hilfe von Beispielabschnitten identifiziert.

Zur Übersicht werden die Metriken energiebasierter Matthews Correlation Coefficient (eMCC), Mittlerer absoluter Fehler (MAE) und relativer Fehler bei der Gesamtenergieverbrauchsschätzung (RETE) betrachtet (siehe Abschnitt 4.8).

Dabei dient der eMCC als allgemeiner Indikator für den Erfolg der Disaggregation, da die Metrik direkte Vergleiche zwischen unterschiedlichen Geräten zulässt, selbst wenn sich die Geräte im durchschnittlichen Energieverbrauch unterscheiden. Die Vergleichbarkeit wird dadurch erreicht, dass für den eMCC die geschätzten und tatsächlichen Energiemengen normiert werden und dadurch der Metrikwert in das Intervall $[-1; 1]$ fällt. Ein perfekter Disaggregationsansatz würde einen eMCC-Wert von 1.0 erreichen während ein zufällig geschätzter Lastgang in der Regel mit einem Metrikwert nahe 0.0 bewertet wird. Negative Werte sind möglich, falls häufig genau dann eine Gerätelast geschätzt wird, wenn das Gerät inaktiv ist. Jedoch erzielen alle evaluierten Ansätze positive Metrikwerte.

Die Metriken MAE und RETE liefern erste Hinweise, welche systematischen Fehler die Ansätze begehen: Der MAE zeigt an, wie groß im Schnitt der Fehlerbetrag bei der Lastschätzung für individuelle Messintervalle ist. Anhand des RETE, der den Anteil der fehlerhaft geschätzten Energiemenge gegenüber der tatsächlich verbrauchten Energiemenge des Gerätes angibt, lässt sich eine systematische Unterschätzung (negativer Wert) oder Überschätzung (positiver Wert) des Gesamtenergieverbrauchs erkennen.

Ein perfekter Disaggregationsansatz würde einen eMCC-Wert von 1.0, einen MAE von 0 W sowie RAETE von 0 % erreichen.

5.2.1 Waschmaschinen und Wäschetrockner

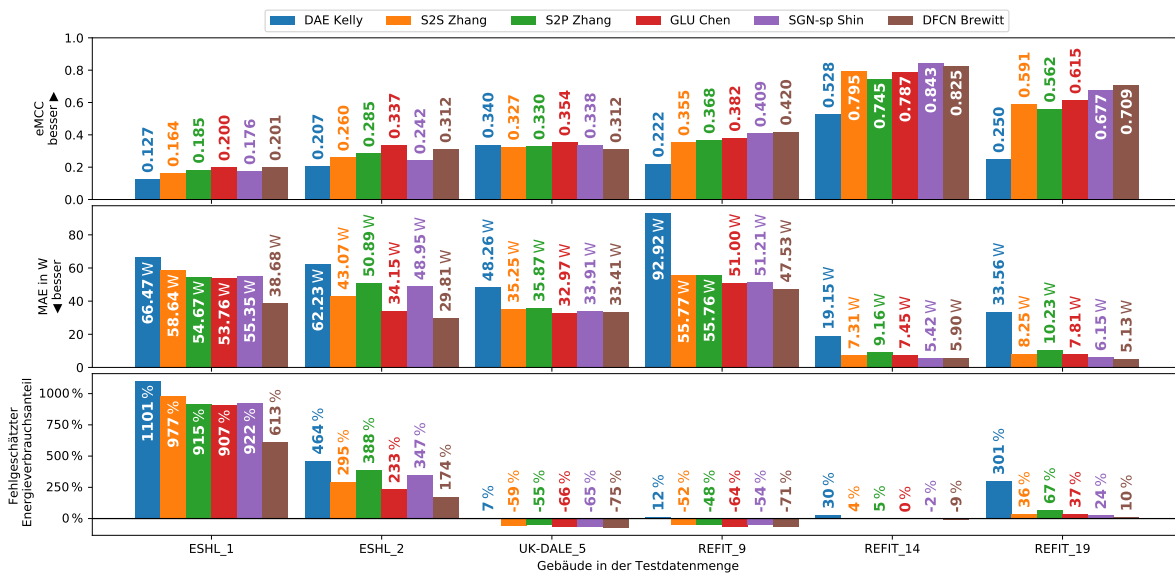


Abbildung 5.1: Matthews Correlation Coefficient (eMCC), Mittlerer absoluter Fehler (MAE) und Fehlgeschätzter Anteil am Energieverbrauch (RETE) bekannter Disaggregationsansätze bei Waschmaschinen.

In Abbildung 5.1 sind die Fehlermetriken für die Disaggregation der Waschmaschinenlast in den Testgebäuden dargestellt. Bei allen Metriken ist zu erkennen, dass die Ergebnisse für die jeweiligen Testgebäude ähnlich sind: Die Ergebnisse verschiedener Testgebäude unterscheiden sich stärker als die Ergebnisse verschiedener Ansätze. Der DAE ist ein Negativ-Ausreißer und ist in allen Metriken mit wenigen Ausnahmen am schlechtesten.

In den Testgebäuden REFIT 14 und 19 erreichen die Ansätze, ausgenommen DAE, mit einem eMCC zwischen 0.56 und 0.84 eine gute Erkennung der Waschmaschinenlast. Hierbei ist auch der MAE zwischen 5.42 W und 10.23 W gering und die Gesamtenergiemenge wird im Schnitt um etwa 20% überschätzt.

Anhand des negativen fehlgeschätzten Energieverbrauchsanteils in den Gebäuden REFIT 9 und UK-DALE 5 lässt sich erkennen, dass die Ansätze etwa die Hälfte des tatsächlichen Energieverbrauchs nicht der Waschmaschine zugeordnet haben. Dadurch liegt, ausgenommen DAE, der mittlere Fehler etwa bei 50 bzw. 35 W und der eMCC liegt zwischen 0.314 und 0.419 im Mittelfeld der Ergebnisse.

DAE schätzt den Gesamtenergieverbrauch mit 7% und 12% zwar relativ genau, dies scheint jedoch zufällig zu sein, da der hohe MAE eine starke Fehlschätzung der Einzellastwerte anzeigt.

In den ESHL-Testdaten wird der Energieverbrauch der Waschmaschine deutlich überschätzt. Der Datensatz ESHL 1 schneidet dabei schlechter ab als ESHL 2.

Gute Disaggregation in REFIT 14 und 19

Abbildung 5.2 stellt beispielhaft die Schätzungen der jeweiligen Ansätze für einen Zeitabschnitt im REFIT 14 dar. In dem Zeitraum ist die Waschmaschine zweimal aktiv, was von allen Ansätzen erkannt wird.

Während der ersten Aktivierung sind wenige andere Geräte im Haushalt parallel im Betrieb und alle Ansätze rechnen die Last größtenteils der Waschmaschine an.

Bei der zweiten Aktivierung wird der Waschmaschinenlastgang durch die Last anderer Geräte überlagert. Dadurch rechnen die Ansätze den Lastblock am Anfang der zweiten Aktivierung (währenddessen die Waschmaschine das Wasser aufheizt) nicht vollständig dem Zielgerät an. Nach der zweiten Aktivierung (ab 21 Uhr) weist der DAE-Ansatz der Waschmaschine fälschlicherweise Energieverbrauch zu, was die anderen Ansätze vermeiden.

Alle Ansätze haben Schwierigkeiten, einen echten Waschmaschinenlastgang zu reproduzieren und erzeugen Ausreißer und Rampen in dem geschätzten Lastgang, die eine echte Waschmaschine nicht produzieren würde.

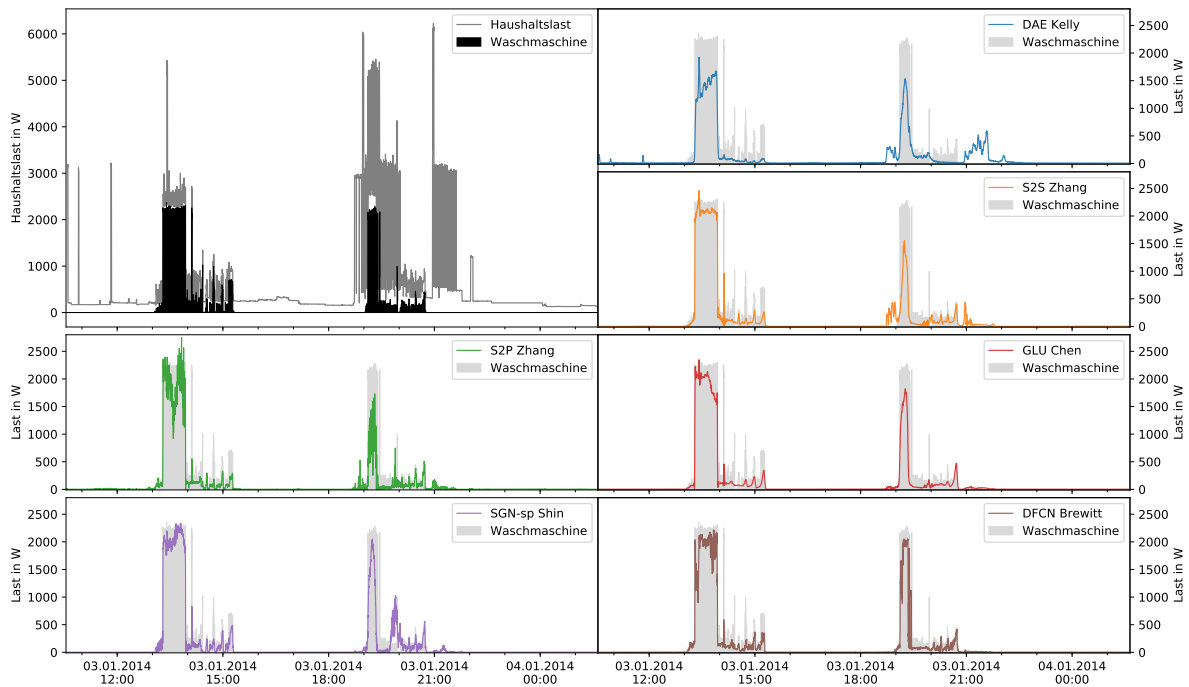


Abbildung 5.2: Disaggregierter Waschmaschinenlastgangsausschnitt aus REFIT 14.

Unterschätzter Energieverbrauch in UK-DALE 5 und REFIT 9

Abbildung 5.3 stellt die Schätzungen der jeweiligen Ansätze für einen ausgewählten Ausschnitt im REFIT 9 dar. Zwei Aktivierungen sind dargestellt: Die erste Aktivierung scheint ein Waschprogramm zu sein. Die zweite Aktivierung verbraucht konstant etwa 250 W, ist jedoch kein Waschprogramm, da weder das Heizen noch die beim Schleudern typische Fluktuation im Energieverbrauch sichtbar sind. Es handelt sich hierbei vermutlich um ein Trockenprogramm. Der letzte Abschnitt der ersten Aktivierung sieht ähnlich zur zweiten Aktivierung aus.

Alle Ansätze folgen dem Lastgang der Waschmaschine bis kurz nach 10 Uhr. Der letzte Abschnitt der ersten Aktivierung wird nicht als Waschmaschinenlast anerkannt. Auch erkennen alle Ansätze die zweite Aktivierung nicht.

Somit wird das Trockenprogramm, welches individuell oder im Anschluss eines Waschprogramms läuft, von allen Ansätzen nicht erkannt und dies führt insgesamt zu einem unterschätzten Energieverbrauch. Dies ist wohl auf die geringe Trainingsdatenmenge für das Trockenprogramm von Waschtrocknern zurückzuführen: In den verwendeten Datensätzen sind nur 4 Waschtrockner vorhanden (siehe Tabelle 4.1), von denen 2 sich in Testgebäuden befinden und nicht zum Training verwendet werden.

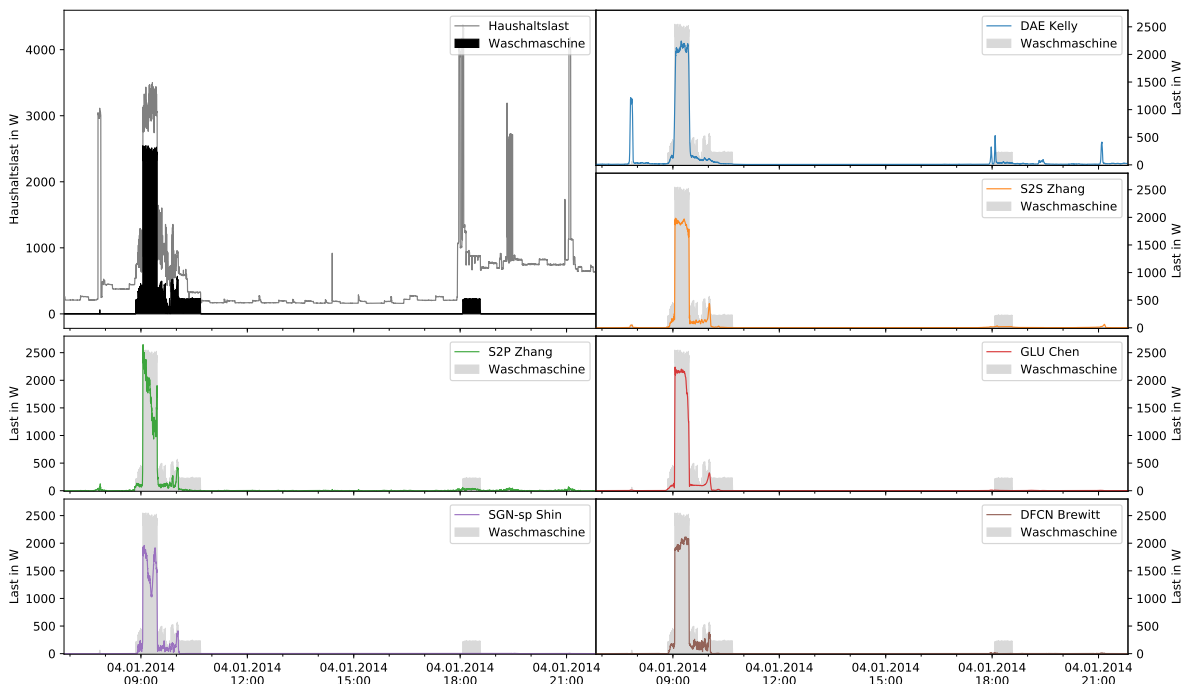


Abbildung 5.3: Disaggregierter Waschmaschinenlastgangsausschnitt aus REFIT 9.

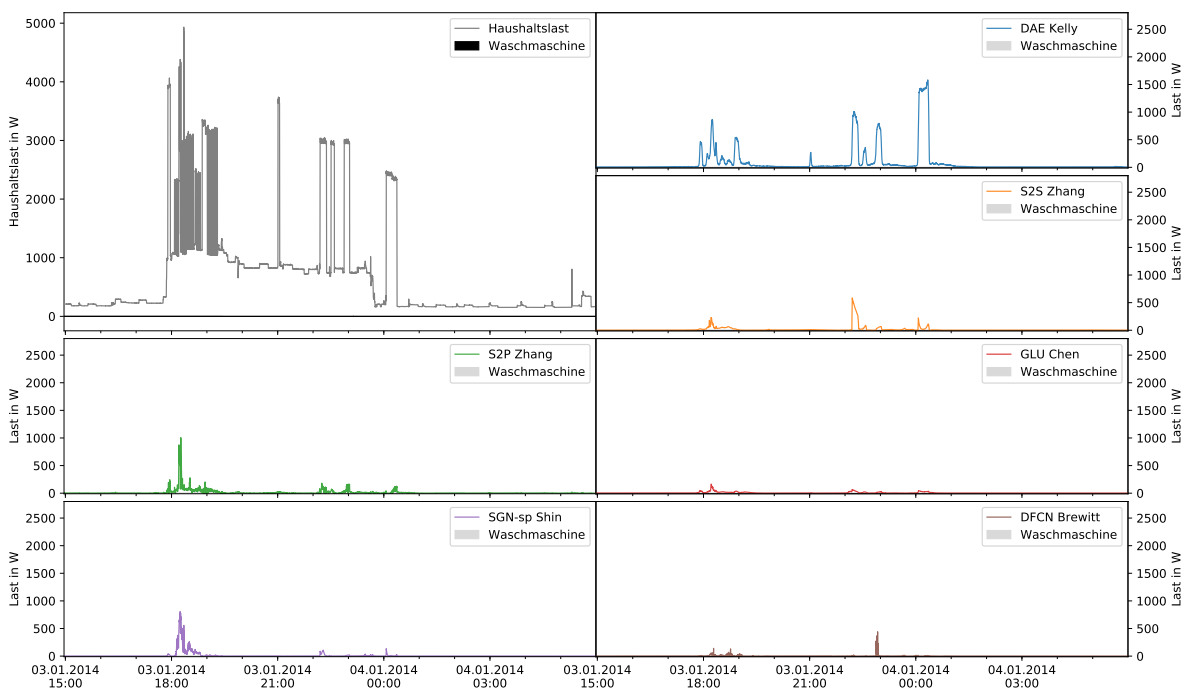


Abbildung 5.4: Ein weiterer disaggregierter Waschmaschinenlastgangsausschnitt aus REFIT 9 mit hoher Gebäudelast.

Der fehlerhaft geschätzte Energiemengenanteil ist bei dem Ansatz DAE geringer als bei den anderen Ansätzen. Wie es Abbildung 5.4 beispielhaft zeigt, liegt dies jedoch daran, dass der DAE-Ansatz größere Energiemengen der Waschmaschine zu den Zeiten zuschreibt, zu denen das Gerät nicht aktiv ist. Die genauere Energieverbrauchsschätzung wird also durch den zufälligen Ausgleich von zwei Fehlern begründet.

Überschätzter Energieverbrauch in ESHL

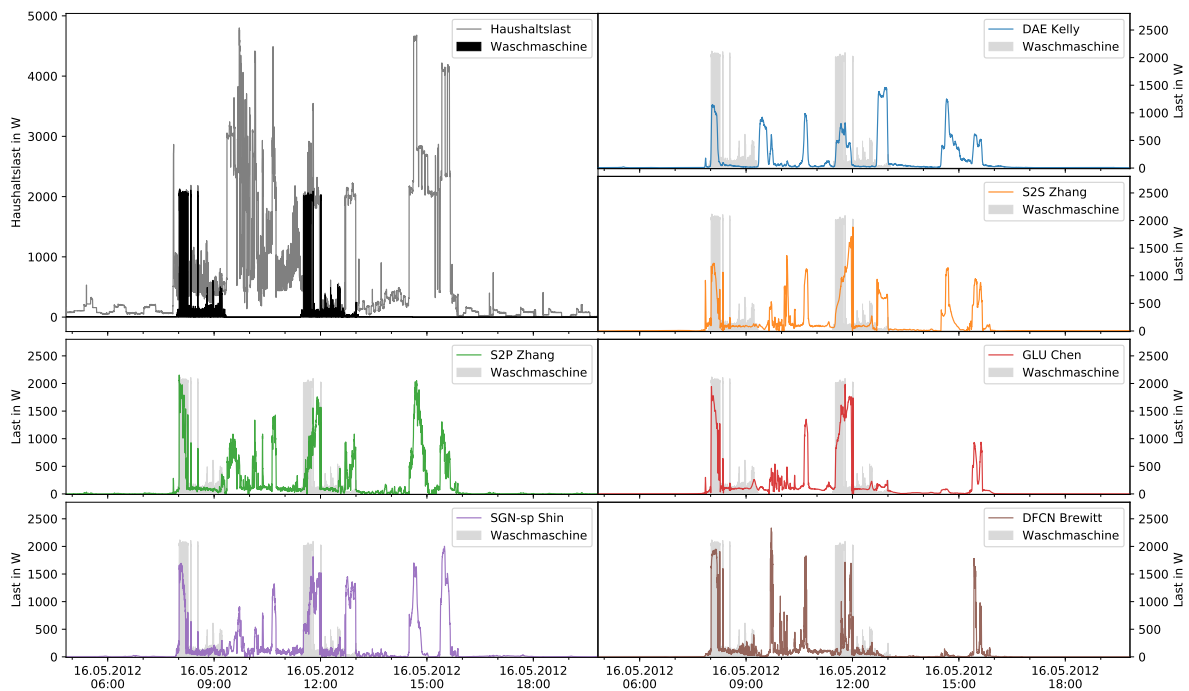


Abbildung 5.5: Disaggregierter Waschmaschinenlastgangsausschnitt aus ESHL 1.

Abbildung 5.5 stellt einen disaggregierten Ausschnitt des Datensatzes ESHL 1 dar, in dem die Waschmaschine zweimal benutzt wurde. In dieser Zeit sind auch andere Haushaltsgeräte parallel aktiv und der Waschmaschinenlastgang ist im Haushaltslastgang schwer erkennbar.

Alle Ansätze haben Schwierigkeiten, insbesondere die zweite Waschmaschinenaktivierung zu erkennen, da während dieser Zeit die Spülmaschine parallel aktiv ist. Außerhalb der beiden Aktivierungen wird der Waschmaschine Energieverbrauch zugeordnet, obwohl das Gerät in der Zeit nicht aktiv ist.

Auch im Datensatz ESHL 2 wird in den Zeitabschnitten mit hoher Haushaltslast häufig fälschlicherweise Waschmaschinenaktivität erkannt, obwohl die Waschmaschine nicht aktiv ist

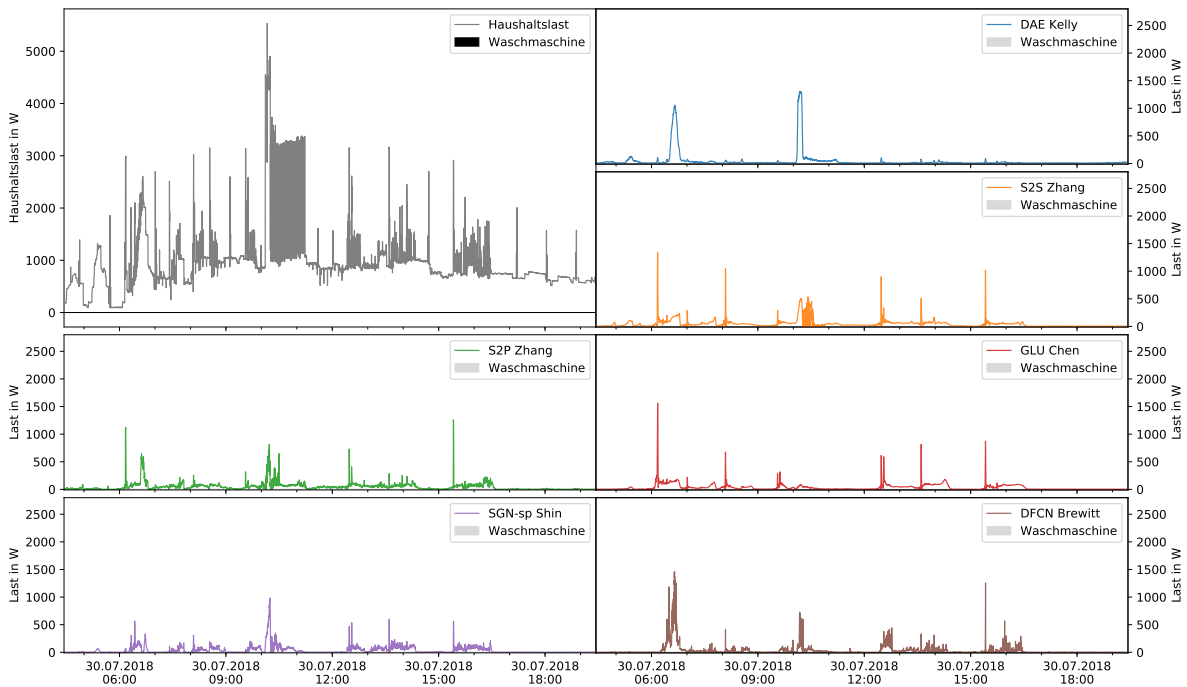


Abbildung 5.6: Disaggregierter Waschmaschinenlastgangsausschnitt aus ESHL 2.

(so wie in Abbildung 5.6 dargestellt). Dies resultiert in den überschätzten Energieverbrauch in den ESHL-Daten.

5.2.2 Kühlgeräte

Die Metriken zur Disaggregation von Kühlgeräten sind in Abbildung 5.7 dargestellt. Grundsätzlich werden alle Kühlgeräte in allen Gebäuden mit einer eMCC-Metrik über 0.53 gut erkannt und die fehlschätzte Energieverbrauchsmenge liegt im gesamten Durchschnitt bei 32,9 %. Lediglich der fehlgeschätzte Energieverbrauch über 134 % und eine relativ hohe MAE einiger Ansätze deuten auf systematische Probleme im Datensatz UK-DALE 5 hin. Die höchsten eMCC-Werte und die niedrigsten MAE-Werte werden am häufigsten durch den Ansatz DFCN erreicht.

In den Gebäuden UK-DALE 5, REFIT 9 und REFIT 14, die über nur ein Kühlgerät verfügen (siehe Tabelle 4.1), wird der Gesamtenergieverbrauch überschätzt. Im ESHL und in REFIT 19, die jeweils einen Kühlschrank und einen Gefrierschrank enthalten, wird der Gesamtenergieverbrauch unterschätzt (mit einem negativen fehlgeschätzten Energieverbrauchsanteil).

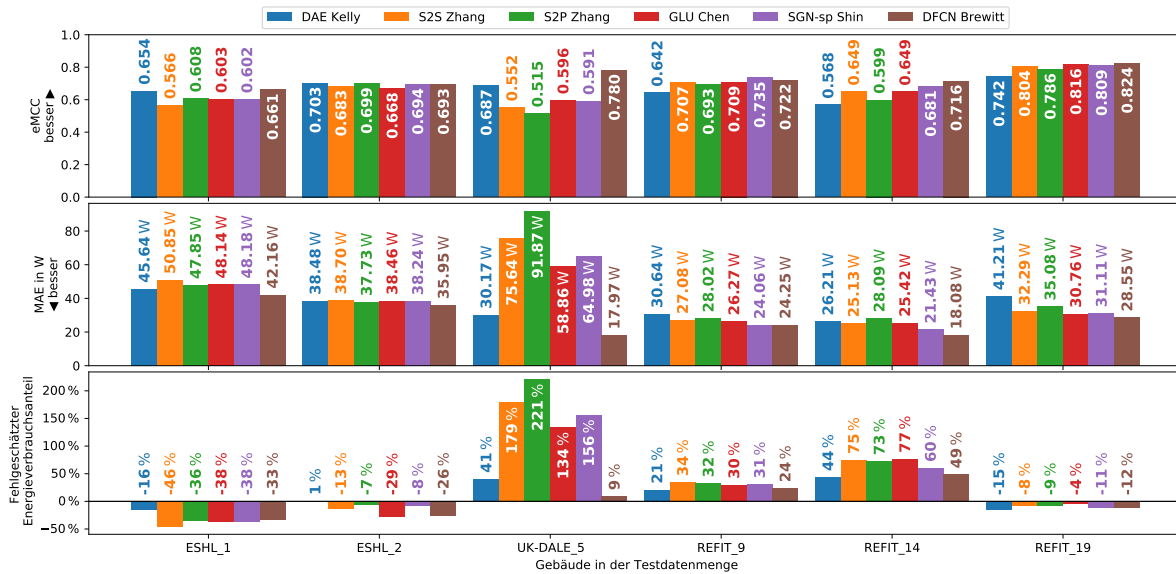


Abbildung 5.7: Matthews Correlation Coefficient (eMCC), Mittlerer absoluter Fehler (MAE) und Fehlgeschätzter Anteil am Energieverbrauch (RETE) bekannter Disaggregationsansätze bei Kühlgeräten.

Fehlgeschätzte Kühlgeräteleast bei hoher Geräteaktivität im Haushalt

Elektrische Geräte werden in der Nacht und oft auch zur Mittagszeit nicht benutzt, wenn die Bewohner außer Haus sind. In den Morgenstunden und Abends sind oft mehrere Geräte parallel aktiv.

In Abbildung 5.8, welches einen Zeitabschnitt aus REFIT 9 zeigt, lässt sich das an der niedrigen Last in der Nacht bis etwa 6 Uhr und ab 10 Uhr erkennen. Zu diesen Zeiten werden Laständerungen hauptsächlich durch die Kühlgeräte verursacht. Zwischen 6 und 10 Uhr werden viele Geräte verwendet und die Kühlgeräte lassen sich schwer im hohen Haushaltslastgang erkennen.

Alle Ansätze disaggregieren in der Regel den Kühlgeräteleastgang zu Zeiten mit geringer Geräteaktivität im Haushalt gut. Zu Zeiten mit höherer Geräteaktivität im Haushalt verschlechtert sich die Erkennung der Kühlgeräteleast und dies führt zu einer Überschätzung (wie zwischen 7 und 10 Uhr in Abbildung 5.8) oder Unterschätzung (wie zwischen 5 und 11 Uhr in Abbildung 5.10) des Energieverbrauchs.

Überschätzter Energieverbrauch in UK-DALE 5

In Abbildung 5.9 ist ein Lastgangsausschnitt des Gebäudes UK-DALE 5 dargestellt. Der Kühlschrank wechselt regelmäßig zwischen Aktivität und Standby mit variablen Verweildauern

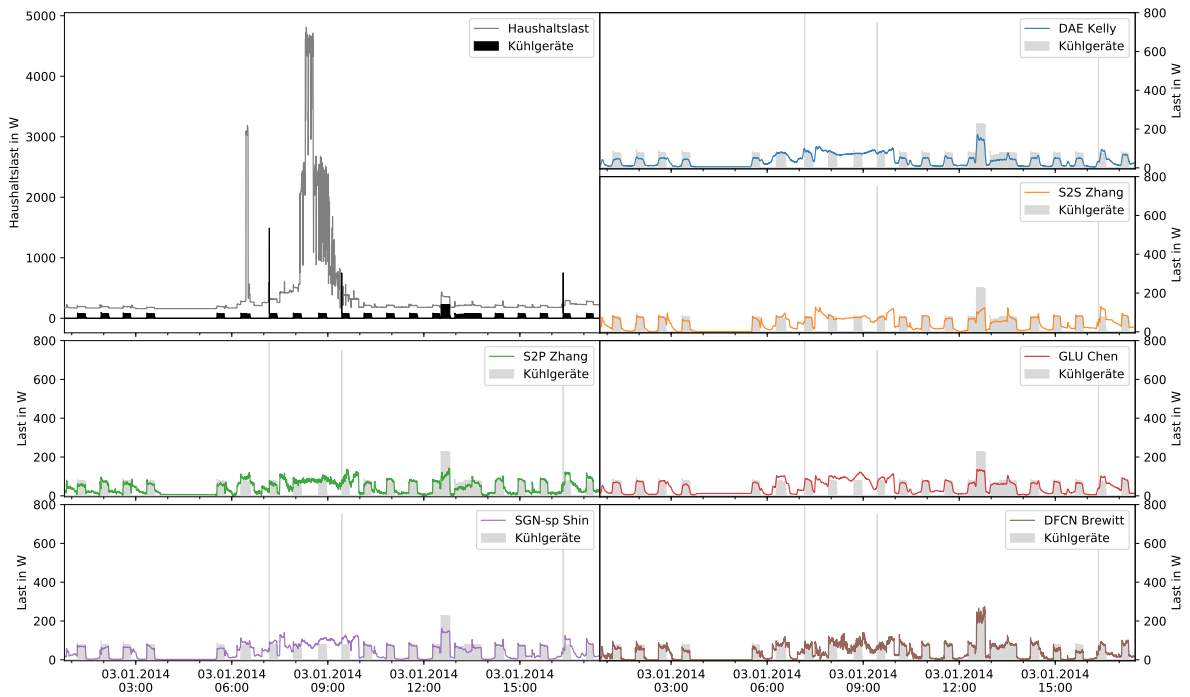


Abbildung 5.8: Disaggregierter Kühlgeräteleistungsausschnitt aus REFIT 9.

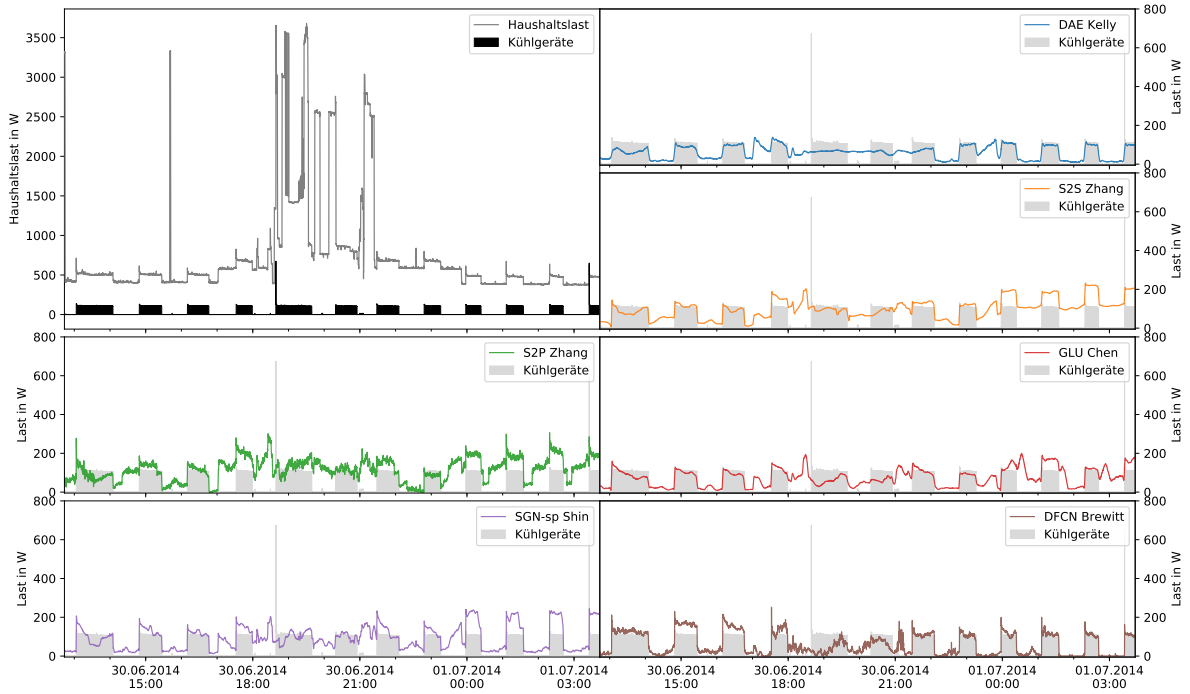


Abbildung 5.9: Disaggregierter Kühlgeräteleistungsausschnitt aus UK-DALE 5.

in den jeweiligen Zuständen. Im Haushalt sind zwischen 13 und 17 Uhr sowie ab 0 Uhr wenige andere Geräte aktiv, sodass hauptsächlich nur die Kühlgeräte für Laständerungen verantwortlich sind.

Im Zeitraum zwischen 17 und 0 Uhr sind weitere Geräte aktiv und sorgen zeitweise für eine hohe Gebäudelast. Die hohe Last erschwert die Erkennung des Kühlgerätelastgangs und alle Ansätze geben ungenaue Schätzungen aus.

Wie in der Abbildung ab 0 Uhr zu erkennen ist, überschätzen die Ansätze S2S, S2P, GLU und SGN jedoch auch die Kühlgerätelast in Zeiten niedriger Last. Daher überschätzen diese Ansätze den gesamten Energieverbrauch der Kühlgeräte zwischen 134 % und 221 %.

Mehrere Kühlgeräte in REFIT 19 und ESHL

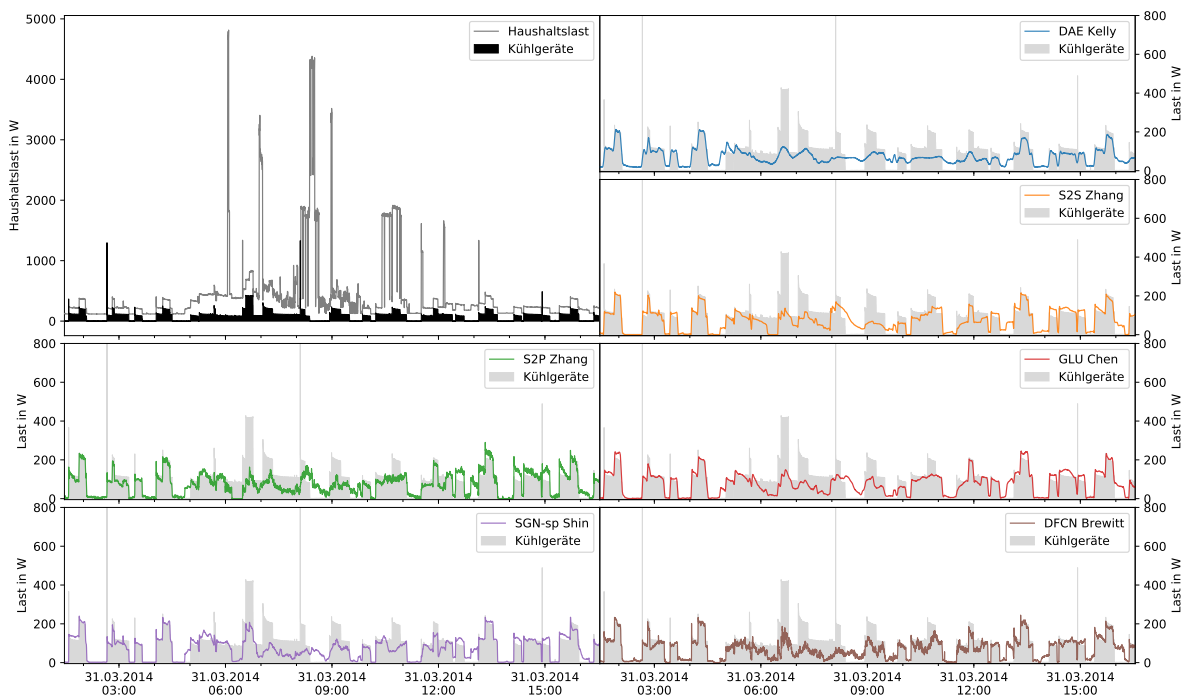


Abbildung 5.10: Disaggregierter Kühlgerätelastgangsausschnitt aus REFIT 19.

In REFIT 19 und im ESHL sind jeweils zwei parallel betriebene Kühlgeräte installiert. In den disaggregierten Kühlgerätelastgängen in Abbildung 5.10 ist zu erkennen, dass der parallele Betrieb der beiden Kühlschränke in REFIT 19 von den Ansätzen erkannt wird. Während in ESHL 2 (in Abbildung 5.11 ausschnittsweise dargestellt) die Ansätze meist nur einen der beide Kühlgeräte erkennt. Die Ursache der schlechteren Erkennung des zweiten Kühlgeräts liegt möglicherweise darin, dass das zweite Kühlgerät kleiner dimensioniert ist und mit

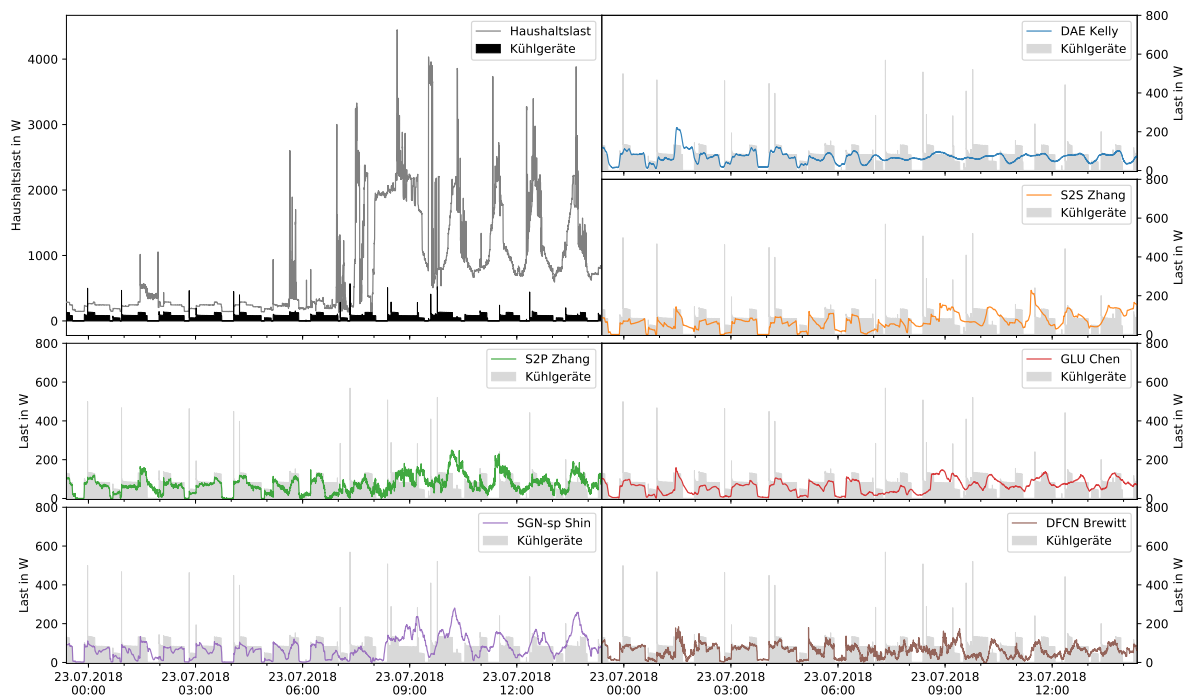


Abbildung 5.11: Disaggregierter Kühlgerätestlastgangsausschnitt aus ESHL 2.

etwa 50 W eine vergleichsweise kleine Maximallast aufweist. Solche Kühlschränke sind möglicherweise nicht in der Trainingsdatenmenge enthalten.

5.2.3 Trockner

Abbildung 5.12 stellt die Ergebnisse der Disaggregation von Wäschetrocknern dar. Die Trocknermodelle in REFIT 14 und REFIT 19 werden mit einem eMCC zwischen 0.49 und 0.82 gut erkannt, wobei der Ansatz SGN-sp die höchsten eMCC-Werte und die niedrigsten MAE-Werte erreicht.

Das Modell im ESHL wird hingegen mit einem eMCC unter 0.39 und einem mittleren Fehler über 33.5 W relativ schlecht disaggregiert.

Gute Disaggregation mit leichter Energieverbrauchsüberschätzung in REFIT 14 und REFIT 19

Wie in Abbildung 5.13 beispielhaft dargestellt, wird die Aktivität der Trocknermodelle in REFIT 14 und REFIT 19 von allen Ansätzen gut erkannt.

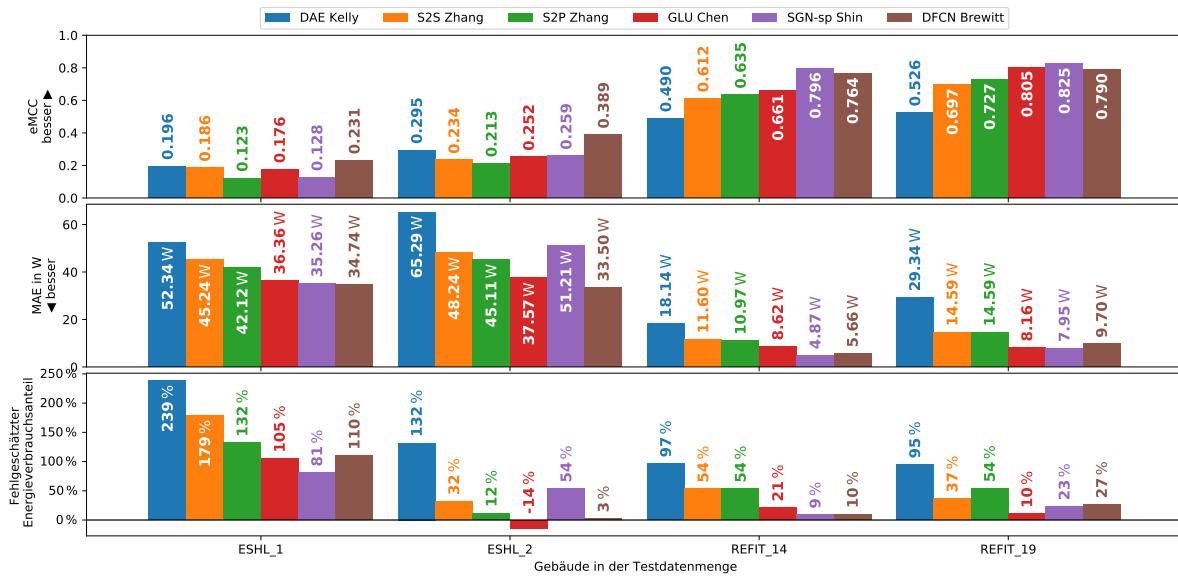


Abbildung 5.12: Matthews Correlation Coefficient (eMCC), Mittlerer absoluter Fehler (MAE) und Fehlgeschätzer Anteil am Energieverbrauch (RETE) bekannter Disaggregationsansätze bei Trocknern.

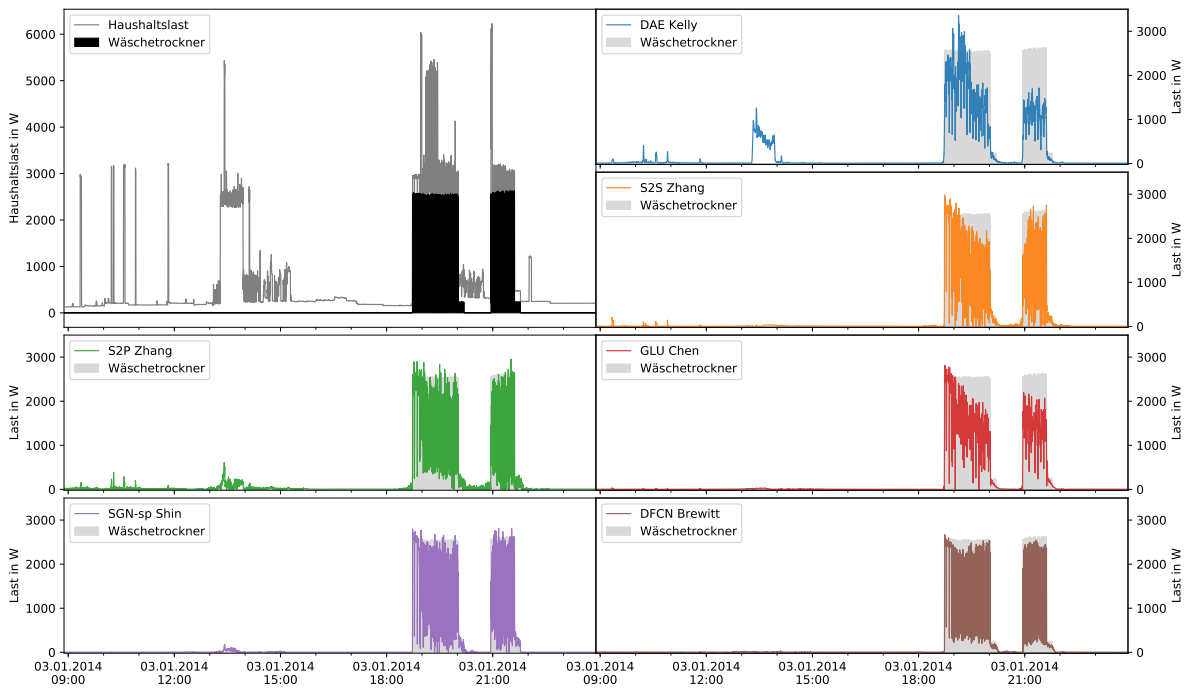


Abbildung 5.13: Disaggregierter Trocknerlastgangsausschnitt aus REFIT 14.

Während des Trockenprogramms wechselt die Last häufig zwischen einer mittleren Last (bei etwa 250 W) und einer Maximallast. Die Lastsprünge sind in den disaggregierten Lastgängen deutlich erkennbar, jedoch werden die tatsächlichen Lastwerte von den Ansätzen DAE, S2S und GLU nicht genau getroffen. Zudem ist bei DAE und S2P am deutlichsten zu erkennen, dass die Last anderer Haushaltsgeräte fälschlicherweise als Trocknerlast erkannt wird. Dies führt insgesamt, wie anhand des positiven Anteils fehlerhaft zugewiesenen Energieverbrauchs in Abbildung 5.12 zu erkennen, zu einem überschätzten Energieverbrauch durch alle Ansätze.

Unerkanntes Gerätemodell im ESHL

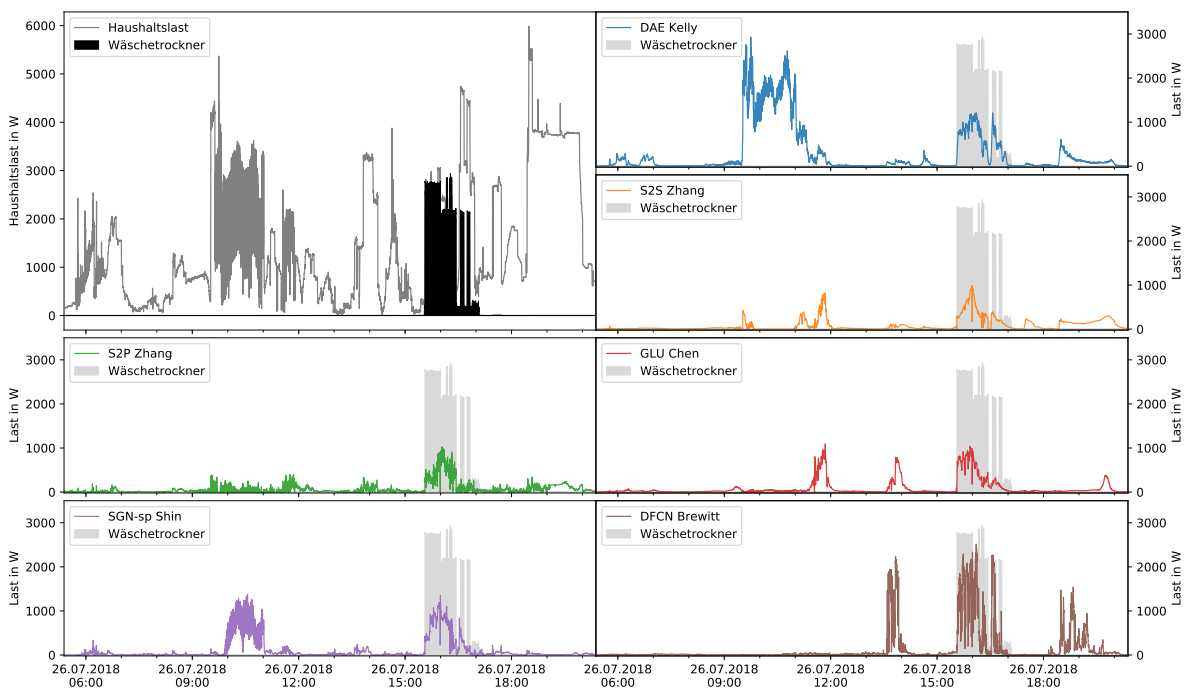


Abbildung 5.14: Disaggregierter Trocknerlastgangsausschnitt aus ESHL 2.

In Abbildung 5.14 ist beispielhaft eine Trockneraktivierung im ESHL dargestellt, während dessen weitere Geräte aktiv sind. Im Vergleich zu den Trocknermodellen der anderen Testgebäude besitzt das Trocknermodell des ESHL eine hohe Maximallast knapp unter 3000 W. Zudem ist im Lastgang neben einer Laststufe bei 200 W eine zusätzliche Laststufe bei 2000 W erkennbar und die Last fällt selten auf die Laststufe bei 200 W ab. Ein Trockner mit einer solchen Lastgangscharakteristik kommt in den Trainingsdaten nicht vor.

Alle Ansätze erkennen daher die Aktivität des Trocknermodells im ESHL nur knapp und weisen dem Gerät nur ein Teil der tatsächlich verbrauchten Energiemenge zu. Gleichzeitig

weisen die Ansätze einen Anteil des Energieverbrauchs anderer Geräte fälschlicherweise dem Trockner zu. In der disaggregierten Energieverbrauchsbilanz überwiegt tendenziell der Anteil fehlerhaft zugewiesenen Energieverbrauchs anderer Geräte.

5.2.4 TV und Unterhaltungselektronik

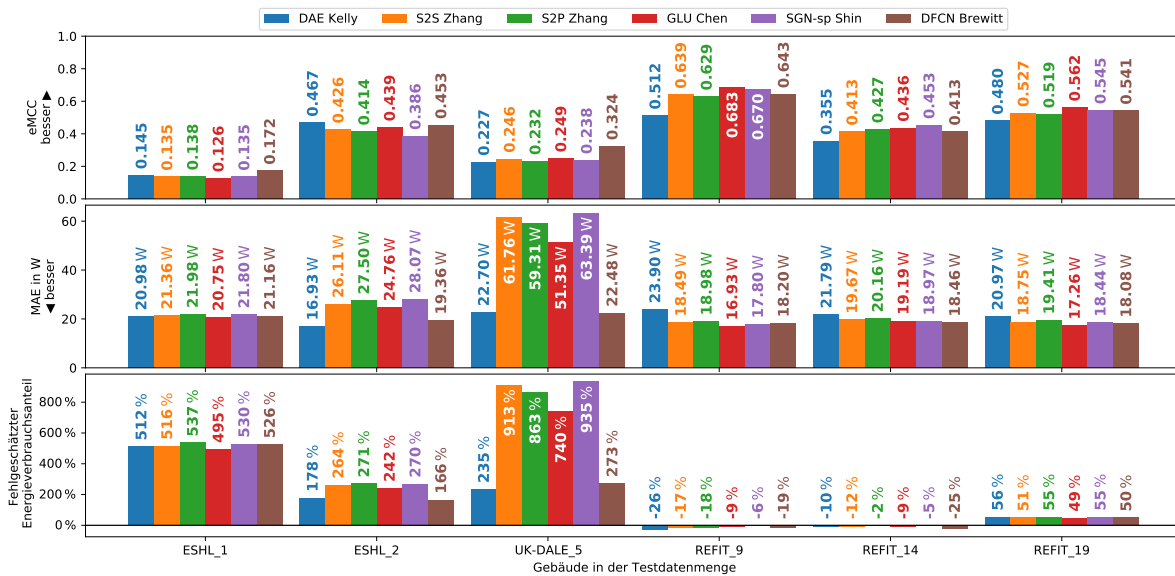


Abbildung 5.15: Matthews Correlation Coefficient (eMCC), Mittlerer absoluter Fehler (MAE) und Fehlgeschätzter Anteil am Energieverbrauch (RETE) bekannter Disaggregationsansätze bei TV und Unterhaltungselektronik.

Abbildung 5.15 stellt die Ergebnisse der Disaggregation von TV und Unterhaltungselektronik dar. Im Vergleich zu den Ergebnissen der anderen Haushaltsgeräte erreicht keiner der Ansätze einen MAE unter 15 W, welches ein relativ hoher Wert ist, da die Spitzenlast dieser Geräteklasse in der Regel unter 100 W liegt.

Dennoch erreichen manche Ansätze in manchen Gebäuden einen Korrelationskoeffizienten von über 0.6 und der Energieverbrauch wird bis auf 2 % genau geschätzt. Der hohe MAE zeigt jedoch ein systematisches Problem bei der Disaggregation von TV und Unterhaltungselektronik an, welches bei der folgenden Betrachtung von Beispielen aufgedeckt wird.

Koinzidenz von Haushaltslast und Aktivität von TV und Unterhaltungselektronik

Von den Testgebäuden weist REFIT 9 die höchsten eMCC-Ergebnisse und gleichzeitig eine der niedrigsten Energiefehlschätzungen auf und stellt somit das Gebäude dar, in dem die

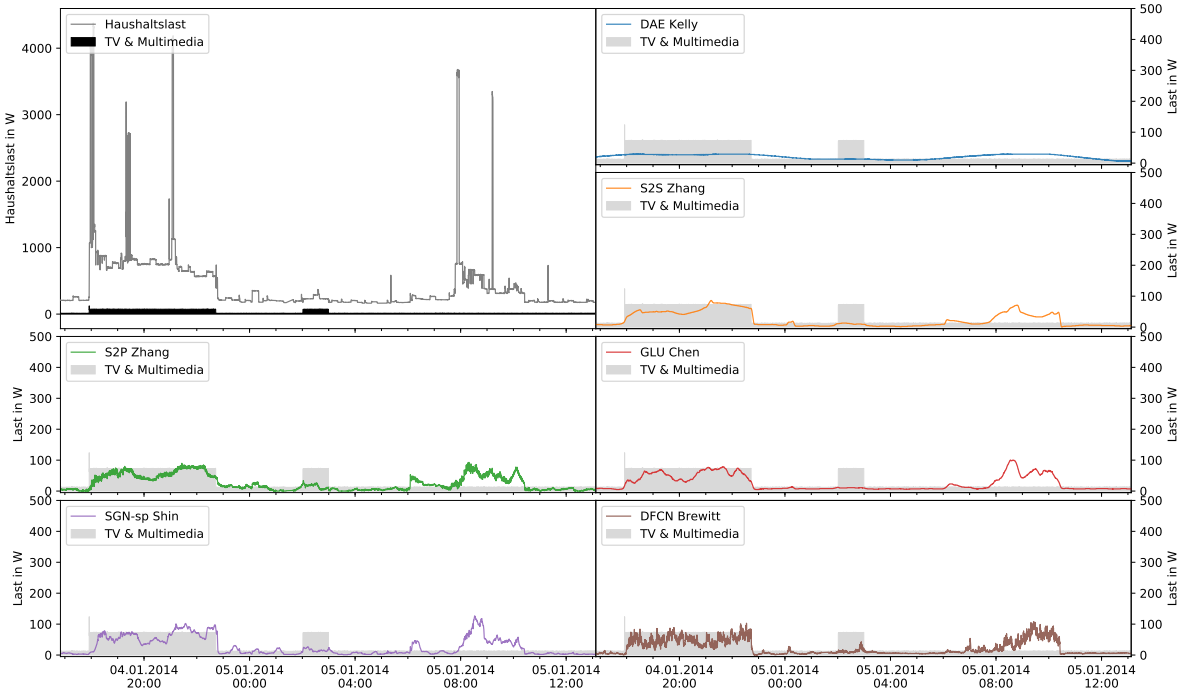


Abbildung 5.16: Disaggregierter Lastgangausschnitt für TV und Unterhaltungselektronik in REFIT 9.

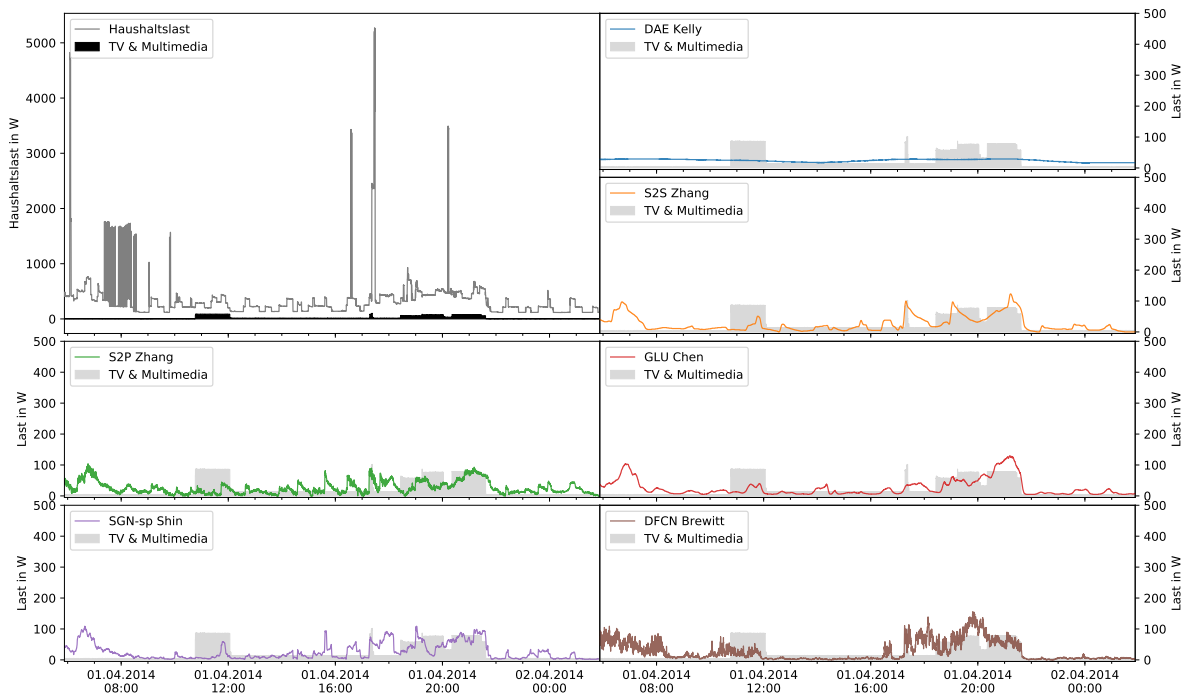


Abbildung 5.17: Disaggregierter Lastgang von TV und Unterhaltungselektronik in REFIT 19.

Disaggregation am erfolgreichsten war. Abbildung 5.16 zeigt einen disaggregierten Abschnitt. Auf die erste Aktivierung ab 18 Uhr reagieren alle Ansätze, während die zweite Aktivierung um 2 Uhr nicht erkannt wird. Ab 8 Uhr steigt die mittelfristige Haushaltsgrundlast ohne eine Aktivität von TV und Unterhaltungselektronik. Fälschlicherweise weisen die Ansätze dieser Geräteklasse Energieverbrauch zu. Abbildung 5.17 stellt einen Abschnitt aus REFIT 19 dar. Auch hier ist kein starker Zusammenhang zwischen tatsächlicher und geschätzter TV- und Unterhaltungselektroniklast erkennbar. Die Lastschätzung korreliert stattdessen mit der mittelfristigen Haushaltsgrundlast.

Dies legt den Schluss nahe, dass die Ansätze die Aktivität von TV und Unterhaltungselektronik nicht primär an gerätespezifischen Merkmalen im Lastgang erkennen, sondern stattdessen die Koinzidenz von Unterhaltungselektronikaktivität mit einer hohen Haushaltsgrundlast ausgenutzt wird.

5.2.5 Mikrowellen

Abbildung 5.18 stellt die Ergebnisse bei der Disaggregation von Mikrowellen dar. Da Mikrowellen in der Regel kurz aktiv sind und somit auch einen geringen Gesamtenergieverbrauch haben, ist es für die Ansätze leichter, einen niedrigen mittleren Fehlerbetrag zu

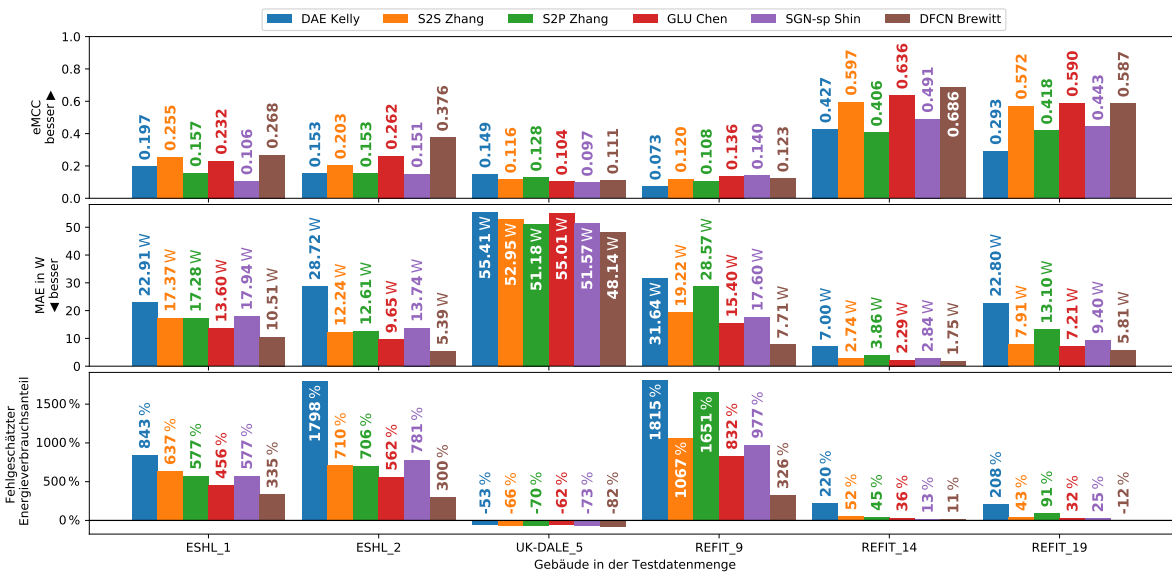


Abbildung 5.18: Matthews Correlation Coefficient (eMCC), Mittlerer absoluter Fehler (MAE) und Fehlgeschätzter Anteil am Energieverbrauch (RETE) bekannter Disaggregationsansätze bei Mikrowellen.

erreichen. Die kurze Aktivitätszeit erschwert jedoch das Erreichen eines hohen Korrelationswerts.

Im besten Fall wird die Mikrowelle wie in REFIT 14 disaggregiert. Der Beispielausschnitt in Abbildung 5.19 aus REFIT 14 zeigt, wie die Last der Mikrowelle zwar erkannt wird, jedoch wird dem Gerät auch im Standby fälschlicherweise Last zugewiesen.

Wie beispielhaft anhand des Ausschnitts aus REFIT 9 in Abbildung 5.20 dargestellt, wird in den Testgebäuden REFIT 9 und ESHL 1 und 2 ein großer Anteil des Energieverbrauchs der Mikrowelle falsch zugewiesen.

Bei dem Mikrowellenlastgang in UK-DALE 5 liegt die Minimallast bei etwa 50 W. Keines der Ansätze sagt eine nennenswerte Grundlast voraus, sodass die Ergebnisse für dieses Testgebäude insgesamt schlecht ausfallen.

5.2.6 Geschirrspüler

Wie die Ergebnisse in Abbildung 5.21 darstellen, werden Geschirrspüler in der Regel gut erkannt. Für jedes Gebäude gibt es mindestens einen Ansatz, der einen Korrelationswert von über 0.72 erreicht.

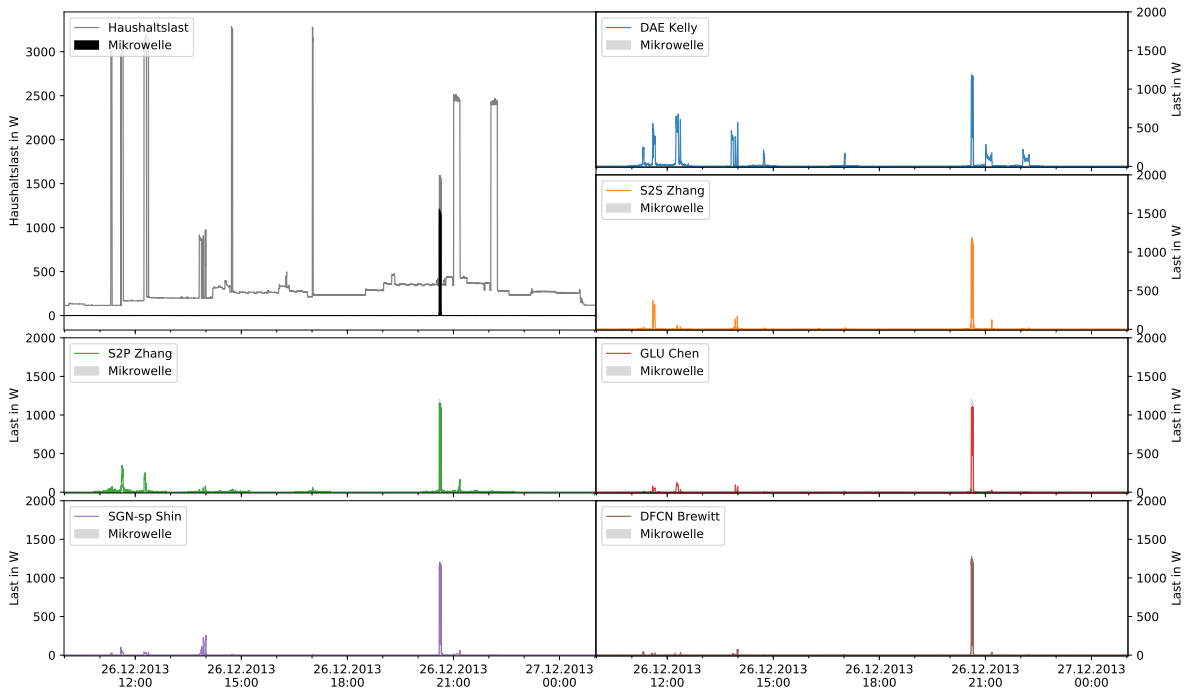


Abbildung 5.19: Disaggregierter Mikrowellenlastgangsausschnitt aus REFIT 14.

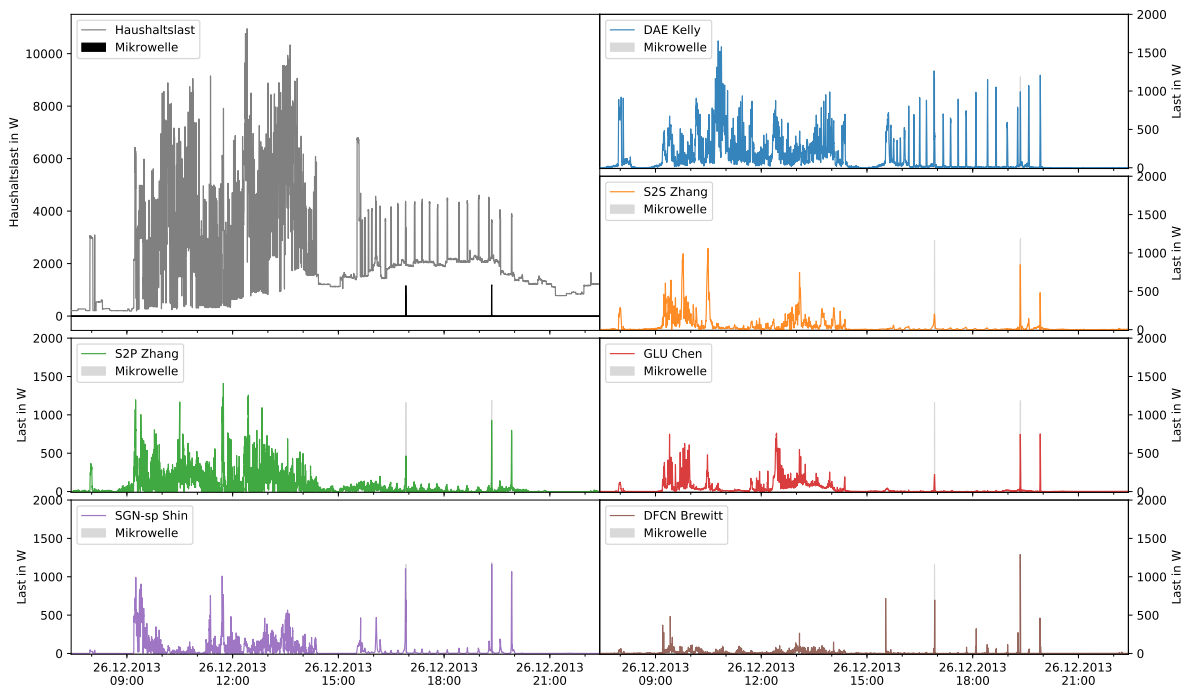


Abbildung 5.20: Disaggregierter Mikrowellenlastgangsausschnitt aus REFIT 9.

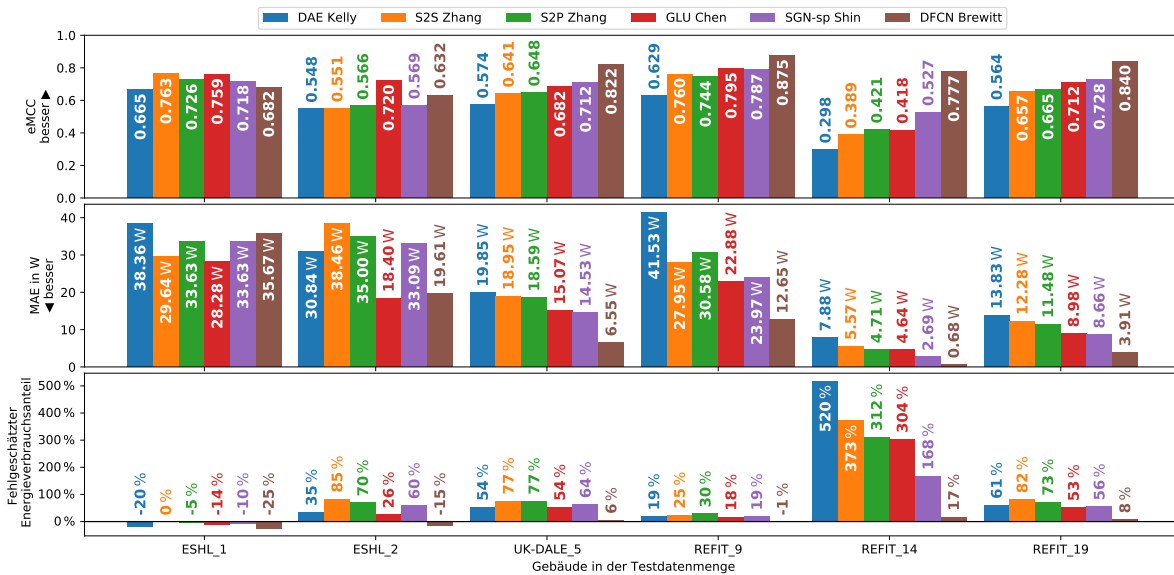


Abbildung 5.21: Matthews Correlation Coefficient (eMCC), Mittlerer absoluter Fehler (MAE) und Fehlgeschätzter Anteil am Energieverbrauch (RETE) bekannter Disaggregationsansätze bei Geschirrspülern.

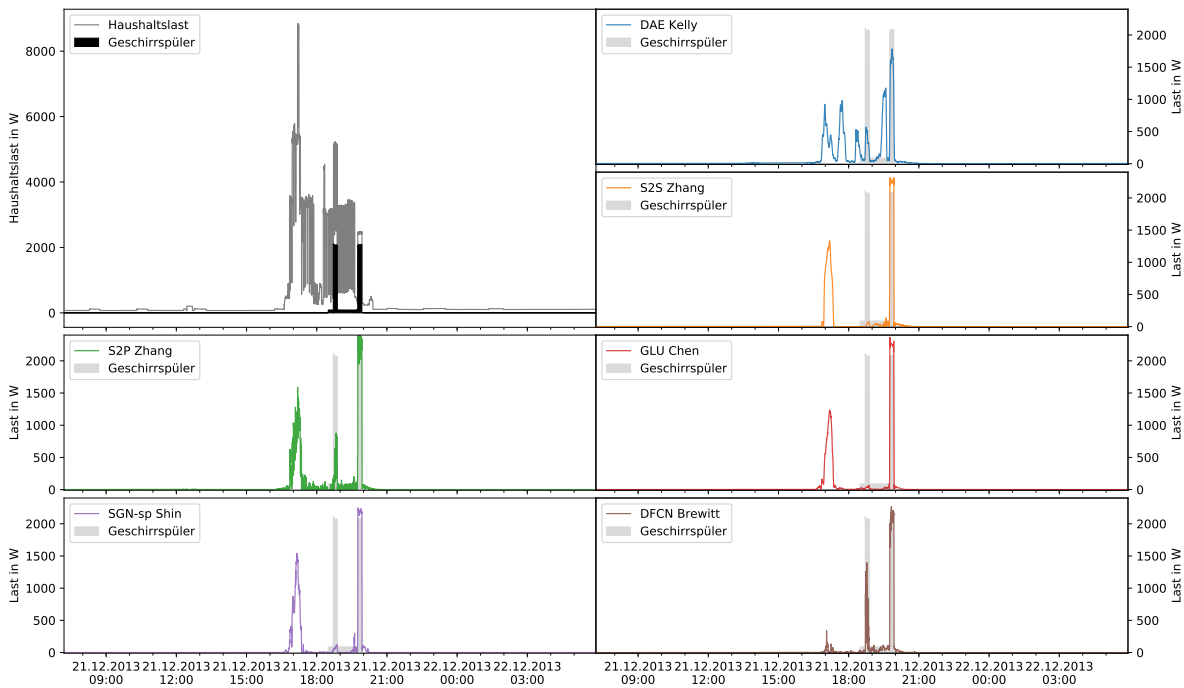


Abbildung 5.22: Disaggregierter Geschirrspülerlastgangsausschnitt aus REFIT 14.

Insbesondere in Gebäude REFIT 14 werden die Unterschiede zwischen den Ansätzen erkennbar. Die Darstellung der disaggregierten Lastgänge in Abbildung 5.22 zeigt, dass alle Ansätze damit kämpfen, die Last des Geschirrspülers aus der Überlagerung anderer Geräte zu extrahieren. Dabei schafft es DFCN besonders gut, andere Gerätelasten nicht fälschlicherweise als Geschirrspülerlast zu erkennen. Dennoch ist auch DFCN nicht in der Lage, den vollständigen Energieverbrauch des ersten Lastblocks der Spülmaschinenaktivierung anzurechnen.

5.2.7 Wasserkocher

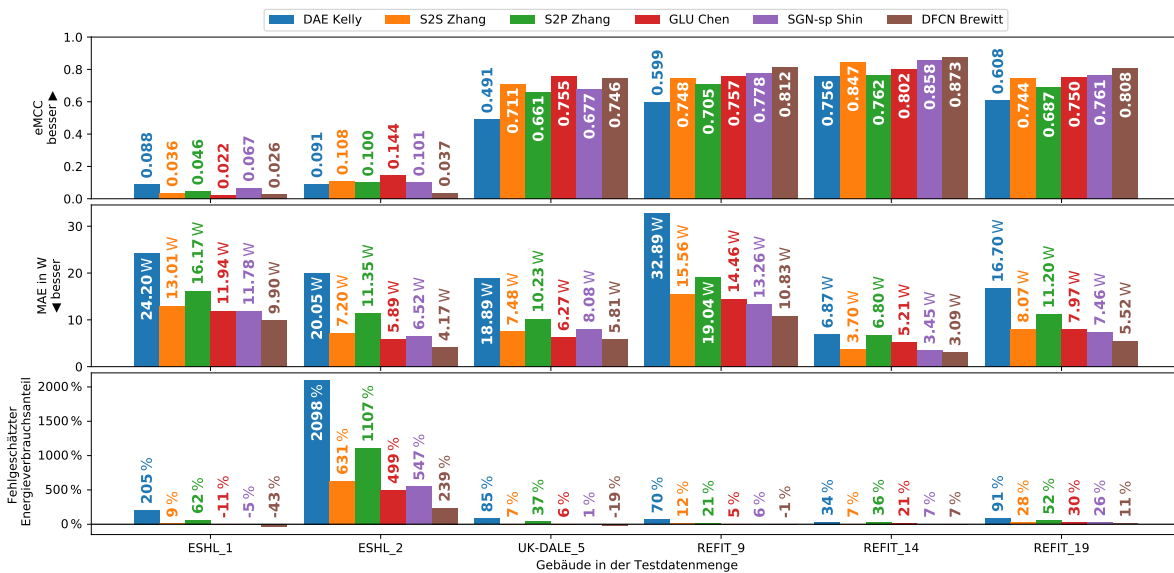


Abbildung 5.23: Matthews Correlation Coefficient (eMCC), Mittlerer absoluter Fehler (MAE) und Fehlgeschätzter Anteil am Energieverbrauch (RETE) bekannter Disaggregationsansätze bei Wasserkochern.

Die Wasserkochermodelle in den Gebäuden aus Großbritannien (REFIT und UK-DALE) werden mit einer hohen eMCC-Wert um 0.8 gut erkannt. Keines der Ansätze erreicht bei dem im ESHL verwendeten Wasserkocher einen Korrelationswert von über 0.15, was anzeigt, dass dieses Modell nicht erkannt wird.

Dies bestätigt sich mit einem Blick auf einen Ausschnitt des disaggregierten Lastgangs von ESHL 2 in Abbildung 5.24: Die beiden Aktivierungen werden nicht konsistent erkannt. Stattdessen werden andere kurzfristige Lastspitzen als Wasserkocheraktivität fehlinterpretiert.

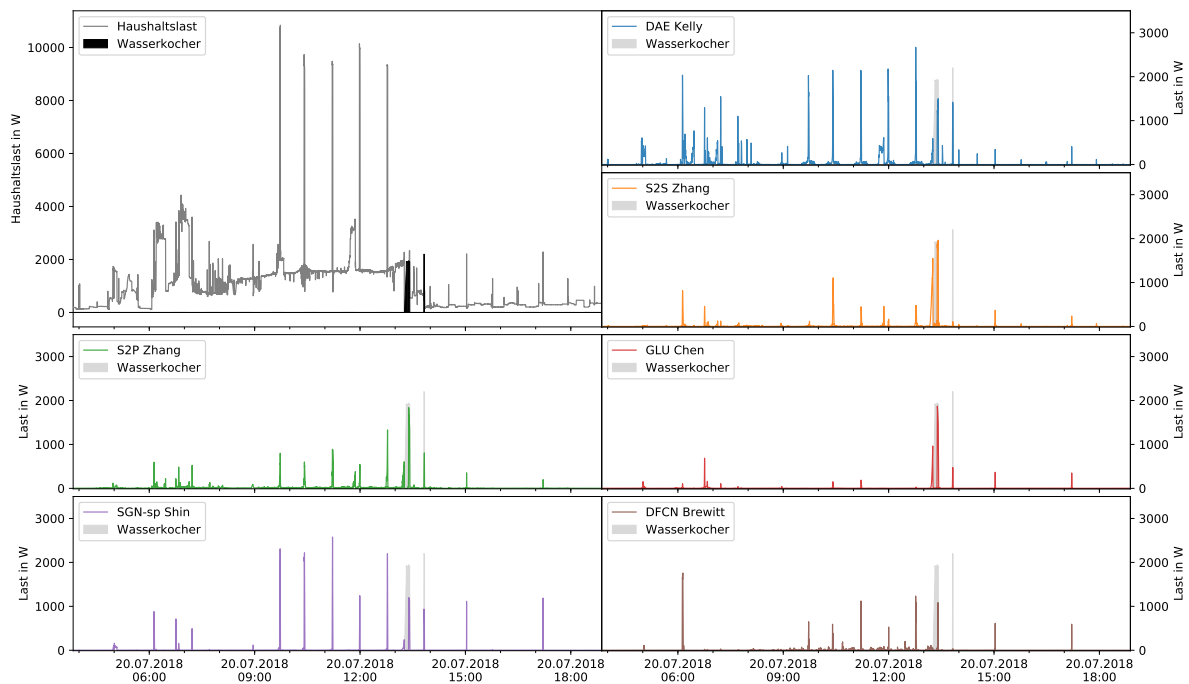


Abbildung 5.24: Disaggregierter Wasserkocherlastgangsausschnitt aus ESHL 2.

Eine Ursache für dieses Ergebnis liegt möglicherweise darin, dass die britisch Wasserkochermodelle eine Spitzenlast von etwa 3000 W aufweisen, während die Spitzenlast des Wasserkochers im ESHL bei 2000 W liegt.

5.3 Zusammenfassung

Zur Analyse des Standes der Technik zu Deep-Learning-basierten Disaggregationsansätzen ist eine Auswahl bestehender Ansätze (siehe Abschnitt 3.11) in einem einheitlichen Experimentaufbau (siehe Kapitel 4) evaluiert worden. Im Folgenden werden die dabei gewonnenen Erkenntnisse und der für die weitere Arbeit identifizierte Forschungsbedarf erläutert.

5.3.1 Schlussfolgerungen aus der Analyse

Bei allen Geräteklassen fällt zunächst auf, dass die Ergebnisse der Ansätze für ein jeweiliges Testgebäude oftmals nahe beieinanderliegen. Nur selten unterscheiden sich die Ergebnisse der Ansätze so stark wie bei der Disaggregation von Geschirrspülern in REFIT 19 (siehe

Abschnitt 5.2.6). Dies deutet auf eine Veranlagung im Experimentaufbau hin und weniger auf die verwendeten neuronalen Netzmodelle.

Nach Analyse der disaggregierten Lastgänge werden folgende Ursachen für Disaggregationsprobleme identifiziert.

Unbekannte Gerätemodelle in den Testgebäuden

Bei manchen Testgebäuden kann kein Ansatz gute Disaggregationsergebnisse liefern. Dies liegt daran, dass in den Trainingsdaten offenbar kein Gerätemodell vorhanden war, welches dem Gerät im Testgebäude ähnlich ist.

Beispielsweise besitzt der Wasserkocher im ESHL im Vergleich zu denen in den anderen Testgebäuden eine wesentlich geringere Leistung und wird bei der Disaggregation nicht erkannt (siehe Abschnitt 5.2.7).

Ein anderes Beispiel betrifft die Waschtrockner in UK-DALE 5 und REFIT 9, bei denen die Last des Trockenprogramms nicht erkannt wird. Offenbar gab es in den Trainingsdaten keine oder nicht genügend Beispiele für Trockenprogramme von Waschtrocknern (siehe Abschnitt 5.2.1).

In anderen Fällen weisen die Ansätze nur ein Teil der Last dem Zielgerät zu, wie bei dem Trockner im ESHL, da vermutlich dieses Trocknermodell mit einer geringen Konfidenz erkannt wird (siehe Abschnitt 5.2.3).

Gleichzeitigkeit von Geräteaktivität in manchen Testgebäuden

Die gleichzeitige Nutzung von elektrischen Haushaltsgeräten erschwert den Disaggregationsprozess, da sich die Lastgänge und die Gerätemerkmale überlagern.

In den Testhaushalten ist dies am deutlichsten bei den Kühlgeräten erkennbar. In der Nacht sorgt meist nur das An- und Ausschalten des Kühlaggregats für Laständerungen im Haushalt und die Ansätze können in der Zeit die Kühlgeräteleast mit hoher Genauigkeit disaggregieren. Sobald am Tag weitere Geräte genutzt werden, können die Ansätze den Schaltzyklus der Kühlgeräte kaum erkennen. (siehe Abschnitt 5.2.2)

Bei allen Geräteklassen sorgen parallele Geräteaktivitäten sowohl für falsch negative als auch falsch positiv zugewiesene Energiemengen. Dies zeigt sich beispielsweise im Ausschnitt in Abbildung 5.5 deutlich, bei dem die Last der ersten Waschmaschinenaktivierung von den Ansätzen gut erkannt wird. Während der zweiten Waschmaschineaktivierung ist die Geschirrspülmaschine gleichzeitig aktiv, sodass die Waschmaschinenlast nicht vollständig erkannt wird. Zudem erkennen manche Ansätze die Geschirrspülmaschinenlast sowie die Last anderer Geräte fälschlicherweise als Waschmaschinenlast.

Fehlende oder wenige gerätespezifische Merkmale im Lastgang

Bei TV und Unterhaltungselektronik sowie bei der Mikrowelle fällt es den Ansätzen schwer, eindeutige Merkmale im Haushaltslastgang für die Aktivität dieser Geräte zu finden.

Im Fall der Mikrowelle reagieren die Ansätze auf Geräteaktivität, jedoch sind die erlernten Merkmale nicht spezifisch genug, um die Aktivität anderer Geräte auszuschließen. Somit wird oft fälschlicherweise Mikrowellenaktivität erkannt, obwohl das Gerät tatsächlich inaktiv ist (so wie in Abbildung 5.21 dargestellt).

Bei der Geräteklasse TV und Unterhaltungselektronik ist nicht erkennbar, dass ein zuverlässiges Merkmal für die Aktivität dieser Geräte gefunden wurde. Der disaggregierte Lastgang zeigt stärkere Korrelationen zur mittelfristigen Haushaltsgrundlast als zum gemessenen Gerätelastgang (wie am Beispiel in Abbildung 5.16 erkennen). Die teils befriedigenden Disaggregationsergebnisse in Abbildung 5.15 basieren lediglich auf der Koinzidenz von Unterhaltungselektronikaktivität und allgemein hoher Aktivität im Haushalt.

Ausgabe von unrealistischen Gerätelastgängen

Die von den Ansätzen disaggregierten Lastgänge enthalten oftmals Merkmale wie Lastrampen, mittelhohe Lastwerte und Rauschen, die in gemessenen Gerätelastgängen nicht vorkommen und von echten Geräten nicht reproduziert werden können. Die Ursache liegt vermutlich in der fehlenden Konfidenz bei der Erkennung von Geräteaktivität, mit der das Modell die Gerätelast zu skalieren scheint. Dies ist besonders bei weniger erfolgreichen Disaggregationen wie beispielsweise bei dem Trocknermodell des ESHL in Abbildung 5.14 ersichtlich.

Dieser Effekt tritt jedoch auch bei erfolgreichen Disaggregationen auf wie bei der Waschmaschine von REFIT 19 in Abbildung 5.2. Hier ist die Differenz zum tatsächlichen Lastgang bereits gering, jedoch lassen sich die disaggregierten Lastgänge anhand der unrealistischen Merkmale als Schätzungen identifizieren.

5.3.2 Forschungsbedarf

Die Ergebnisse in Abschnitt 5.2 zeigen, dass mindestens mit dem DFCN ein Ansatz existiert, welches hochqualitative Disaggregationen durchführen kann. Dafür sind jedoch, wie in Abschnitt 5.3.1 aufgeführt, ähnliche Gerätemodelle in den Trainingsdaten und eindeutige Lastgangsmerkmale für die jeweilige Geräteklasse erforderlich.

Um einige der in Abschnitt 5.3.1 beschriebenen Probleme bei der Disaggregation zu beheben, werden folgende Ansätze verfolgt.

Vermeidung von unrealistischen Gerätelastgängen Die neuronalen Netze sind selten in der Lage, realistische Gerätelastgänge auszugeben. Mit der Integration des Konzepts von realistischen Lastgängen in das neuronale Gerätemodell wäre eine Disaggregation möglicherweise einfacher, da unrealistische Varianten ausgeschlossen werden können. Auch ist eine Korrektur von unrealistischen Gerätelastgängen vorstellbar, um die Qualität der ausgegebenen disaggregierten Lastgänge zu erhöhen.

Auswirkungen der Kombination mehrerer Datensätze Um die Anzahl der von den Ansätzen bekannten Gerätemodelle zu vergrößern, ist eine Erhöhung der verwendeten Datensätze wünschenswert. Jedoch wurde bereits zu Beginn eine im Vergleich zu bestehenden Arbeiten größere Anzahl an Datensätzen kombiniert. Zur Verifikation der These, dass eine niedrige Disaggregationsqualität durch fehlende Gerätemodelle im Training verursacht wird, kann die Anzahl der verwendeten Datensätze reduziert werden.

Ansätze zur Optimierung der Disaggregationsqualität

Bei der qualitativen Beurteilung vorhandener Ansätze ist beobachtet worden, dass die neuronalen Netze besonders im Fehlerfall keine von Geräten realisierbare Lastgänge ausgeben. Die fehlerhaften Lastgänge enthalten zum einen Lastwerte, die in gemessenen Lastgängen nicht vorkommen. Zum anderen sind in den Lastgängen nicht die charakteristischen Programmabfolgen des entsprechenden Gerätes erkennbar.

Unter der Annahme, dass ein fehlerhaft geschätzter Lastgang eine Ähnlichkeit zum tatsächlichen Gerätelastgang aufweist, kann ein fehlerhafter Lastgang korrigiert werden, wenn das neuronale Netz diesen fehlerhaften Lastgang auf den ähnlichsten realisierbaren Gerätelastgang abbildet.

Mit dieser Motivation verfolgen die in diesem Kapitel vorgestellten drei Ansätze das Ziel, bei der Schätzung die Ausgabe von Lastgängen zu erzwingen, die von einem realen Gerät umsetzbar sind.

6.1 Ansatz A: Unüberwacht trainierter Generator für Gerätelastgänge

Dieser Ansatz verfolgt das Ziel, ein neuronales Netz zu erstellen, das in der Lage ist, realisierbare Lastgänge einer bestimmten Geräteklasse mit einer hohen Wahrscheinlichkeit zu erzeugen. Wie in Abbildung 6.1 dargestellt, soll während des Disaggregationsprozesses ein solcher Lastgangsgenerator als zusätzlicher Verarbeitungsschritt (Schritt 4b) zwischen Disaggregation (Schritt 4a) und Nachverarbeitung (Schritt 5) eingeführt werden.

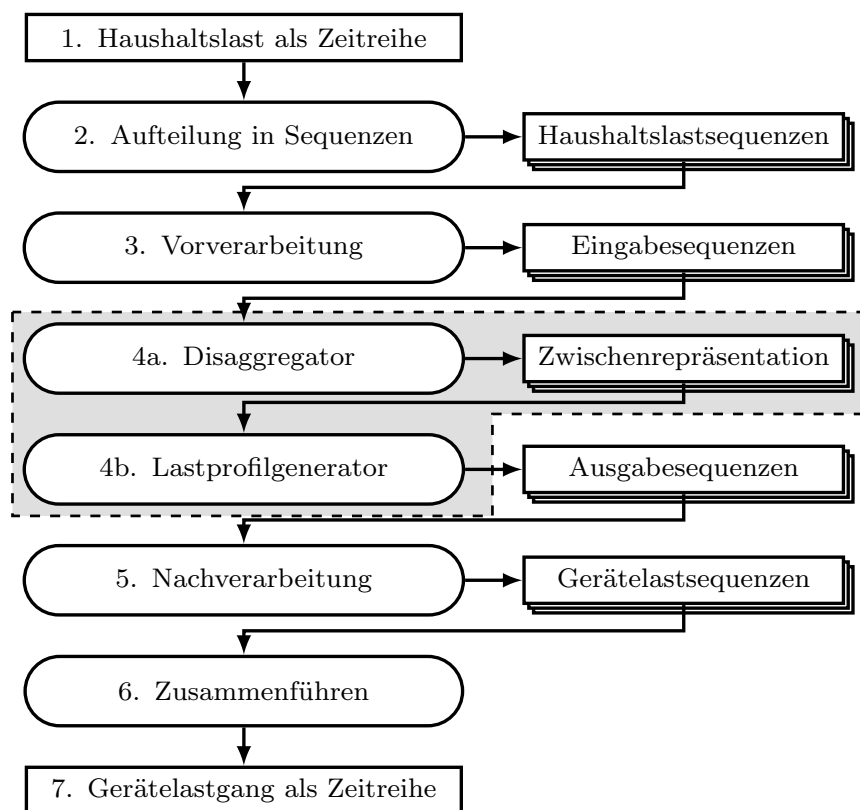


Abbildung 6.1: Integration des Gerätelastsequenzgenerators in den Disaggregationsprozess. Änderungen zum bisherigen Disaggregationsprozess sind grau hinterlegt.

Dazu wird der Disaggregator dahingehend modifiziert, dass er eine Zwischenrepräsentation des Lastgangs ausgibt, aus dem der Lastprofilgenerator erst im folgenden Schritt eine Gerätelastsequenz erzeugt. Dieser zusätzliche Verarbeitungsschritt soll Fehler bei der Disaggregation korrigieren und die Ausgabe eines realistischen Lastgangs begünstigen.

Eine Beschreibung des vollständigen Raums aller realisierbaren Lastgänge einer Geräteklasse ist mühselig zu erstellen. Daher wird in diesem Ansatz ein Generative Adversarial Network (GAN) (siehe Abschnitt 2.5.9) basierend auf aufgezeichneten Gerätelastgängen trainiert, um diesen Raum näherungsweise in ein neuronales Netzmodell zu überführen.

Als Ergebnis entsteht ein neuronales Netz, welches annähernd realisierbare Gerätelastgänge erzeugt (Generator) und ein weiteres neuronales Netz, welches realisierbare von nicht realisierbaren Gerätelastgängen unterscheidet (Diskriminator). Im Anschluss wird nur der Generator im Disaggregationsprozess weiterverwendet.

Dieses Konzept sowie erste Experimente sind in einer gemeinsamen Arbeit (Bao, Ibrahimov et al., 2018) veröffentlicht. In dieser Arbeit wird die Fragestellung nach dem Mehrwert dieses Ansatzes ergänzend aufgeklärt.

6.1.1 Vorgehensweise

Im ersten Schritt werden Generatoren basierend auf dem GAN-Konzept entworfen, die augenscheinlich in der Lage sind, alle realisierbaren Gerätelastprofile zu erzeugen. (Abschnitte 6.1.3 bis 6.1.6)

Die Güte eines erstellten Generators ist jedoch schwer direkt zu bestimmen. Gäbe es dafür eine Metrik, könnte diese direkt zur Erstellung eines neuronalen Modells eingesetzt werden und würde den Einsatz des GAN überflüssig machen. Zumindest können anhand einiger zufällig generierter Beispiellastsequenzen offensichtlich dysfunktionale Generatoren leicht erkannt werden.

Um die Einsatzfähigkeit dieser Generatoren für das vorgeschlagene Disaggregationskonzept dennoch zu überprüfen, wird im zweiten Schritt evaluiert, wie genau der Testdatensatz mit diesen Generatoren reproduziert werden kann. Somit werden die Limitierungen quantifiziert, die durch den Einsatz eines dedizierten Lastprofilgenerators entstehen. (Abschnitt 6.1.7)

Als finaler Schritt werden die Generatoren in einen vollständigen Disaggregationsansatz integriert und evaluiert. Der Vergleich mit den Ergebnissen aus dem zweiten Schritt ermöglicht eine Aussage über das Potential des Ansatzes. (Abschnitt 6.1.8)

6.1.2 Einschränkung auf Waschmaschinenlastgänge

Im Folgenden wird lediglich die Geräteklasse Waschmaschine betrachtet, da unter den sieben Geräteklassen die Lastgänge der Waschmaschine eine Vielzahl augenscheinlich prüfbar Merkmale aufweisen: Ein Waschprogramm beginnt meist mit dem Aufheizen des Wassers, was anhand einer längeren Maximallast von etwa 2000W zu erkennen ist. Darauf erfolgt ein zyklischer Lastverlauf bei niedrigen Lastwerten, der durch das Umwälzen zustande kommt. Zwischenzeitlich ist Nachheizen des Wassers bei Maximallast möglich. Am Ende des Programms ist das Schleudern durch eine höhere Last und einer steigenden Lastrampe zu erkennen. (vgl. Abschnitt 4.2.2)

6.1.3 Erstellung des Lastsequenzgenerators

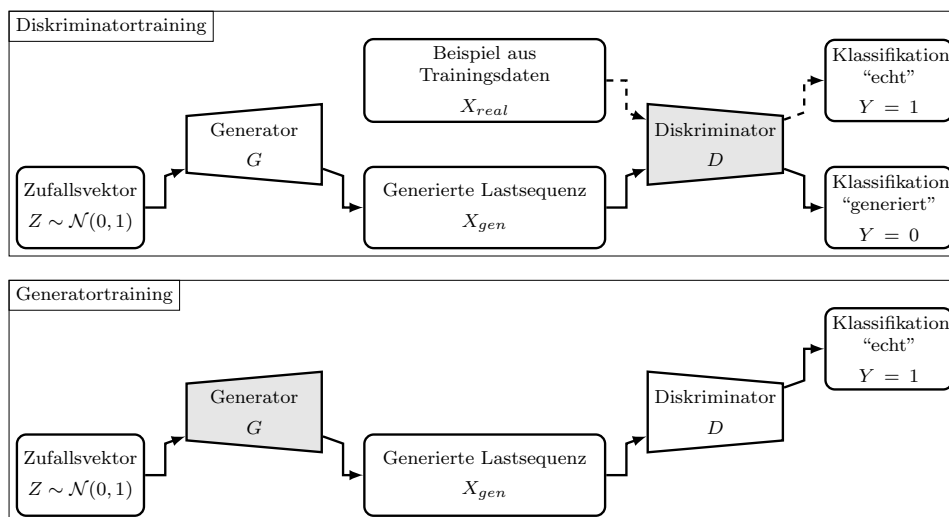


Abbildung 6.2: Datenpfade beim Training eines Generative Adversarial Network.

Der Lastsequenzgenerator wird als Bestandteil eines GAN (siehe Abschnitt 2.5.9) erstellt. Zum Training genügen Beispiele von echten Lastgangssequenzen des Zielgeräts. Wie in Abbildung 6.2 dargestellt, wird der Generator darauf optimiert, ausschließlich realisierbare Lastgänge zu erzeugen. Dies wird durch den Einsatz eines zweiten neuronalen Netzes als Diskriminator erreicht, der parallel zum Generator zu dem Zweck trainiert wird, zwischen generierten Lastgängen und realisierbaren Lastgängen aus dem Trainingsdatensatz zu unterscheiden. Nach dem Training erzeugt der Generator im Idealfall ausschließlich Lastgänge, die der Diskriminator nicht von echten Gerätelastgängen unterscheiden kann.

Während des Trainings dient ein Zufallsgenerator als Entropiequelle für Variationen im Gerätelastgang. Damit wird der durch den Zufallsgenerator abgedeckte Raum zu einer

Zwischenrepräsentation für Lastgänge einer bestimmten Geräteklasse. Die Freiheitsgrade werden somit datengetrieben erlernt und eine aufwendige manuelle Konstruktion eines Gerätemodells entfällt.

Trainingsdaten

Als Grundlage für das Training der Lastsequenzgeneratoren dienen die gleichen Datensätze, die auch für das Training der Disaggregationsansätze verwendet werden (siehe Abschnitte 4.3 und 4.5).

Aus der Trainingsdatenmenge erzeugt der in Abschnitt 4.6.2 beschriebene „Zufallssegmentgenerator mit ausbalancierter Gebäude- und Aktivitätshäufigkeit“ Trainingspaare bestehend aus Lastsequenzen des Haushalts und Lastsequenzen des Zielgeräts, wobei für das GAN-Training nur die Gerätelastsequenzen verwendet werden. Da nur Gerätelastsequenzen relevant sind, sind weitere Methoden zur Trainingspaarerzeugung redundant.

Die Parametrisierung des „Zufallssegmentgenerators mit ausbalancierter Gebäude- und Aktivitätshäufigkeit“ zur Erstellung der Trainingspaare erfolgt in den folgenden drei Varianten:

Disaggregatortraining (DT) Analog zum Training von Disaggregationsansätzen (siehe Abschnitt 4.6.5) kommt mit einer Wahrscheinlichkeit von 50% ein aktives Zielgerät in der Lastsequenz vor. Die Aktivierung muss dabei nicht zwingend vollständig innerhalb der ausgegebenen Lastsequenz liegen.

Unvollständige Aktivierungen (UA) In dieser Konfiguration werden ausschließlich Lastsequenzen mit Geräteaktivität erzeugt. Dabei muss sich die Aktivierung nicht zwingend vollständig innerhalb der Lastsequenz befinden und kann früher anfangen bzw. später zum Ende kommen.

Vollständige Aktivierungen (VA) Auch in dieser Variante werden zum Training nur Lastsequenzen mit Geräteaktivität erzeugt, wobei eine Aktivierung vollständig innerhalb der Lastsequenz zufällig platziert wird. Sollte die Aktivierung länger als die Lastsequenz sein, wird ein zufällig gewählter Abschnitt der Aktivierung verwendet.

Mit der Parametrisierung „Disaggregatortraining“ werden somit Daten erzeugt, die den geringsten Anteil an Geräteaktivität besitzen. Die Parametrisierung „Vollständige Aktivierungen“ erzeugt Trainingsdaten mit dem höchsten Anteil an Geräteaktivität.

Als Länge für die Gerätelastsequenzen wird 102,4 Minuten gewählt, da die meisten Waschmaschinenaktivierungen im UK-DALE-Datensatz kürzer als diese Zeitspanne sind (siehe Abschnitt 3.3.2).

Vorverarbeitung

Für die neuronalen Netze werden die Gerätestastwerte aus dem Bereich $[0W; 2500W]$ auf das Intervall $[-1; 1]$ normiert. Dabei liegt $2500W$ über der Maximallast der Waschmaschine und $(-1; 1)$ ¹ entspricht dem Bild von Tangens Hyperbolicus, der Aktivierungsfunktion in der letzten Schicht des Lastsequenzgenerators (siehe Abschnitt 6.1.4).

6.1.4 Netzarchitektur des Diskriminators und des Generators

Die neuronalen Netzarchitekturen für den Lastsequenzdiskriminator und den Generator basieren auf dem Ansatz Deep Convolutional Generative Adversarial Network (DC-GAN) von Radford et al. (2015) und sind auf Lastsequenzen mit nur einer zeitlichen Dimension adaptiert.

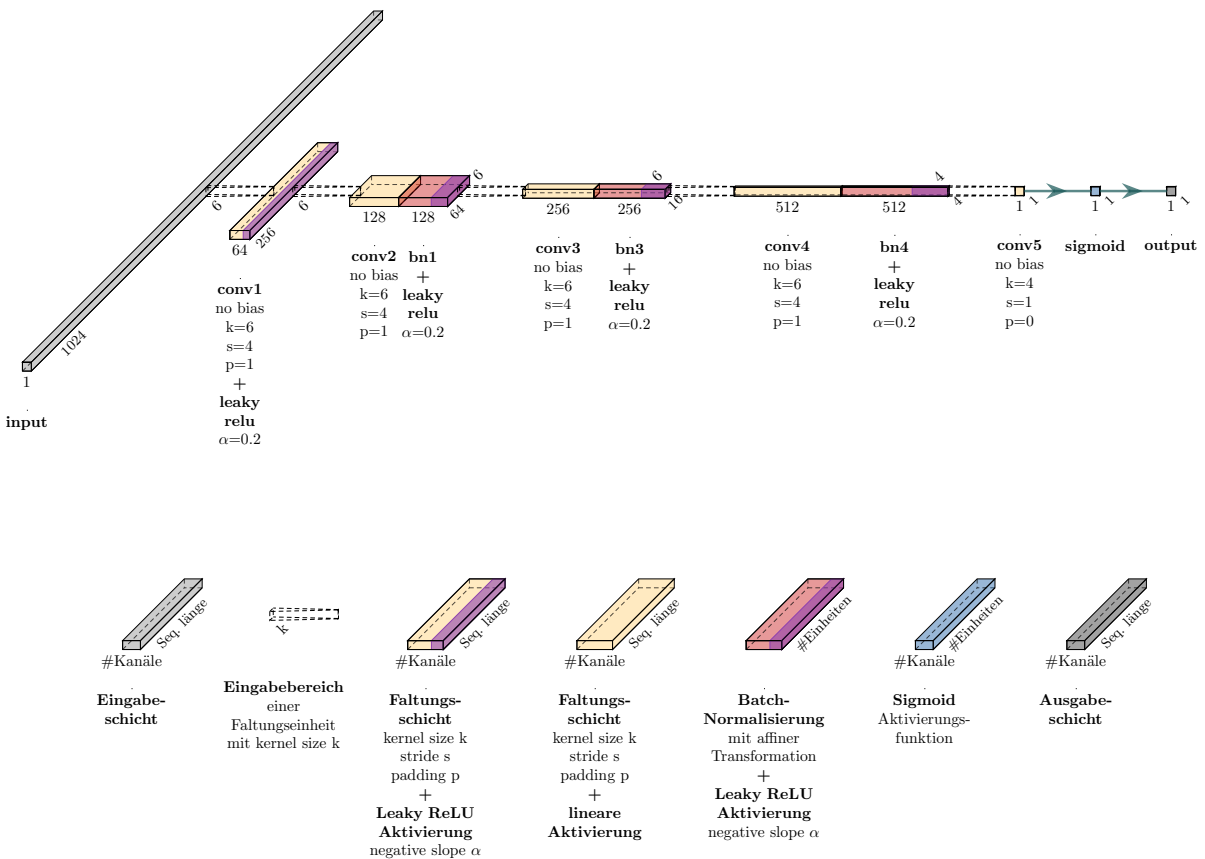


Abbildung 6.3: Netzarchitektur des Diskriminators basierend auf dem Ansatz Deep Convolutional Generative Adversarial Network (DC-GAN).

¹bzw. aufgrund der Fließkommazahlenrundung tatsächlich $[-1; 1]$.

Der Diskriminator für Lastsequenzen besteht, so wie in Abbildung 6.3 dargestellt, aus fünf Faltungsblöcken. Die Faltungsblöcke reduzieren die zeitliche Auflösung nach jeder Faltung auf ein Viertel der Eingabeauflösung. Bei den drei Faltungsblöcken in der Mitte der Netztopologie werden auf die Zwischenergebnisse Batch-Normalisierungen (siehe Abschnitt 2.5.5) angewendet.

Wie bei dem Diskriminatoraufbau von Radford et al. (2015) wird als innere Aktivierungsfunktion Leaky-ReLU verwendet. Mit dieser Aktivierungsfunktion existiert auch bei negativen Werten ein reduzierter Gradient. Als finale Aktivierungsfunktion wird die Sigmoidfunktion verwendet, da die Aufgabe des Netzes ein binärer Klassifikator ist.

Der Aufbau des Lastsequenzgenerators ist in Abbildung 6.4 dargestellt: Aus einer 100-dimensionalen Zwischenrepräsentation erzeugt es in sechs Schritten eine Lastsequenz. Mit der gewählten Größe des Eingabevektors wird somit implizit angenommen, dass die Anzahl der Freiheitsgrade von realisierbaren Waschmaschinenlastgängen unter 100 liegt. Die transponierten Faltungen erhöhen dabei schrittweise die zeitliche Auflösung und reduzieren gleichzeitig die Anzahl der Kanäle bzw. Merkmale.

In jedem Schritt wird eine transponierte Faltung mit anschließender Batch-Normalisierung durchgeführt, gefolgt von der Anwendung der ReLU-Aktivierungsfunktion. Die Faltungsschicht addiert keine Versatzwerte (bias) auf das Faltungsergebnis, wohingegen die Batch-Normalisierung pro Kanal eine affine Transformation mit Skalierungsfaktor und Versatzwert erlernt.

In ersten Experimenten wurde für den Generator zunächst eine zum Diskriminator spiegel-symmetrische Architektur mit fünf Blöcken verwendet. Das Training mit diesem Aufbau ist jedoch nicht stabil und der Generator erzeugt am Ende je nach Durchlauf unbefriedigende oder keine erkennbaren Waschmaschinenlastgänge (siehe Abschnitt 6.1.5). Ebenso ist die Batch-Normalisierung eine entscheidende Komponente in der Netzarchitektur, ohne die in Vorexperimenten kein stabiles Training möglich war.

Initialisierung

Von dem DC-GAN-Ansatz von Radford et al. (2015) übernommen, werden die erlernbaren Modellparameter für Faltungen aus zufälligen Werten aus $\mathcal{N}(0, 0.02)$ initialisiert. Für den erlernbaren Skalierungsfaktor bei der Normalisierung wird ein zufälliger Wert aus $\mathcal{N}(1, 0.02)$ gewählt. Alle Versatzwerte (bias) werden mit 0 initialisiert.

Trainingsparameter und Verlustfunktion

Wie bei dem Training der Disaggregationsnetze werden alle GAN-Ansätze mit dem A Method for Stochastic Optimization (ADAM)-Optimierer für 100'000 Iterationen mit einer initialen

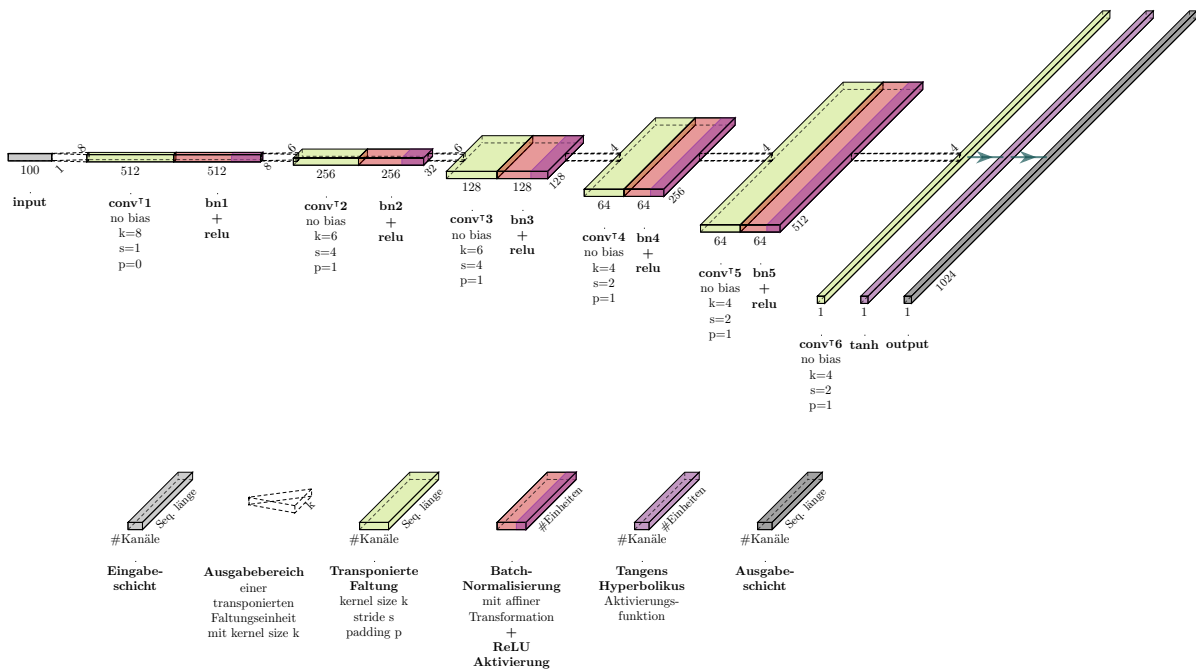


Abbildung 6.4: Netzarchitektur des Generators basierend auf dem Ansatz Deep Convolutional Generative Adversarial Network (DC-GAN).

Lernrate von 10^{-4} trainiert. Nach 50'000 Iterationen wird die Lernrate auf 10^{-5} , nach 25'000 weiteren Iterationen auf 10^{-6} reduziert.

Als Verlustfunktion für den Diskriminator wird der Negative Log-Likelihood bzw. die Binary Cross Entropy für Logits verwendet, welches die Sigmoid-Aktivierungsfunktion in der letzten Schicht implizit mitberechnet und ersetzt. Das Generatortraining nutzt indirekt über den Diskriminator die gleiche Verlustfunktion.

6.1.5 Trainingskonvergenz bei alternativen Netzmodellen

Das GAN-Training ist im Allgemeinen schwer kontrollierbar, da die gegenseitige Beeinflussung des Generators und des Diskriminators beide Netze in einen unerwünschten Zustand konvergieren lassen kann, in denen die Netze nicht ihre angedachte Aufgabe erfüllen.

In Abbildung 6.5 ist beispielsweise ein instabiler Trainingsdurchgang dargestellt, bei dem ein Generator mit fünf Faltungsblöcken verwendet wird. Die Abbildung zeigt für sieben zu Beginn des Trainings zufällig gewählte Vektoren aus $\mathcal{N}(0, 1)$, welche Ergebnisse der Generator im Verlauf des Trainings daraus erstellt. Nach 10'000 Iterationen sind waschmaschinenartige Lastgänge erkennbar, jedoch destabilisiert sich das Training nach insgesamt 25'000

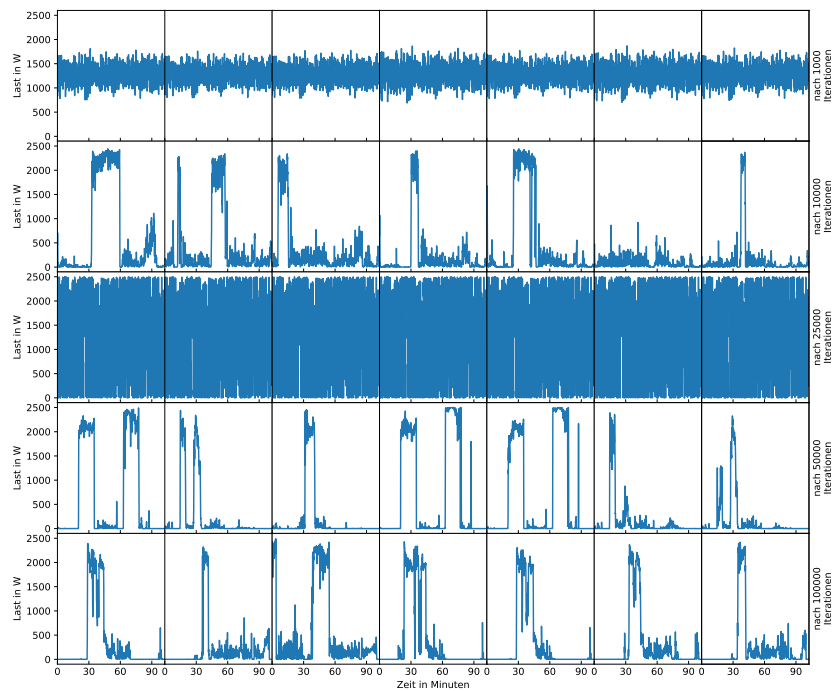


Abbildung 6.5: Generierte Lastgänge eines DC-GAN mit 5 Faltungsblöcken im Verlauf des Trainings.

Iterationen und der Generator erzeugt in Folge nur noch scheinbar zufällige Werte. Bis zum Abschluss des Trainings werden wieder waschmaschinenartige Lastgänge erzeugt, jedoch besitzen die Lastgänge keine befriedigende Qualität.

Bei Experimenten, einen Generator zu erstellen, wurde für die zuvor beschriebene Netzarchitektur versucht, die Batch-Normalisierung durch Instanz-Normalisierung (siehe Abschnitt 2.5.5) zu ersetzen oder auch vollständig auf die Normalisierung zu verzichten. Der Trainingsverlauf war jedoch nicht stabil und nach dem Training konnten keine erkennbaren Waschmaschinenlastgänge erzeugt werden. Auch der Austausch der letzten Aktivierungsfunktion mit einer linearen Aktivierung oder ReLU-Aktivierung erwies sich nicht als erfolgreich.

Die zuvor beschriebenen Netzarchitekturen und Trainingsparameter haben sich für den verwendeten Datensatz jedoch als zuverlässig erwiesen.

Ein erfolgreiches Training ist in Abbildung 6.6 dargestellt: Für die sieben zufällig aus $\mathcal{N}(0, 1)$ gewählten Vektoren werden waschmaschinenartige Lastgänge erzeugt, die qualitativ bereits sehr ähnlich zu tatsächlich gemessenen Lastgängen sind.

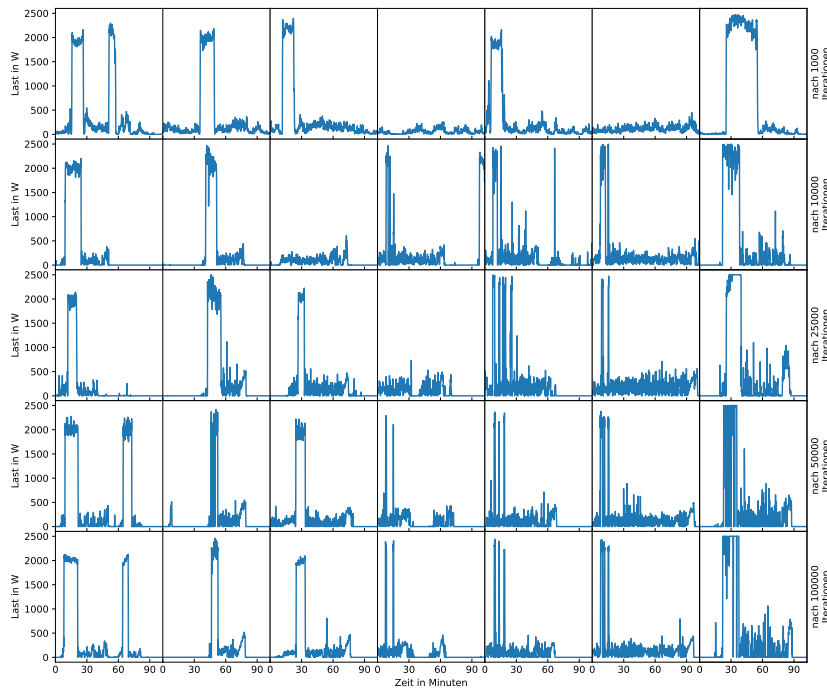


Abbildung 6.6: Generierte Waschmaschinellastgänge der beschriebenen DC-GAN-Architektur im Verlauf des Trainings.

6.1.6 Wasserstein GAN

Da bereits geringfügige Änderungen an der Architektur zu unzuverlässigen Trainingsverhalten führen kann, wird neben dem für DC-GAN eingesetzten konventionellem Trainingsverfahren die Trainingsmethode Wasserstein-GAN (W-GAN) (Arjovsky et al., 2017) verwendet.

Dabei wird die gleiche Netzarchitektur für Generator und Diskriminator benutzt. Das Diskriminatornetz wird jedoch nicht mehr als Klassifikator genutzt, sondern als Schätzer für die Wasserstein-Distanzmetrik zwischen der Verteilung der Trainingsbeispiele und der Verteilung der generierten Daten. Die Sigmoid-Aktivierungsfunktion im letzten Schritt des Diskriminatornetzes entfällt. Intuitiv gibt die Wassersteindistanz oder Earth-Mover-Distanz an, wieviel Arbeit in Form von Verschieben von Wahrscheinlichkeitsmasse mindestens aufgebracht werden muss, um eine Wahrscheinlichkeitsverteilung in eine andere zu überführen (siehe Arjovsky et al., 2017).

Der Diskriminator wird so trainiert, dass dessen Ausgabewert für Trainingsbeispiele maximiert wird und für generierte Beispiele minimiert wird. Nach jedem Gradientenabstieg werden die Modellparameter auf ein festes Intervall begrenzt, womit das Diskriminatornetz zu einer K -Lipschitz begrenzten Funktion wird, wobei K von dem Intervall abhängt, der

konkrete Wert jedoch nicht relevant ist. Mit diesem Aufbau nähert sich der Diskriminator einer Funktion an, die proportional zur gesuchten Distanzmetrik ist. (siehe Arjovsky et al., 2017)

Zum Optimieren des Generators wird die vom Diskriminator geschätzte Distanz minimiert. In Kombination mit der harten Begrenzung der Modellparameter führt die Minimierung zur Annäherung der Wassersteindistanz. (siehe Arjovsky et al., 2017).

Die Wassersteindistanz bietet stabile Gradienten, sodass das Training häufiger konvergiert. Jedoch ist die Abschätzung dieser Distanzfunktion schwer, sodass in jeder Iteration der Diskriminator häufiger optimiert wird als der Generator (siehe Arjovsky et al., 2017). In den folgenden Experimenten wird der Diskriminator pro Iteration fünf Mal optimiert, der Generator nur ein Mal pro Iteration.

6.1.7 Validierung des Gerätelastprofilgenerators

Für die zuvor erstellten Generatoren ist es schwierig zu bestimmen, wie ähnlich die generierten Lastgänge zu tatsächlich realisierbaren Waschmaschinenlastgängen sind und ob tatsächlich der vollständige Raum aller möglichen Waschmaschinenlastgänge abgedeckt ist.

Hilfsweise kann jedoch geprüft werden, ob die Generatoren dazu fähig sind, die Gerätelastsequenzen aus der Testdatenmenge nachzubilden. Dazu wird ein weiteres neuronales Netz trainiert, das die Umkehrabbildung des Generators annähern soll und für eine gegebene Gerätelastsequenz die passende Zwischenrepräsentation ausgibt. Wie in Abbildung 6.7 abgebildet, wird statt des Disaggregators dieses neuronale Netz als Lastsequenzkodierer eingesetzt. Statt Haushaltslastsequenzen werden Gerätelastsequenzen eingelesen, sodass als Aufgabe lediglich die Enkodierung der Lastsequenz zu bewältigen ist.

Training des Lastsequenzkodierers

Als Lastsequenzkodierer wird eine Variation des Diskriminatornetzes verwendet. Wie in Abbildung 6.8 dargestellt, wird eine 100-dimensionale Zwischenrepräsentation ausgegeben und die Sigmoid-Aktivierungsfunktion am Ende entfällt.

Der Lastsequenzkodierer wird auf der *Testdatenmenge* trainiert, da nur die Existenz einer Zwischenrepräsentation nachgewiesen werden soll, mit dem der Generator hinreichend genau die Testdaten reproduzieren kann.

Zum Training wird die vollständige Testdatenmenge mit allen Testgebäuden mit einem gleitenden Fenster mit Schrittweite 16 (1,6 Minuten) in Lastsequenzen aufgeteilt. Um das

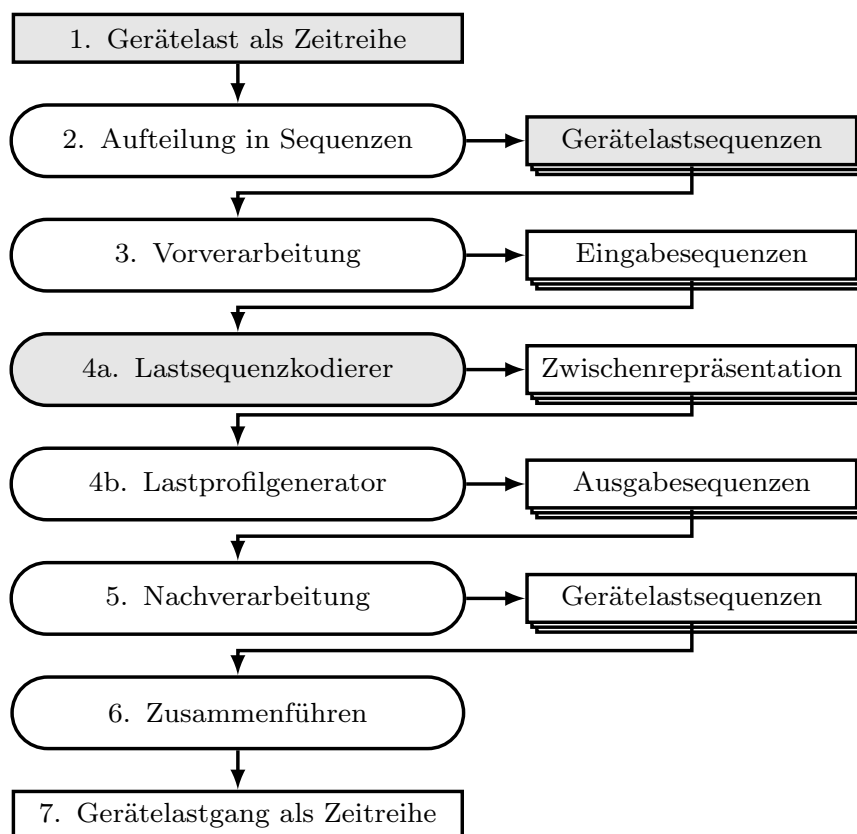


Abbildung 6.7: Methode zur Validierung des Gerätelastsequenzgenerators.

6.1 Ansatz A: Unüberwacht trainierter Generator für Gerätelestgänge

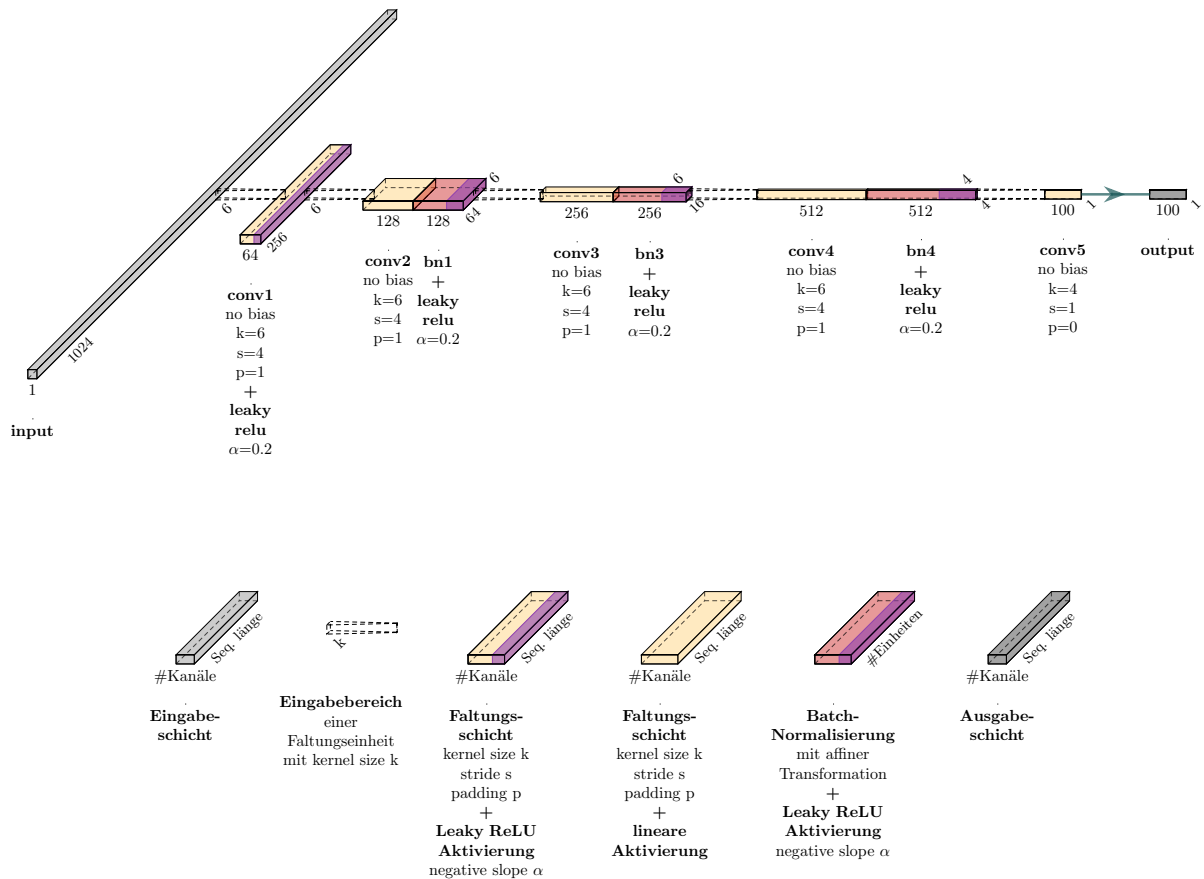


Abbildung 6.8: Netzarchitektur des Lastsequenzkodierers zur Validierung des Gerätelestproflgenerators.

Training zu beschleunigen, werden nur die Lastsequenzen verwendet, in denen eine Geräteaktivität zu erkennen ist. Als Ersatz sind 15'000 Lastsequenzen ohne Geräteaktivität in der Trainingsmenge enthalten, was etwa 10% der Trainingsmenge entspricht. Mit einer Mini-Batch-Größe von 256 Lastsequenzen ist die Trainingsmenge nach etwa 650 Iterationen vollständig verarbeitet. Zu Beginn jeder Epoche werden die Lastsequenzen zufällig durchmischt.

Das Training des Lastsequenzkodierers erfolgt mit dem ADAM-Optimierer für 100 Epochen mit einer initialen Lernrate von 10^{-4} . Nach 50 Epochen wird die Lernrate auf 10^{-5} , nach 25 weiteren Epochen auf 10^{-6} reduziert.

Als Verlustfunktion wird die L_2 -Metrik verwendet.

Ergebnisse

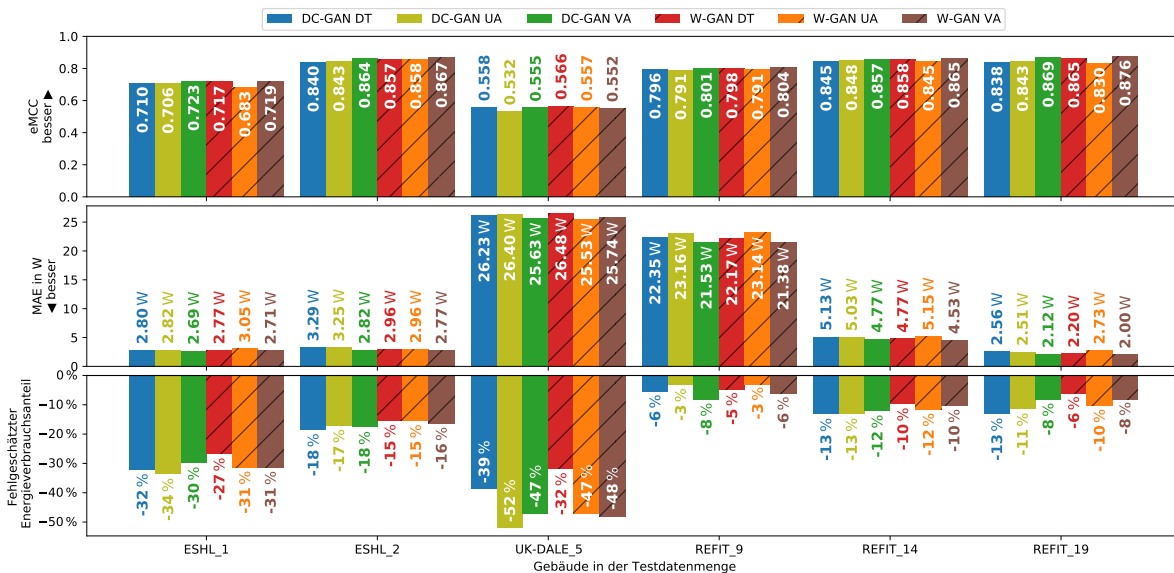


Abbildung 6.9: Fehlermetriken für die reine Generierung von Waschmaschinenlastgängen auf dem Testdatensatz.

Wie bei der Bewertung der Disaggregation von vorhandenen Ansätzen in Abschnitt 5.2 wird die Qualität der Lastsequenzreproduktion mit den Metriken (siehe Abschnitt 4.8) energiebasierter Matthews Correlation Coefficient (eMCC), Mittlerer absoluter Fehler (MAE) und relativer Fehler bei der Gesamtenergieverbrauchsschätzung (RETE) bewertet.

Das Ergebnis des Vergleichs zwischen reproduziertem Gerätelastgang und ursprünglichem Lastgang ist in Abbildung 6.9 dargestellt. Die Suffixe DT, UA und VA stehen dabei für die verwendeten Trainingsdaten.

Die Korrelationswerte liegen, mit Ausnahme des Testgebäudes UK-DALE 5, auf einem hohen Niveau. Auch der mittlere Fehler ist bei den meisten Testgebäuden im niedrigen einstelligen Bereich. Dabei bilden jedoch die Testgebäude UK-DALE 5 und REFIT 9 eine Ausnahme mit verhältnismäßig hohen MAE. Der in allen Fällen negative RETE zeigt an, dass zu geringe Lasten ausgegeben werden.

Im Vergleich zwischen den Trainingsdatenerzeugungsmethoden erreicht der Ansatz, ausschließlich mit vollständigen Aktivierungen zu trainieren, im Schnitt die beste Bewertung (hohe eMCC und niedrige MAE).

Die mittels Wasserstein-GAN erstellten Generatoren können die Lastgänge geringfügig genauer nachbilden.

Analyse

Die hohen Korrelationswerte und die niedrigen absoluten Fehler zeigen zunächst an, dass sich mithilfe eines Lastgangsgenerators Waschmaschinenlastgänge näherungsweise reproduzieren lassen können. Jedoch bleibt überall ein deutlicher Abstand zur Höchstpunktzahl 1.0. Dies zeigt an, dass die Gerätelastgangsnachbildung nicht perfekt ist.

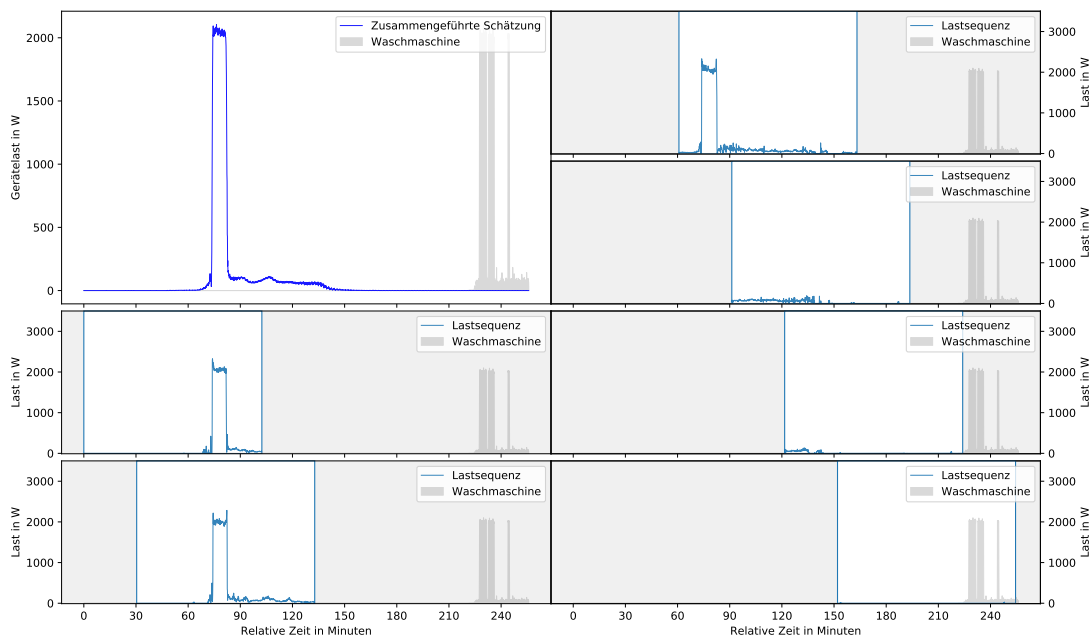


Abbildung 6.10: Beispiel für die Reproduktion von Waschmaschinenlastgängen in ESHL 1 mit W-GAN, das mit vollständigen Aktivierungen trainiert wurde.

Weiteren Aufschluss über die Lastsequenzgenerierung liefert ein Beispielabschnitt aus dem Testgebäude ESHL 1 in Abbildung 6.10. Die Abbildung stellt die Reproduktion eines Lastgangsausschnitts mithilfe des W-GAN dar, welches mit vollständigen Aktivierungen trainiert wurde. In den kleinen Abbildungen sind einzelne, von dem Generator ausgegebene Lastsequenzen dargestellt. Die Lastsequenzen werden zu der in der großen Abbildungen dargestellten Endschätzung zusammengefasst.

Der erste Lastblock wird korrekt wiedergegeben, jedoch fehlen in den generierten Lastsequenzen der kurze Leistungseinbruch im ersten Lastblock sowie die zwei kurzen Lastspitzen zum späteren Zeitpunkt. In den generierten Lastsequenzen ist die leicht erhöhte Last während des Schleudergangs nicht eindeutig wiedererkennbar.

Die einzelnen Lastsequenzen scheinen untereinander nicht konsistent zu sein, denn der zusammengeführte Lastgang ist nicht als typischer Waschmaschinenlastgang erkennbar.

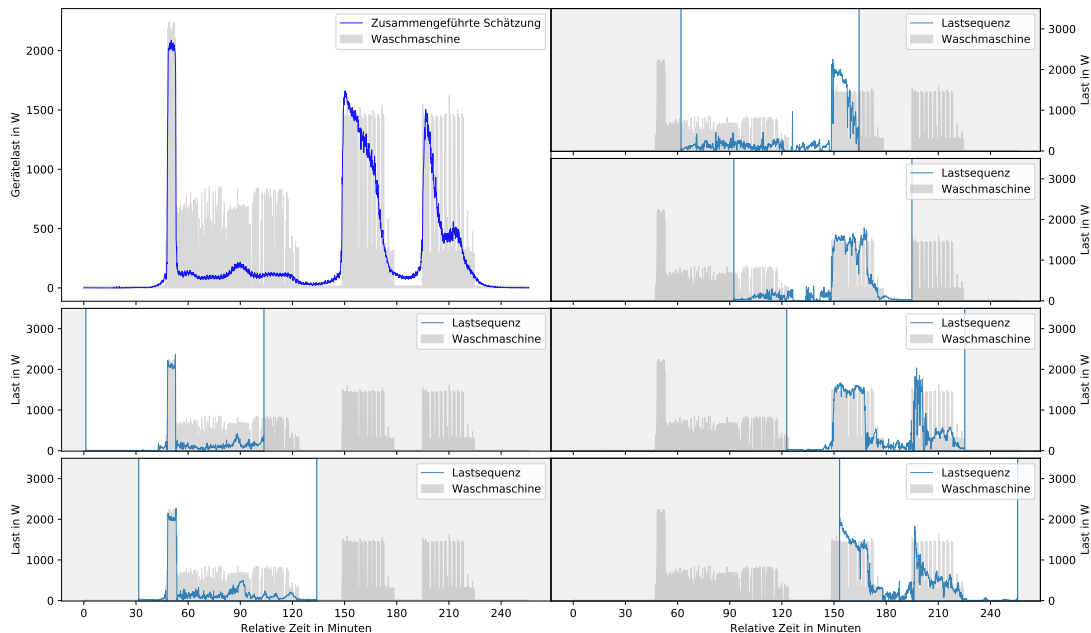


Abbildung 6.11: Beispiel für die Reproduktion von Waschmaschinenlastgängen in UK-DALE 5 mit W-GAN, das mit vollständigen Aktivierungen trainiert wurde.

In Abbildung 6.11 ist dargestellt, wie ein Abschnitt aus dem Testgebäude UK-DALE 5 reproduziert wird. Der Lastgang lässt auf ein Waschprogramm mit zwei anschließenden Trockenvorgängen schließen. Das Waschprogramm wird durch die einzelnen Lastsequenzen im Mittel reproduziert, jedoch können die betragsmäßig hohen Lastsprünge nicht nachgefahren werden. Die Lastgänge der Trockenvorgänge können nicht reproduziert werden. Dabei ähneln die ausgegeben Lastsequenzen keinen typischen Waschmaschinenlastgängen.

6.1 Ansatz A: Unüberwacht trainierter Generator für Gerätelastgänge

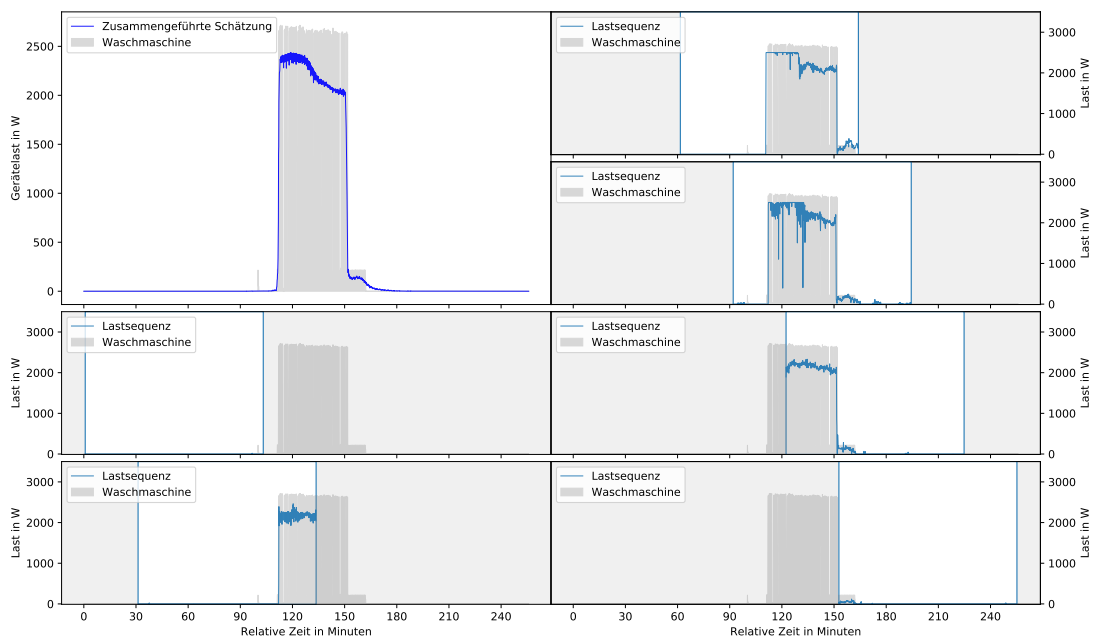


Abbildung 6.12: Beispiel für die Reproduktion von Waschmaschinenlastgängen in REFIT 9 mit W-GAN, das mit vollständigen Aktivierungen trainiert wurde.

Ein weiteres Beispiel, wie das W-GAN, das mit vollständigen Aktivierungen trainiert wurde, die Waschmaschinenlastgänge des Testgebäudes REFIT 9 reproduziert, ist in Abbildung 6.12 dargestellt. Wie in den vorigen Fällen tritt nicht der gewünschte Effekt ein, dass die einzelnen Lastsequenzen stets realisierbare Waschmaschinenlastgänge sind. Im direkten Vergleich zu dem tatsächlichen Lastgang werden die häufigen Lastsprünge zwischen 250 W und 2600 W nicht wiedergegeben. Zudem liegt die Maximallast dieser Aktivierung über der angenommenen Maximallast von 2500 W und kann von dem Generator nicht erzeugt werden.

Im Gegensatz zu den von Bao, Ibrahimov et al. (2018) beschriebenen ersten Experimenten gelingt es hier sowohl mit dem DC-GAN als auch mit dem W-GAN-Ansatz, Lastgangssequenzen ohne Geräteaktivität zu erzeugen. Daher entfällt die Notwendigkeit, eine GAN-Variante wie dem Auxiliary Classifier GAN (Odena et al., 2016) zu verwenden, um gezielt Lastgänge ohne Geräteaktivität reproduzieren zu können.

Zusammenfassend zeigen die Ergebnisse, dass durch den Einsatz eines Lastsequenzgenerators der Raum der möglichen Ausgabesequenzen eingeschränkt wird. Wie die beispielhaft dargestellten Lastsequenzen bereits zeigen, erfolgt jedoch die Einschränkung nicht genau auf den Raum realisierbarer Gerätelastgänge, da in diesen Beispielen der Generator dazu gebracht werden kann, auch nicht realisierbare Lastgänge zu erstellen. Im nächsten Schritt wird untersucht, ob ein solcher Generator trotzdem für eine Disaggregation hilfreich ist.

6.1.8 Disaggregation mit Gerätelastsequenzgeneratoren

Für die Evaluierung, ob eine durch den Lastsequenzgenerator erzwungene Einschränkung des Ausgaberaums für eine Disaggregation hilfreich ist, wird nun, wie in Abbildung 6.1 dargestellt, ein Gerätelastsequenzgenerator in den Disaggregationsprozess integriert.

Da der Generator eine Lastsequenz ausgibt, sind Generatoren nur in die Ansätze integrierbar, die ebenfalls Lastsequenzen ausgeben. Der Disaggregator muss jeweils modifiziert und neu trainiert werden. Für diese Evaluation wird der Denoising Autoencoder (DAE) von Kelly und Knottenbelt (2015a) und der Sequence-to-Sequence-Ansatz (S2S) von Zhang et al. (2016) verwendet (siehe Abschnitt 5.1).

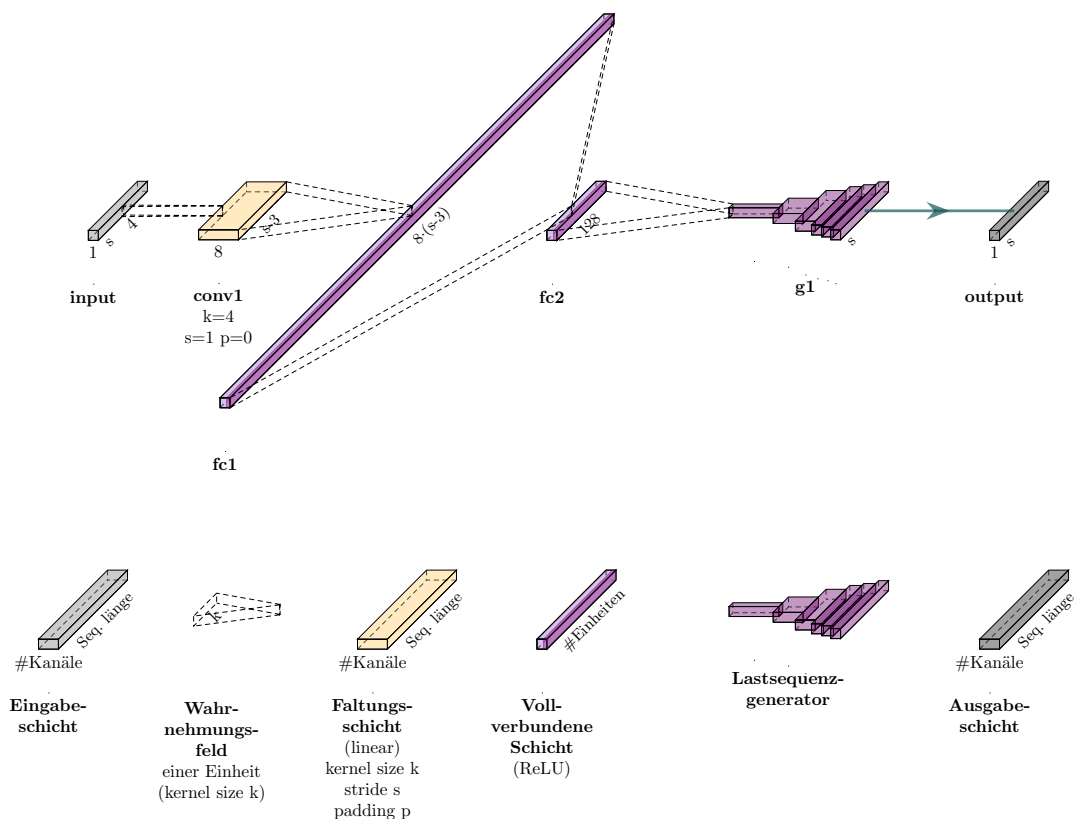


Abbildung 6.13: Netzarchitektur des Denoising Autoencoders mit integriertem Lastsequenzgenerator.

Die Architektur des Denoising Autoencoders bleibt, wie in Abbildung 6.13 dargestellt, bis zum Flaschenhals unverändert. Danach wird die 128-dimensionale Zwischenrepräsentation auf die 100-dimensionale Zwischenrepräsentation des Generators abgebildet.

Die Netzarchitektur des Sequence-to-Sequence-Ansatzes wird, wie in Abbildung 6.14 dargestellt, zunächst so verbreitert, damit eine 102,4-minütige Lastsequenz eingelesen und

6.1 Ansatz A: Unüberwacht trainierter Generator für Gerätelastgänge

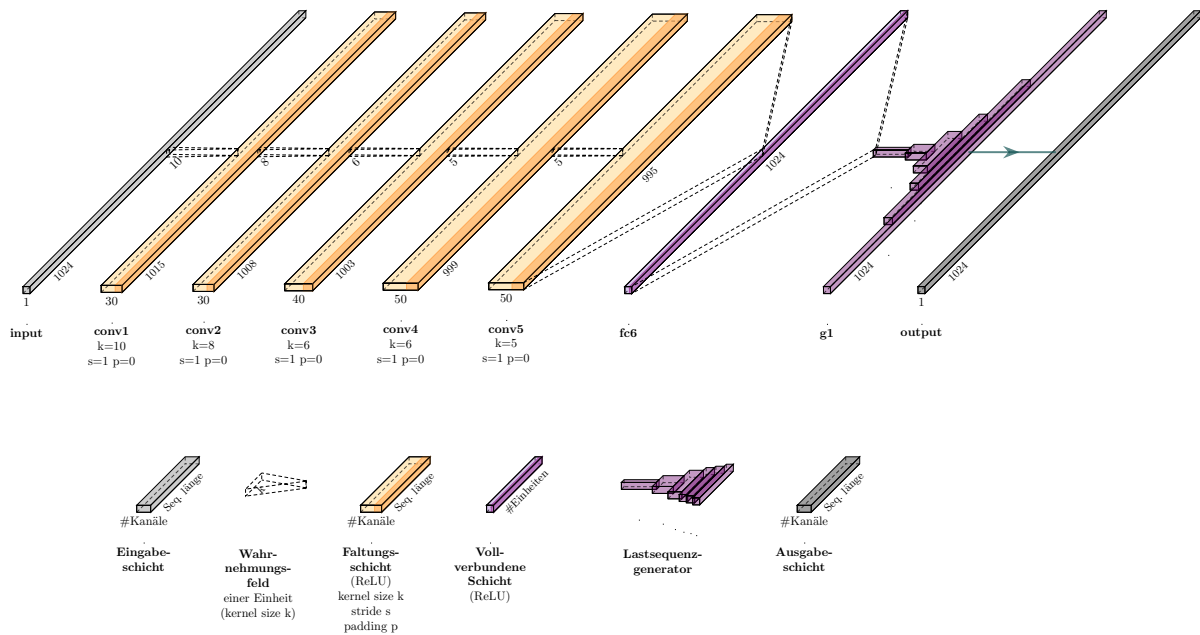


Abbildung 6.14: Netzarchitektur des Sequence-to-Sequence-Ansatz mit integriertem Lastsequenzgenerator.

ausgegeben werden kann. Am Ende wird anstelle der Lastsequenz die 100-dimensionale Zwischenrepräsentation des Generators ausgegeben.

Das Training der Netze erfolgt identisch mit dem in Abschnitt 5.1 angegebenen Parametern mit den modifizierten Netzen. Während des Trainings bleibt das Generatornetz unverändert.

Im Folgenden werden ausschließlich diejenigen Generatoren verwendet, die mit vollständigen Aktivierungen trainiert wurden, da mit diesen Generatoren die Gerätelastgänge genauer reproduziert werden können.

Ergebnisse

In Abbildung 6.15 sind die Fehlermetriken energiebasierter Matthews Correlation Coefficient (eMCC), Mittlerer absoluter Fehler (MAE) und relativer Fehler bei der Gesamtenergieverbrauchsschätzung (RETE) für die Disaggregation mit und ohne Lastsequenzgeneratoren dargestellt.

Der mittlere absolute Fehler ist mit dem Einsatz eines Lastsequenzgenerators stets höher. Bezüglich des Korrelationskoeffizienten verbessert sich dieser in ESHL 1 und UK-DALE 5, wenn der DC-GAN-basierte Lastsequenzgenerator verwendet wird. Im allgemeinen verringert sich jedoch der eMCC mit dem Einsatz eines Lastsequenzgenerators. Der Einsatz

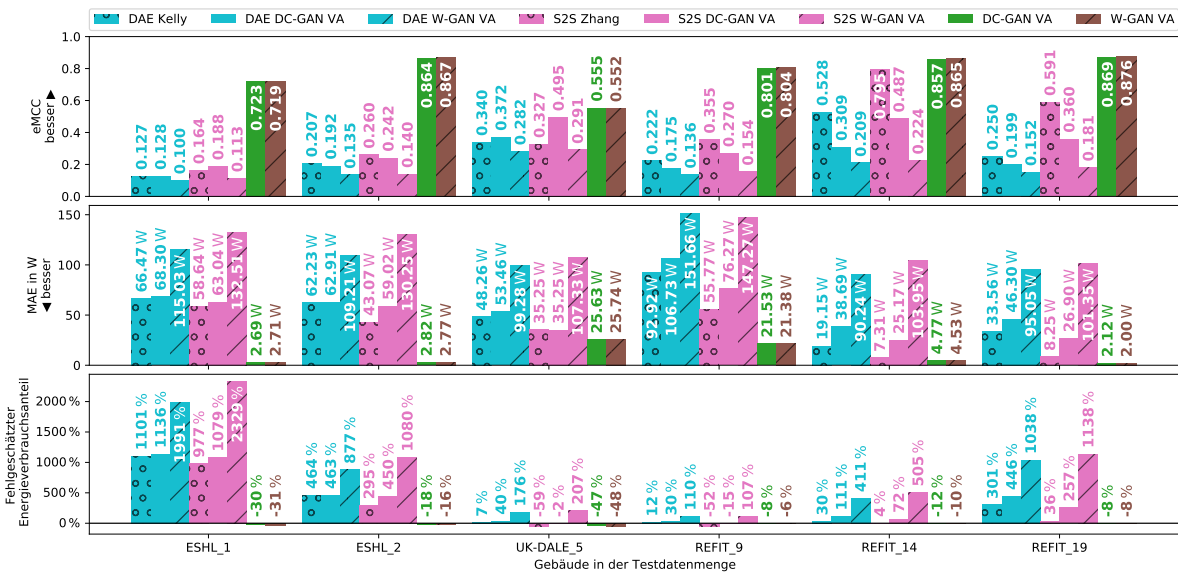


Abbildung 6.15: Fehlermetriken bei der Disaggregation von Waschmaschinenlastgängen mit Gerätelastsequenzgeneratoren.

des W-GAN-basierten Lastsequenzgenerators führt laut allen Metriken zu einer deutlich schlechteren Disaggregationsqualität.

Im Vergleich zu den Metrikwerten, die bei der reinen Reproduktion der Gerätelastsequenzen erreicht wurden, schöpfen die Disaggregationsansätze das Potential der Lastsequenzgeneratoren bei weitem nicht aus.

Analyse

In Abbildung 6.16 ist dargestellt, wie ein Abschnitt aus dem Testgebäude ESHL 1 disaggregiert wird. Neben der zusammengeführten Schätzung sind einige Lastsequenzen, die von dem Generator ausgegeben werden, dargestellt. Es zeigt sich, dass die generierten Lastsequenzen untypisch für Waschmaschinen sind.

Anhand dieses Beispiels zeigt sich, dass der Einsatz des Lastsequenzgenerators also nicht zu dem gewünschten Effekt führt, dass ausschließlich realisierbare Waschmaschinenlastgänge ausgegeben werden. Die Integration des Lastsequenzgenerators scheint das Training des Disaggregators zu erschweren und mindert die Disaggregationsqualität.

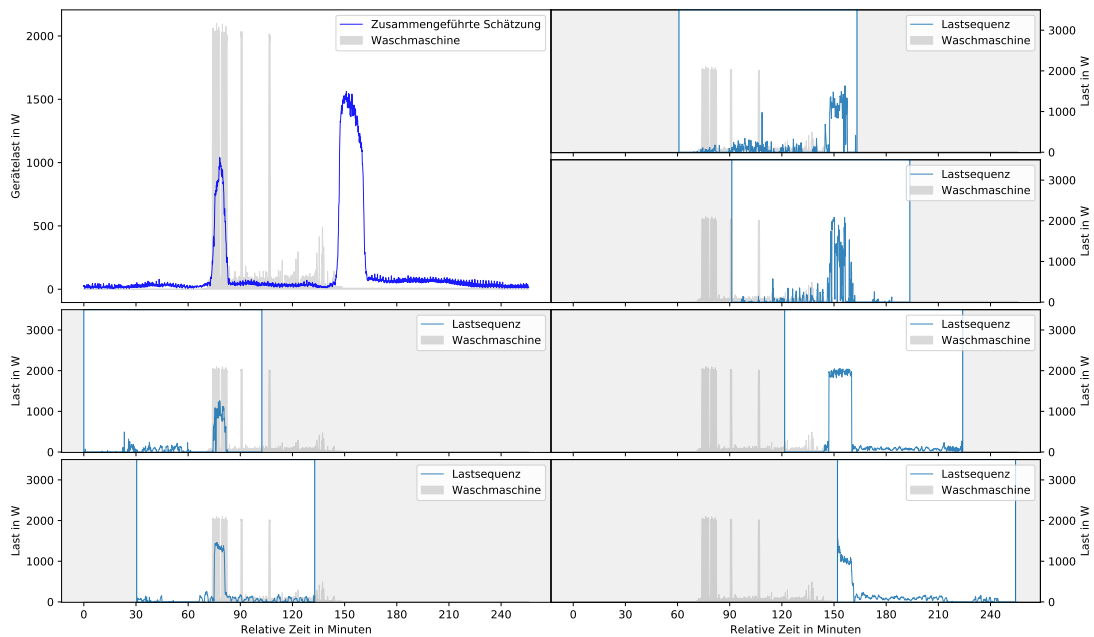


Abbildung 6.16: Beispielabschnitt von der Disaggregation von Waschmaschinenlastgängen in ESHL 1 mit DC-GAN, das mit vollständigen Aktivierungen trainiert wurde.

6.1.9 Schlussfolgerungen

Die Experimente in diesem Abschnitt haben gezeigt, dass mit dem GAN-Konzept ein Lastsequenzgenerator erstellt werden kann, der Waschmaschinenlastsequenzen näherungsweise erzeugen kann.

Da nicht alle Testdaten mit einem solchen Generator reproduziert werden können, wurde zwar gezeigt, dass der Ausgaberaum des Generators Einschränkungen unterliegt. Dennoch ist es durchaus möglich, mit den Generatoren offensichtlich nicht realisierbare Gerätelastgänge zu erstellen. Der anvisierte Effekt, dass der Generator ausschließlich realisierbare Gerätelastgänge ausgibt, ist nicht eingetreten.

Der Einsatz eines dedizierten Gerätelastprofilgenerators zum Zweck der Disaggregation führt im Endeffekt zu keiner Verbesserung der Disaggregationsqualität.

6.2 Ansatz B: Diskriminator für Gerätelastgänge

Durch die Integration eines dedizierten Lastsequenzgenerators in den Disaggregationsprozess werden im vorigen Ansatz keine Verbesserungen erzielt. Obwohl zufällig gezogene Zwischenrepräsentationsvektoren auf augenscheinlich realisierbare Lastsequenzen abgebildet werden, bewirkt der Lastsequenzgenerator letztendlich nicht die gewünschte Einschränkung der Ausgabe auf ausschließlich realisierbare Lastsequenzen.

Mit dem in diesem Abschnitt beschriebenen Ansatz wird weiterhin darauf abgezielt, die Ausgabehäufigkeit von realisierbaren Lastgängen zu erhöhen. Anstatt des Anhängens eines Lastsequenzgenerators während der Disaggregationsphase soll dies jedoch mit einer strukturierten Verlustfunktion während der Trainingsphase erreicht werden. Während des Trainings soll dabei der Diskriminator eines GAN eine erlernte Verlustfunktion bilden, die sowohl die Korrektheit der Disaggregation als auch die Realisierbarkeit der Gerätelastsequenzen bewertet.

Die Vorgehensweise ist von Isola et al. (2016) inspiriert, die diese Lösungsstrategie erfolgreich für die Transformation von Bilddaten eingesetzt haben. Sie demonstrieren u. a. die Übersetzung von Luftbilddaten in eine schematische Kartenrepräsentation und umgekehrt. Während der Generator die Übersetzung durchführt, erhält der Diskriminator das ursprüngliche Bild überlagert mit dem transformierten Bild und bewertet die Korrektheit der Transformation.

Hier wird diese Strategie auf die Disaggregation übertragen. Dabei wird ein disaggregierendes neuronales Netz eines vorhandenen Ansatzes (siehe Kapitel 3), wie in Abbildung 6.17 dargestellt, als Generator verwendet. Statt einem Zufallsvektor erhält der Generator den Haushaltslastgang als Quelle für Variationen. Der Diskriminator erhält den Haushaltslastgang und den disaggregierten Gerätelastgang und bewertet die Korrektheit der Disaggregation.

Wie bei Isola et al. (2016) wird eine kombinierte Verlustfunktion bestehend aus dem GAN-Klassifikationsverlust und einer konventionellen Verlustfunktion verwendet, die die durchschnittliche Wertedifferenz zwischen geschätzter und tatsächlicher Gerätelast abbildet. Der Wertedifferenzverlust wird mit einem Faktor α skaliert, damit der Wertebereich der Verlustwerte in der gleichen Größenordnung liegt.

6.2.1 Vorgehensweise

Als Basis für die Evaluation dieses Ansatzes werden zunächst die Referenzansätze Denoising Autoencoder von Kelly und Knottenbelt (2015a) und Sequence-to-Sequence von Zhang et al. (2016) verwendet. Weiterhin wird der im Schnitt beste Disaggregationsansatz, Dilated Fully

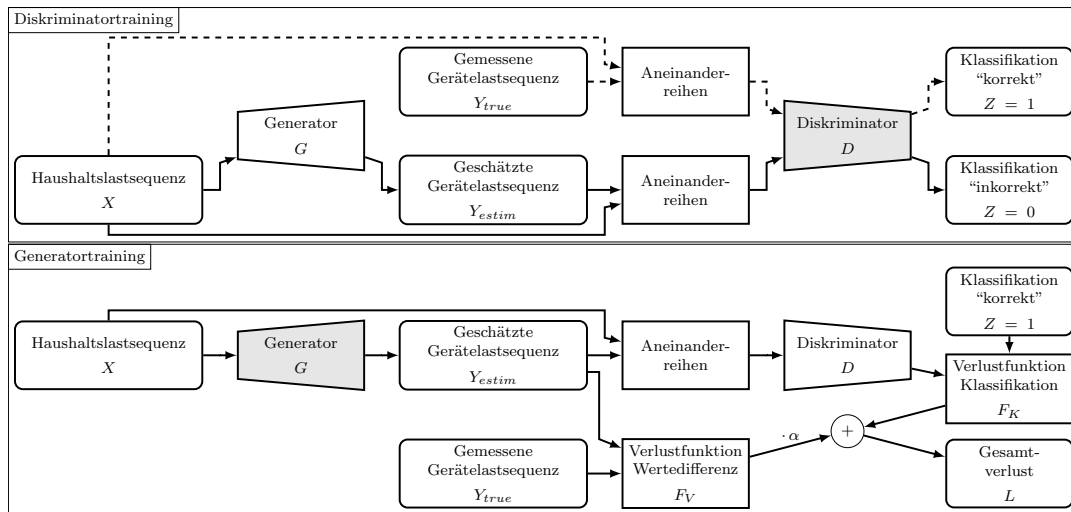


Abbildung 6.17: Training des Disaggregators mit einer GAN-basierten strukturierten Verlustfunktion.

Convolutional Network von Brewitt und Goddard (2018), als Grundlage genommen. (vgl. Kapitel 7)

Basierend auf den neuronalen Netzen zur Disaggregation wird ein jeweiliges Klassifikationsnetz entworfen, welches als Diskriminator verwendet wird. Als Generator wird das ursprüngliche neuronale Netz des Ansatzes eingesetzt.

Während des Trainingsprozesses werden abwechselnd der Diskriminator und der Generator trainiert. Der Generator wird darauf optimiert, Lastsequenzpaare bestehend aus Haushaltslast und gemessener Gerätelast von Lastsequenzpaaren aus Haushaltslast und geschätzter Gerätelast zu unterscheiden.

Damit ergeben sich einige Kombinationsmöglichkeiten für die endgültige Verlustfunktion. Letztendlich werden alle Ansätze mit folgenden kombinierten Verlustfunktionen evaluiert:

L_2 Alle bestehenden Ansätze aus der Literatur nutzen die L_2 -Verlustfunktion, um die geschätzte Gerätelastsequenz mit der gemessenen Gerätelastsequenz zu vergleichen (siehe Kapitel 3). Mit dieser Verlustfunktion gleicht dieser Aufbau den in der Literatur beschriebenen Ansätzen. Die Evaluationsergebnisse sind aus Kapitel 7 übernommen.

L_2 + GAN-Verlust Hier fließt neben der L_2 -Verlustfunktion die Bewertung des Diskriminators als weiterer Faktor in das Training des disaggregierenden neuronalen Netzes ein. Der L_2 -Verlust wird mit einem Faktor von $\alpha = 100$ skaliert, um die Größenordnung des L_2 -Verlustes an die des Klassifikationsverlustes anzugleichen.

- L₁** Isola et al. (2016) berichten, dass die Nutzung des L₁-Verlustes zu schärferen Bildern mit deutlicheren Kanten führt. Übertragen auf die Disaggregation sind auch hier klarere Lastsprünge in den geschätzten Lastgängen wünschenswert. Daher wird in den Experimenten statt der L₂-Verlustfunktion auch diese Variante evaluiert, bei dem die einzelnen Lastwerte mit der L₁-Metrik verglichen werden.
- L₁ + GAN-Verlust** Zur Vollständigkeit wird die Kombination von L₁-Verlust mit dem Klassifikationsverlust untersucht. Dabei wird der L₁-Verlust mit dem Faktor $\alpha = 100$ multipliziert, um die Größenordnung der beiden Verlustwerte anzugleichen.

6.2.2 Architektur der Diskriminatoren

Für die drei Ansätze orientieren sich die als Diskriminatoren verwendeten neuronalen Netze an der Architektur der ursprünglichen Verfahren. Der hintere Teil der Architektur wird jeweils so modifiziert, dass das gesamte Netz einen einzigen Wert ausgibt. Auf diesen Wert wird schließlich die Sigmoid-Aktivierungsfunktion angewandt.

Bei allen Netzmodellen werden ReLU-Aktivierungsfunktionen durch Leaky-ReLU ersetzt, da Vorexperimente gezeigt haben, dass der Diskriminator bereits nach wenigen Iterationen in einen Fehlerzustand gelangen kann, in dem es kaum mehr in der Lage ist, korrekte und inkorrekte Disaggregationen zu unterscheiden. Dies kann bis zum Ende des Trainings nicht mehr korrigiert werden.

Vermutlich hängt dies mit der Eigenschaft von ReLU zusammen, für negative Werte einen Gradienten von 0 aufzuweisen. Dadurch können Einheiten „absterben“, wenn für den kompletten Trainingsdatensatz die Aktivierung im negativen Bereich liegt. Die Parameter dieser Einheiten werden in Folge des Trainings nicht mehr verändert. Leaky-ReLU beseitigt dieses Problem, indem auch für negative Werte ein geringer Gradient garantiert wird.

Denoising Autoencoder

Die Netzarchitektur des Diskriminators für den Denoising Autoencoder von Kelly und Knottenbelt (2015a) ist in Abbildung 6.18 dargestellt. Nach dem Flaschenhals wird die zeitliche Dimension nicht erhöht. Stattdessen reduziert das Netz die Zwischenergebnisse auf einen einzelnen Wert und wendet die Sigmoid-Funktion an. Alle ReLU-Aktivierungen sind durch Leaky-ReLU ersetzt.

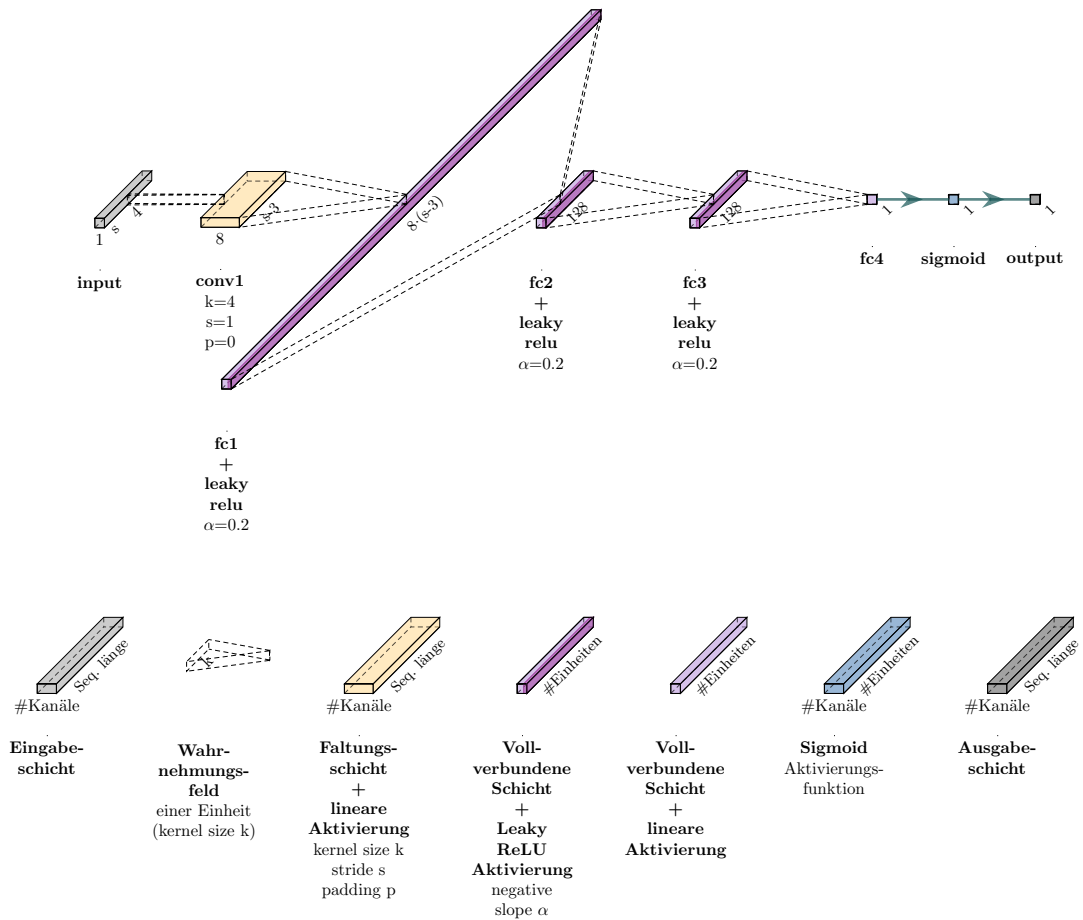


Abbildung 6.18: Netzarchitektur des Diskriminators für den Ansatz Denoising Autoencoder.

Sequence-to-Sequence

Die Netzarchitektur des Diskriminators für den Sequence-to-Sequence-Ansatz ist in Abbildung 6.19 dargestellt. Diese basiert auf der Architektur des Sequence-to-Point-Ansatzes von Zhang et al. (2016). ReLU wurde durch Leaky-ReLU ersetzt und am Ende wurde die Sigmoid-Aktivierungsfunktion angehängt.

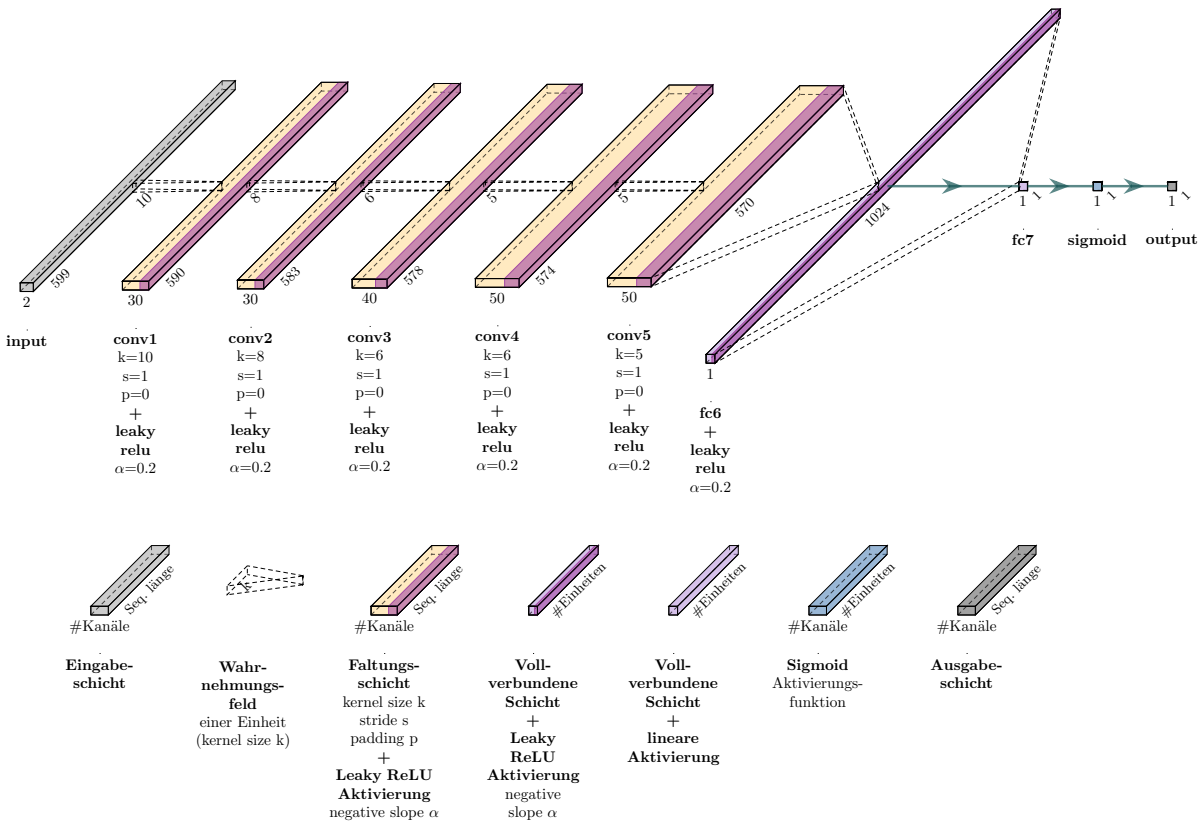


Abbildung 6.19: Netzarchitektur des Diskriminators für den Sequence-to-Sequence-Ansatz.

Dilated Fully Convolutional Network

Bei der Netzarchitektur des Diskriminators zum Training des Dilated Fully Convolutional Network (DFCN) werden die dilatierten Faltungen durch Faltungen mit doppelter Schrittgröße (stride) ersetzt. Dies reduziert den Rechenaufwand, da nur eine Bewertung für die Eingabesequenz durchgeführt werden soll. Die Berechnungen des Diskriminatornetzes sind vergleichbar mit denen des ursprünglichen Ansatzes für genau einen Ausgabewert. Das resultierende Netz ist in Abbildung 6.20 abgebildet.

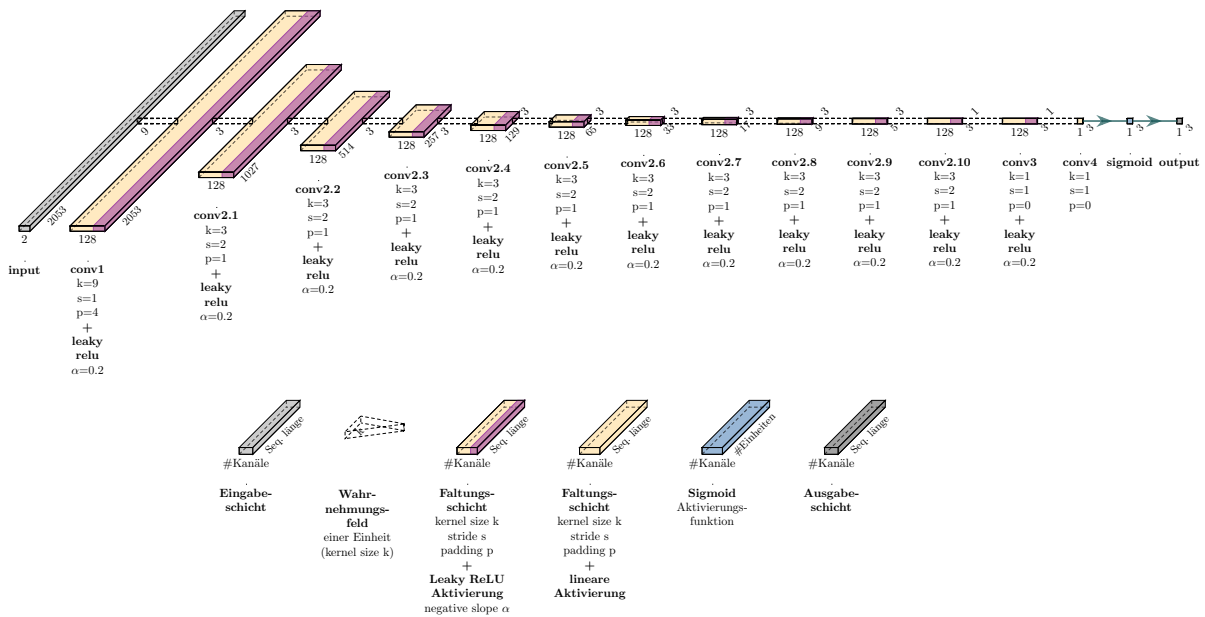


Abbildung 6.20: Netzarchitektur des Diskriminators für das Dilated Fully Convolutional Network.

Aufgrund der vergleichsweise großen Eingabelänge wird bei diesem Ansatz die Eingabesequenz in drei Bereiche unterteilt und das Netz bildet auf drei statt einem Wert ab. Dadurch bewertet der Diskriminator individuell für die drei Bereiche, ob die Disaggregation korrekt ist.

Wie bei den anderen Diskriminatornetzen wird ReLU durch Leaky-ReLU ersetzt und auf das Endergebnis wird die Sigmoid-Aktivierungsfunktion angewandt.

6.2.3 Trainingsparameter

Das Training der neuronalen Netze wird größtenteils analog zu Abschnitt 5.1 durchgeführt.

Für den Denoising Autoencoder wird Stochastic Gradient Descent mit Nesterov-Momentum verwendet, für die anderen Ansätze ADAM. Die anfängliche Lernrate liegt für den Denoising Autoencoder ohne Diskriminator bei 10^{-1} , bei Einsatz des Diskriminators bei 10^{-3} . Für den Sequence-to-Sequence-Ansatz wird immer eine initiale Lernrate von 10^{-3} verwendet, für das Dilated Fully Convolutional Network in allen Varianten 10^{-4} .

Alle Ansätze durchlaufen 100'000 Trainingsiterationen, wobei in Iteration 50'000 und 75'000 die Lernrate jeweils um Faktor 10 reduziert wird. Es werden die gleichen Trainingsdaten wie in Kapitel 4 beschriebenen verwendet.

Zur Initialisierung der Gewichtsmatrizen bei Ansätzen, die mit einem Diskriminator trainiert werden, wird eine Normalverteilung nach dem Verfahren von Glorot und Bengio (2010) verwendet. Bei den Ansätzen ohne Diskriminator bleibt die Initialisierungsmethode unverändert zu der in der Literatur beschriebenen Parametern (siehe Kapitel 4).

Zur Verbesserung der Trainingsstabilität werden bei Ansätzen mit Diskriminator übermäßig große Gradienten normiert und somit begrenzt. Unter Beibehaltung der Richtung werden Gradienten mit einer L_2 -Norm über 100 auf eine Maximallänge von 100 gekürzt. Dieses *Gradient Clipping* genannte Verfahren vermeidet große Veränderungen der Netzparameter und sorgt damit für eine höhere Trainingsstabilität. (vgl. Goodfellow, Bengio et al., 2016, Kapitel 10.11.1)

6.2.4 Ergebnisse

Wie bereits in Kapitel 5 festgestellt, hängen die Disaggregationsergebnisse besonders von der generellen Disaggregierbarkeit der Geräteklassen und von den Gerätemodellen im jeweiligen Testgebäude ab. Daher werden die Ergebnisse im Folgenden für jede Geräteklasse einzeln diskutiert. Die Durchschnittsmetrik über alle Geräteklassen ist damit nur mit Vorsicht zu interpretieren.

In den folgenden Graphiken werden die einzelnen Metrikwerte nach Geräteklasse und Testgebäude gruppiert als Punkte dargestellt. Der Mittelwert je Geräteklasse über alle Testgebäude wird durch einen diamantförmigen Punkt repräsentiert. Die vollständigen Ergebnisse sind im Anhang in Abschnitt A.1 aufgeführt.

Denoising Autoencoder

Abbildung 6.21 stellt den energiebasierten Matthews Correlation Coefficient (eMCC) und den Mean Absolute Error (MAE) dar, wenn der Denoising Autoencoder (DAE) mit alternativen Verlustfunktionen trainiert wird.

Durch den Einsatz der L_1 -Verlustfunktion wird bei 5 der 7 Geräteklassen der eMCC-Wert im Durchschnitt erhöht und dabei gleichzeitig der MAE reduziert. Nur bei Waschmaschine und bei der TV- und Unterhaltungselektronik verschlechtert sich der Korrelationskoeffizient, obwohl der mittlere absolute Fehler sinkt.

Die Hinzunahme des Diskriminatoransatzes wirkt sich nur in einzelnen Fällen positiv auf die Disaggregationsqualität aus. Kühlgeräte und Mikrowellen können dadurch besser erkannt werden. Anhand der stärkeren Steuerung der eMCC-Werte bei Ansätzen mit Diskriminator wird deutlich, dass Mikrowellenmodelle offenbar stärker differenziert werden. In einem Testgebäude wird ein eMCC-Wert von 0 berechnet, da das neuronale Netz keine Energie der

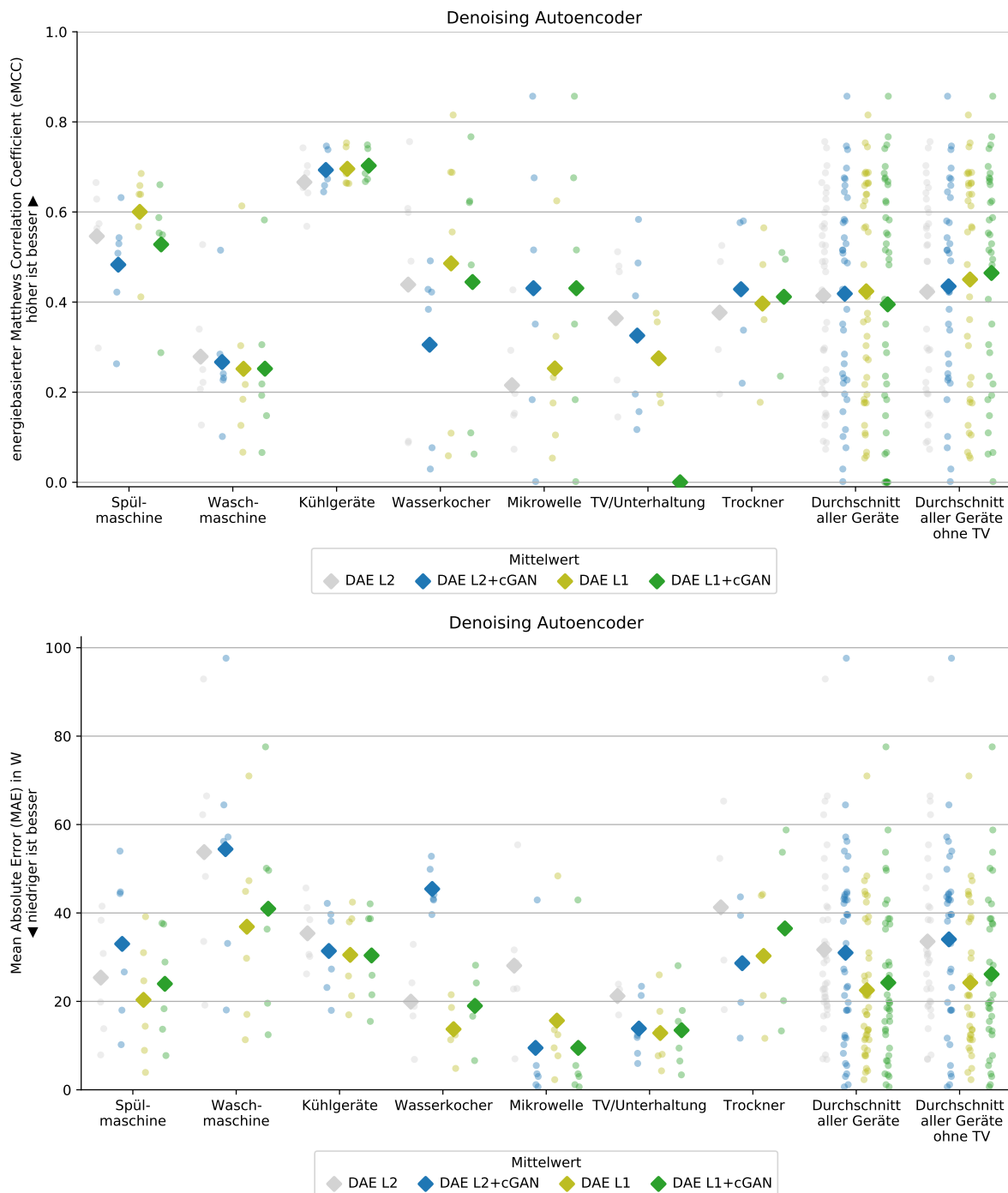


Abbildung 6.21: Metriken eMCC und MAE bei der Disaggregation mit dem Denoising Autoencoder, der mit alternativen Verlustfunktionen trainiert wurde.

Mikrowelle zugeordnet hat. Auch der MAE bewertet dieses Verhalten bei der Mikrowelle als vorteilhaft.

Das mit L_1 -Metrik und Diskriminator trainierte Netz erkennt in keinem Testgebäude Energieverbrauch von TV- und Unterhaltungselektronik. Dies führt zu einem eMCC-Wert von 0, obwohl der mittlere Fehler niedrig ist.

Der mittlere absolute Fehler verringert sich mit L_1 durchschnittlich um 32,6% und in der Kombination von L_1 mit Diskriminator um 25,8%. Der Einsatz von L_2 mit Diskriminator erhöht den Fehler bei der Mehrzahl der Geräte.

Mit keinem der alternativen Verlustfunktionen wird in allen Fällen ein niedrigerer Fehler erzielt. Im Mittel erzielt das Training mit der L_1 -Verlustfunktion die besten Ergebnisse.

Sequence-to-Sequence

Abbildung 6.22 stellt die Veränderung der Fehlermetriken bei der Verwendung von alternativen Verlustfunktionen mit dem Sequence-to-Sequence-Ansatz dar.

Das mit L_1 -Metrik trainierte Netz erreicht bessere Korrelationskoeffizienten bei 4 von 7 Geräten. Wird ein Diskriminator verwendet, sinkt in den meisten Fällen durch dessen Einsatz der Korrelationskoeffizient.

Der mittlere Fehler wird bei der Mehrheit der Geräte durch den Einsatz der alternativen Verlustfunktionen verringert.

Bei Wasserkochern führt die Verwendung jeder alternativen Verlustfunktion zu einer schlechteren Disaggregation. Wird bei dieser Geräteklasse die L_1 -Metrik verwendet, erkennt das trainierte Netz in den Testgebäuden keinen Energieverbrauch und dies führt zu einem eMCC-Wert von 0. Auch das mit L_1 -Metrik und Diskriminator trainierte Netz für TV- und Unterhaltungselektronik erkennt keine einzige Geräteaktivität in den Testgebäuden.

Bei Kühlgeräten wird, im Gegensatz zum Denoising Autoencoder, bei der Verwendung eines Diskriminators die Disaggregation von Kühlgeräten verschlechtert.

Mit dem L_1 -Verlust werden mit die besten Ergebnisse erzielt: Der mittlere Fehler ist im Vergleich zur ursprünglichen Variante um 11,9% reduziert und die meisten Geräte erreichen mit dieser Verlustfunktion höhere eMCC-Werte.

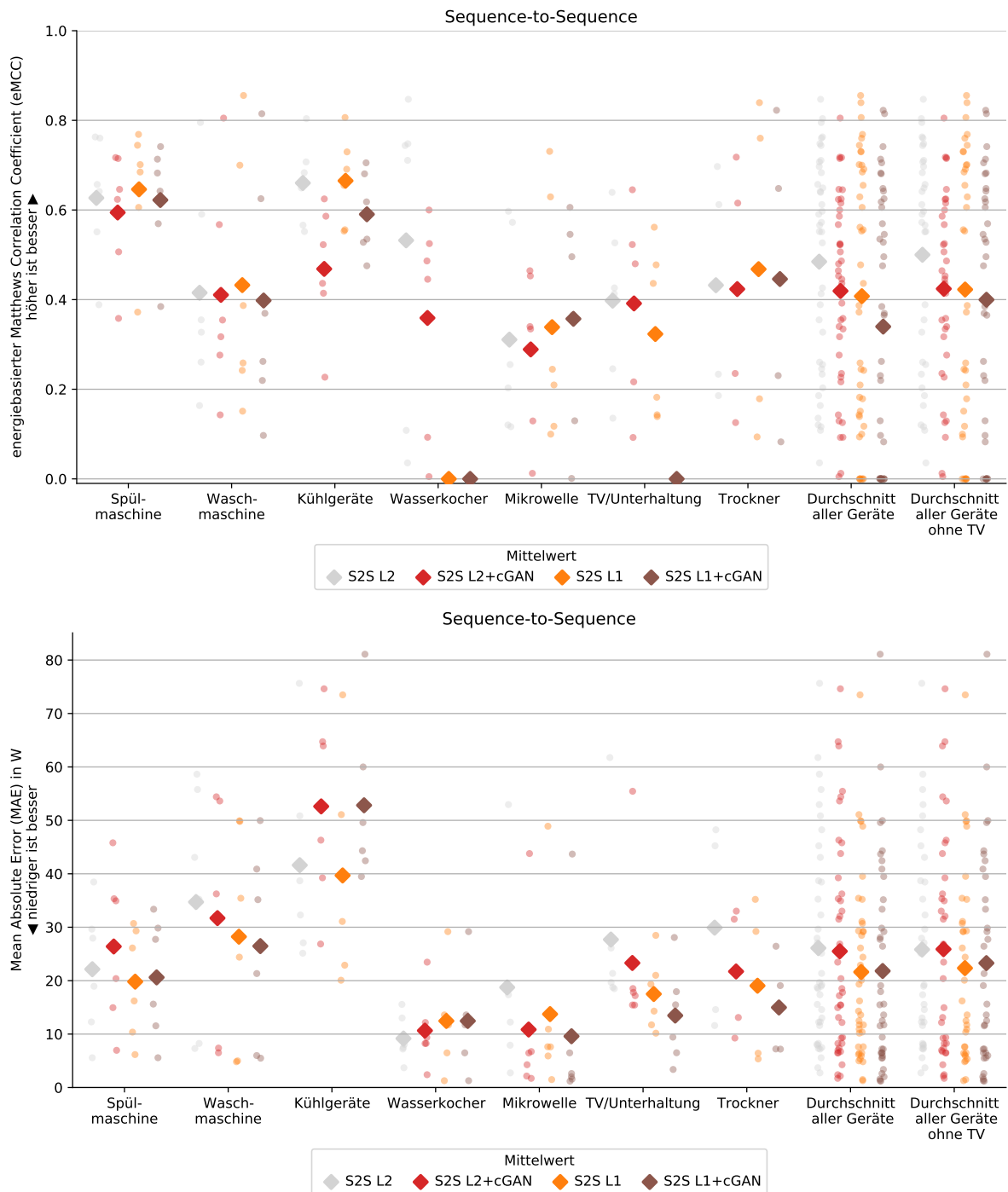


Abbildung 6.22: eMCC und MAE im Vergleich zum ursprünglichen Ansatz (rote Linie) bei der Disaggregation mit dem Sequence-to-Sequence-Ansatz.

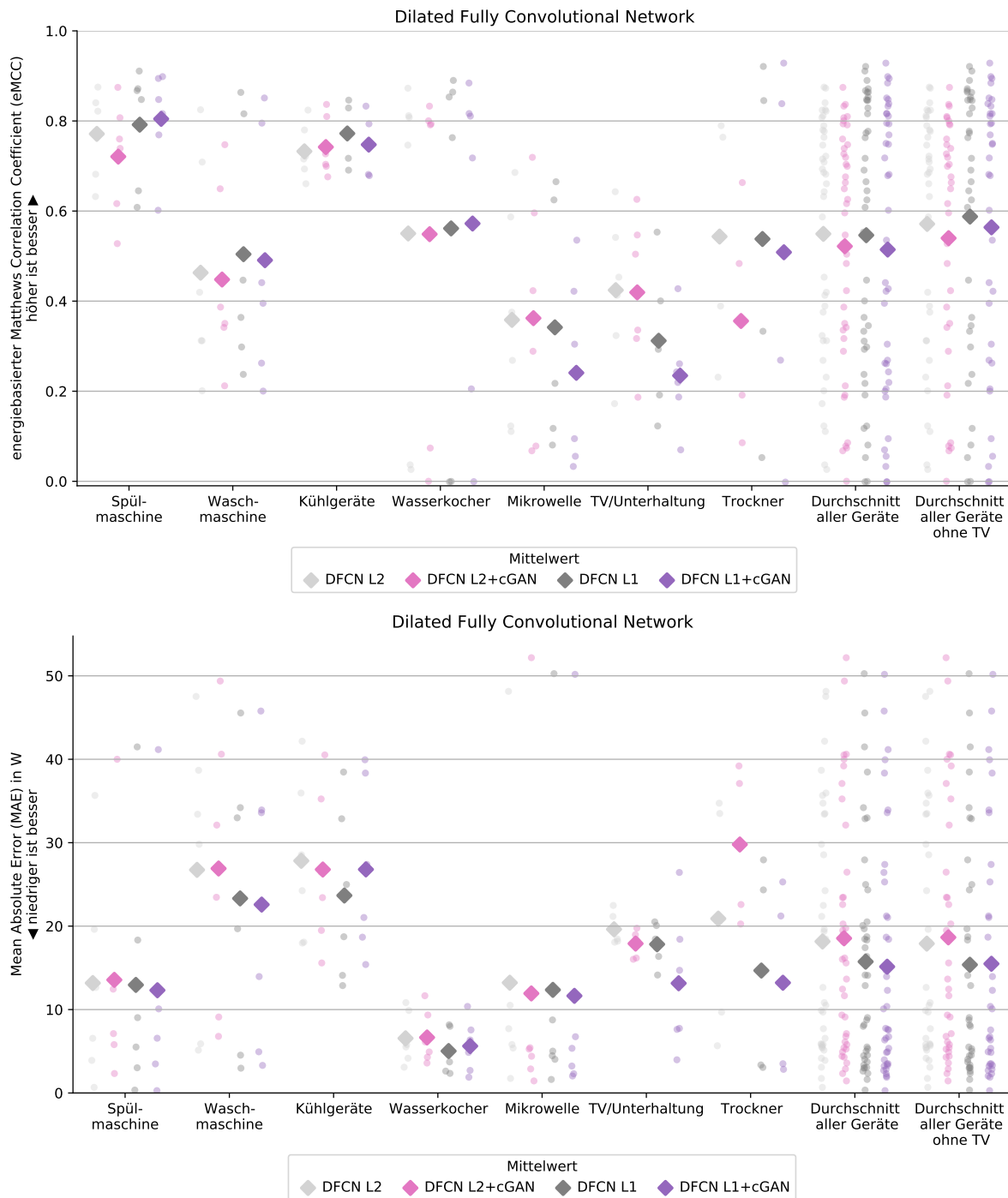


Abbildung 6.23: eMCC und MAE im Vergleich zum ursprünglichen Ansatz (rote Linie) bei der Disaggregation mit dem Dilated Fully Convolutional Network.

Dilated Fully Convolutional Network

In Abbildung 6.22 ist die Entwicklung der Fehlermetriken bei der Verwendung von alternativen Verlustfunktionen mit dem Dilated Fully Convolutional Network dargestellt.

Wenn das Netz mit der L_1 -Metrik trainiert wird, erhöht sich der Korrelationskoeffizient im Mittel bei 4 der 7 Geräteklassen. Der durchschnittliche Fehler sinkt in jedem Fall mit dem Einsatz der L_1 -Metrik.

Der Einsatz des Diskriminators führt oft zu einer Verschlechterung insbesondere der Korrelationsmetrik. Auf Basis der L_1 -Verlustfunktion kann der Diskriminator den mittleren Fehler in 5 von 7 Fällen nochmals senken.

Der Wechsel von L_2 - auf den L_1 -Verlust sorgt am zuverlässigsten für eine Erhöhung des Korrelationskoeffizienten bei gleichzeitiger Reduktion des mittleren Fehlers.

Der niedrigste mittlere Fehler ungeachtet der Korrelationswerte wird mit dem L_1 -Verlust und einem Diskriminator erreicht. Gegenüber dem ursprünglichen Ansatz wird der Fehler im Schnitt um 20,9 % reduziert. Wird nur die L_1 -Metrik verwendet, wird im Schnitt eine Fehlerreduktion um 19,3 % erreicht.

6.2.5 Fazit

Der Einsatz eines Diskriminators während des Trainings, der die Korrektheit der Disaggregation erlernt und dessen Bewertung als zusätzlicher Faktor in der Verlustfunktion des Generators/Disaggregators berücksichtigt wird, kann bei manchen Geräteklassen und Ansätzen die Disaggregationsqualität erhöhen. Im Allgemeinen wird jedoch dadurch nicht sichergestellt, dass durch dieses Konzept eine bessere Disaggregation erreicht wird. Dies wird daran deutlich, dass die eMCC-Metrik im Mittel, verglichen zum gleichen Ansatz ohne Diskriminator, auf dem gleichen oder gar einem schlechteren Niveau liegt.

Wie mit dem Dilated Fully Convolutional Network demonstriert, kann bei manchen Geräten die Kombination aus Diskriminator und L_1 -Verlust die Disaggregationsqualität im Schnitt erhöhen. Jedoch wird diese Verbesserung nicht zuverlässig in allen Situationen bzw. bei allen Geräteklassen erreicht.

Wird hingegen während des Trainings die L_1 - statt der L_2 -Metrik als Verlustfunktion verwendet, verbessern sich in den meisten Fällen die eMCC-Metrik und gleichzeitig sinkt der mittlere absolute Fehler. Somit wird in der Mehrzahl der Fälle mit der L_1 -Metrik ein besserer Disaggregator trainiert. Das Ergebnis scheint dabei nicht von dem verwendeten Netzmodell abzuhängen.

Sowohl der Einsatz der L_1 -Verlustfunktion als auch der des Diskriminator scheint sich polarisierend auf die Lasterkennung auszuwirken. Gerätemodelle und Geräteklassen, die bereits

mit der L_2 -Verlustfunktion gut disaggregiert werden können, werden mit L_1 -Verlustfunktion und einem Diskriminator noch besser erkannt. Bei unbekanntem Gerätemodellen und bei schlecht disaggregierbaren Geräteklassen wird mit diesen alternativen Verlustfunktionen in der Regel die Disaggregation noch schlechter.

6.3 Ansatz C: U-Net mit Dilertation

Der letzte in dieser Studie evaluierte Ansatz untersucht das Potential der U-Net-Architektur (Ronneberger et al., 2015) für Disaggregationsaufgaben. Dieser Ansatz ist von den Arbeiten von Isola et al. (2016), Barsim und Yang (2018) und Brewitt und Goddard (2018) inspiriert.

In Abbildung 6.24 ist das dafür entworfene U-Net dargestellt. Die U-Net-Architektur ist unterteilt in einen Enkodierblock, einen Reduktionsblock und einen Dekodierblock. Im Enkodierblock und im Reduktionsblock wird die Auflösung der Eingabedaten zunächst schrittweise reduziert, um zunehmend abstraktere und zeitlich ausgedehnte Merkmale zu erkennen. Im Dekodierblock wird aus der niedrigaufgelösten Zwischenrepräsentation eine hochaufgelöste Darstellung des disaggregierten Gerätestrang zurückgewonnen.

Die Besonderheit der U-Net-Architektur besteht in dem zusätzlichen Informationspfad zwischen den Schichten des Enkodier- und Dekodierblocks, die die Daten auf der gleichen Auflösungsstufe verarbeiten. Damit können großräumigere und abstraktere Informationen durch die Reduktionsschichten verarbeitet werden. Gleichzeitig werden kleinräumige Informationen, wie beispielsweise detailliertere Lastgangstexturparameter, den Dekodierschichten direkt bereitgestellt und müssen nicht durch alle Schichten durchgereicht werden.

Die Netzarchitektur verwendet dilatierte Faltungen, analog zum Ansatz Dilated Fully Convolutional Network (DFCN) von Brewitt und Goddard (2018). Damit entspricht der Informationsfluss im Enkodierer einem Sequence-to-Point-Ansatz, der parallel an mehreren Stellen durchgeführt wird. Der Dekodierer expandiert die disaggregierten Informationen mittels transponierten Faltungen auf einen kleinen Bereich. Von der letzten Netzschicht wird der Randbereich ausgeschnitten, da für Werte dieses Randbereichs eine reduzierte Menge an Eingabewerten einfließen.

Barsim und Yang (2018) verwenden erstmalig eine U-Net-Architektur für Disaggregationsaufgaben. Allerdings wird dabei nur die Gerätezustandserkennung evaluiert und nicht die allgemeinere Lastgangsrekonstruktion. Ein weiterer Unterschied liegt in dem exponentiell wachsenden Dilatationsfaktor, um einen großräumigen Kontext für die Geräteerkennung bereitzustellen.

In Vorexperimenten wurden Architekturvarianten mit Residuenblöcken sowie unterschiedlichen Normalisierungsschichten erprobt. Für den Anwendungsfall der Disaggregation hat der Einsatz dieser Elemente keine messbaren Vorteile erbracht. In der Regel war der Einsatz dieser Architekturelemente sogar schädlich für die Disaggregationsleistung.

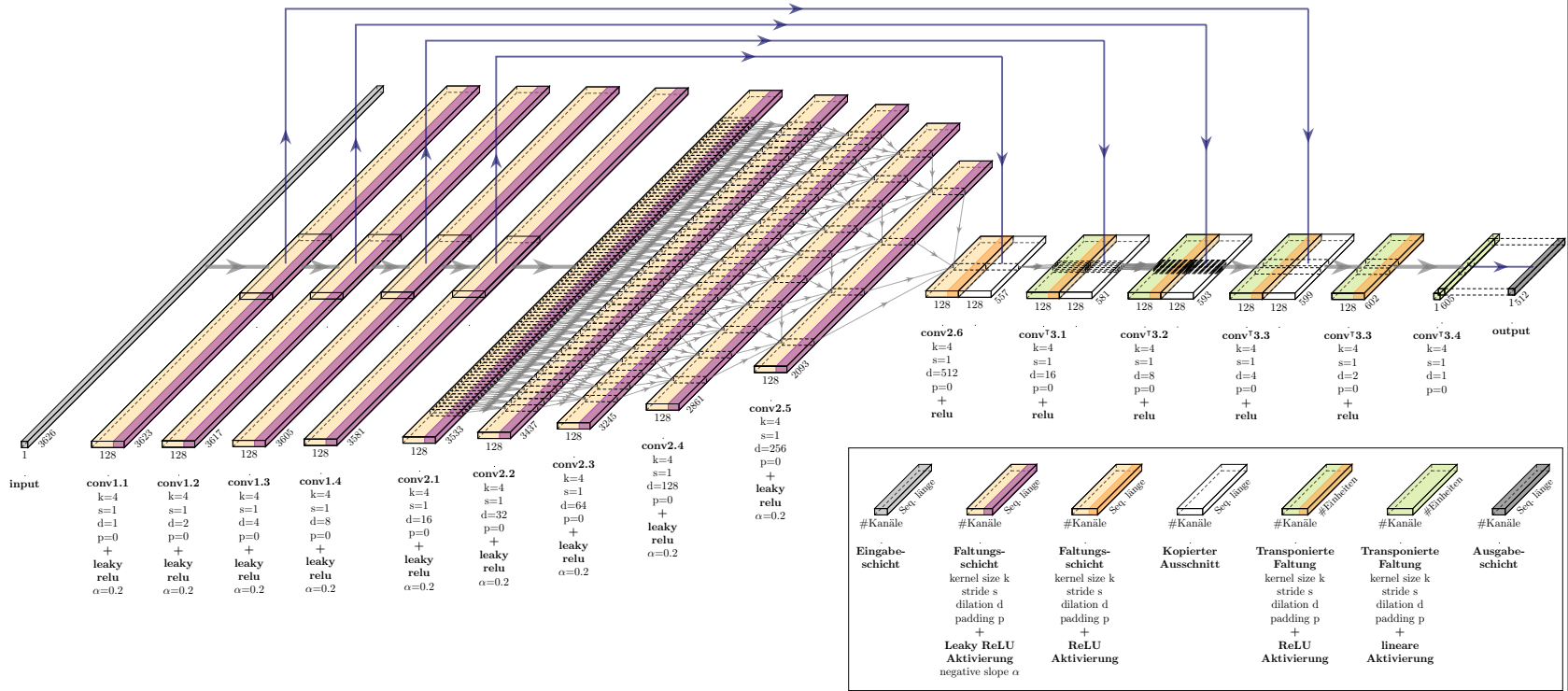


Abbildung 6.24: Architektur des U-Net

6.3.1 Trainingsparameter

Das Training der Ansätze und die Datenvorverarbeitung orientieren sich an der durchgeführten Evaluation des DFCN in Abschnitt 5.1: Alle neuronalen Netze werden mit ADAM 100'000 Iterationen optimiert. Die Mini-Batch-Größe beträgt bei allen Ansätzen 64 Trainingspaare. Die initiale Lernrate 10^{-4} wird in der Iteration 50'000 und 75'000 jeweils um Faktor 10 reduziert.

Wie bei dem Ansatz von Brewitt und Goddard (2018) wird der Haushaltslastgang durch 500 W dividiert und der Mittelwert der individuellen Eingangslastsequenzen auf 0 gesetzt. Dabei repräsentiert 500 W die Größenordnung, in der die Standardabweichung des Haushaltslastgangs liegt. Die Gerätelastsequenz wird durch die durchschnittliche Last der Zielgeräteklasse dividiert, die während Aktivierungen im Trainingsdatensatz beobachtet wird.

Bei der Disaggregation wird der Gerätelastgang in 512 Messintervalle lange, überlappungsfreie Sequenzen partitioniert. Die Länge entspricht 51,2 Minuten. Für jede Sequenz und jedes Messintervall ist nur eine Inferenz des neuronalen Netzes nötig, sodass die Zwischenergebnisse nur aneinandergereiht werden müssen. Die Haushaltslastsequenzen werden so ausgeschnitten, dass die 512 Ausgangswerte genau in der Mitte liegen und das gesamte Wahrnehmungsfeld des neuronalen Netzes ausgefüllt ist. Damit werden jeweils 1557 Messintervalle vor und nach dem 512 Werte langen Gerätelastgang betrachtet und für eine Inferenz wird eine Gebäudelastsequenz mit insgesamt 3626 Messintervalle eingelesen.

Als Verlustfunktion wird die L_2 - und die L_1 -Metrik sowie das L_1 -Diskriminator-Konjugat evaluiert. Für letzteres wird der in Abbildung 6.25 abgebildete Netzarchitektur mit dilatierten Faltungen verwendet. Bei dem Diskriminatornetz wird keine Lastgangssequenz erzeugt, daher gibt es hier keine zum U-Net analogen Querverbindungen.

6.3.2 Ergebnisse

In Abbildung 6.26 ist oben der energiebasierte Korrelationskoeffizient (eMCC) für die Disaggregation der unterschiedlichen Geräteklassen dargestellt. Im Vergleich zwischen den Verlustmetriken L_2 und L_1 ist bei dem DFCN festzustellen, dass bei der Mehrzahl der Geräteklassen (4 von 7) der Korrelationskoeffizient durch den Einsatz der L_1 -Metrik gesteigert werden kann. Bei Trocknern, Mikrowellen und bei der TV- und Unterhaltungselektronik sinkt jedoch der Korrelationskoeffizient mit dem Einsatz der L_1 -Metrik, obwohl in den selben Fällen der durchschnittliche Fehler reduziert wird.

Durch den Einsatz der U-Net-Architektur werden, verglichen zum DFCN mit L_1 -Metrik, bei Waschmaschinen, Kühlgeräten und Wasserkochern sowohl eine nochmals höhere eMCC-Metrik als auch eine noch niedrigere absolute Fehler erzielt. Bei Mikrowellen sinkt der Korrelationskoeffizient, obwohl der absolute Fehler abnimmt. Laut den Metriken wird die

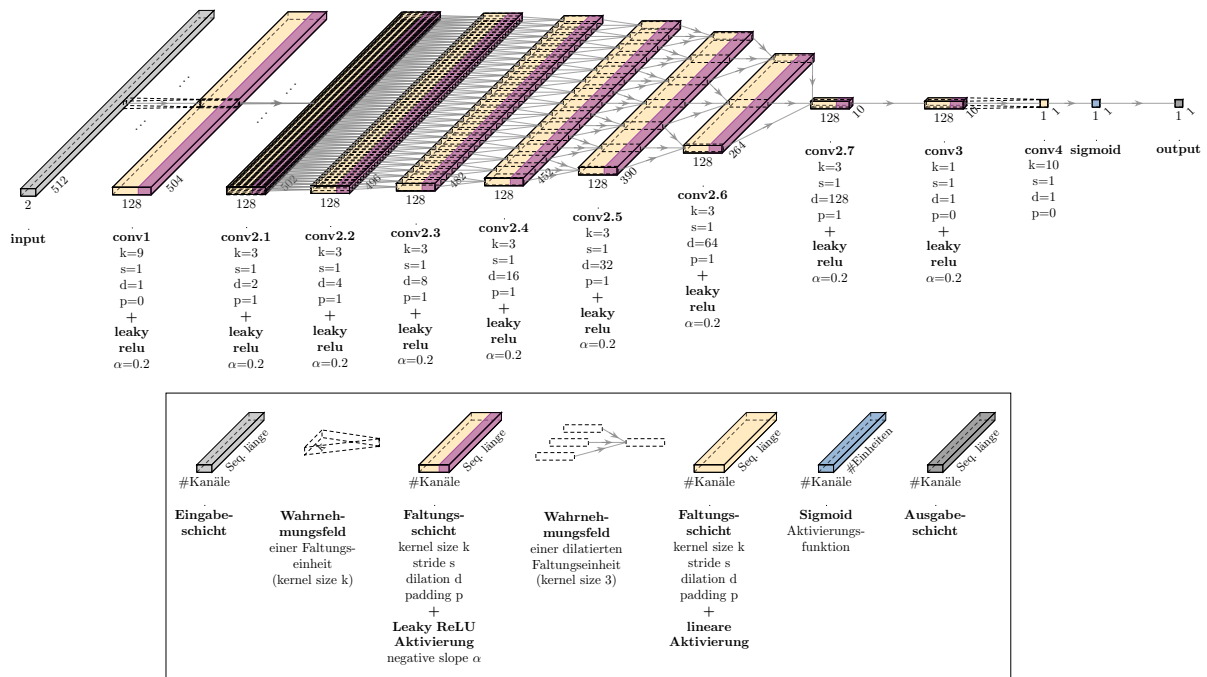


Abbildung 6.25: Architektur des Diskriminators, der bei dem Training einer Variante des U-Net verwendet wird.

Disaggregation von Spülmaschinen, Trocknern und TV- und Unterhaltungselektronik durch die U-Net-Architektur schlechter durchgeführt.

Der mittlere absolute Fehler (MAE) ist unten in Abbildung 6.26 aufgezeigt. Durch die Nutzung der L_1 statt der L_2 -Metrik bei DFCN wird der Fehler bei allen Geräteklassen reduziert. In der Mehrzahl der Fälle verringert die U-Net-Architektur den mittleren Fehler nochmals bei Waschmaschinen, Kühlgeräten, Wasserkochern und Mikrowellen. Bei Spülmaschinen, Trocknern sowie bei TV- und Unterhaltungselektronik steigt jedoch der Fehler im Mittel mit der U-Net-Architektur.

Die vollständigen Ergebnisse und Zahlenwerte sind im Anhang in Abschnitt A.2 aufgeführt.

6.3.3 Fazit

Mit der vorgeschlagenen U-Net-Architektur mit dilatierten Faltungen wird ein für 512 Messintervalle parallel durchgeführter Sequence-to-Point-Ansatz umgesetzt, der die DFCN-Architektur von Brewitt und Goddard (2018) um die U-Net-typischen Informationspfade ergänzt. Dadurch müssen Detailparameter des Lastgangs nicht durch alle Schichten durchgeschleust werden und können über diesen zweiten Informationspfad direkt in den

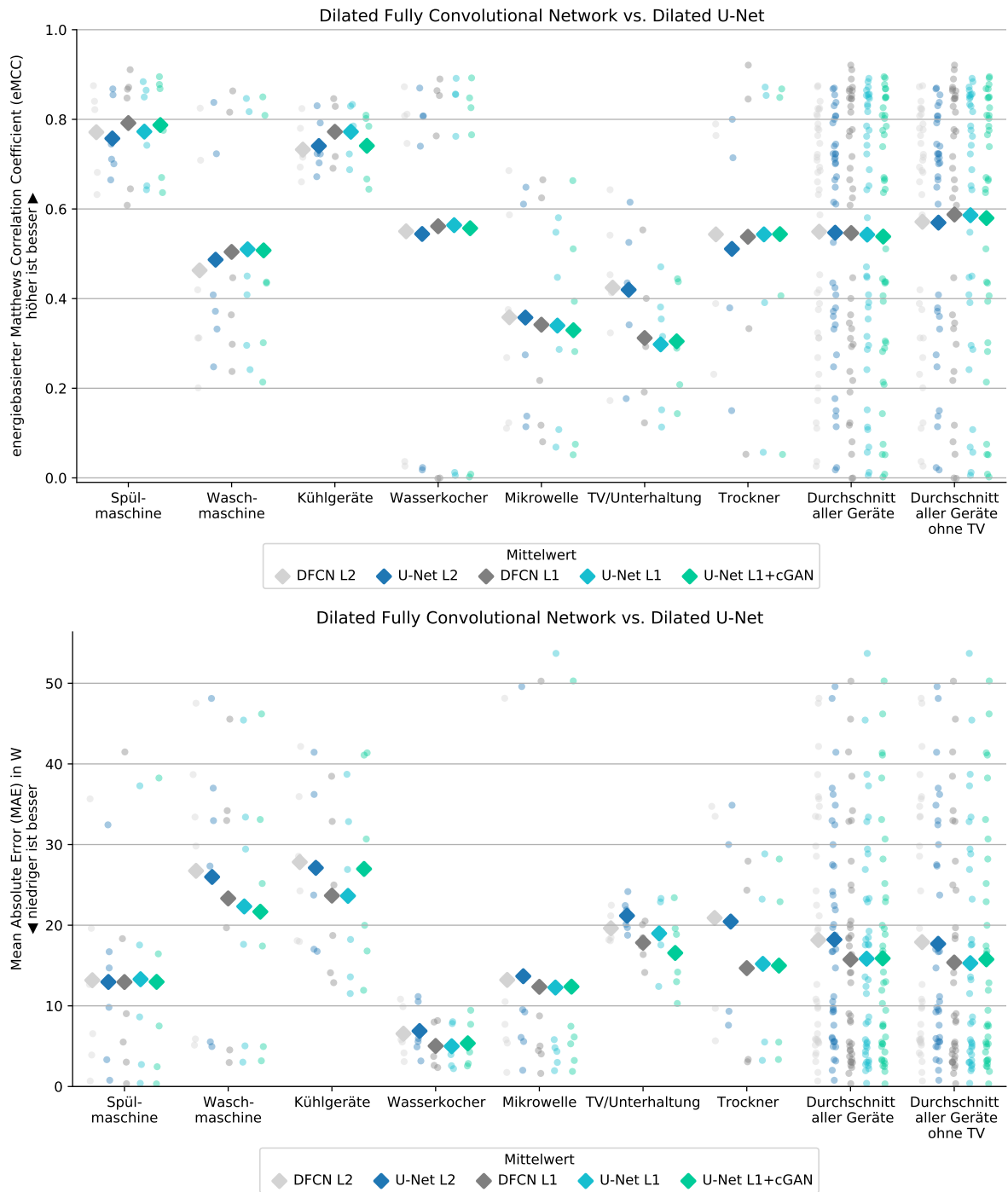


Abbildung 6.26: DFCN vs. Dilated U-Net

Dekodierer fließen, der die Gerätelastgänge reproduziert. Das besondere an dieser neu entwickelten Architektur ist, dass der Sequence-to-Point-Charakter trotz U-Net-Form beibehalten wird, sodass, wie bei DFCN, die Zusammenfassung der Teilergebnisse von dem Netz selbst durchgeführt wird.

Im Ergebnis verbessert die U-Net-Architektur die Disaggregationsqualität bei den Haushaltsgeräteklassen, die bereits zuvor von bestehenden Ansätzen gut disaggregiert werden können. Bei Haushaltsgeräten, die auch von vorhandenen Ansätzen schlecht disaggregiert werden können, erzielt die U-Net-Architektur in der Regel noch schlechtere Ergebnisse.

Letzteres kann vermutlich kompensiert werden, sofern mehr Gerätemodelle und Haushaltsgrundlasten in den Trainingsdatensatz aufgenommen werden. Diese Sensitivität der vorgeschlagenen Netzarchitektur gegenüber mehr und qualitativ besseren Datenquellen zeigt die Wegrichtung für die Nutzbarmachung von größeren Datensätzen in der Zukunft.

6.4 Zusammenfassung

Ausgehend von der Motivation, Disaggregationsansätze zu verbessern, indem die Ähnlichkeit von Lastgängsschätzungen zu gemessenen Lastgängen erhöht wird, sind in diesem Kapitel drei Ansätze vorgestellt und evaluiert worden.

Der erste Ansatz hängt einen Lastgangsgenerator als Nachverarbeitungstufe an ein bestehendes disaggregierendes neuronale Netz an. Dabei wird der Lastgangsgenerator zuvor als Teil eines GAN erstellt. Obwohl nachgewiesen wird, dass der Lastgangsgenerator die Testdaten mit einer befriedigenden Qualität reproduzieren kann, erzielt das Konstrukt keinerlei Verbesserungen in der Disaggregationsqualität.

Der zweite Ansatz trainiert vorhandene Ansätze mit einer zusätzlichen Bewertung eines Lastgangsdiskriminators. Dessen Aufgabe ist die Bewertung, ob eine durchgeführte Disaggregation korrekt ist. Der Lastgangsdiskriminator wird dabei als Teil eines GAN unüberwacht trainiert. Dieser Ansatz führt jedoch nur bei manchen Geräteklassen zu einer Verbesserung der Disaggregationsqualität. Die Ergebnisse zeigen weiterhin, dass die gängige Praxis, zum Training ausschließlich die L_2 -Verlustfunktion zu verwenden, überdacht werden muss. In der Regel erzielt der Austausch der L_2 -Metrik mit der L_1 -Metrik eine höhere Disaggregationsqualität.

Ein weiterer Ansatz evaluiert das Potential einer U-Net-förmigen Architektur, das als Weiterentwicklung des Dilated Fully Convolutional Network konzipiert ist. Bei diesem Ansatz werden Gerätemodelle, die von bestehenden Ansätzen bereits gut erkannt werden, noch präziser disaggregiert. Jedoch werden bei denen Gerätemodellen, bei denen bestehende Ansätze Schwierigkeiten haben, die Last noch schlechter geschätzt.

Insgesamt deuten die Ergebnisse darauf hin, dass bereits das DFCN-Architektur eine ausreichende Kapazität verfügt, um die gängigen Haushaltsgeräte disaggregieren zu können, sofern diese durch Merkmale im Haushaltslastgang erkennbar sind. Vermutlich bewirken umfangreichere Datensätze mit Aufzeichnungen von diverseren Gerätemodellen deutlich stärkere Steigerungen in der Disaggregationsqualität als Modifikationen an Netzarchitekturen oder Trainingsmethoden.

Kapitel 7

Ergebnisse

In diesem Kapitel werden die Ergebnisse der bestehenden Ansätze aus Kapitel 5 sowie die der neuen Ansätze aus Kapitel 6 zusammengefasst und miteinander verglichen.

Da die Energieverbrauchsdisaggregation für unterschiedliche Zielsetzungen eingesetzt werden kann, müssen zum Vergleich jeweils andere, anwendungsspezifische Bewertungsmetriken herangezogen werden. Die folgenden in der Literatur am häufigsten genannten Anwendungsszenarien werden betrachtet:

Reproduktion des Gerätelastgangs

Hierbei steht der exakte zeitliche Verlauf des Gerätelastgangs im Fokus. Jegliche Abweichung des geschätzten Lastwertes von dem gemessenen Lastwert im jeweiligen Messintervall wird als Fehler gewertet. Zum Abbilden dieser Fehlerdefinition ist der *Mean Average Error* (MAE) eine offensichtliche Wahl. Da jedoch bei dieser Metrik ein direkter Vergleich zwischen unterschiedlichen Haushaltsgeräten aufgrund der Leistungsunterschiede schwierig ist, wird der energiebasierte *Matthews Correlation Coefficient* (eMCC) als weitere Vergleichsmetrik zum Einsatz gebracht.

Ein Ansatz muss sowohl höhere eMCC-Werte als auch niedrigere MAE aufweisen, um gegenüber einem anderen Ansatz als besser zu gelten. Wenn sich nur eine Metrik verbessert und sich zeitgleich die andere Metrik verschlechtert, werden die Ansätze als gleichwertig bewertet und dieser Fall wird genauer untersucht.

Dieser Anwendungsfall ist von den hier genannten der Allgemeinste, da aus den disaggregierten Lastgangsverläufen die für die verbleibenden Anwendungsfälle benötigten Daten leicht berechnet werden können.

Energieverbrauchsbilanz eines Gerätes

Für diesen Anwendungsfall soll eine gerätescharfe Energieverbrauchsübersicht innerhalb eines Gebäudes generiert werden. Dabei spielt der zeitliche Verlauf eines Gerätelastgangs weniger eine Rolle als vielmehr die Bilanz des Gesamtstromverbrauchs innerhalb eines längeren (Abrechnungs-)Zeitraums.

Dieser Anwendungsfall wird mit der Metrik *Relative Error in Total Energy* (RETE) repräsentiert. Diese Metrik drückt die Differenz zwischen geschätzter und tatsächlicher Energieverbrauchsmenge im gesamten Evaluationszeitraum aus und setzt diese in Bezug zur tatsächlichen Energieverbrauchsmenge. Dabei ist es akzeptabel, wenn sich zeitweises Unter- und Überschätzen der Last ausgleichen.

Detektion von Geräteaktivität

Mit dem dritten Anwendungsfall wird auf eine binäre Klassifikation abgezielt, bei dem in jedem Messintervall bestimmt werden soll, ob das Zielgerät aktiv war oder sich im Ruhezustand befand. Bezüglich einer Privatsphärenbetrachtung wird angenommen, dass die Geräteaktivität mit der Aktivität von Haushaltsbewohnern korreliert, wodurch die Geräteaktivität selbst zu einem sensitiven, personenbezogenen Datum wird.

Zur Bewertung wird in dieser Arbeit der *Matthews Correlation Coefficient* (MCC) bevorzugt. Im Gegensatz zu der in der Literatur oft verwendeten F_1 -Metrik ist der MCC auch bei den Haushaltsgeräten aussagekräftig, die nicht überwiegend inaktiv sind.

Tatsächlich sind im Endeffekt die Zahlenwerte von MCC und F_1 ähnlich. Nur bei Kühlschränken, die etwa die Hälfte der Zeit aktiv sind, unterscheiden sich die Metriken stark.

Ein perfektes Disaggregationsverfahren würde in allen Metriken und Anwendungsfällen gleichzeitig die besten Werte erreichen. Jedoch sind praktische Ansätze fehlerbehaftet, sodass sich bei einzelnen Metriken und damit bei den entsprechenden Zwecken von Non-Intrusive Appliance Load Monitoring (NILM) Unterschiede ergeben. Daher werden im Folgenden alle Verfahren hinsichtlich dieser drei unterschiedlichen Anwendungsfälle getrennt evaluiert.

7.1 Reproduktion des Gerätelastgangs

Bei diesem Anwendungsfall soll das Disaggregationsverfahren den Verlauf des Gerätelastgangs exakt schätzen. Mit den Metriken *Mean Average Error* (MAE) und energiebasierter *Matthews Correlation Coefficient* (eMCC) werden in jedem Messintervall Abweichungen von der gemessenen Gerätelast gewertet.

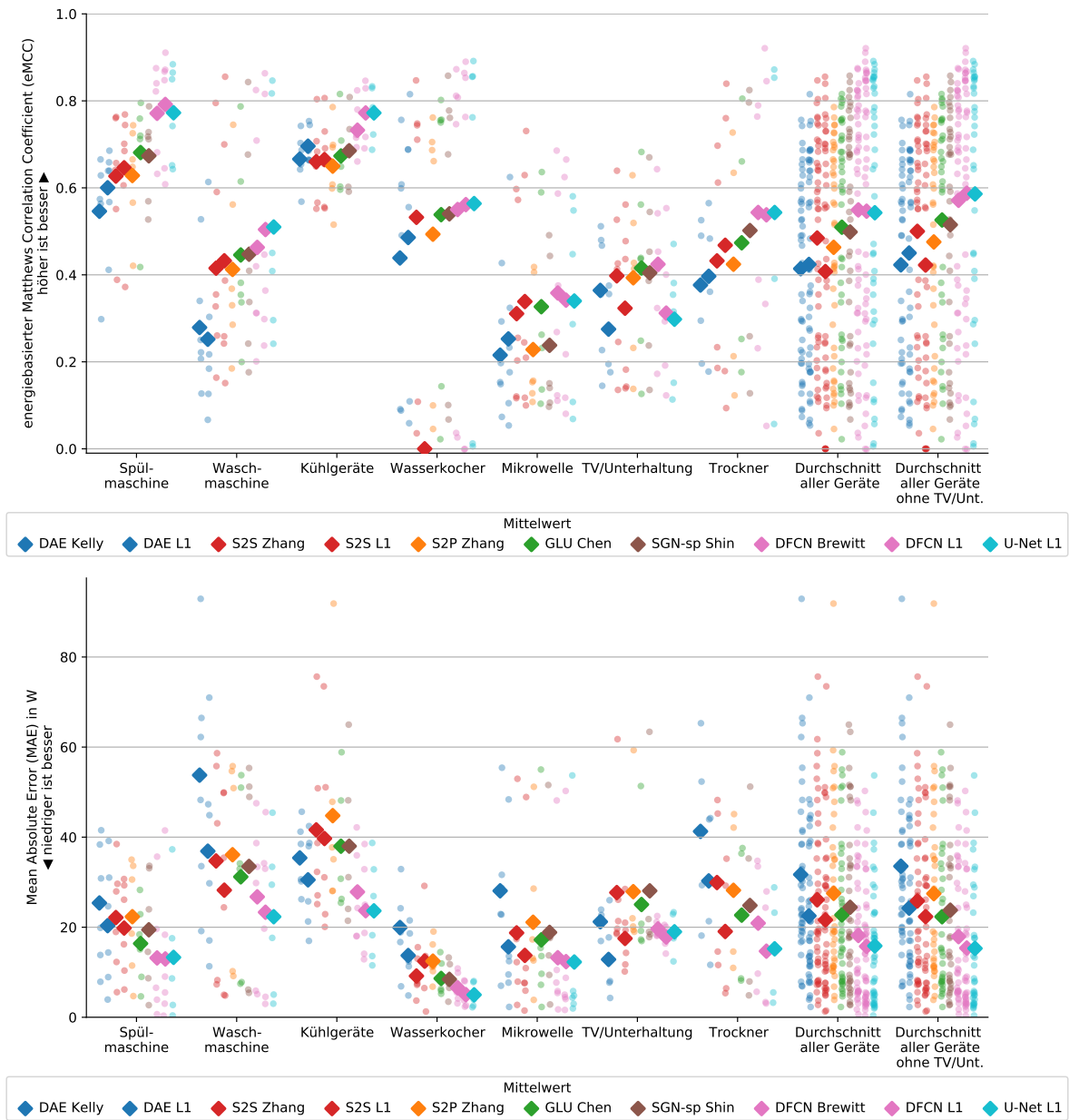


Abbildung 7.1: Die Metriken energiebasierter Matthews Correlation Coefficient (eMCC) und Mean Average Error (MAE) der wesentlichen Ansätze in dieser Arbeit.

Diese beiden Metriken sind in Abbildung 7.1 nach Geräteklasse gruppiert dargestellt. Das Ergebnis eines Testgebäudes ist als ein Punkt dargestellt. Der arithmetische Mittelwert über alle Testgebäude ist als Raute dargestellt.

Das älteste Verfahren, Denoising Autoencoder (DAE), stellt zunächst die Vergleichsbasis dar. Im Vergleich dazu wird sowohl mit Sequence-to-Sequence (S2S) als auch mit Sequence-to-Point (S2P) der Disaggregationsfehler reduziert, wobei S2P im Mittel schlechter abschneidet als S2S.

Der Einsatz der L_1 -Verlustfunktion bei DAE und S2S reduziert zwar den MAE, die Korrelation zwischen geschätztem und tatsächlichem Energieverbrauch verbessert sich jedoch nur bei dem DAE-Ansatz. Auffällig ist bei S2S mit L_1 , dass der Wasserkocher in keinem Testgebäude erkannt wird, was zu einem Korrelationswert von 0 führt.

Der Ansatz mit Gated Linear Units (GLU) und das Subtask Gated Network (SGN) zeigen verglichen zu DAE und S2S/S2P eine höhere Disaggregationsqualität. Nochmals besser ist das Dilated Fully Convolutional Network (DFCN).

Wird der DFCN mit der L_1 -Verlustfunktion trainiert, verringert sich der Korrelationskoeffizient geringfügig. Der MAE wird jedoch deutlich reduziert.

Das evaluierte U-Net mit L_1 -Verlustfunktion hat ebenfalls einen etwas niedrigeren Korrelationskoeffizienten als die DFCN-Varianten. Bei einigen Geräten, wie der Waschmaschine, liefert es jedoch die besten Disaggregationsergebnisse.

Im Durchschnitt über alle Geräte liefert das DFCN mit einer L_1 -Verlustfunktion die genauesten Disaggregationen.

Auffällig beim Einsatz der L_1 -Verlustfunktion bei DFCN und U-Net ist, dass diese Ansätze gerade bei schwer disaggregierbaren Geräten wie TV- und Unterhaltungselektronik gemessen am Korrelationskoeffizienten versagen. Dennoch liefern diese Ansätze deutlich geringere absolute Fehler.

Im folgenden werden die Ergebnisse für die einzelnen Geräteklassen diskutiert.

Waschmaschinen und Waschtrockner

In Abbildung 7.2 sind die Metriken zur Beurteilung der Disaggregationsqualität bei Waschmaschinen dargestellt. Bei den meisten Testgebäuden sowie im Durchschnitt erreicht das U-Net mit L_1 -Verlustfunktion die höchsten Korrelationswerte und gleichzeitig die niedrigsten absoluten Fehler.

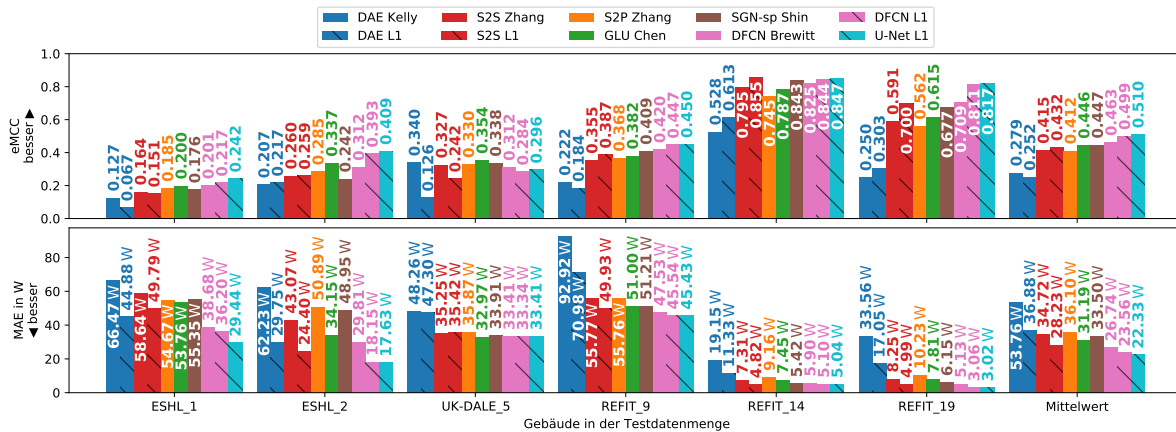


Abbildung 7.2: Energiebasierter Matthews Correlation Coefficient (eMCC) und Mean Average Error (MAE) für Verfahren zur Disaggregation von Waschmaschinenlast.

Ausnahmen bilden das Testgebäude 5 des UK-DALE-Datensatzes, in dem der Ansatz mit Gated Linear Unit (GLU) von Chen, Wang et al. (2018) bessere Werte erreicht. Wie in Abschnitt 5.2.1 bereits festgestellt, ist den neuronalen Gerätemodellen vermutlich das Trockenprogramm der Wäschetrockner aus dem Training nicht bekannt.

Kühlgeräte

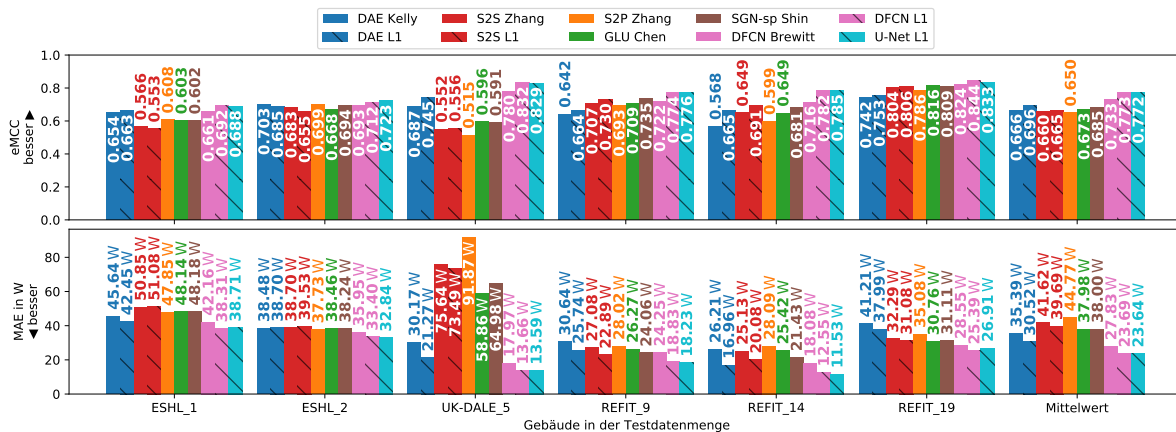


Abbildung 7.3: Energiebasierter Matthews Correlation Coefficient (eMCC) und Mean Average Error (MAE) für Verfahren zur Disaggregation von Kühlgeräteleast.

Die Bewertung der Disaggregation von Kühlgeräten als Regressionsproblem ist in Abbildung 7.3 dargestellt. In allen Testgebäuden werden die höchsten eMCC-Bewertungen und die niedrigsten MAE von den beiden Ansätzen U-Net und Dilated Fully Convolutional

Network (DFCN) mit L_1 -Verlustfunktion erreicht. Dabei erreicht das U-Net einen etwas geringeren absoluten Fehler.

Trockner

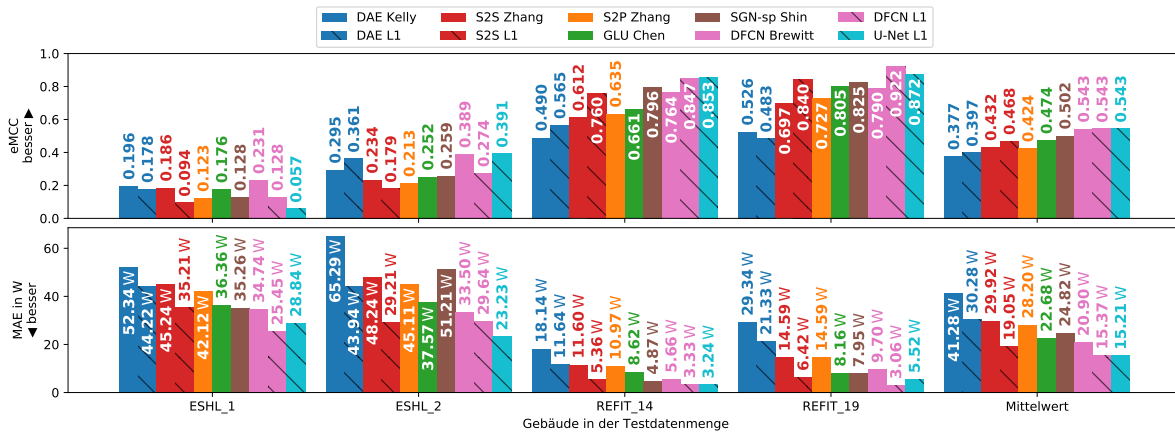


Abbildung 7.4: Energiebasierter Matthews Correlation Coefficient (eMCC) und Mean Average Error (MAE) für Verfahren zur Disaggregation von Trocknerlast.

In Abbildung 7.4 ist die Disaggregationsqualität bei Wäschetrocknern dargestellt. Bei den zwei Messkampagnen im ESHL stellt das Dilated Fully Convolutional Network mit L_2 -Verlustfunktion bezüglich des eMCC das beste Verfahren dar. Für die REFIT-Testgebäude erreichen mal das DFCN, mal das U-Net, jeweils mit einer L_1 -Verlustfunktion, die besten Ergebnisse. Die niedrigsten absoluten Fehler werden abwechselnd von dem DFCN und dem U-Net mit L_1 -Verlustfunktion erreicht.

TV und Unterhaltungselektronik

In Abbildung 7.5 ist die Disaggregationsleistung bei der Geräteklasse TV und Unterhaltungselektronik dargestellt. Die Metriken befinden sich bei vielen Ansätzen auf einem vergleichbaren Niveau.

Auffällig sind die vergleichsweise schlechten eMCC-Werte bei den Ansätzen, die mit einer L_1 -Verlustfunktion trainiert wurden. Die im Schnitt schlechteren eMCC-Werte dieser Ansätze stehen im Widerspruch zu den niedrigeren absoluten Fehlern. Hierbei erreicht der DAE mit L_1 -Verlustfunktion im Mittel den niedrigsten Wert.

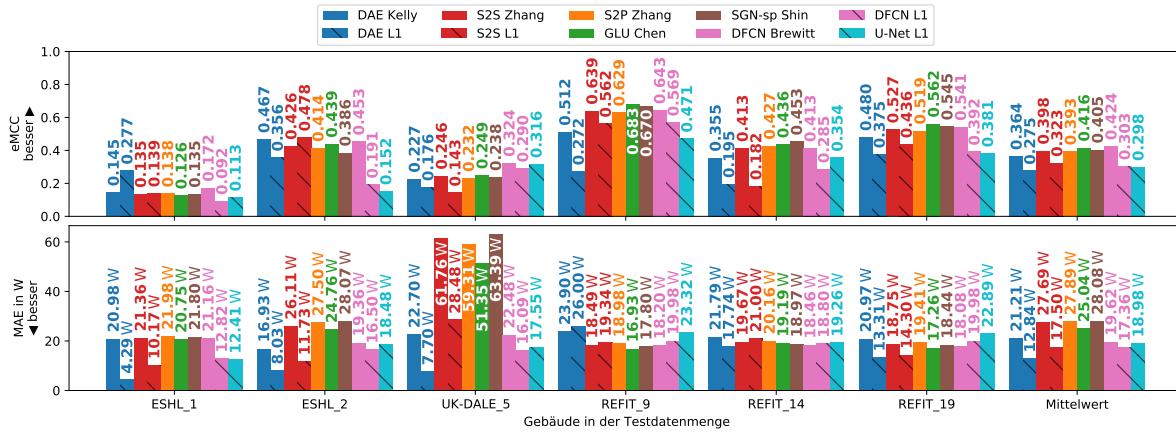


Abbildung 7.5: Energiebasierter Matthews Correlation Coefficient (eMCC) und Mean Average Error (MAE) für Verfahren zur Disaggregation der Last von TV und Unterhaltungselektronik.

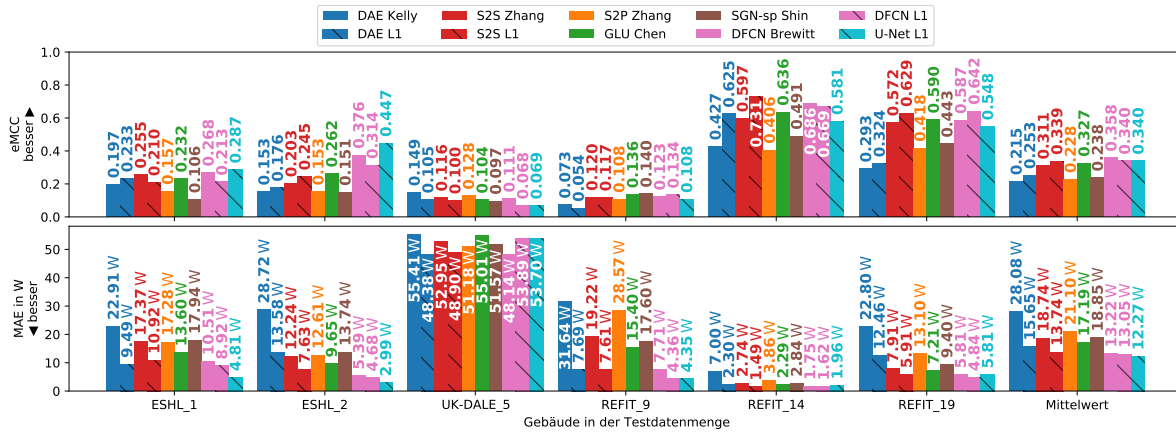


Abbildung 7.6: Energiebasierter Matthews Correlation Coefficient (eMCC) und Mean Average Error (MAE) für Verfahren zur Disaggregation von Mikrowellenlast.

Mikrowellen

In Abbildung 7.6 ist die Regressionsleistung der Disaggregationsverfahren bei Mikrowellen dargestellt. Verglichen mit den Ergebnissen bei der Disaggregation der zuvor betrachteten Haushaltsgeräte ist der energiebasierte Matthews Correlation Coefficient von allen Ansätzen deutlich niedriger. Nur die Mikrowellen in den REFIT-Gebäuden 14 und 19 können von manchen Ansätzen mit einem eMCC von über 0,6 disaggregiert werden. Der eMCC liegt bei allen anderen Testgebäuden unter 0,5.

Gemessen am durchschnittlichen eMCC disaggregiert das Dilated Fully Convolutional Network mit L_2 -Verlustfunktion die Mikrowelle am besten. Den geringsten durchschnittlichen Fehler bei den meisten Testgebäuden macht das U-Net mit einem L_1 -Verlust.

Geschirrspüler

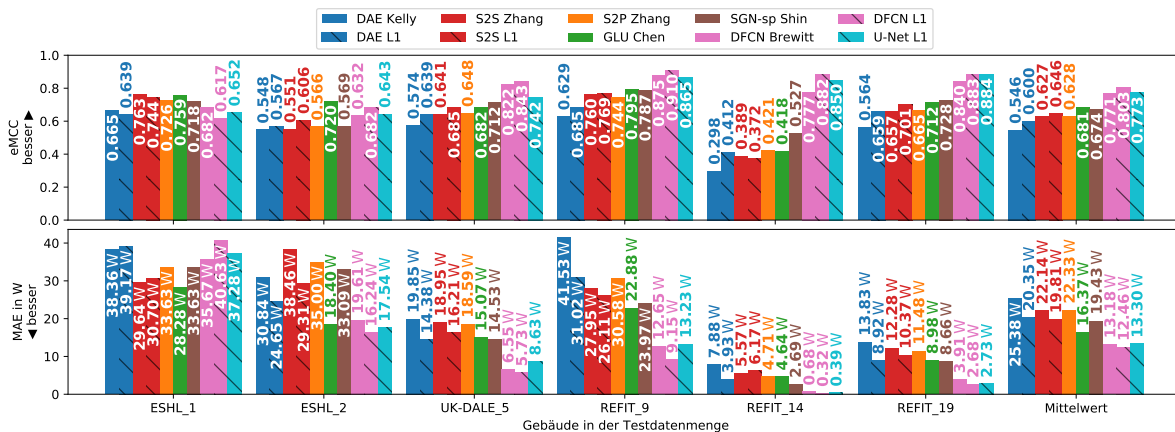


Abbildung 7.7: Energiebasierter Matthews Correlation Coefficient (eMCC) und Mean Average Error (MAE) für Verfahren zur Disaggregation von Geschirrspülmaschinenlast.

Die Ergebnisse für die Regression des Lastgangs der Geschirrspülmaschine sind in Abbildung 7.7 dargestellt. Im Schnitt wird ein Spülmaschinenlastgang von einem Dilated Fully Convolutional Network mit L_1 -Verlust am genauesten disaggregiert. Bei den Testgebäuden 9, 14 und 19 des REFIT-Datensatzes und 5 des UK-DALE-Datensatzes erreicht es die besten Bewertungen.

Mit dem Spülmaschinenmodell im Energy Smart Home Lab scheint das DFCN jedoch Probleme zu haben. In diesem Fall führt das GLU-Netz von Chen, Wang et al. (2018) die bessere Disaggregation durch.

Wasserkocher

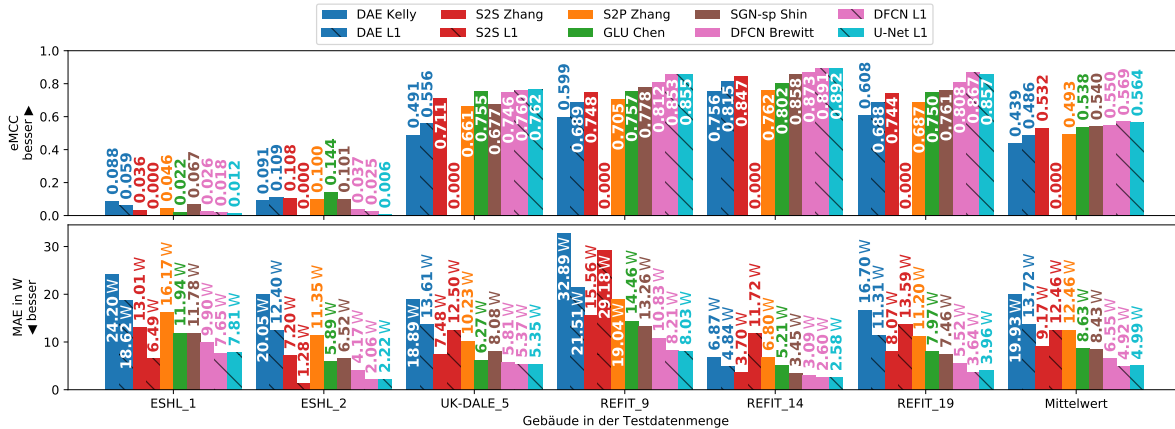


Abbildung 7.8: Energiebasierter Matthews Correlation Coefficient (eMCC) und Mean Average Error (MAE) für Verfahren zur Disaggregation von Wasserkochern.

In Abbildung 7.8 ist die Regressionsleistung der Disaggregationsverfahren bei Wasserkochern dargestellt. Der energiebasierte Matthews Correlation Coefficient ist bei sämtlichen Verfahren bei den beiden ESHL-Gebäuden kleiner als 0,2 und damit deutlich niedriger als bei den verbleibenden Testgebäuden.

Bemerkenswert ist der eMCC-Wert von 0 bei dem Sequence-to-Sequence-Ansatz, der mit einem L₁-Verlust trainiert wurde. Der Ansatz erkennt keine einzige Wasserkocheraktivität. Im ESHL führt dieses konservative Verhalten zum niedrigsten absoluten Fehler unter den Ansätzen.

Im Schnitt führt das Dilated Fully Convolutional Network mit L₁-Verlust die beste Disaggregation durch, mit den im Mittel höchsten eMCC und niedrigsten MAE-Werten.

7.2 Energieverbrauchsbilanz eines Gerätes

Bei diesem Anwendungsfall wird evaluiert, wie genau der Gesamtenergieverbrauch über den gesamten Messzeitraum geschätzt wird.

Zur Bewertung wird in Abbildung 7.9 der Betrag des Relative Error in Total Energy nach Geräteklassen gruppiert dargestellt. Jedes Ergebnis für ein Testgebäude ist als ein Punkt dargestellt. Der arithmetische Mittelwert über alle Testgebäude ist als Raute dargestellt. Aufgrund der Streuung in höhere Werte ist der y-Achsen-Abschnitt bis 600 % nochmals in der unteren Graphik dargestellt.

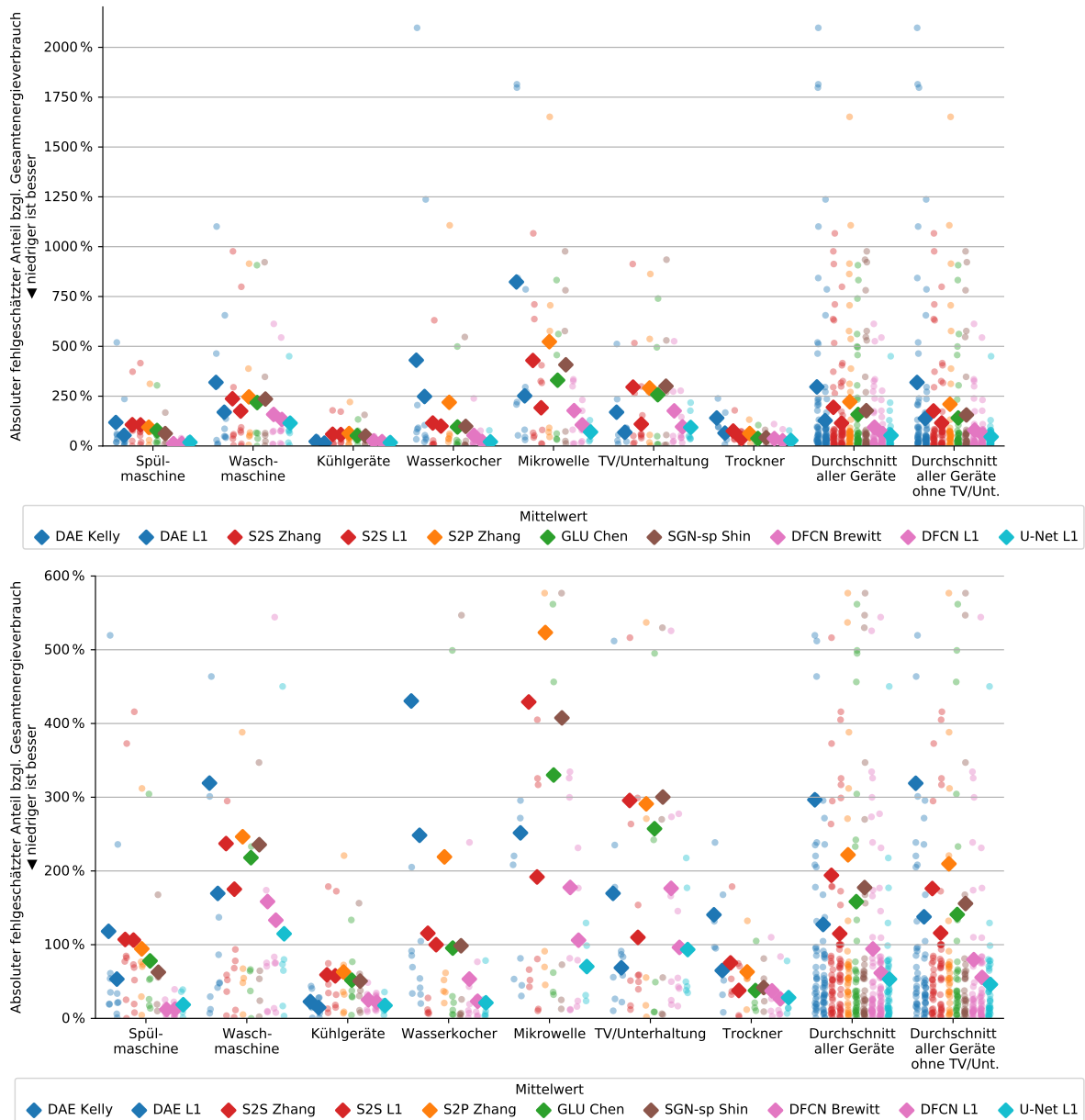


Abbildung 7.9: Absoluter Relative Error in Total Energy von allen Ansätzen. Die untere Graphik zeigt nochmals den y-Achsen-Ausschnitt bis 600 %.

Zunächst zeigt sich, dass gerade bei Geräten mit niedrigem Gesamtenergieverbrauch wie Wasserkocher und Mikrowelle eine Fehlschätzung von über 2000 % möglich ist. Aber auch der Energieverbrauch von Geräten mit einem mittleren Gesamtenergieverbrauch wie der Waschmaschine wird stark überschätzt.

Die mit L_1 -Verlust trainierten Ansätze schätzen den Gesamtenergieverbrauch genauer als die gleichen Ansätze, die mit einem L_2 -Verlust trainiert worden sind. Dabei erreicht das U-Net mit L_1 -Verlust im Mittel die niedrigsten Fehlschätzungen bezüglich der Energieverbrauchsbilanz.

Bei Kühlgeräten ist es bemerkenswert, dass der Denoising Autoencoder neben DFCN und dem U-Net sehr gute Ergebnisse liefert, jedoch im Mittel bei allen anderen Geräten schlecht schätzt.

Im Folgenden werden die Ergebnisse für die einzelnen Geräteklassen anhand des vorzeichenbehafteten Relative Error in Total Energy (RETE) diskutiert. Am Vorzeichen ist erkennbar, ob der Energieverbrauch unter- oder überschätzt wurde. Für die Bildung des Mittelwertes in der letzten Spalte werden die Beträge der RETE-Werte verwendet.

Waschmaschinen und Waschtrockner

In Abbildung 7.10 ist die Energiebilanzschätzung der Verfahren bei Waschmaschinen dargestellt. In den Testgebäuden UK-DALE 5 und REFIT 9 und 14 wird der Energieverbrauch der Waschmaschine eher unterschätzt während alle Ansätze den Waschmaschinenenergieverbrauch im ESHL deutlich überschätzen.

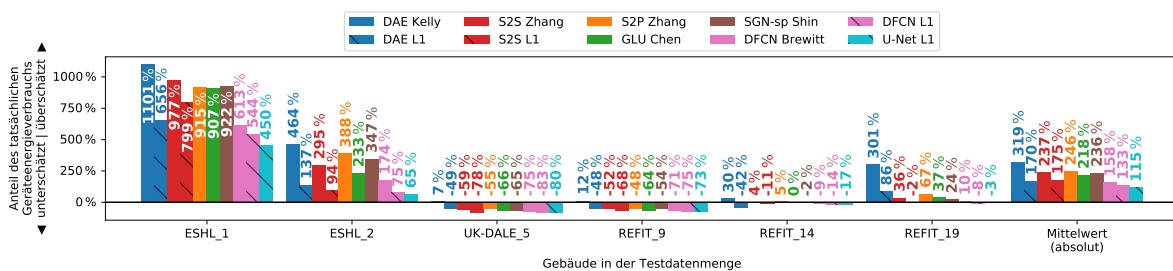


Abbildung 7.10: Anteil fehlgeschätzten Gesamtenergieverbrauchs von Waschmaschinen.

Kühlgeräte

Die Bewertung der Energiebilanzschätzung bei Kühlgeräten ist in Abbildung 7.11 mit der RETE-Metrik dargestellt. Der Gesamtenergieverbrauch der Kühlgeräte im UK-DALE-Gebäude 5 wird von einigen Ansätzen stark überschätzt. In den meisten Fällen ist die

Schätzung jedoch bis auf einen Fehler von bis zu 30 % korrekt. Im Mittel erreicht der DAE mit L_1 -Verlust die genaueste Energiebilanzschätzung.

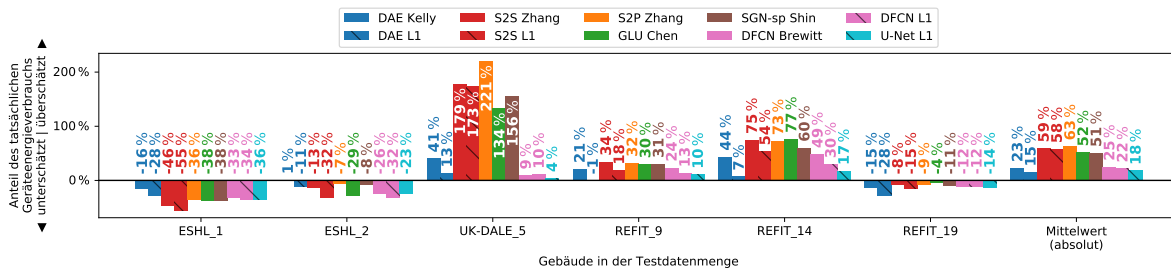


Abbildung 7.11: Anteil fehlgeschätzten Gesamtenergieverbrauchs von Kühlgeräten.

Trockner

In Abbildung 7.12 ist das Ergebnis der Energiebilanzschätzung bei Wäschetrocknern dargestellt. Der Einsatz der L_1 -Verlustfunktion führt zu einer Reduktion einer überschätzten Energiebilanz. Bei der ESHL-Messkampagne 2 wird dadurch der Energieverbrauch teilweise unterschätzt. Im Mittel liefert hier der Ansatz DFCN mit L_1 -Verlust die genauesten Energiebilanzschätzungen.

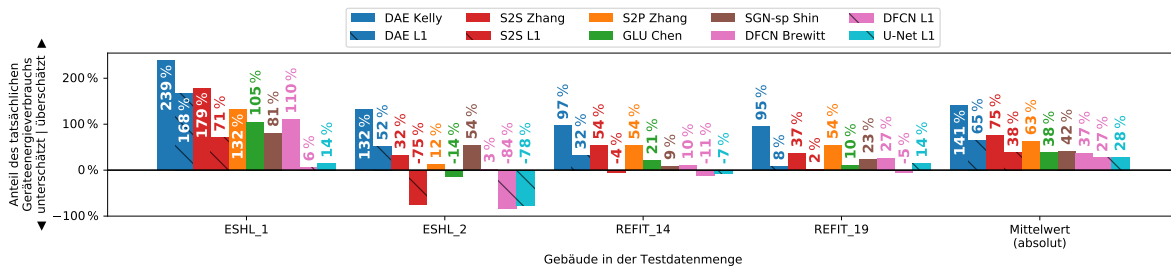


Abbildung 7.12: Anteil fehlgeschätzten Gesamtenergieverbrauchs von Trocknern.

TV und Unterhaltungselektronik

In Abbildung 7.13 ist das Ergebnis der Gesamtenergieverbrauchsschätzung bei der Geräteklasse TV und Unterhaltungselektronik dargestellt. Im ESHL und im UK-DALE-Gebäude 5 wird der Energieverbrauch häufig überschätzt. In den REFIT-Gebäuden ist die Energieverbrauchsschätzung akzeptabel.

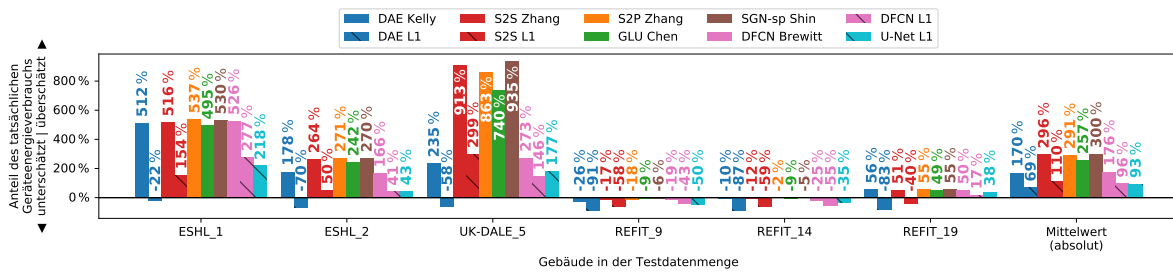


Abbildung 7.13: Anteil fehlgeschätzten Gesamtenergieverbrauchs von TV- und Unterhaltungselektronik.

Mikrowellen

In Abbildung 7.14 ist das Ergebnis der Energiebilanzschätzung bei Mikrowellen dargestellt. Nur der Gesamtenergieverbrauch der Mikrowellen in den Gebäuden UK-DALE 5, REFIT 14 und 19 werden von den Ansätzen akzeptabel geschätzt. In den verbleibenden Testgebäuden wird der Energieverbrauch deutlich überschätzt. Im Mittel liefert das mit L₁-Verlust trainierte U-Net die genauesten Energiebilanzschätzungen, da es auch bei den schwierigen Gebäuden den Energieverbrauch nicht extrem überschätzt.

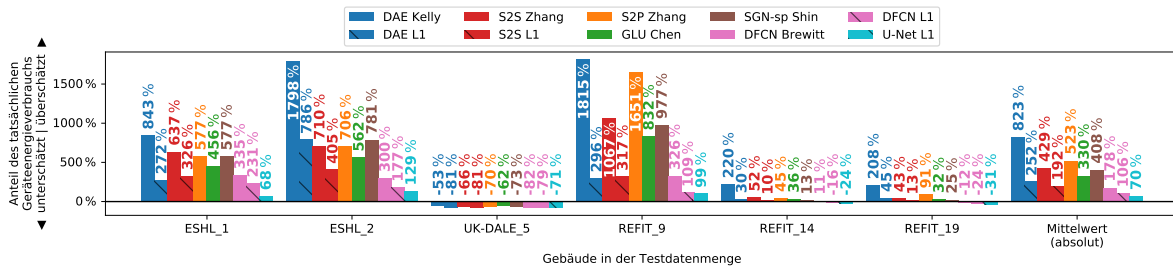


Abbildung 7.14: Anteil fehlgeschätzten Gesamtenergieverbrauchs von Mikrowellen.

Geschirrspüler

Die Ergebnisse der Energiebilanzschätzung bei Geschirrspülmaschinen sind in Abbildung 7.15 dargestellt. Auffällig ist die starke Überschätzung des Gesamtenergieverbrauchs in REFIT-Gebäude 14. Im Mittel erreicht der mit L₂-Metrik trainierte DFCN-Ansatz die genauesten Energiebilanzschätzungen.

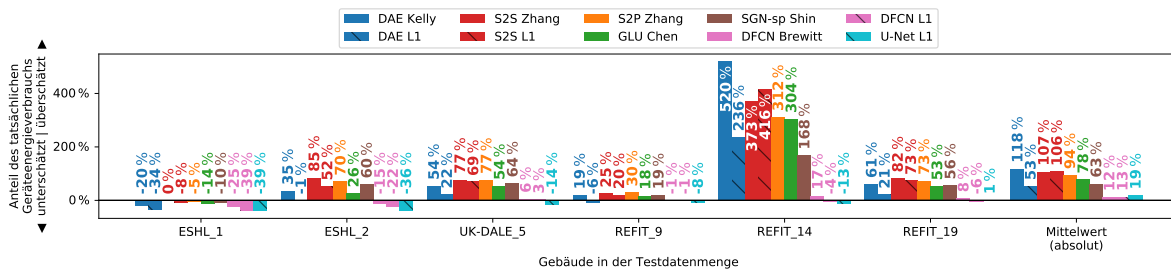


Abbildung 7.15: Anteil fehlgeschätzten Gesamtenergieverbrauchs von Geschirrspülmaschinen.

Wasserkocher

In Abbildung 7.16 ist die Gesamtenergieverbrauchsschätzung der Disaggregationsverfahren bei Wasserkochern dargestellt. Der mit L_1 -Verlust trainierte Sequence-to-Sequence-Ansatz (S2S) erkennt keines der Wasserkocher in den Testgebäuden, was zu einer Unterschätzung von 100 % führt. Im ESHL wird von vielen Ansätzen zuviel Energieverbrauch dem Wasserkocher zugeordnet. Im Mittel kann das mit L_1 -Verlust trainierte U-Net den Gesamtenergieverbrauch des Wasserkochers am genauesten abschätzen.

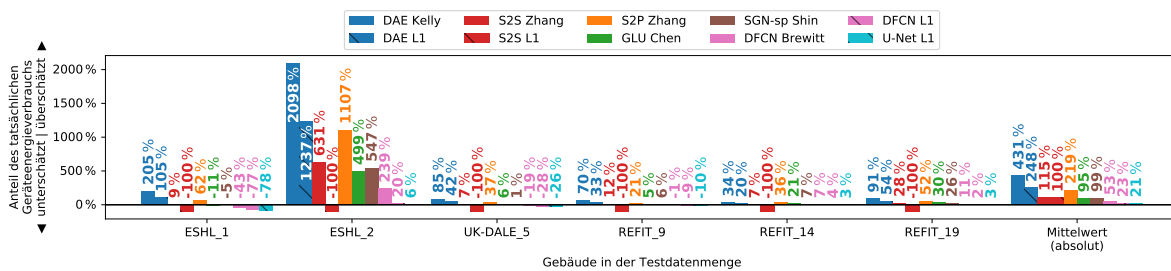


Abbildung 7.16: Anteil fehlgeschätzten Gesamtenergieverbrauchs von Wasserkochern.

7.3 Detektion von Geräteaktivität

In diesem Anwendungsfall wird im zeitlichen Verlauf klassifiziert, ob die jeweilige Geräteklasse aktiv ist. Klassische NILM-Ansätze werden in der Literatur auf diesen Anwendungsfall evaluiert. Da dabei meistens diejenigen Haushaltsgeräte disaggregiert werden, die hauptsächlich inaktiv sind, wird in der Literatur die F_1 -Metrik zur Evaluation verwendet. Bei Geräten wie dem Kühlschrank trifft die Annahme der überwiegenden Inaktivität jedoch nicht zu. Daher wird in dieser Arbeit hauptsächlich mit der (binären) MCC-Metrik evaluiert, die eine balancierte Klassifikationsmetrik auch ohne eine solche Annahme ist.

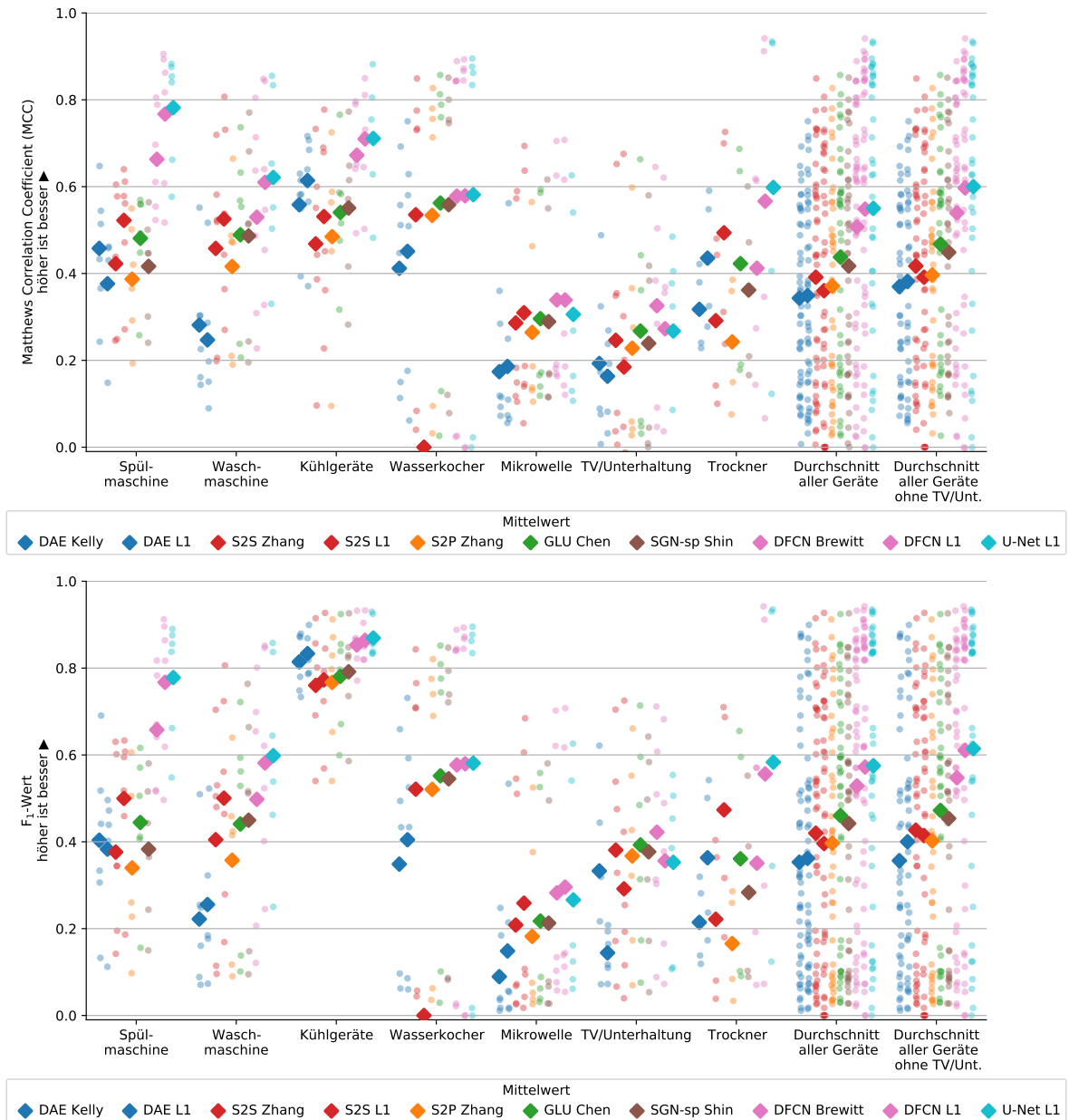


Abbildung 7.17: MCC und F1 von allen Ansätzen

In Abbildung 7.17 sind dennoch die MCC-Metrik und die F_1 -Metrik zum Vergleich dargestellt. Die Zahlenwerte sind tatsächlich bei fast allen Geräten ähnlich. Nur bei Kühlgeräten unterscheiden sich die Metrikwerte stärker: Mit der F_1 -Metrik erscheint die Klassifikationsgenauigkeit bei Kühlgeräten unberechtigt besser als mit der MCC-Metrik.

Mit beiden Metriken ist erkennbar, dass tendenziell die neueren Ansätze die Geräteaktivität besser klassifizieren. Der Einsatz der L_1 -Verlustfunktion verbessert die Klassifikationsleistung bei DFCN deutlich. Mit dem U-Net wird die Klassifikationsleistung nochmals geringfügig gesteigert.

Waschmaschinen und Wäschetrockner

In Abbildung 7.18 ist die Klassifikationsgenauigkeit der Verfahren bei Waschmaschinen dargestellt. In fast allen Testgebäuden erreicht zumindest ein Ansatz einen MCC-Wert von über 0,5. Nur die Aktivität der Waschmaschine während der ersten Messkampagne im ESHL wird schlecht erkannt. In den Gebäuden REFIT 14 und 19 wird die Waschmaschine besonders gut erkannt. Von den Ansätzen erreicht das mit L_1 -Verlust trainierte U-Net im Mittel die beste Klassifikationsbewertung.

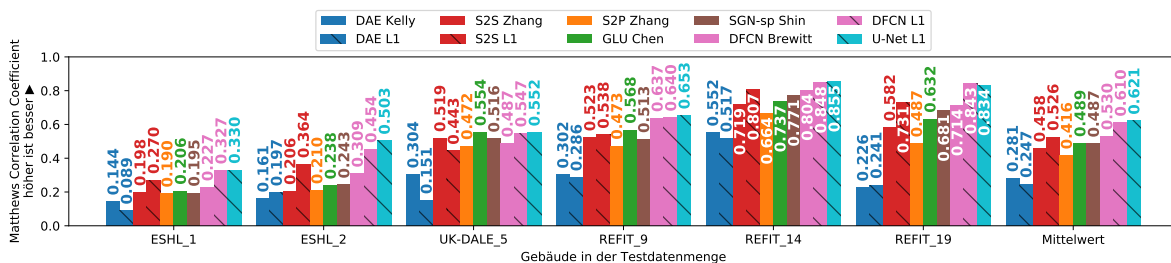


Abbildung 7.18: Bewertung der Klassifikationsleistung von Disaggregationsverfahren bei Waschmaschinen.

Kühlgeräte

Die Bewertung der Aktivitätserkennung bei Kühlgeräten ist in Abbildung 7.19 mit der MCC-Metrik dargestellt. Dabei wird die Kühlgeräteaktivität in der Regel mit einem MCC-Wert über 0,6 gut erkannt. Nur bei den Daten der zweiten Messkampagne des ESHL ist die Erkennungsrate bei allen Ansätzen niedriger. Im Mittel erreicht auch hier das mit L_1 -Verlust trainierte U-Net die beste Bewertung.

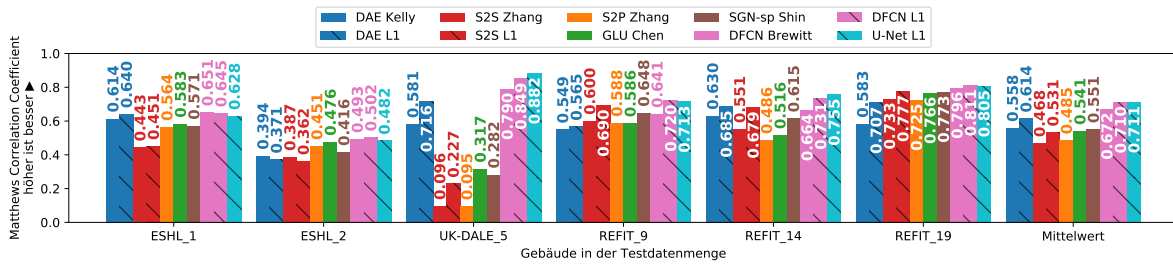


Abbildung 7.19: Matthews Correlation Coefficient zur Bewertung von Verfahren zur Erkennung der Aktivitäten von Kühlgeräten.

Trockner

In Abbildung 7.20 ist die Klassifikationsgenauigkeit bei Wäschetrocknern dargestellt. Die Ergebnisse zeigen, dass die Trocknermodelle von REFIT 14 und 19 bereits aus dem Training bekannt sind. Die Aktivität des Trockners im ESHL wird von allen Ansätzen knapp erkannt. Der beste Ansatz im Mittel ist auch hier wieder das U-Net.

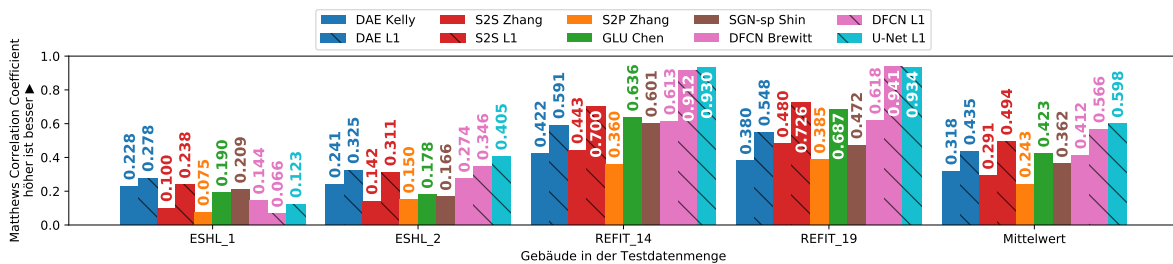


Abbildung 7.20: Bewertung der Klassifikationsleistung von Disaggregationsverfahren bei Trocknern.

TV und Unterhaltungselektronik

In Abbildung 7.21 ist die Aktivitätserkennung bei der Geräteklasse Fernseher und Unterhaltungselektronik dargestellt. Wie bei der Analyse in Abschnitt 5.2 bereits festgestellt, können die Disaggregationsansätze die Aktivität von TV und Unterhaltungselektronik nur anhand deren Koinzidenz mit vorhandener Gesamtgebäudelast erkennen. In REFIT 9, 14 und 19 ist diese Strategie erfolgreich. In den verbleibenden Gebäuden wird diese Geräteklasse praktisch nicht erkannt. Im Mittel erreicht der Ansatz DFCN mit L₂-Verlust die beste Bewertung.

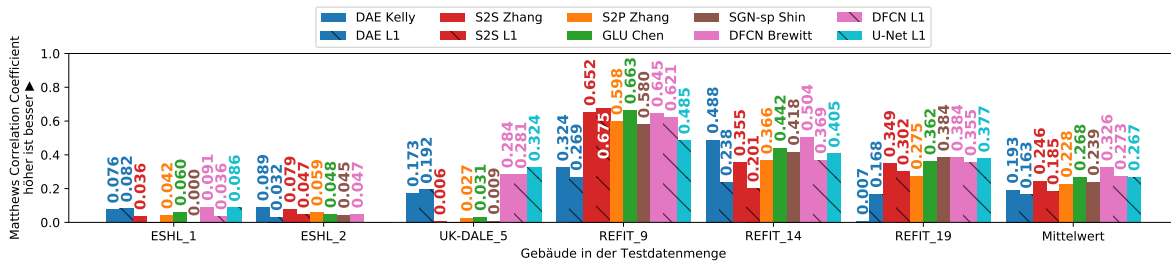


Abbildung 7.21: Bewertung der Klassifikationsleistung von Disaggregationsverfahren bei Fernsehern und Unterhaltungselektronik.

Mikrowellen

In Abbildung 7.22 ist die Aktivitätserkennung der Disaggregationsverfahren bei Mikrowellen dargestellt. In den Testgebäuden REFIT 14 und 19 scheint das Mikrowellenmodell den Ansätzen bekannt zu sein und hier werden MCC-Werte über 0,5 erreicht. Bei den übrigen Testgebäuden wird die Mikrowelle kaum erkannt und der MCC-Wert liegt stets unter 0,3. Im Durchschnitt liefert der DFCN-Ansatz mit L_1 -Verlust die beste Erkennung der Mikrowellenaktivität.

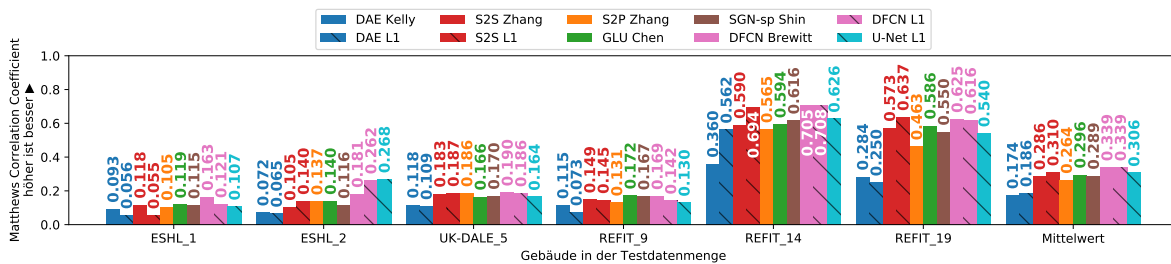


Abbildung 7.22: Bewertung der Klassifikationsleistung von Disaggregationsverfahren bei Mikrowellen.

Geschirrspüler

Die Ergebnisse der Klassifikation von Geräteaktivität bei der Geschirrspülmaschine sind in Abbildung 7.23 dargestellt. Bei allen Gebäuden existiert mindestens ein Ansatz, der die Geschirrspülmaschine mit einem MCC-Wert von über 0.6 gut erkennt. Im Mittel über alle Testgebäude erreicht das mit L_1 -Verlust trainierte U-Net die höchste Klassifikationsleistung.

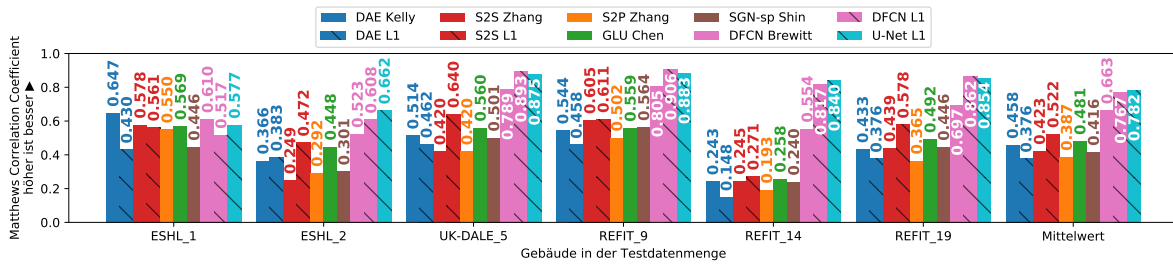


Abbildung 7.23: Matthews Correlation Coefficient für Verfahren zur Erkennung von Geschirrspülmaschinenaktivität.

Wasserkocher

In Abbildung 7.24 ist die Klassifikationsgenauigkeit bei Wasserkochern dargestellt. Die Wasserkocheraktivität im ESHL wird kaum erkannt. Einige Ansätze schätzen sogar eine Aktivität, die geringfügig negativ zur tatsächlichen Aktivität korreliert ist¹. In allen übrigen Testgebäuden wird die Nutzung des Wasserkochers mit einem MCC-Wert über 0,8 bei einigen Ansätzen hingegen sehr gut erkannt. Möglicherweise liegt das an dem Unterschied zwischen deutschen und englischen Wasserkochermodellen.

Der mit L_1 -Verlust trainierte Sequence-to-Sequence-Ansatz (S2S) schätzt stets Inaktivität und dies führt zu einem Korrelationskoeffizient von 0. Im Durchschnitt über alle Testgebäude erreicht das mit L_1 -Verlust trainierte U-Net die beste Aktivitätsklassifikation bei Wasserkochern.

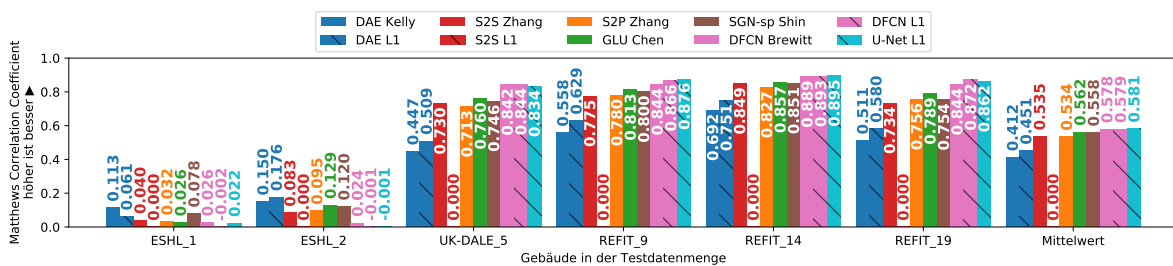


Abbildung 7.24: Bewertung der Klassifikationsleistung von Disaggregationsverfahren bei Wasserkochern.

¹Ein negativer Korrelationskoeffizient tritt ein, wenn vorhandene Geräteaktivität nicht erkannt wird (hohe Falsch-Negativ-Rate) und stattdessen dem inaktiven Gerät fälschlicherweise Last zugeordnet wird (hohe Falsch-Positiv-Rate).

7.4 Analyse

Zunächst soll die Frage beantwortet werden, inwieweit Deep-Learning-basierte Disaggregationsverfahren auch bei unbekanntem Haushalten akzeptable Ergebnisse liefern (Abschnitt 7.4.1). In Kapitel 6 wird nicht abschließend beantwortet, ob es durch einen der Ansätze gelingt, ausschließlich realisierbare Lastgänge als Gerätestrangsschätzungen auszugeben. Dies wird in Abschnitt 7.4.2 untersucht. Abschnitt 7.4.3 geht dem Einfluss der L_1 -Verlustfunktion auf die Disaggregationsqualität nach.

Abschließend wird in Abschnitt 7.4.4 ermittelt, wieviel Rechenaufwand das Training und die Disaggregation mit den evaluierten Netzmodellen in Anspruch nimmt.

7.4.1 Übertragbarkeit der Disaggregationsverfahren auf unbekannte Haushalte

Im oberen Diagramm von Abbildung 7.1 ist bei den Geräteklassen Spülmaschine, Waschmaschine, Kühlgeräte, Wasserkocher sowie Trockner zu erkennen, dass es mehrere Ansätze gibt, die in manchen Gebäuden einen energiebasierten Matthews Correlation Coefficient über 0,8 erreichen. Da in allen Experimenten ausschließlich Testgebäude evaluiert worden sind, deren Daten während des Trainings nicht verwendet wurden, kann somit bestätigt werden, dass Deep-Learning-basierte Disaggregationsverfahren grundsätzlich auch auf unbekannte Haushalte generalisieren.

Die Aussage hängt jedoch davon ab, wie ähnlich sich die erlernten Gerätemodelle aus dem Trainingsdatensatz und die Modelle der Testgebäude sind. So betreffen die niedrigen Korrelationskoeffizienten (unter 0,2) bei Wasserkochern in Abbildung 7.1 das für den deutschen Markt produzierte Wasserkochermodell im KIT Energy Smart Home Lab (ESHL), das eine niedrigere Leistung aufweist als die englischen Wasserkocher im Trainingsdatensatz. Auch bei Waschmaschinen scheint es größere Modellunterschiede zu geben. Viele Modelle werden nur befriedigend mit einem Korrelationskoeffizienten um 0,3 erkannt. Hingegen werden die Kühlgeräte und Spülmaschinen auch in den Testgebäuden gut erkannt: Selten fällt der Korrelationskoeffizient unter 0,5.

Wenn die Ergebnisse nach Geräteklasse und Testhaushalten gruppiert werden, fällt der vergleichsweise geringe Unterschied zwischen den Ansätzen auf. Entweder werden die Geräte der Testhaushalte systematisch von allen Ansätzen schlecht erkannt oder sie werden von allen Ansätzen gut erkannt. Dabei schneiden neuere Ansätze relativ betrachtet besser ab.

Mit TV und Unterhaltungselektronik wird eine Geräteklasse evaluiert, die grundsätzlich schwer zu disaggregieren ist. In Abschnitt 5.2 wurde bereits vermutet, dass bei der Erkennung dieser Klasse nicht Lastprofilmerkmale entscheidend sind, sondern lediglich die Korrelation

zwischen hoher Gebäudelast und Aktivität dieser Geräteklasse. Entsprechend liefern alle Ansätze schlechte Evaluationsergebnisse.

Anhand der Ergebnisse bei Trocknern ist der limitierende Effekt von zu wenig Trainingsdaten auf die Generalisierbarkeit besonders deutlich erkennbar. Alle verwendeten Datensätze zusammengenommen haben eine geringe Anzahl an Trocknermodellen. Zudem wurden viele der Modelle als Testdaten statt als Trainingsdaten verwendet. Daher gibt es eine starke Streuung der Ergebnisse mit einzelnen sehr guten und einzelnen sehr schlechten Korrelationskoeffizienten.

7.4.2 Reproduktion realistischer Lastgänge

In Abschnitt 6.1.8 wurde bereits exemplarisch gezeigt, dass der Einsatz eines neuronalen Lastsequenzgenerators nicht dazu führt, dass ausschließlich realisierbare Lastgänge erzeugt werden. Tatsächlich sind sowohl Zwischenergebnisse als auch Endergebnisse als eindeutig nicht realisierbare Gerätelastgänge identifizierbar. Dies ist der Fall, obwohl für zufällig gezogene Eingabevektoren in der Regel augenscheinlich realisierbare Gerätelastgänge erzeugt werden. Somit ist der Versuch gescheitert, durch einen dedizierten Lastsequenzgenerator die Disaggregation zu verbessern.

Mit den weiteren neu entwickelten Ansätzen in Kapitel 6, insbesondere der zusätzlichen Trainingsbewertung mit einem Diskriminator und dem U-Net, werden nur kleine Verbesserungen in der Disaggregationsqualität erreicht. Der Einsatz der L_1 -Verlustfunktion führt hingegen zu einer deutlichen Verbesserung der Disaggregation. In diesem Abschnitt wird untersucht, inwiefern diese Maßnahmen mit realistischeren Lastgängen korrelieren.

Wie in Abschnitt 5.2 bereits festgestellt, können Kühltischlastgänge in der Nacht sehr genau nachgefahren werden. Sobald jedoch eine höhere Gebäudelast herrscht, wird wie in Abbildung 7.25 ab etwa 7 Uhr, der Kühltischlastgang nur noch schwer erkannt. Die endgültige Schätzung der bekannten Ansätze entspricht in diesen Fällen keinen realisierbaren Kühlgerätelastgängen.

Rechts unten in der Abbildung 7.25 sind die Schätzungen der Ansätze dargestellt, die mit dem L_1 -Diskriminator-Konjugat trainiert wurden. Tatsächlich sind diese Lastgänge über die meiste Zeit schwerer unterscheidbar von tatsächlich gemessenen Lastgängen und, nicht wie bei den anderen Schätzung, sofort als Disaggregationsschätzungen erkennbar. Bei U-Net L_1 +cGAN bricht erst bei anhaltender hoher Last das Modell zusammen und es produziert eindeutig unrealisierbare Kühlgerätelastgänge.

Bei den Ansätzen mit Diskriminator sehen die Ergebnisse zwar plausibler aus, jedoch sind die Schätzungen genauso wenig korrekt wie die der Ansätze mit eindeutig nicht realisierbaren Lastgängen. In der numerischen Bewertung der Ansätze mit Diskriminator in Kapitel 6

ergibt sich kein Vorteil bezüglich der Disaggregationsmetriken. Die Darstellung von realisierbaren Lastgängen kann jedoch einen großen Unterschied in der Benutzerakzeptanz von Produkten oder Diensten ausmachen, die auf Disaggregation beruhen.

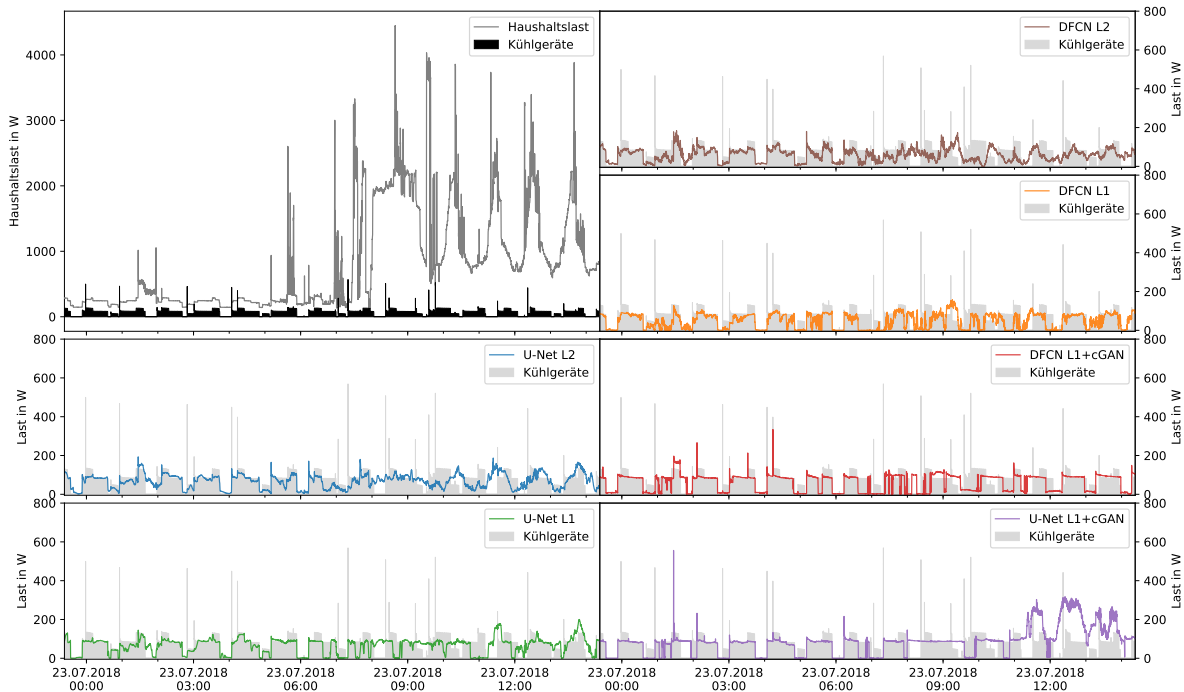


Abbildung 7.25: Mit neuen Ansätzen disaggregierter Kühlgeräteleistungsausschnitt aus ESHL 2. Der Ausschnitt ist identisch zu dem in Abb. 5.11.

Die größte Verbesserung durch den Austausch der L_2 - mit der L_1 -Verlustfunktion bei dem DFCN-Modell wird beim Trockner in Testgebäude REFIT 19 erreicht. Der eMCC-Wert steigt dabei von 0,790 auf 0,922. In Abbildung 7.26 ist die Verbesserung der Disaggregationsqualität visuell nur im Detail erkennbar.

Keines der in Kapitel 6 beschriebenen Ansätze garantiert die Ausgabe von ausschließlich realisierbaren Lastgängen. Der Trockner im ESHL ist den neuronalen Modellen nicht bekannt. In einem Ausschnitt mit hoher Gebäudelast wie in Abbildung 7.27 ist eindeutig zu sehen, dass keines der Ansätze realisierbare Geräteleistungen erzeugen kann.

Damit steht fest, dass alle neu entwickelten Ansätze den Ausgaberaum der neuronalen Netze nicht auf ausschließlich realisierbare Lastgänge einschränken können. Beispiele aus der Forschung zum Thema Adversarial Training zeigen ebenfalls, dass es grundsätzlich schwer ist, künstliche neuronale Netze auf einen mathematisch schwer formulierbaren Raum zu beschränken (siehe beispielsweise Szegedy et al., 2014). Am Beispiel der Kühlgeräte ist erkennbar, dass das Training mit dem L_1 -Diskriminator-Konjugat dennoch einen Schritt in diese Richtung darstellt. Das Reproduzieren von realisierbaren und plausibel aussehenden

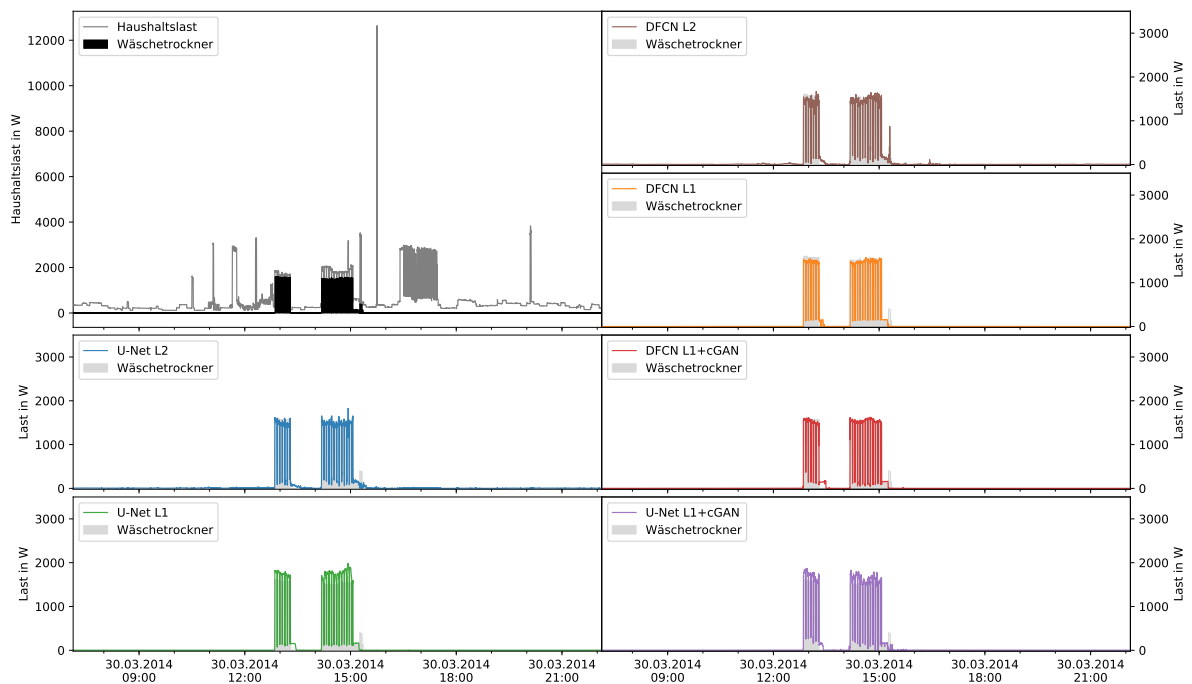


Abbildung 7.26: Mit neuen Ansätzen disaggregierter Trocknerlastgang aus REFIT 19.

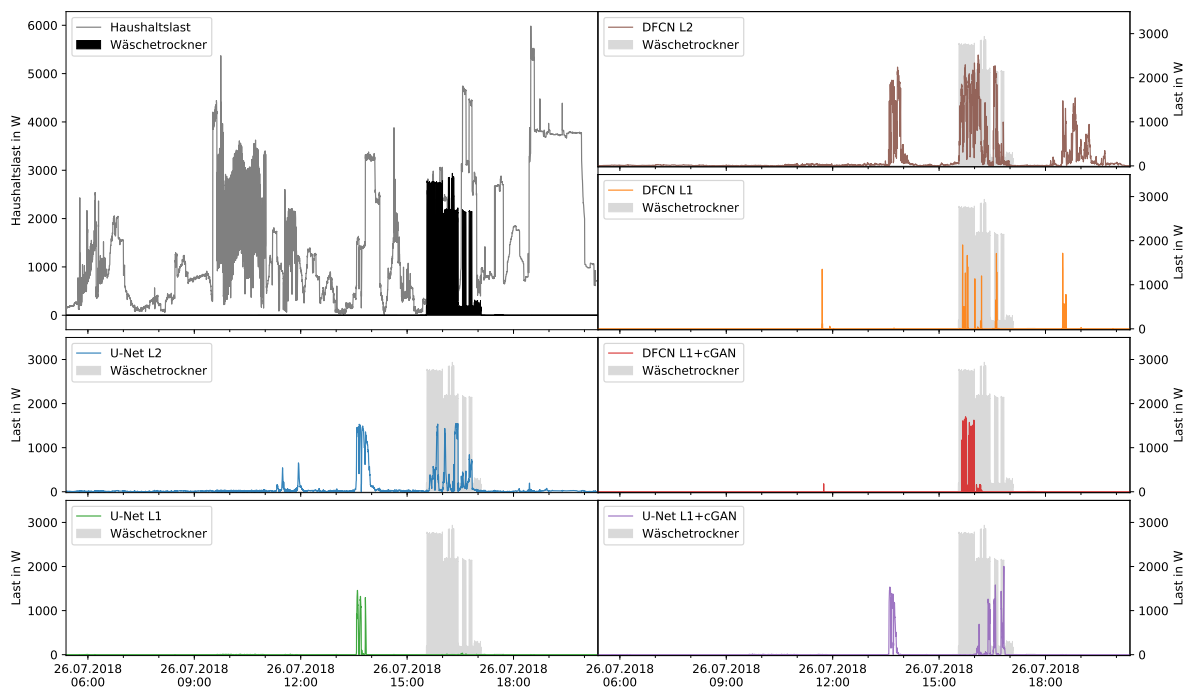


Abbildung 7.27: Mit neuen Ansätzen disaggregierter Trocknerlastgangsausschnitt aus ESHL 2. Der Ausschnitt ist identisch zu dem in Abb. 5.14.

Gerätelastgängen führt jedoch nicht zu der erwarteten Verbesserung der Disaggregationsleistung.

Die schlechte Disaggregationsqualität wird in vielen Fällen daher vermutlich nicht durch die Unfähigkeit der neuronalen Modelle verursacht, realisierbare Lastgänge zu produzieren, sondern entspringt der Herausforderung, die Geräteaktivität auch bei mehreren parallel laufenden Geräten zu erkennen.

7.4.3 Effekt der L_1 -Verlustfunktion auf das Training der Disaggregationsansätze

Der Einsatz der L_1 -Verlustfunktion hat einen positiven Effekt auf die Disaggregationsqualität, was vor allem am höheren energiebasierten Matthews Correlation Coefficient der entsprechenden Ansätze in Abbildung 7.1 zu sehen ist. Ein sinkender mittlerer absoluter Fehler (MAE) alleine reicht für diese Erkenntnis nicht aus, da L_1 -Verlust und MAE essentiell die gleiche Funktion sind und dies erwartbar war. Aber auch bei den Metriken der anderen Anwendungsfälle (siehe Abb. 7.9 und 7.17) ist erkennbar, dass die Ansätze, die mit einer L_1 - statt der L_2 -Verlustfunktion trainiert worden sind, bessere Ergebnisse liefern. Über die Ursache dieses Phänomens wird an dieser Stelle nur spekuliert:

Möglichweise basieren die besseren Ergebnisse auf der höheren Robustheit von L_1 -trainierten Modellen gegenüber Ausreißern im Trainingsdatensatz. Da hier viele unterschiedliche Datensätze kombiniert worden sind, ist es wahrscheinlich, dass trotz Bereinigung einige fehlerhafte Trainingsdaten verblieben sind. Der Prognosefehler, der durch diese Ausreißer erzeugt wird, wird durch den L_2 -Verlust besonders stark gewichtet und so bekommen die Ausreißer einen unerwünscht hohen Einfluss auf die Modellparameter.

Eine andere Erklärungsmöglichkeit liefern möglicherweise die optimalen Maximum-Likelihood-Schätzer von maschinellen Lernmodellen, die mit einem L_2 - bzw. L_1 -Verlust erstellt sind. Für ein jeweiliges Eingabedatum entspricht der Maximum-Likelihood-Schätzer eines Modells bezüglich des L_2 -Verlusts dem arithmetischen Mittelwert der Ausgabedaten (siehe Goodfellow, Bengio et al., 2016, Abschnitt 5.5.1). Bei einem L_1 -Verlust entspricht der Maximum-Likelihood-Schätzer dem Median der Ausgabedaten. Übertragen auf Lastgänge sind die Schätzung der mit L_1 -Verlust trainierten Modelle insofern realistischer, als dass – zumindest für jedes Messintervall unabhängig betrachtet – die geschätzte Last näher an einer tatsächlich gemessenen Last liegt.

7.4.4 Rechenaufwand

Im Folgenden wird der Rechenaufwand der evaluierten Ansätze abgeschätzt. Dazu wird die theoretische Anzahl der Multiply-and-Accumulate-Operationen (MAC) gezählt, die benötigt

wird, um eine Lastsequenz durch das neuronale Netz zu verarbeiten.

Die für die Anwendung von neuronalen Netzen benötigten Matrizen- bzw. Tensorenmultiplikationen werden mit diesen Operationen durchgeführt. Auf modernen Prozessoren und Graphikkarten ist dies eine elementare Operation, die nicht mehr Zyklen benötigt als eine Fließkommamultiplikation.

Aus der Anzahl der MACs für die Inferenz eines Lastgangs wird abgeschätzt, wie viele Operationen zur Disaggregation eines Messintervalls durchschnittlich benötigt werden. Auch wird daraus die Anzahl der Operationen für das Training des neuronalen Netzmodells abgeschätzt.

In dieser Rechnung wird nicht berücksichtigt, wie viele Rechenoperationen die Bereitstellung von Trainingspaaren in Anspruch nimmt. Dies ist eine additive Konstante und bei fast allen Ansätzen gleich, da während des Trainings die gleiche Mini-Batch-Größe und Anzahl an Trainingsiterationen verwendet wird. Nur beim Training eines W-GAN-Generators werden mehr Trainingsbeispiele pro Iteration verwendet.

Ebenfalls nicht angegeben ist die Anzahl der Rechenoperationen für die Berechnung des Gradienten und für die Aktualisierung der Parameter gemäß des jeweiligen Optimierungsverfahrens. Die Anzahl der dafür benötigten Operationen ist proportional zur Anzahl der Modellparameter.

Disaggregationsaufwand

Der Disaggregationsaufwand ergibt sich aus der Anzahl der Rechenoperationen, die durchschnittlich für die Berechnung eines Messintervalls benötigt wird. Dazu muss berücksichtigt werden, dass aufgrund des Sliding-Window-Ansatzes mehrere Lastsequenzen durch das neuronale Modell inferiert werden, bevor das Endergebnis zusammengeführt wird.

In Tabelle 7.1 ist die theoretische Anzahl an Rechenoperationen zur Berechnung eines Messintervalls aufgeführt. Multipliziert mit der Testdatenmenge ergibt dies die theoretische Rechenoperationsmenge für die in dieser Arbeit durchgeführten Evaluationen, wie in Abbildung 7.28 graphisch dargestellt ist.

Die Anzahl der Rechenoperationen für die Vorverarbeitung der Haushaltlastsequenz, das Zusammenführen der Lastsequenzen sowie die Nachbearbeitung der Gerätelastsequenzen ist im Vergleich zur Anwendung des neuronalen Netzes vernachlässigbar klein und wird hier nicht berücksichtigt.

Tabelle 7.1: Der Rechenaufwand beim Training ist proportional zu den MACs pro Inferenz. Der Aufwand für die Disaggregation ist proportional zu den MACs pro Messintervall.

Ansatz	Sequenzlänge	Parameter / 10^6	MACs pro Inferenz / 10^6	Inferenzen pro Messintervall	MACs pro Messintervall / 10^6
DAE	128	1,31	1,32	1 / 16	0,08
	288	5,90	5,93	1 / 16	0,37
	512	17,83	17,88	1 / 16	1,12
	1024	69,22	69,31	1 / 16	4,33
	1536	154,17	154,30	1 / 16	9,64
DAE+GAN	128	2,64	72,08	1 / 16	4,51
	288	7,06	76,51	1 / 16	4,78
	512	18,76	88,22	1 / 16	5,51
	1024	69,62	139,11	1 / 16	8,69
	1536	154,04	223,55	1 / 16	13,97
GLU	800	1,20	50,10	1 / 5	10,02
S2P	599	30,71	53,19	1	53,19
S2S	599	31,32	53,81	1	53,81
S2S+GAN	599	32,25	124,18	1	124,18
SGN-sp	432	44,40	87,68	1	87,68
DFCN	4105	0,46	1661,82	1 / 2053	0,81
U-Net	3626	1,12	2156,14	1 / 512	4,21

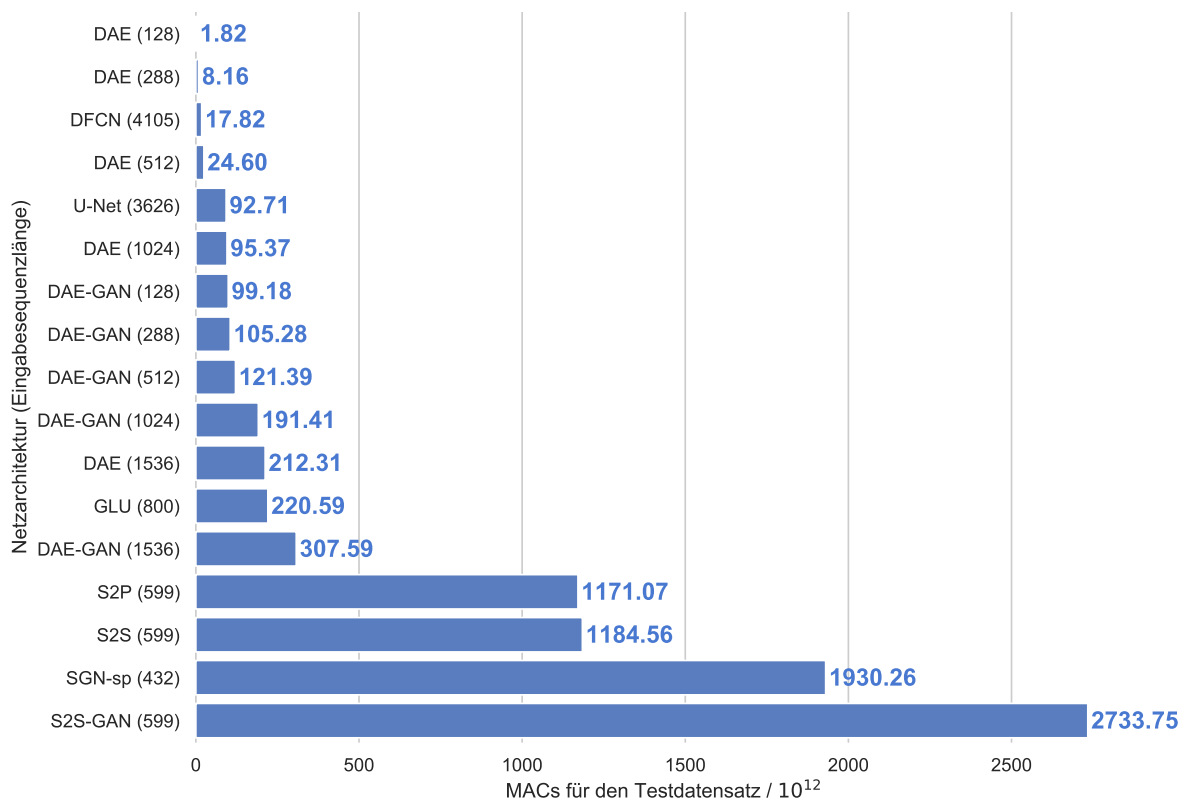


Abbildung 7.28: Theoretische Anzahl an Rechenoperationen für die Disaggregation des Testdatensatzes.

Trainingsaufwand

Für das Training durchläuft das neuronale Netz eine feste Anzahl an Iterationen. In dieser Arbeit gibt es keinen frühzeitigen Abbruch des Trainings bei Erfüllung eines bestimmten Kriteriums.

Der gesamte Trainingsaufwand ist somit zunächst proportional zu dem in Tabelle 7.1 aufgeführten Rechenaufwand für eine Inferenz. Bei Ansätzen mit Generative Adversarial Network (GAN) muss noch dessen Trainingsaufwand dazugezählt werden. Dazu ist der Aufwand für das GAN-Training in Tabelle 7.2 aufgeführt.

Abschließend ergibt sich für die Ansätze ein Gesamttrainingsaufwand, der in Abbildung 7.29 graphisch dargestellt ist.

Tabelle 7.2: Rechenaufwand für das Training und die Nutzung von GAN.

	MACs		Parameter		Durchläufe		MACs Gesamt
	D	G	D	G	D	G	
DC-GAN_1024	9,60	70,88	1,04	1,04	3	1	99,67
W-GAN_1024	9,60	70,88	1,04	1,04	11	5	459,97
DAE-cGAN_1024	68,26	69,31	68,18	68,18	3	1	274,11
S2S-cGAN_599	51,00	53,81	29,22	29,22	3	1	206,80

7.5 Zusammenfassung

Aus den vorherigen Betrachtungen ergeben sich folgende wesentlichen Erkenntnisse:

Die Reproduktion des Gerätelastgangs wird, je nach Gerät, durch einen der drei Ansätze Dilated Fully Convolutional Network (DFCN) mit L_2 - oder L_1 -Verlust und U-Net mit L_1 -Verlust am besten durchgeführt. Der durchschnittliche energiemengenbasierte Matthews Correlation Coefficient (eMCC) liegt bei allen drei Ansätzen auf einem ähnlichen Niveau. Mit DFCN und L_1 -Verlust werden im Mittel die niedrigsten absoluten Fehler (MAE) erreicht. Auch werden bei den meisten Geräten mit dem DFCN L_1 die höchsten energiemengenbasierten Korrelationskoeffizienten erreicht.

Die genauesten Energieverbrauchsbilanzen werden im Schnitt durch das U-Net mit L_1 -Verlust erstellt.

Auch die Klassifikation bzw. Detektion von Geräteaktivität wird von dem U-Net und dem DFCN, jeweils mit L_1 -Verlust, am genauesten durchgeführt.

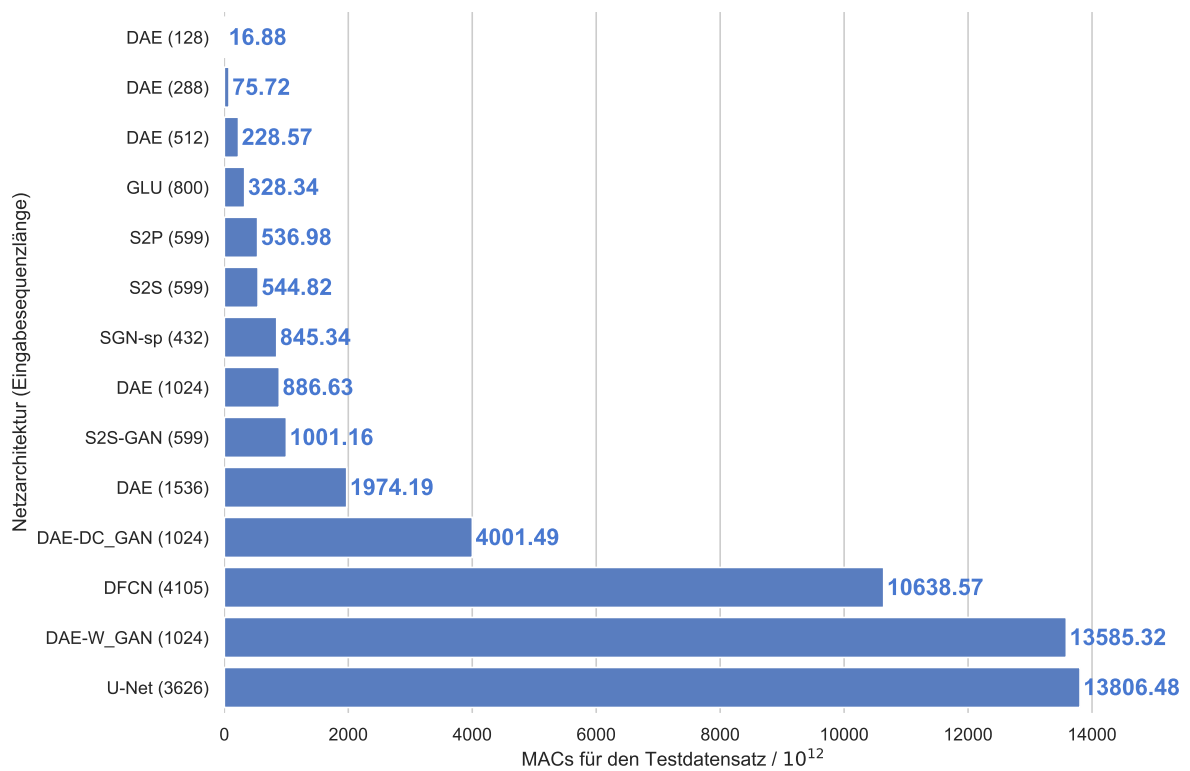


Abbildung 7.29: Theoretische Anzahl an Rechenoperationen für das Training eines Netzmodells.

Der Einsatz des L_1 -Verlustes sollte in zukünftigen Disaggregationsexperimenten berücksichtigt werden. Bei Geräten, die von bekannten Ansätzen bereits schlecht erkannt worden sind, kann die Verwendung der L_1 -Verlustfunktion allerdings auch zu schlechteren Ergebnissen führen.

Das U-Net verbessert die Disaggregationsleistung nochmals um einen kleinen Betrag. Es hilft besonders bei Geräten, die ohnehin mit bekannten Ansätzen gut disaggregiert werden konnten.

Die Übertragbarkeit der Disaggregationsverfahren auf unbekannte Gebäude hängt davon ab, ob die Gerätemodelle bereits aus dem Training bekannt sind. Ebenfalls hängt die Disaggregationsqualität stark davon ab, wieviele Geräte gleichzeitig benutzt werden.

Mit dem L_1 -Diskriminator-Konjugat als Verlustfunktion erzeugen die Disaggregationansätze öfter realisierbare Gerätelastgänge, jedoch verbessern sich dadurch die Disaggregationsmetriken nicht wesentlich. Was zu einer Verbesserung der Disaggregationsmetriken führt, ist alleine auf den L_1 -Verlust zurückzuführen.

Der Ansatz DFCN und das in dieser Arbeit entworfene U-Net mit dilatierten Faltungen benötigen zwar einen hohen Rechenaufwand beim Training. Dafür benötigt die Disaggregation nur einen Durchlauf, sodass der Anwendungsaufwand niedriger liegt als bei den meisten älteren Ansätzen.

Zusammenfassung und Ausblick

In dieser Dissertation werden Non-Intrusive Appliance Load Monitoring (NILM) genannte Verfahren zur Aufschlüsselung des Energieverbrauchs einzelner elektrischer Geräte anhand des Gesamthaushaltsverbrauchs zur Energieverbrauchsvisualisierung und damit der Bereitstellung einer Datengrundlage zur Energieoptimierung bzw. zur Energieverbrauchsreduktion untersucht. Die evaluierten NILM-Ansätze basieren auf dem Einsatz künstlicher neuronaler Netze zur Erkennung der Energieverbrauchsmuster einzelner Geräteklassen aus der Gesamthaushaltsenergiemessung und kommen somit ohne Installation zusätzlicher Hardware an den einzelnen Geräten aus. Existierende, auf Deep Learning basierende NILM-Ansätze wurden insbesondere im Hinblick auf deren Fehleranfälligkeit bei der Energieverbrauchsdisaggregation bei unbekanntem Haushalten und unbekanntem Haushaltsgeräten, die nicht bereits aus dem Training bekannt waren, anhand eines einheitlichen Experimentaufbaus untersucht. Als typische Disaggregationsprobleme konnten die folgenden Phänomene identifiziert werden (siehe Kapitel 5):

Unbekannte Gerätemodelle in den Testgebäuden

Da bei manchen Testgebäuden kein bekannter Ansatz gute Disaggregationsergebnisse liefern kann, liegt eine aktuelle Limitation bekannter NILM-Verfahren in der Tatsache, dass in den Trainingsdaten Gerätemodelle vorhanden sein müssen, welche den Geräten im Testgebäude ähnlich sind. Geräte mit deutlich geringerer Leistung als Vergleichsgeräte können bspw. nicht erkannt werden, wenn diese nicht auch im Trainingsdatensatz vorkommen. Um NILM großflächig einsetzen zu können, ist daher für eine große Bandbreite an Gerätemodellen im Trainingsdatensatz zu sorgen.

Gleichzeitigkeit von Geräteaktivität in manchen Testgebäuden

Wenn sich bei gleichzeitiger Nutzung von elektrischen Geräten in Testgebäuden Last-

gänge und Gerätemerkmale überlagern, leidet die Disaggregationsqualität. Dies ist besonders deutlich bei der Schätzung von Kühlgerätelestgängen erkennbar. Der Schaltzyklus dieser Geräte wird zwar in der Nacht gut erkannt, jedoch nicht am Tag, während vermehrter Aktivität im Gebäude. Bei allen Geräteklassen sorgen parallele Geräteaktivitäten sowohl für falsch negativ als auch falsch positiv zugewiesene Energie.

Fehlende oder wenige gerätespezifische Merkmale im Lastgang

Bei einigen Geräteklassen, wie TV und Unterhaltungselektronik sowie Mikrowellen, fehlen eindeutige gerätespezifische Merkmale im Haushaltslastgang, sodass es den Disaggregationsansätzen erschwert wird, die Aktivität dieser Geräte zu erkennen. Teilweise basieren befriedigende Disaggregationsergebnisse lediglich auf der Koinzidenz von Unterhaltungselektronikaktivität und allgemein erhöhter Aktivität im Haushalt.

Ausgabe von unrealistischen Gerätelestgängen

Die von den NILM-Ansätzen disaggregierten Lastgangsschätzungen enthalten oftmals Merkmale, die in gemessenen Gerätelestgängen nicht vorkommen und von echten Geräten nicht reproduziert werden können. Hierbei handelt es sich beispielsweise um Lastrampen, Zwischenlastwerte und Rauschen. Als Ursache wird eine fehlende Konfidenz bei der Erkennung von Geräteaktivität angenommen, mit der das Modell die Geräteleast zu skalieren scheint. Dieser Effekt tritt auch bei erfolgreichen Disaggregationen auf, sodass Lastgangsschätzungen leicht als solche erkannt werden können.

8.1 Vorgehensweise

8.1.1 Ermittlung des Standes der Technik bestehender Deep-Learning-Ansätze

Da sich bisher kein einheitlicher Aufbau im Hinblick auf Datensatz oder regelmäßig stattfindende Wettbewerbe für Disaggregationsexperimente etabliert hat, ist der Stand der Technik basierend auf der reinen Literatur schwer zu bestimmen. Zudem ist die Nichtverfügbarkeit eines ausreichend großen Datensatzes mit vielen unterschiedlichen Geräteklassen und einer Vielzahl unterschiedlicher Gerätemodelle pro Geräteklasse ein weiteres Hindernis. Daher wurde basierend auf der Literatur verglichen, welche Ansätze für das Unbekannte-Gebäude-Szenario niedrige absolute Fehler (MAE) aufweisen. Als Referenzdienten der

- Denoising Autoencoder (DAE) von Kelly und Knottenbelt (2015a), sowie
- Sequence-to-Sequence (S2S) und Sequence-to-Point (S2P) von Zhang et al. (2016).

Berücksichtigt wurden nur Ansätze, die verglichen zu den Referenzansätzen im gleichen Szenario geringere MAE bei der Disaggregation aufweisen. Hierzu zählen (siehe Abschnitt 3.11):

- Dilated Fully Convolutional Network (DFCN) von Brewitt und Goddard (2018),
- Subtask Gated Network (SGN) von Shin et al. (2018) und
- Gated Linear Unit (GLU) von Chen, Wang et al. (2018).

8.1.2 Szenario und Aufbau der Experimente

Bei den durchgeführten Experimenten wurde hauptsächlich evaluiert, wie genau sich die Lastgänge von ausgewählten Geräteklassen durch Disaggregationsansätze rekonstruieren lassen, wenn ausschließlich der Haushaltslastgang zur Verfügung steht. Die Schätzung des Geräteaktivitätsverlaufs und die Schätzung des Gesamtenergieverbrauchs einer Geräteklasse waren sekundäre Evaluationsszenarien.

Die Auswahl der betrachteten Geräteklassen basierte aus praktischen Gründen auf deren Verfügbarkeit in öffentlichen Datensätzen. In dieser Studie wurden sieben Geräteklassen untersucht. Das Disaggregationsziel waren stets Geräteklassen, da Lastgänge von Geräten mit einem ähnlichen Funktionsprinzip ein kaum unterscheidbares Lastgangsprofil besitzen und so kaum zwischen Geräteinstanzen unterschieden werden kann. Die Bildung von Geräteklassen stellte dabei auch sicher, dass in vielen Gebäuden mindestens eine Geräteinstanz vorhanden ist. Die untersuchten Geräteklassen waren: (1) Kühl- und Gefriergeräte, (2) Waschmaschinen und Wäschetrockner, (3) Wäschetrockner, (4) Geschirrspülmaschinen, (5) Wasserkocher, (6) Mikrowelle sowie (7) TV- und Unterhaltungselektronik (siehe Abschnitt 4.2). TV und Unterhaltungselektronik sowie Trockner werden dabei selten in anderen Arbeiten verwendet.

In dieser Studie wurden öffentlich verfügbare Datensätze genutzt und miteinander in Kombination verwendet. Die Kombination aus mehreren Datensätzen erfolgt nur selten in anderen Arbeiten. Oft wird Training und Test auf einzelnen Datensätzen durchgeführt. Eine Zusammenlegung von Datensätzen ist aber notwendig, um ein real einsetzbares NILM-Verfahren zu erstellen. Die Auswahl der Datensätze wurde durch folgende Parameter motiviert:

- Verfügbarkeit von synchron gemessenen Haushalts- und Einzellastgängen,
- Einzel- und Haushaltslastgänge mit einem Messintervall im Bereich von 1 bis 8 Sekunden, um einen einheitlichen Bereich für die zeitliche Lastgangauflösung als Vergleichsgrundlage festzulegen,
- Auslegung der Haushaltsgeräte auf das europäische Energiesystem.

Anhand dieser Kriterien wurden folgende Datensätze ausgewählt (siehe Abschnitt 4.3):

- Dutch Residential Energy Dataset (DRED),
- Electricity Consumption and Occupancy Dataset (ECO),

- Energiemessdaten des KIT Energy Smart Home Lab (ESHL),
- UK Domestic Appliance-Level Electricity (UK-DALE) und
- Personalised Retrofit Decision Support Tools for UK Homes using Smart Home Technology (REFIT).

Im Rahmen gemeinsamer Vorverarbeitungsschritte wurden Zeiträume mit fehlenden Messwerten identifiziert und die Datensätze um diese Messlücken bereinigt, um eine einheitliche Datenqualität zu erhalten und fehlerhafte Lastgänge während des Trainings und der Evaluation der Disaggregationsansätze größtenteils zu vermeiden (siehe Abschnitt 4.4.1). Messintervalle mit Messlücken wurden übersprungen, allerdings mussten solche unter 2 Minuten toleriert und durch das Auffüllen mit Ersatzwerten behandelt werden, um nicht zu viel Datenmaterial auszuschließen.

Trainingsaufbau

Zur überschneidungsfreien Trennung in Trainings- und Testdatenmenge wurden für den Test Gebäude mit einem möglichst vollständigen Satz der zuvor ausgewählten Haushaltsgeräteklassen bevorzugt. Zudem wurden Gebäude mit möglichst geringerer Datenmenge für das Testen bevorzugt, um der Trainingsphase möglichst wenig Daten zu entziehen (siehe Abschnitt 4.5).

Zum Training der neuronalen Netze wurden Datenpaare bestehend aus Eingabe- und Zielvektoren mit einer einheitlichen Vorgehensweise erzeugt, um die Vergleichbarkeit der untersuchten Verfahren untereinander sicherzustellen (siehe Abschnitt 4.6). Basierend auf den bestehenden Trainingspaargeneratoren (Balancierte Auswahl gemessener Lastgänge, Randomisierte Ausschnitte realer Lastgänge und Haushaltlastgangsynthese aus Aktivierungen) wurden zwei Modifikationen entworfen:

- Es wurde ein *Zufallssegmentgenerator mit ausbalancierter Gebäude- und Aktivierungshäufigkeit* eingesetzt, um eine unkontrollierte Gewichtung, welche nicht die Vorkommenshäufigkeit der Geräte in Haushalten repräsentiert, zu vermeiden. Auf diese Weise wurden die Lastsegmente zufällig so gewählt, dass alle Gebäude der Trainingsdatenmenge mit der gleichen Häufigkeit vertreten sind.
- Gegenüber der Haushaltlastgangsynthese aus Aktivierungen wurde die alternative Datenanreicherungsstrategie der *Haushaltlastgangsynthese durch Austausch von Gerätelastprofilen* verwendet, die die Zielgeräteleast von der Haushaltlast subtrahiert und Geräteaktivierungen zufällig neuplatziert, damit die Last aller anderen Geräte – auch jene, die nicht zu den ausgewählten Geräten zählen – im Haushaltlastgang erhalten bleibt.

Eine Trainingsepoche wurde durch eine feste Anzahl an Trainingsiterationen definiert, um sicherzustellen, dass die komplette Trainingsdatenmenge vollständig abgearbeitet wurde.

8.1.3 Analyse der Deep-Learning-Ansätze

Die Evaluation erfolgte auf Gebäudedaten, die nicht Bestandteil des Trainingsdatensatzes waren. Als Evaluationsmetriken wurden der Matthews Correlation Coefficient in Form einer energiebasierten Klassifikationsmetrik (eMCC), die Regressionsmetrik Mean Average Error (MAE) und der relative Fehler bei der Gesamtenergieverbrauchsschätzung (RETE) als geeignet ausgewählt (siehe Abschnitt 4.8).

Zur qualitativen Begutachtung der Disaggregationsqualität wurden die in Abschnitt 3.11 identifizierten Stand-der-Technik-Ansätze mit demselben in Kapitel 4 beschriebenen Experimentaufbau trainiert. Dabei ist die individuelle Parametrisierung dieser Ansätze für die Disaggregationsexperimente in Abschnitt 5.1 beschrieben.

8.2 Adressierung der Forschungsfragen

Im Verlauf der Arbeit wurde Bezug zu den Forschungsfragen genommen. Hier sind die wesentlichen Erkenntnisse und Verweise zur Beantwortung der Forschungsfragen zusammengetragen.

1. In welchem Maße generalisieren Disaggregationsverfahren auf unbekannte Haushalte?

Wie in Abschnitt 7.4.1 genauer erläutert, generalisieren alle untersuchten Deep-Learning-basierten Disaggregationsverfahren auch auf unbekannte Haushalte, sofern die darin installierten Gerätemodelle ähnlich zu einem der Gerätemodelle aus den Trainingsgebäuden sind. Die Daten aus dem KIT Energy Smart Home Lab (ESHL) werden beispielsweise ausschließlich als Testdaten verwendet. Die neuronalen Modelle konnten hier lediglich die Kühlgeräte und den Geschirrspüler gut erkennen.

Aus der Analyse der Disaggregation wird deutlich, dass die Disaggregation zusätzlich dadurch erschwert wird, wenn im Haushalt viele Aktivitäten gleichzeitig stattfinden. Dabei überlagert sich die Last vieler Haushaltsgeräte und damit überlagern sich auch die Lastgangsmerkmale der Geräte. Einzelne Geräte werden dadurch von den Ansätzen schwerer erkannt. Unbekannte Haushalte mit geringer gleichzeitiger Bewohneraktivität sind damit leichter zu disaggregieren.

2. Wie können bekannte Disaggregationsverfahren weiterentwickelt werden, um die Disaggregationsqualität auch auf unbekanntem Haushalten zu verbessern?

Dazu wird in Kapitel 5 zunächst das Phänomen beobachtet, dass vorhandene Ansätze oft Lastgänge als Zwischen- und Endergebnisse liefern, die offensichtlich nicht von realen Geräten stammen. In Kapitel 6 wird mit drei aufeinander aufbauenden Ansätzen versucht, diesem Phänomen zu begegnen.

2a. Wie kann sichergestellt werden, dass Ergebnisse bei der Disaggregation ausschließlich realisierbare Lastgänge enthalten und welchen Einfluss hat diese Strategie auf die Qualität der Disaggregation?

Wie in Abschnitt 7.4.2 analysiert, gelingt es mit Ansatz B aus Kapitel 5, die Anzahl realisierbarer Lastgänge in den Zwischen- und Endergebnissen zu erhöhen. In diesen Fällen bewirken die realisierbaren Lastgänge jedoch keine Verbesserung der Disaggregationsqualität.

Zudem verbleiben immer noch Situationen, in denen die Ausgabe von realisierbaren Lastgängen nicht erzwungen werden kann.

2b. Kann das U-Net-Konzept die Disaggregationsqualität verbessern?

Die Verwendung der U-Net-Architektur hat, verglichen mit dem Ansatz DFCN, keinen großen Einfluss auf die Disaggregationsqualität. Wie in Abschnitt 6.3 beschrieben, kann bei Geräten, die durch das DFCN ohnehin schon gut disaggregiert werden, die Disaggregationsqualität noch geringfügig gesteigert werden.

Einen viel stärkeren Einfluss auf die Disaggregationsqualität hat die Verwendung der L_1 -Verlustfunktion, wie in Abschnitt 7.4.3 näher erörtert.

8.3 Beiträge

8.3.1 Evaluation bestehender Ansätze

Ein Beitrag dieser Arbeit zur Weiterentwicklung von NILM-Verfahren liegt in der Evaluation der State-of-the-Art-Ansätze unter gleichen Trainings- und Evaluationsbedingungen. Hierfür wurden die ausgewählten Ansätze nachgebaut und anhand einer Kombination aus Datensätzen trainiert, sowie auf Gebäudedaten evaluiert, die nicht bereits aus dem

Training bekannt sind, um reale Herausforderungen eines Praxiseinsatzes zu simulieren. Somit konnte anhand von unterschiedlichen Geräteklassen, welche in der Literatur bisher nicht alle berücksichtigt wurden, eine Vergleichsbasis hergestellt werden. Aufgrund der Variationen des Experimentaufbaus oder der verwendeten Datensätze in den in der Literatur beschriebenen Ansätzen fehlte bisher eine solche Ausgangsbasis für Optimierungsvorhaben.

Die Experimente haben auch gezeigt, dass der Einfluss der NILM-Ansätze auf das Disaggregationsergebnis im Vergleich zu dem der verwendeten Daten relativ gering ist. Die erreichte Disaggregationsqualität liegt bei den betrachteten Ansätzen (DAE, S2S, DFCN, SGN und GLU) eng beieinander, sofern bekannte Geräteklassen mit spezifischen Lastprofilen betroffen sind, und sinkt rapide bei unbekanntem Gerätemodellen, unspezifischen oder untypischen Lastprofilen sowie hoher Gleichzeitigkeit von Geräteaktivitäten.

8.3.2 Kombination von Datensätzen

Als Besonderheit dieser Arbeit kann herausgehoben werden, dass zur Annäherung an einen realen Praxiseinsatz und zur Sicherstellung einer einheitlich guten Datenqualität nicht nur ein homogener Datensatz verwendet wurde, sondern mehrere Datensätze aus unterschiedlichen Quellen innerhalb des europäischen Kulturkreises kombiniert wurden (Großbritannien, Niederlande, Deutschland). Hier hat sich auch gezeigt, dass eine länderübergreifende Adaption erfolgreicher NILM-Verfahren die Möglichkeit von länderspezifischen Gerätemodellen mit unterschiedlichen Lastprofilen berücksichtigen muss.

Eine weitere Herausforderung im Rahmen dieser Arbeit lag in der Aufbereitung der Datensätze, um diese in den Disaggregationsexperimenten nutzen zu können. Den größten Aufwand erforderte hier das effektive Aussieben von Abschnitten mit schlechter Datenqualität in den verfügbaren Datensätzen.

8.3.3 Ansätze zur Vermeidung unrealistischer Gerätelastgänge

Im Rahmen dieser Arbeit wurden entsprechend der Zielsetzung, Disaggregationsansätze dadurch zu verbessern, indem die Ähnlichkeit von Lastgangsschätzungen zu gemessenen Lastgängen maximiert wird, drei Ansätze entworfen und evaluiert.

Ansatz A: Unüberwacht trainierter Generator für Gerätelastgänge

Im ersten Versuch wurde ein dedizierter, vortrainierter Lastgangsgenerator als letzter Verarbeitungsblock an ein bestehendes disaggregierendes neuronales Netz angehängt. Der Generator wird dabei aus einem Generative Adversarial Network (GAN) gewonnen, der nur mit

Beispielen von Gerätelastgängen trainiert wird. Dieser Aufbau wurde bereits in einer gemeinsamen Arbeit (Bao, Ibrahimov et al., 2018) vorgestellt. (siehe Abschnitt 6.1)

Um die Fähigkeit eines solchen Generators zu überprüfen, Gerätelastgänge aus einer Testdatenmenge nachzubilden, erfolgte eine erweiterte Analyse dieses Ansatzes. Hierfür wurde ein weiteres neuronales Netz überwacht trainiert, das die Umkehrabbildung des Generators annähern soll und für eine gegebene Gerätelastsequenz die passenden Eingabevektoren für den Generator ausgeben soll. Diese Approximation der Umkehrabbildung wird anstelle des Disaggregators als Lastsequenzkodierer eingesetzt.

Durch die mit wenigen Ausnahmen erzielten hohen Korrelationsmetrikwerte und niedrigen absoluten Fehler konnte gezeigt werden, dass sich mithilfe eines Lastgangsgenerators für die Geräteklasse Waschmaschine Lastgänge näherungsweise reproduzieren lassen können. Allerdings besteht noch ein deutlicher Abstand zum Zielmetrikwert von 1,0, welches in keinem der Fälle erreicht werden konnte.

Im Vergleich zwischen unterschiedlichen Methoden, wie Trainingsbeispiele erstellt werden können, erreicht dabei der Ansatz, ausschließlich mit vollständigen Aktivierungen zu trainieren, im Schnitt die beste Bewertung.

Wie erwartet, konnten die mittels Wasserstein-GAN erstellten Generatoren die Lastgänge etwas genauer nachbilden als die konventionell trainierten Generatoren.

Zusammenfassend zeigen die Ergebnisse zwar, dass durch den Einsatz eines Lastsequenzgenerators der Raum der ausgegebenen Lastsequenzen eingeschränkt wird und die realisierbaren Lastgänge des Zielgeräts näherungsweise umfasst werden. Diese Einschränkung erfolgt allerdings nicht genau auf den Raum realisierbarer Gerätelastgänge, sodass die Gerätelastgangsnachbildung nicht die anvisierte Qualität erreicht.

Bei der Evaluation, ob solche Generatoren nichtsdestotrotz für die Disaggregationsaufgabe hilfreich sind, wurden die Disaggregationsansätze Denoising Autoencoder (DAE) und Sequence-to-Sequence-Ansatz (S2S) mit einem Generator erweitert und neu trainiert. Dabei musste festgestellt werden, dass der Einsatz des Lastsequenzgenerators ebenfalls nicht zu dem gewünschten Effekt führt, dass ausschließlich realisierbare Waschmaschinenlastgänge ausgegeben werden. Die Integration des Lastsequenzgenerators scheint das Training des Disaggregators sogar zu erschweren und mindert damit die Disaggregationsqualität.

Ansatz B: Diskriminator für Gerätelastgänge

Der zweite Versuch, den Ausgaberaum der Disaggregationsansätze auf den Raum realisierbarer Gerätelastgänge zu beschränken, wurde durch die Arbeiten von Isola et al. (2016) inspiriert. Wie bei dem ersten Ansatz wird das GAN-Konzept als Trainingsmethode verwendet. Jedoch liegt der Fokus, im Gegensatz zum ersten Ansatz, bei dem Diskriminator

und die Beschränkung wird in der Trainingsphase erwirkt. In diesem Ansatz bewertet der Diskriminator, ob eine Disaggregation korrekt war. Dazu erhält der Diskriminator den Haushaltlastgang überlagert mit der disaggregierten Gerätelastgangsschätzung und kann somit diese Kombination bewerten.

Eine Analyse der Ergebnisse in Abschnitt 7.4.2 zeigt, dass das gesetzte Ziel teilweise erreicht wird. Exemplarisch ist anhand von Kühlgerätelastgängen zu erkennen, dass durch das Training mit einem Diskriminator die Ausgabe öfter realisierbaren Lastgängen entspricht.

Die Ergebnisse in Abschnitt 6.2 zeigen allerdings, dass dies nicht dazu führt, dass die Disaggregationsqualität im Hinblick auf die Metriken besser wird. Nur die Geräte, die ohne einen Diskriminator bereits gut disaggregiert werden können, werden durch ein mit Diskriminator trainiertes neuronales Netz minimal besser disaggregiert.

Hingegen führt der Austausch der bei bekannten Ansätzen üblicherweise verwendeten L_2 -Verlustfunktion mit der L_1 -Verlustfunktion zu einer deutlichen Verbesserung der Disaggregationsqualität. Auch hier wird der Effekt beobachtet, dass bereits gut disaggregierbare Geräte mit dem Austausch noch besser disaggregiert werden können. Szenarien mit bereits schlechten Metriken führen nach dem Austausch oft zu noch schlechteren Ergebnissen.

Auch wenn der Einsatz des Diskriminators im Mittel zu keinen messbaren Verbesserungen führt, so ist es dennoch möglich, dass die Ausgabe aufgrund der plausibler aussehenden Schätzungen eine höhere Benutzerakzeptanz hat. Für messbare Verbesserungen der Disaggregationsqualität ist das Training mit einer L_1 -Verlustfunktion zu wählen.

Ansatz C: U-Net mit Dilertation

Der dritte im Rahmen dieser Arbeit entwickelte Ansatz ist von den Arbeiten von Isola et al. (2016) sowie Barsim und Yang (2018) inspiriert. Dieser Ansatz kombiniert die U-Net-Architektur (Ronneberger et al., 2015) mit dem Entwurfsprinzip der exponentiell größer werdenden Dilatationsabstände von Brewitt und Goddard (2018).

Die Nutzung der U-Net-Architektur wird dadurch motiviert, dass bei Lastgängen zwischen kleinräumigen Parametern bezüglich der Lastprofiltextur und großräumigen Lastgangsmerkmalen unterschieden werden kann. Die U-Net-Architektur enthält separate Informationspfade für beide Parametergruppen. Insbesondere die kleinräumigen Parameter müssen dadurch nicht durch alle Schichten durchgeschleust werden, da diese Informationen direkt im Dekodierblock zur Verfügung stehen. Dadurch wird erwartet, dass die geschätzten Gerätelastgänge eine höhere Detailtreue aufweisen.

Der vorgeschlagene Entwurf behält gleichzeitig das Sequence-to-Point-Prinzip von DFCN. Dadurch werden bei der Disaggregation mit einer einzigen Inferenz der Lastwert für mehrere

Zeitpunkte parallel berechnet und eine Zusammenführung durch Mittelwertbildung entfällt. Die Eingabelänge ist entsprechend so gewählt und die Ausgabesequenz wird so gestutzt, dass es keine Randeffekte gibt. Dennoch wird wie bei dem DFCN für jeden Ausgabewert ein breiter Kontext eingelesen, was durch die exponentiell wachsenden Dilatationsabstände erreicht wird. Das Netzmodell kann mit den selben Parametern und bei gleichbleibender Kontextlänge beliebig in der zeitlichen Dimension verlängert werden.

Die Evaluation der mit L_1 -Verlust trainierten Ansätze ergab, dass bei 4 von 7 Geräteklassen der Korrelationskoeffizient gesteigert wird. Bei den verbleibenden 3 Geräteklassen sinkt der Korrelationskoeffizient, obwohl der durchschnittliche Fehler reduziert wird.

Insofern betrifft die Verbesserung der Disaggregationsqualität wiederum genau die Geräteklassen, die bereits zuvor von bestehenden Ansätzen gut disaggregiert werden konnten. Bei den Problemfällen erzielt der U-Net-Ansatz wiederum noch schlechtere Ergebnisse, sodass im Durchschnitt über alle Geräte der U-Net-Ansatz schlechter abschneidet als das DFCN.

8.3.4 Vergleich der Disaggregationsansätze

Da der Einsatz von NILM-Verfahren unterschiedlichen Zielsetzungen folgen kann, wurden in Kapitel 7 die bekannten NILM-Ansätze sowie die neu entwickelten Ansätze dieser Arbeit anhand der einzelnen Geräteklassen im Hinblick auf die Fähigkeit

- einen Gerätelastgang zu reproduzieren,
- die Energieverbrauchsbilanz korrekt zu schätzen, und
- eine Geräteaktivität zu detektieren,

verglichen. In Tabelle 8.1 sind die jeweils im Durchschnitt besten NILM-Ansätze zusammengetragen.

Weiterhin wurde der Trainings- und Disaggregationsaufwand abgeschätzt.

8.3.5 Weitere Beiträge

Technische Verbesserungen

Für diese Arbeit wurden folgende Anpassungen am Non-Intrusive Load Monitoring Toolkit (NILMTK)¹ und NeuralNILM² durchgeführt. Diese sind zusammen mit allen vorhandenen

¹<https://github.com/KaibinBao/nilmk>

²<https://github.com/KaibinBao/neuralnilm>

Tabelle 8.1: Im Durchschnitt bester Disaggregationsansatz je Geräteklasse und Anwendungsszenario. In Klammern steht die verwendete Verlustfunktion für das Training.

	Reproduktion Gerätelastgang	Bilanz Energie- verbrauch	Detektion Geräteaktivität
Waschmaschine	U-Net (L ₁)	U-Net (L ₁)	U-Net (L ₁)
Kühlgerät	U-Net (L ₁) / DFCN (L ₁)	DAE (L ₁)	U-Net (L ₁)
Trockner	U-Net (L ₁) / DFCN (L ₁)	DFCN (L ₁)	U-Net (L ₁)
TV/Unterhaltung	DAE (L ₁)	DAE (L ₁)	DFCN (L ₂)
Mikrowelle	DFCN (L ₂) / U-Net (L ₁)	U-Net (L ₁)	DFCN (L ₁)
Geschirrspüler	DFCN (L ₁) / GLU (L ₂)	DFCN (L ₂)	U-Net (L ₁)
Wasserkocher	DFCN (L ₁)	U-Net (L ₁)	U-Net (L ₁)

und neu entworfenen Ansätzen, die in dieser Arbeit verwendet wurden, veröffentlicht³, um die Reproduzierbarkeit der Ergebnisse zu gewährleisten. (für weitere Details siehe Abschnitt 4.9.1)

Korrekturen in NILMTK für die bereinigte Version von REFIT In der in dieser Arbeit verwendeten bereinigten Veröffentlichung des Datensatzes REFIT haben sich im Vergleich zur ursprünglichen Veröffentlichung einige Gerätezuordnungen geändert. Diese Änderungen wurden im Datensatzkonversionswerkzeug von NILMTK noch nicht vollständig nachvollzogen. Für diese Arbeit wurde dies entsprechend nachgeholt. Weiterhin wurde ein Fehler beim Laden der Datensätze behoben, der bei sich knapp überlappenden Zeitreihensegmenten auftreten kann.

Reimplementierung der Ansätze für PyTorch Für die bekannten Ansätze standen größtenteils keine Implementierungen für aktuelle Deep-Learning-Frameworks quelloffen zur Verfügung. Alle bestehenden und neu entworfenen Ansätze wurden daher neu und einheitlich für das Framework PyTorch implementiert.

Profiling und Parallelisierung von NeuralNILM Beim Training von kleinen neuronalen Netzen stellt die Trainingspaargenerierung auf dem Zentralprozessor im Hinblick auf den Durchsatz den limitierenden Faktor dar. Daher wurde die Trainingspaargenerierung optimiert und parallelisiert.

Homogenisierung der Datenqualität der verwendeten Datensätze Die Datenqualität der verwendeten Datenbank wurde beurteilt und mit einem gemeinsamen Qualitätsmaßstab bereinigt. Zudem wurde mit heuristisch bestimmten Grenzwerten versucht, eindeutig fehlerhafte Lastgangsabschnitte aus dem Datensatz zu entfernen. Der Bereinigungsprozess sowie ein Zwischenspeichermechanismus, um eine wiederholte

³<https://github.com/KaibinBao/neuralnilm-pytorch>

Ausführung der Bereinigung zu vermeiden, wurde in den Trainingspaargeneratoren in NeuralNILM implementiert.

Neuimplementierung der Messdatenerfassung im KIT Energy Smart Home Lab

Da das Messsystem des ESHL als einzigartiger Experimentierraum und Forschungslabor für Gebäudeenergiemanagementsysteme Bestandteil von Experimenten mit unterschiedlichen thematischen Ausrichtungen war, war die Verfügbarkeit der Messdatenaufzeichnung in der Vergangenheit nicht sichergestellt. Aufgrund der zahlreichen Messausfälle wurde daher die Neuimplementierung des Messsystems erforderlich. Über eine architekturbedingte Entkopplung konnte eine unterbrechungsfreie Messdatenaufzeichnung bei paralleler sowie erleichterter Durchführbarkeit von Experimenten erreicht werden (siehe Abschnitt 4.3.5). Die dafür vorgenommenen Arbeiten wurden ausführlich von Bao, Mauser et al. (2016) und Bao, Fiebig et al. (2016) beschrieben.

8.4 Ausblick

Es ist zu erwarten, dass die Disaggregationsqualität der untersuchten NILM-Ansätze mit einer Verbreiterung der Datenbasis zunächst im Hinblick auf Quantität, aber auch bezüglich der Diversität der gemessenen Geräteklassen zunehmen wird. Hierfür sollten nicht nur Privathaushaltsszenarien, sondern auch Industriekontexte, Bürogebäude und Co-Working-Spaces als interessante Datenlieferanten berücksichtigt werden. Für Forschende stellt sich die Herausforderung, eine entsprechende Datenbasis mit Einzelmessungen der Geräteklassen aufzubauen.

Hierbei müssen sich Informatiker*innen auch künftig vermehrt datenschutzrechtlichen Anforderungen stellen, um auch im Nachhinein von einzelnen Forschungsprojekten die Legitimation zur Datenanalyse sowie Veröffentlichung der Daten in öffentlich zugänglichen Forschungsdatenbanken sicherzustellen. Projekte wie das Energy Smart Home Lab ermöglichen Forschenden einzigartige Forschungsumgebungen, in denen die Grenzen zwischen Forscher und Forschungsobjekt diffundieren.

In dieser Arbeit wurde u. a. ein Ansatz aus der Bilderkennung auf das Anwendungsszenario NILM adaptiert. Im Rahmen der Weiterentwicklung von NILM-Ansätzen sind die Möglichkeiten der Überführung moderner Ansätze aus anderen Fachbereichen noch nicht ausgereizt. Allerdings sind diese aktuell kaum nutzbar bzw. es sind keine signifikanten Qualitätsverbesserungen gegenüber den bekannten Verfahren zu erwarten, solange noch zu wenig Daten für Training und Evaluation zur Verfügung stehen.

Eine weitere wichtige Hilfestellung bei der NILM-Forschung wären Werkzeuge zur Kombination und Homogenisierung von unterschiedlichen Datensätzen sowie die Etablierung von

Evaluationsprotokollen oder Wettbewerben. Die aktuelle Arbeit zeigt, dass die gängige Praxis, nur einen Datensatz zu verwenden, nicht mehr genügt, um auch in der Realität anwendbare NILM-Modelle zu trainieren. Einen Teil der Lösung könnten Generatoren für synthetische Datensätze und Referenzimplementierungen, so wie sie Reinhardt und Klemenjak (2020) vorgeschlagen haben, sein.

Eine interessante Forschungsrichtung ist die Abkehr von dem von Kelly und Knottenbelt (2015a) eingeführten Paradigma, genau ein neuronales Netzmodell pro Gerät oder Geräteklasse zu erstellen. Yu, Jiang et al. (2019) führen dazu erste Untersuchungen durch. Alternativ versuchen D’Incecco et al. (2019), das Wissen aus der Disaggregation eines Gerätes auf andere Geräte zu übertragen.

Ein Faktor für die Akzeptanz bei potentiellen Kunden von NILM-Ansätzen könnte die Plausibilität der Ergebnisse sein. Zukünftige Arbeiten könnten sich auch stärker auf die Kundenakzeptanz konzentrieren und dafür kleine Einschnitte in der Disaggregationsqualität in Kauf nehmen.

Anhang A

Einzelne Ergebnisse bei der Evaluation der Ansätze zur Optimierung der Disaggregationsqualität

A.1 Ergebnisse cGAN (Anhang)

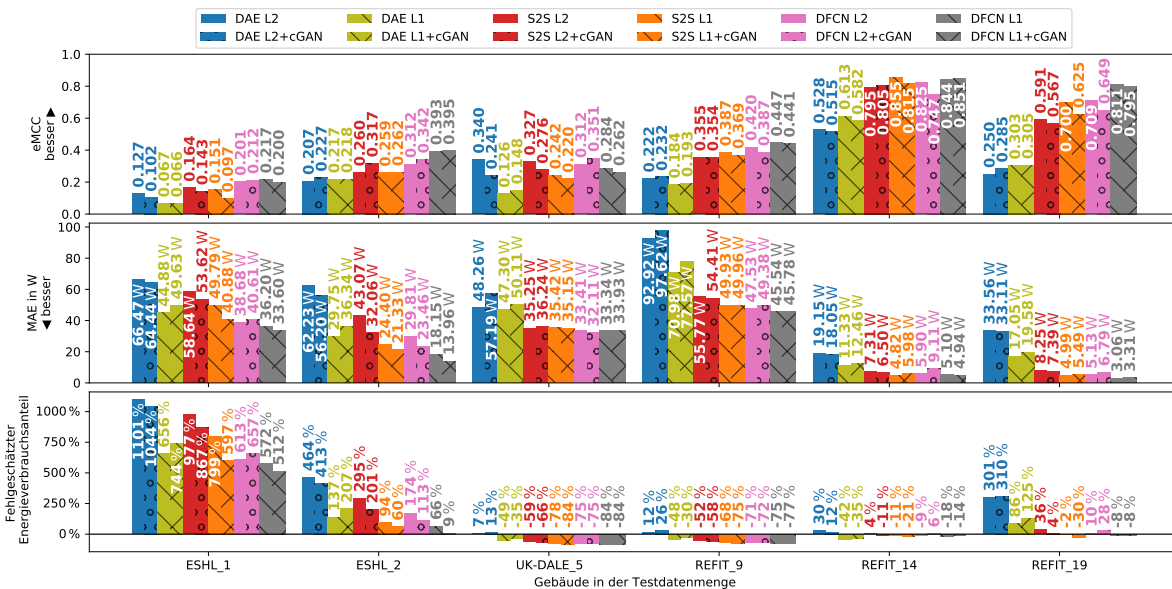


Abbildung A.1: Fehlermetriken bei der Disaggregation von Waschmaschinen mit Ansätzen mit unterschiedlichen Verlustfunktionen.

Anhang A Einzelne Ergebnisse bei der Evaluation der Ansätze zur Optimierung der Disaggregationsqualität

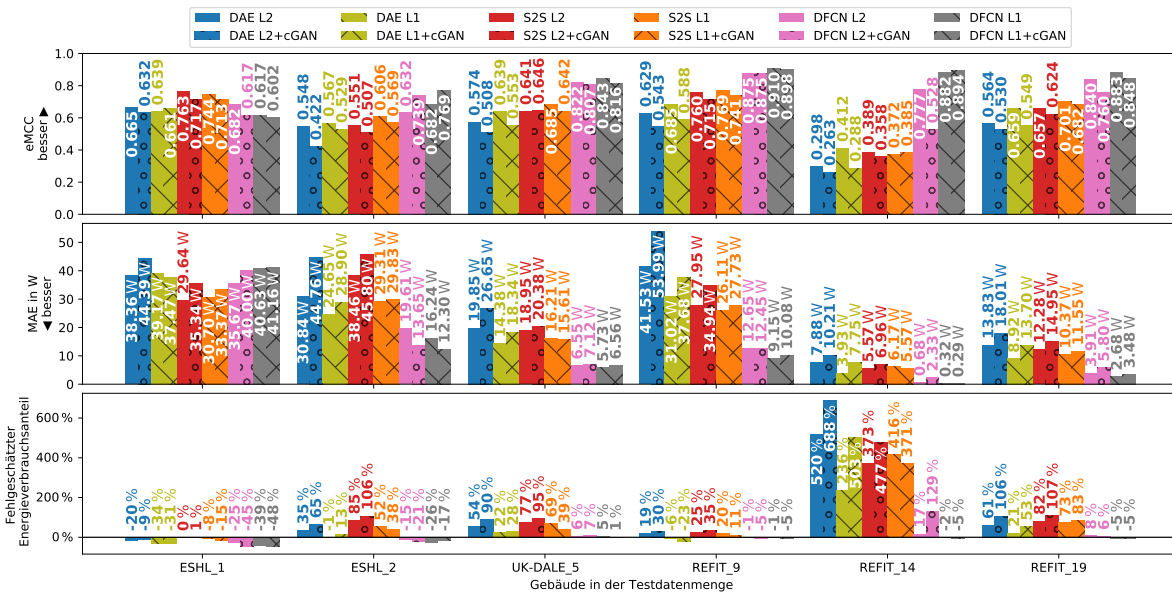


Abbildung A.2: Fehlermetriken bei der Disaggregation von Geschirrspülern mit Ansätzen mit unterschiedlichen Verlustfunktionen.

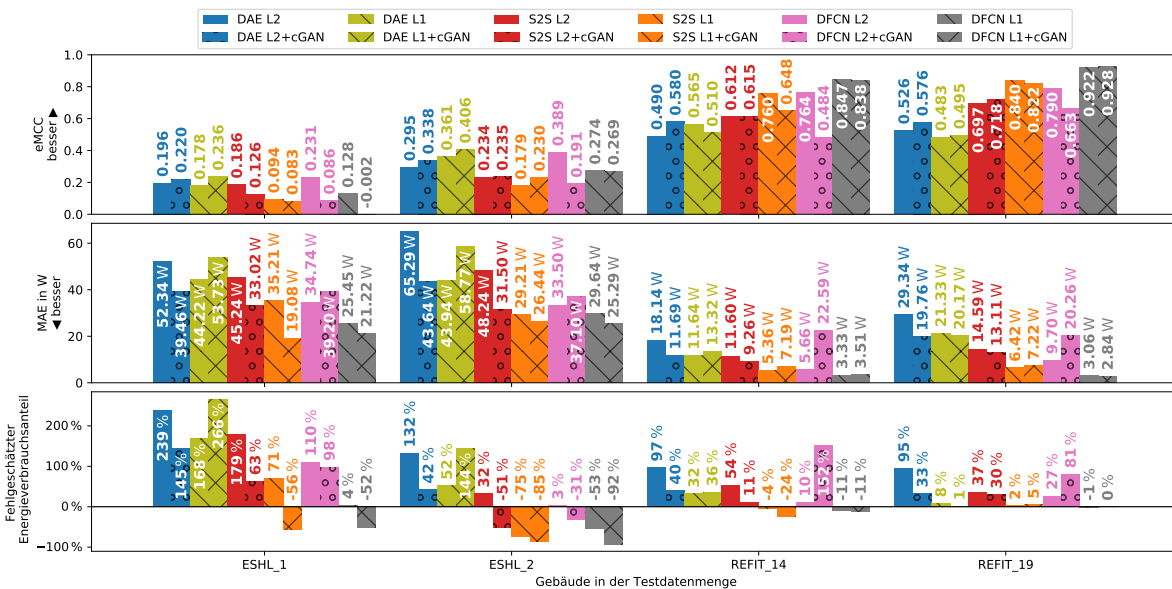


Abbildung A.3: Fehlermetriken bei der Disaggregation von Trocknern mit Ansätzen mit unterschiedlichen Verlustfunktionen.

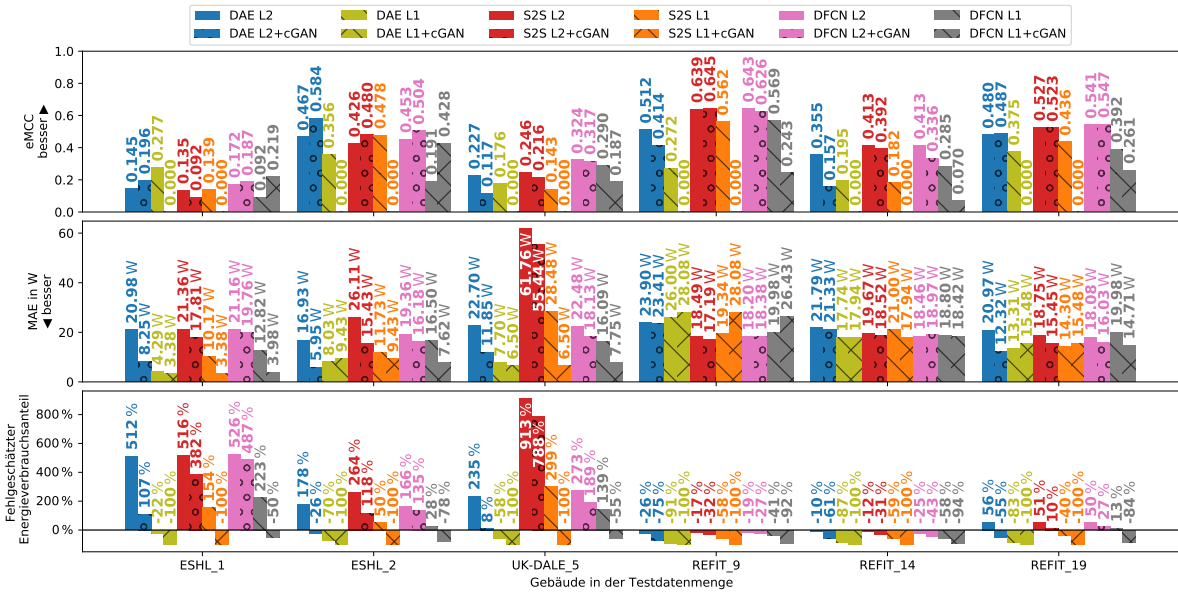


Abbildung A.4: Fehlermetriken bei der Disaggregation von Fernseher und Unterhaltungselektronik mit Ansätzen mit unterschiedlichen Verlustfunktionen.

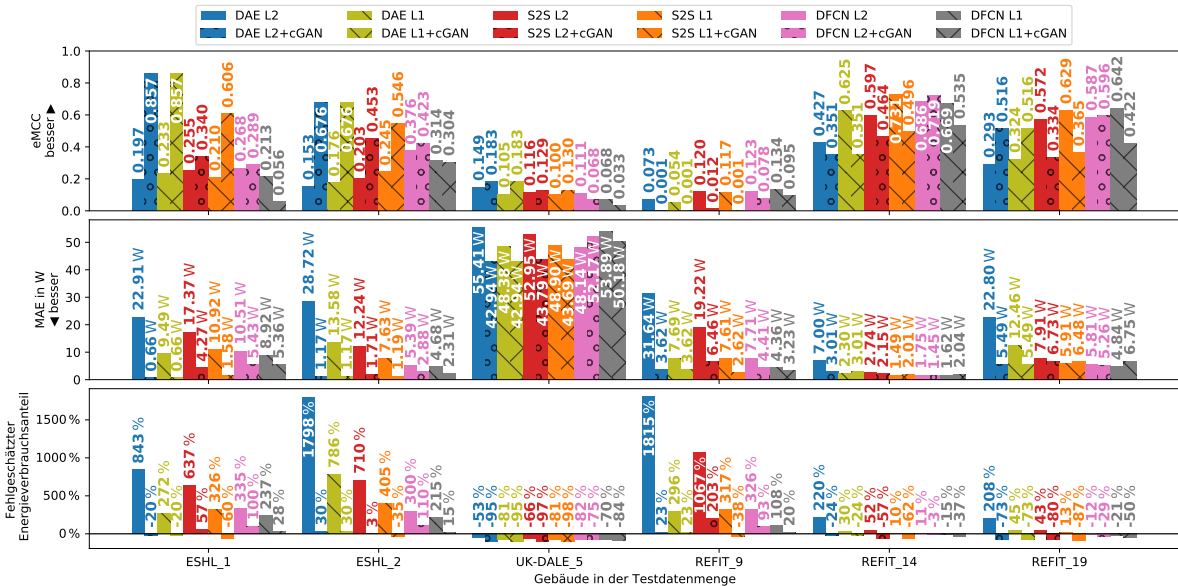


Abbildung A.5: Fehlermetriken bei der Disaggregation von Mikrowellen mit Ansätzen mit unterschiedlichen Verlustfunktionen.

Anhang A Einzelne Ergebnisse bei der Evaluation der Ansätze zur Optimierung der Disaggregationsqualität

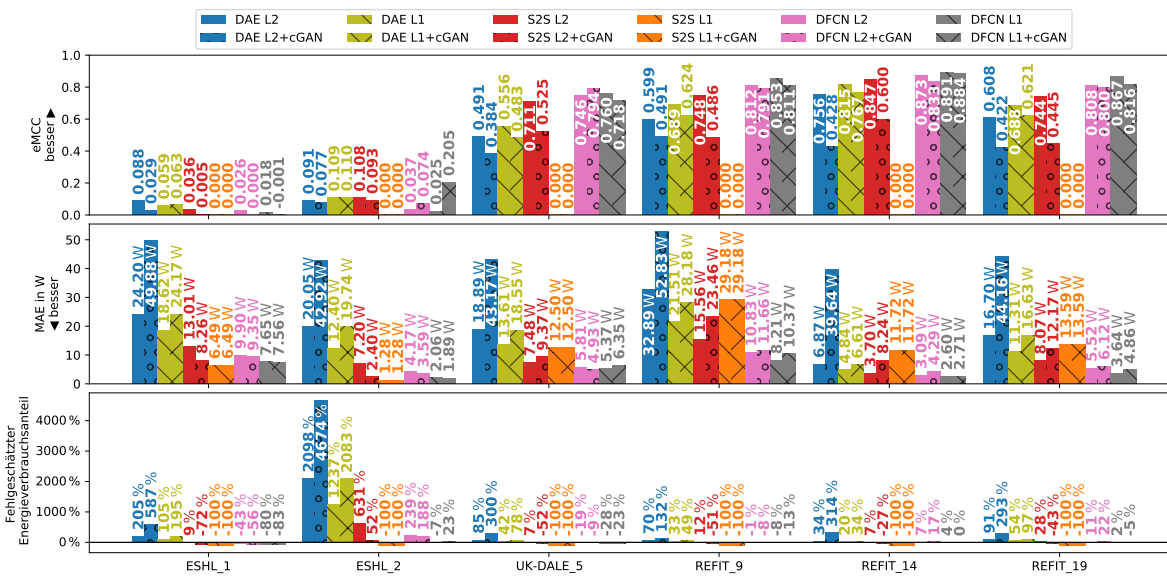


Abbildung A.6: Fehlermetriken bei der Disaggregation von Wasserkochern mit Ansätzen mit unterschiedlichen Verlustfunktionen.

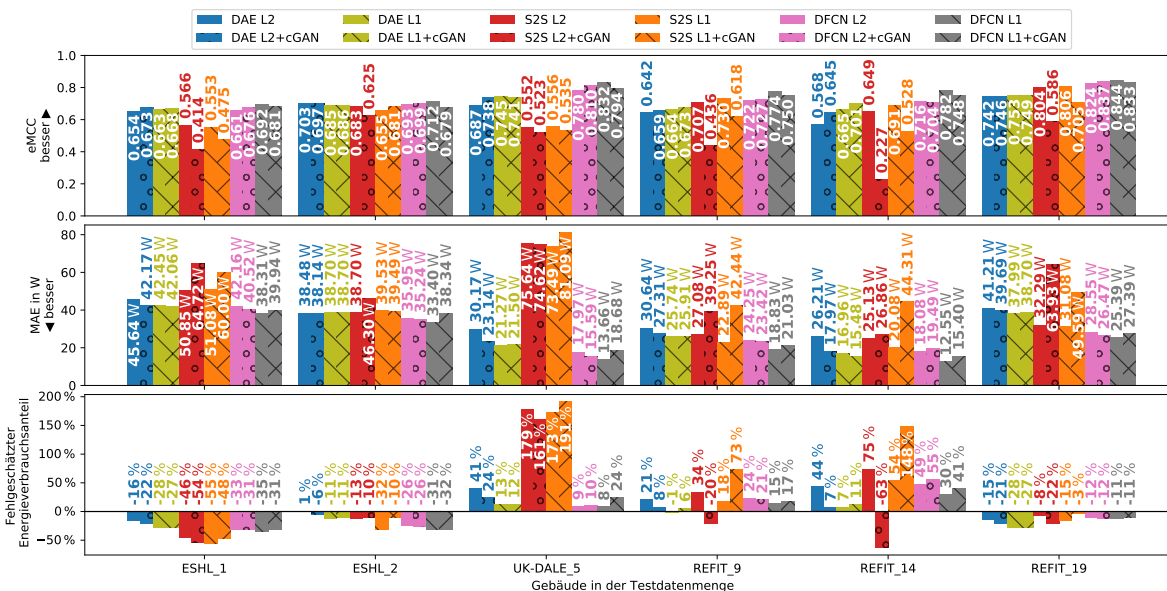


Abbildung A.7: Fehlermetriken bei der Disaggregation von Kühlgeräten mit Ansätzen mit unterschiedlichen Verlustfunktionen.

A.2 Ergebnisse U-Net (Anhang)

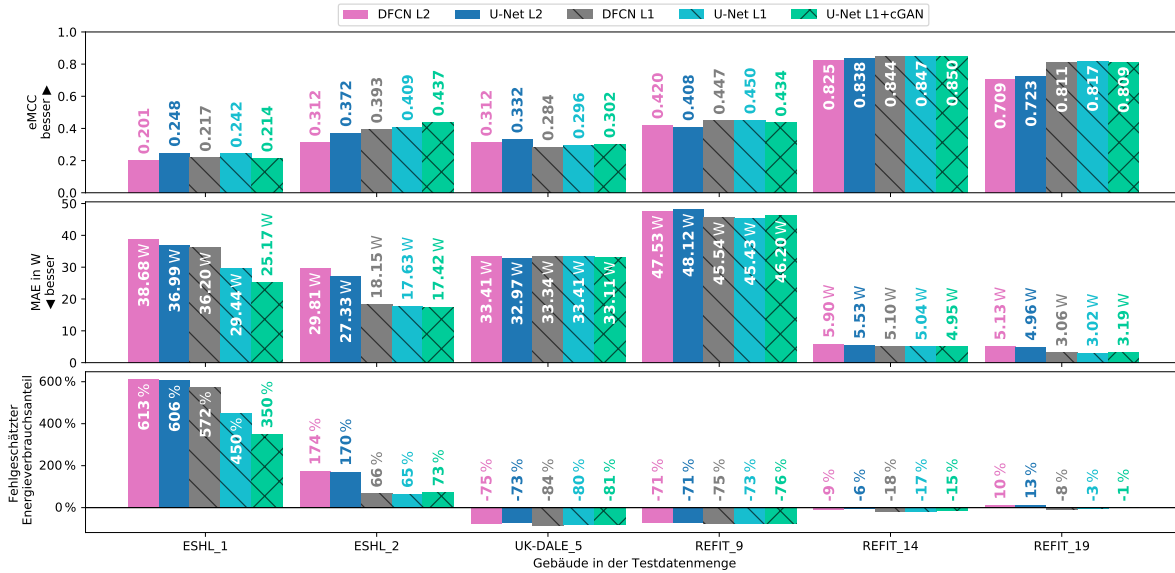


Abbildung A.8: Fehlermetriken bei der Disaggregation von Waschmaschinen mit der U-Net-Architektur.

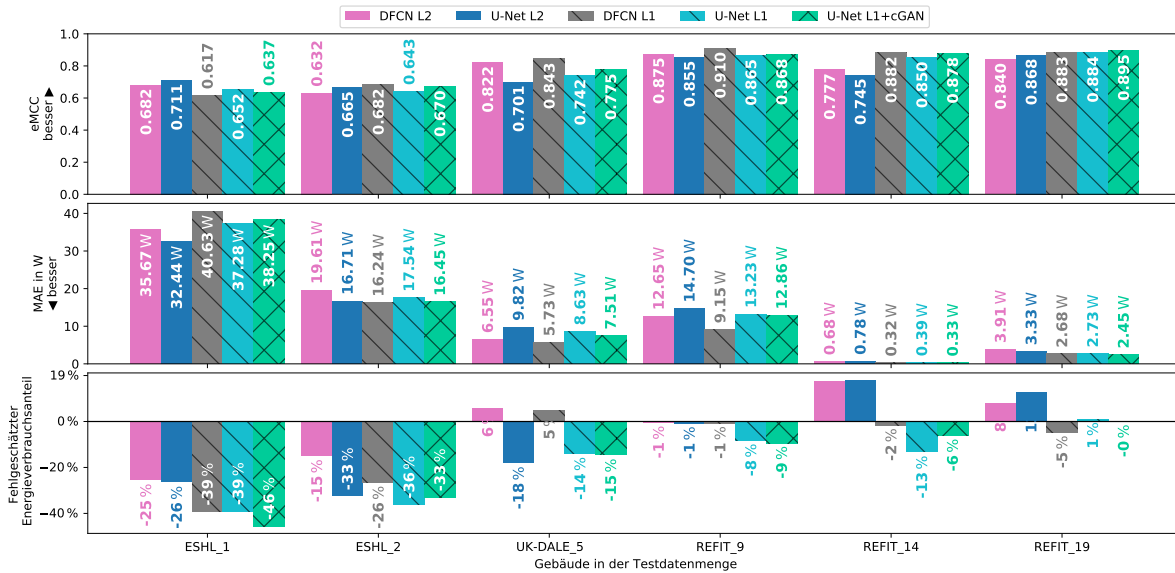


Abbildung A.9: Fehlermetriken bei der Disaggregation von Geschirrspülern mit der U-Net-Architektur.

Anhang A Einzelne Ergebnisse bei der Evaluation der Ansätze zur Optimierung der Disaggregationsqualität

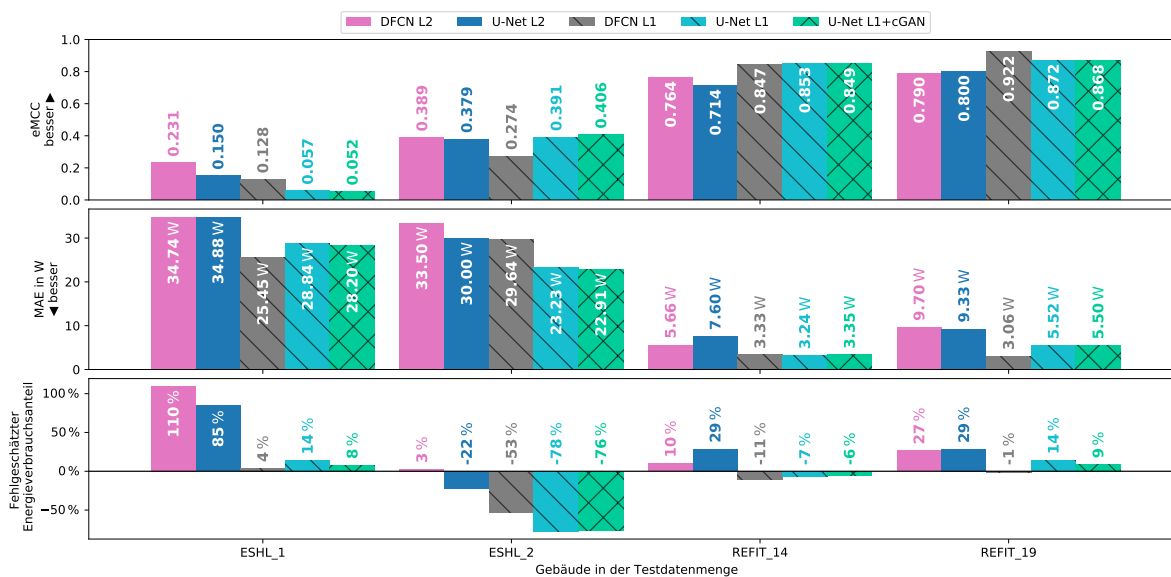


Abbildung A.10: Fehlermetriken bei der Disaggregation von Trocknern mit der U-Net-Architektur.

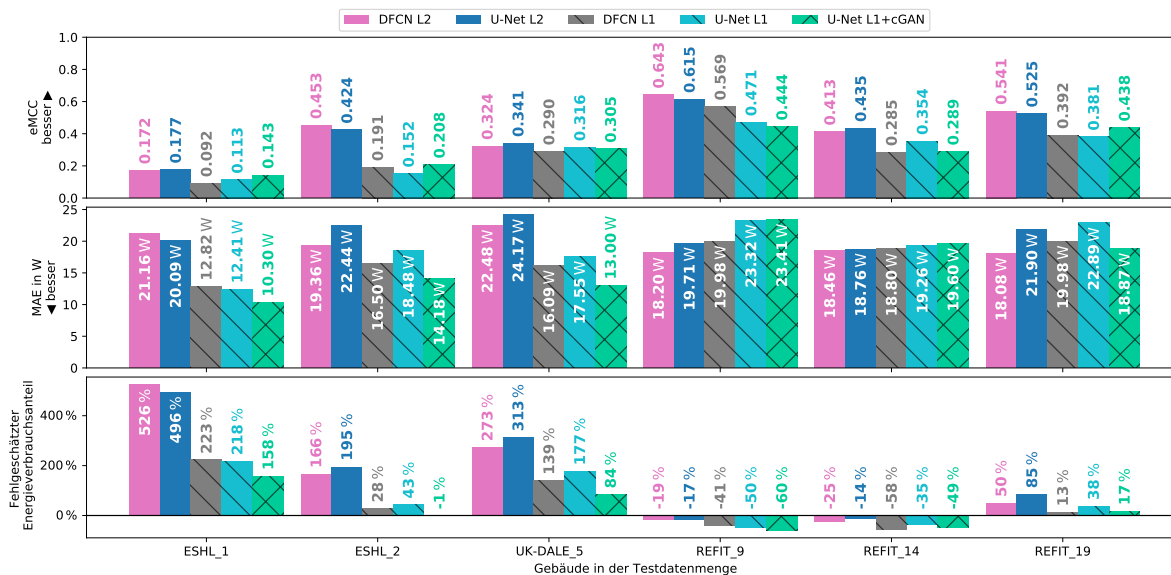


Abbildung A.11: Fehlermetriken bei der Disaggregation von Fernseher und Unterhaltungselektronik mit der U-Net-Architektur.

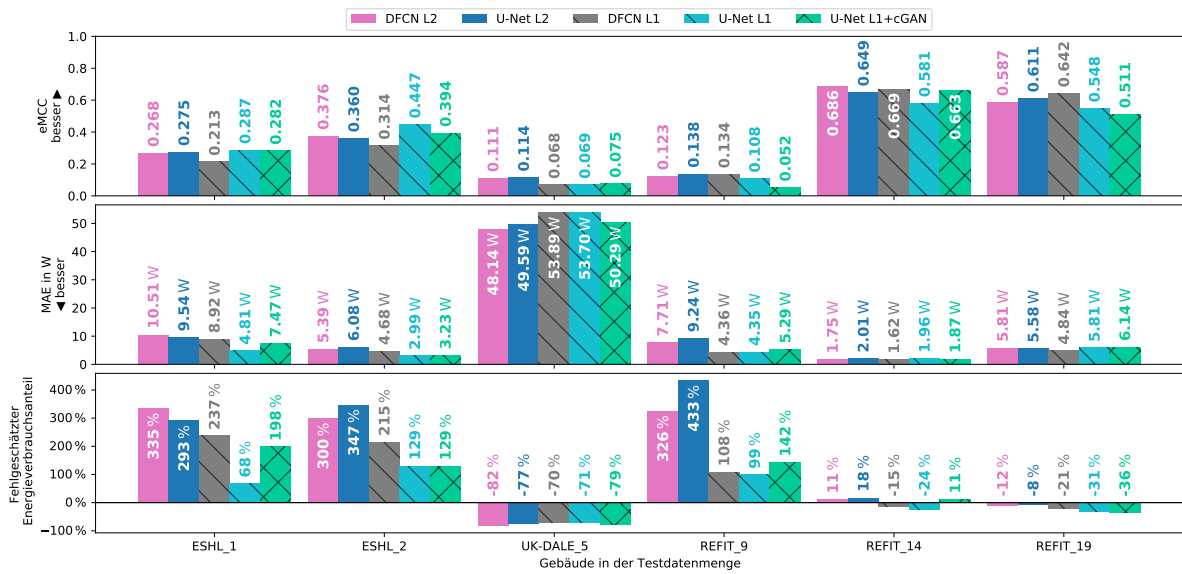


Abbildung A.12: Fehlermetriken bei der Disaggregation von Mikrowellen mit der U-Net-Architektur.



Abbildung A.13: Fehlermetriken bei der Disaggregation von Wasserkochern mit der U-Net-Architektur.

Anhang A Einzelne Ergebnisse bei der Evaluation der Ansätze zur Optimierung der Disaggregationsqualität

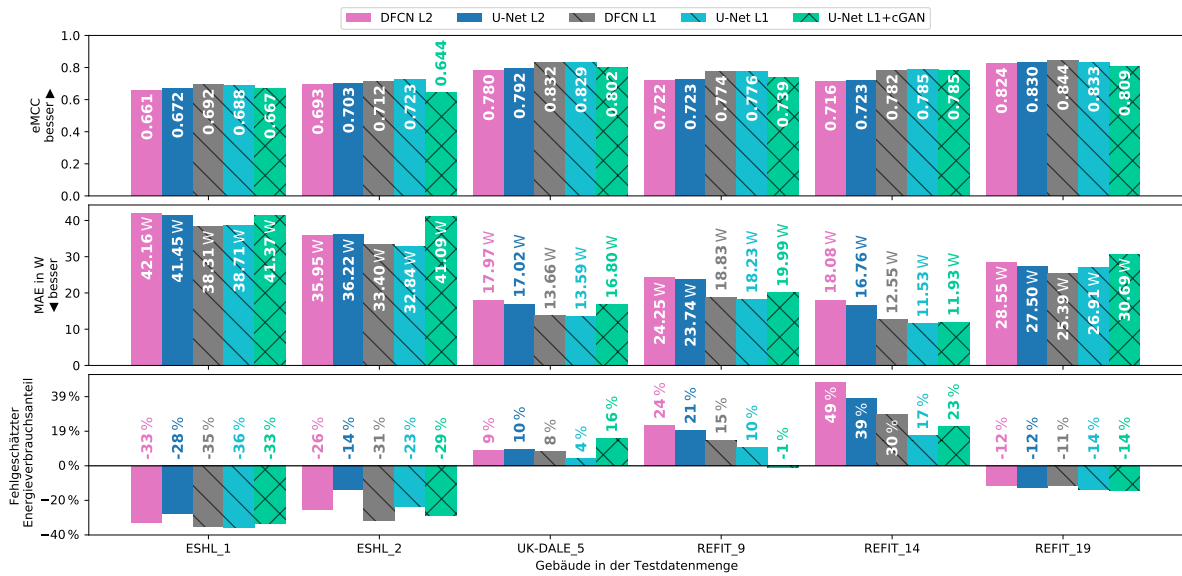


Abbildung A.14: Fehlermetriken bei der Disaggregation von Kühlgeräten mit der U-Net-Architektur.

Literatur

- Ács, G. und Castelluccia, C. (2011). „I Have a DREAM! (Differentially privatE smArt Metering)“. In: *Information Hiding*. Bd. 6958. Springer, S. 118–132. DOI: 10.1007/978-3-642-24178-9_9.
- Alcalá, J. M., Ureña, J., Hernández, Á. und Gualda, D. (2017). „Assessing Human Activity in Elderly People Using Non-Intrusive Load Monitoring“. In: *Sensors* 17.2, S. 351. DOI: 10.3390/s17020351.
- Allerding, F. und Schmeck, H. (2011). „Organic Smart Home: Architecture for Energy Management in Intelligent Buildings“. In: *Proceedings of the 2011 Workshop on Organic Computing*. ACM, S. 67–76. DOI: 10.1145/1998642.1998654.
- Allerding, F. (2013). „Organic Smart Home - Energiemanagement für Intelligente Gebäude“. KIT Scientific Publishing, Karlsruhe. 251 S. DOI: 10.5445/KSP/1000038928.
- Altrabalsi, H., Liao, J., Stankovic, L. und Stankovic, V. (2014). „A Low-Complexity Energy Disaggregation Method: Performance and Robustness“. In: *2014 IEEE Symposium on Computational Intelligence Applications in Smart Grid (CIASG)*. Orlando, FL, USA: IEEE, S. 1–8. DOI: 10.1109/CIASG.2014.7011569.
- An, G. (1996). „The Effects of Adding Noise During Backpropagation Training on a Generalization Performance“. In: *Neural Computation* 8.3, S. 643–674. DOI: 10.1162/neco.1996.8.3.643.
- Arjovsky, M., Chintala, S. und Bottou, L. (2017). „Wasserstein GAN“. In: arXiv: 1701.07875.
- Azzini, H. A. D., Torquato, R. und da Silva, L. C. P. (2014). „Event Detection Methods for Non-intrusive Load Monitoring“. In: *2014 IEEE PES General Meeting | Conference & Exposition. 2014 IEEE Power & Energy Society General Meeting*. National Harbor, MD, USA: IEEE, S. 1–5. DOI: 10.1109/PESGM.2014.6939797.
- Bambauer, J., Muralidhar, K. und Sarathy, R. (2014). „Fool’s Gold: An Illustrated Critique of Differential Privacy“. In: *Vanderbilt Journal of Entertainment and Technology Law* 16.4, S. 55.
- Bao, K., Bräuchle, T. und Schmeck, H. (2015). „Towards Privacy in Monitored Shared Environments“. In: *10th Future Security Conf*, S. 469–472.
- Bao, K., Fiebig, F., Frank, L., Kochanneck, S., Mauser, I. und Schmeck, H. (2016). „Smarter Energy by an Intranet of Things in Buildings“. In: *VDE-Kongress 2016 – Internet Der Dinge*. VDE, S. 6.

- Bao, K., Ibrahimov, K., Wagner, M. und Schmeck, H. (2018). „Enhancing Neural Non-Intrusive Load Monitoring with Generative Adversarial Networks“. In: *Energy Informatics 1.1*, S. 18. DOI: 10.1186/s42162-018-0038-y.
- Bao, K., Mauser, I., Kochanneck, S., Xu, H. und Schmeck, H. (2016). „A Microservice Architecture for the Intranet of Things and Energy in Smart Buildings: Research Paper“. In: *Proceedings of the 1st International Workshop on Mashups of Things and APIs. MOTA '16*. New York, NY, USA: ACM, 3:1–3:6. DOI: 10.1145/3007203.3007215.
- Bao, K., Valev, H., Wagner, M. und Schmeck, H. (2017). „A Threat Analysis of the Vehicle-to-Grid Charging Protocol ISO 15118“. In: *Computer Science - Research and Development*. DOI: 10.1007/s00450-017-0342-y.
- Baranski, M. und Voss, J. (2003). „Non-Intrusive Appliance Load Monitoring Based on an Optical Sensor“. In: *2003 IEEE Bologna Power Tech Conference Proceedings*. Bd. 4. Bologna, Italy: IEEE, S. 267–274. DOI: 10.1109/PTC.2003.1304732.
- Barker, S., Kalra, S., Irwin, D. und Shenoy, P. (2013). „Empirical Characterization and Modeling of Electrical Loads in Smart Homes“. In: *2013 International Green Computing Conference Proceedings*, S. 1–10. DOI: 10.1109/IGCC.2013.6604512.
- Barker, S., Kalra, S., Irwin, D. und Shenoy, P. (2014a). „Empirical Characterization, Modeling, and Analysis of Smart Meter Data“. In: *IEEE Journal on Selected Areas in Communications* 32.7, S. 1312–1327. DOI: 10.1109/JSAC.2014.2332107.
- Barker, S., Kalra, S., Irwin, D. und Shenoy, P. (2014b). „PowerPlay: Creating Virtual Power Meters through Online Load Tracking“. In: *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*. ACM, S. 60–69. DOI: 10.1145/2674061.2674068.
- Barker, S., Mishra, A., Irwin, D., Cecchet, E., Shenoy, P. und Albrecht, J. (2012). „Smart*: An Open Data Set and Tools for Enabling Research in Sustainable Homes“. In: *Proceedings of the 2nd KDD Workshop on Data Mining Applications in Sustainability (SustKDD)*. Bd. 111, S. 112.
- Barsim, K. S. und Yang, B. (2018). „On the Feasibility of Generic Deep Disaggregation for Single-Load Extraction“. In: arXiv: 1802.02139.
- Batra, N., Gulati, M., Singh, A. und Srivastava, M. B. (2013). „It’s Different: Insights into Home Energy Consumption in India“. In: *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings*. BuildSys’13. New York, NY, USA: ACM, 3:1–3:8. DOI: 10.1145/2528282.2528293.
- Batra, N., Kelly, J., Parson, O., Dutta, H., Knottenbelt, W., Rogers, A., Singh, A. und Srivastava, M. (2014). „NILMTK: An Open Source Toolkit for Non-Intrusive Load Monitoring“. In: *Proceedings of the 5th International Conference on Future Energy Systems*. ACM, S. 265–276. DOI: 10.1145/2602044.2602051.
- Batra, N., Parson, O., Berges, M., Singh, A. und Rogers, A. (2014). „A Comparison of Non-Intrusive Load Monitoring Methods for Commercial and Residential Buildings“. In: arXiv: 1408.6595.
- Batra, N., Singh, A. und Whitehouse, K. (2015). „If You Measure It, Can You Improve It? Exploring The Value of Energy Disaggregation“. In: *Proceedings of the 2Nd ACM International*

- Conference on Embedded Systems for Energy-Efficient Built Environments* (Seoul, South Korea). BuildSys '15. New York, NY, USA: ACM, S. 191–200. DOI: 10.1145/2821650.2821660.
- Beckel, C., Kleiminger, W., Cicchetti, R., Staake, T. und Santini, S. (2014). „The ECO Data Set and the Performance of Non-Intrusive Load Monitoring Algorithms“. In: ACM Press, S. 80–89. DOI: 10.1145/2674061.2674064.
- Beckel, C., Sadamori, L., Staake, T. und Santini, S. (2014). „Revealing Household Characteristics from Smart Meter Data“. In: *Energy* 78, S. 397–410. DOI: 10.1016/j.energy.2014.10.025.
- Becker, B., Kellerer, A. und Schmeck, H. (2012). „User Interaction Interface for Energy Management in Smart Homes“. In: *2012 IEEE PES Innovative Smart Grid Technologies (ISGT)*, S. 1–8. DOI: 10.1109/ISGT.2012.6175616.
- Becker, V., Kleiminger, W., Coroamă, V. C. und Mattern, F. (2018). „Estimating the Savings Potential of Occupancy-Based Heating Strategies“. In: *Energy Informatics* 1.1, S. 52. DOI: 10.1186/s42162-018-0022-6.
- Bonfigli, R., Severini, M., Squartini, S., Fagiani, M. und Piazza, F. (2016). „Improving the Performance of the AFAMAP Algorithm for Non-Intrusive Load Monitoring“. In: *2016 IEEE Congress on Evolutionary Computation (CEC)*. 2016 IEEE Congress on Evolutionary Computation (CEC), S. 303–310. DOI: 10.1109/CEC.2016.7743809.
- Bonfigli, R., Felicetti, A., Principi, E., Fagiani, M., Squartini, S. und Piazza, F. (2018). „Denoising Autoencoders for Non-Intrusive Load Monitoring: Improvements and Comparative Evaluation“. In: *Energy and Buildings* 158, S. 1461–1474. DOI: 10.1016/j.enbuild.2017.11.054.
- Brewitt, C. und Goddard, N. (2018). „Non-Intrusive Load Monitoring with Fully Convolutional Networks“. In: arXiv: 1812.03915 [cs, stat].
- BSI-CC-PP-0073 (2014). *BSI Schutzprofil Smart Meter Gateway BSI-CC-PP-0073*. Bundesamt für Sicherheit in der Informationstechnik (BSI).
- Buchmann, E., Böhm, K., Burghardt, T. und Kessler, S. (2013). „Re-Identification of Smart Meter Data“. In: *Personal and Ubiquitous Computing* 17.4, S. 653–662. DOI: 10.1007/s00779-012-0513-6.
- Buneeva, N. und Reinhardt, A. (2017). „AMBAL: Realistic Load Signature Generation for Load Disaggregation Performance Evaluation“. In: *2017 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, S. 443–448. DOI: 10.1109/SmartGridComm.2017.8340657.
- Carrie Armel, K., Gupta, A., Shrimali, G. und Albert, A. (2013). „Is Disaggregation the Holy Grail of Energy Efficiency? The Case of Electricity“. In: *Energy Policy*. Special Section: Transition Pathways to a Low Carbon Economy 52, S. 213–234. DOI: 10.1016/j.enpol.2012.08.062.
- Chan, W. L., So, A. T. P. und Lai, L. L. (2000). „Wavelet Feature Vectors for Neural Network Based Harmonics Load Recognition“. In: *2000 International Conference on Advances in Power System Control, Operation and Management, APSCOM-00*. Bd. 2, 511–516 vol.2. DOI: 10.1049/cp:20000453.

- Chandna, P., Miron, M., Janer, J. und Gómez, E. (2017). „Monoaural Audio Source Separation Using Deep Convolutional Neural Networks“. In: *Latent Variable Analysis and Signal Separation*. Hrsg. von P. Tichavský, M. Babaie-Zadeh, O. J. Michel und N. Thirion-Moreau. Lecture Notes in Computer Science. Springer International Publishing, S. 258–266. DOI: 10.1007/978-3-319-53547-0_25.
- Chang, H.-H., Yang, H.-T. und Lin, C.-L. (2007). „Load Identification in Neural Networks for a Non-Intrusive Monitoring of Industrial Electrical Loads“. In: *International Conference on Computer Supported Cooperative Work in Design*. Springer, S. 664–674.
- Chen, D., Irwin, D., Shenoy, P. und Albrecht, J. (2014). „Combined Heat and Privacy: Preventing Occupancy Detection from Smart Meters“. In: *2014 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, S. 208–215. DOI: 10.1109/PerCom.2014.6813962.
- Chen, K., Wang, Q., He, Z., Chen, K., Hu, J. und He, J. (2018). „Convolutional Sequence to Sequence Non-Intrusive Load Monitoring“. In: arXiv: 1806.02078 [cs, stat].
- Chen, X., Duan, Y., Houthoofd, R., Schulman, J., Sutskever, I. und Abbeel, P. (2016). „InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets“. In: arXiv: 1606.03657 [cs, stat].
- Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H. und Bengio, Y. (2014). „Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation“. In: arXiv: 1406.1078 [cs, stat].
- D’Incecco, M., Squartini, S. und Zhong, M. (2019). „Transfer Learning for Non-Intrusive Load Monitoring“. In: arXiv: 1902.08835 [cs, stat].
- De Baets, L., Ruysinck, J., Develder, C., Dhaene, T. und Deschrijver, D. (2018). „Appliance Classification Using VI Trajectories and Convolutional Neural Networks“. In: *Energy and Buildings* 158, S. 32–36. DOI: 10.1016/j.enbuild.2017.09.087.
- Drenker, S. und Kader, A. (1999). „Nonintrusive Monitoring of Electric Loads“. In: *IEEE Computer Applications in Power* 12.4, S. 47–51. DOI: 10.1109/67.795138.
- Dwork, C. (2006). „Differential Privacy“. In: *Automata, Languages and Programming*. Hrsg. von M. Bugliesi, B. Preneel, V. Sassone und I. Wegener. Bearb. von D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi und G. Weikum. Bd. 4052. Berlin, Heidelberg: Springer Berlin Heidelberg, S. 1–12. DOI: 10.1007/11787006_1.
- Dwork, C. (2008). „Differential Privacy: A Survey of Results“. In: *Theory and Applications of Models of Computation*. Hrsg. von M. Agrawal, D. Du, Z. Duan und A. Li. Bd. 4978. Berlin, Heidelberg: Springer Berlin Heidelberg, S. 1–19. DOI: 10.1007/978-3-540-79228-4_1.
- Dwork, C. und Roth, A. (2013). „The Algorithmic Foundations of Differential Privacy“. In: *Foundations and Trends® in Theoretical Computer Science* 9.3-4, S. 211–407. DOI: 10.1561/04000000042.
- Egarter, D., Prokop, C. und Elmenreich, W. (2014). „Load Hiding of Household’s Power Demand“. In: arXiv: 1406.2534 [cs].

- Eibl, G., Bao, K., Grassal, P.-W., Bernau, D. und Schmeck, H. (2018). „The Influence of Differential Privacy on Short Term Electric Load Forecasting“. In: *Energy Informatics* 1.1, S. 48. DOI: 10.1186/s42162-018-0025-3.
- Eibl, G. und Engel, D. (2014). „Influence of Data Granularity on Nonintrusive Appliance Load Monitoring“. In: ACM Press, S. 147–151. DOI: 10.1145/2600918.2600920.
- Engel, D. (2013). „Wavelet-Based Load Profile Representation for Smart Meter Privacy“. In: *Innovative Smart Grid Technologies (ISGT), 2013 IEEE PES. Innovative Smart Grid Technologies (ISGT), 2013 IEEE PES*, S. 1–6. DOI: 10.1109/ISGT.2013.6497835.
- Ephrat, A., Mosseri, I., Lang, O., Dekel, T., Wilson, K., Hassidim, A., Freeman, W. T. und Rubinstein, M. (2018). „Looking to Listen at the Cocktail Party: A Speaker-Independent Audio-Visual Model for Speech Separation“. In: *ACM Transactions on Graphics* 37.4. Comment: Accepted to SIGGRAPH 2018. Project webpage: <https://looking-to-listen.github.io>, S. 1–11. DOI: 10.1145/3197517.3201357. arXiv: 1804.03619.
- Faustine, A., Mvungi, N. H., Kaijage, S. und Michael, K. (2017). „A Survey on Non-Intrusive Load Monitoring Methodies and Techniques for Energy Disaggregation Problem“. In: arXiv: 1703.00785 [cs].
- FERC (2015). *Energy Primer, a Handbook of Energy Market Basics*.
- Filip, A. (2011). „Blued: A Fully Labeled Public Dataset for Event-Based Nonintrusive Load Monitoring Research“. In: *Proceedings of the 2nd KDD Workshop on Data Mining Applications in Sustainability (SustKDD)*, S. 2012.
- Finster, S. (2013). „Smart Meter Speed Dating, Short-Term Relationships for Improved Privacy in Smart Metering“. In: *2013 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, S. 426–431. DOI: 10.1109/SmartGridComm.2013.6687995.
- Finster, S. und Baumgart, I. (2014). „SMART-ER: Peer-Based Privacy for Smart Metering“. In: *2014 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHP)*, S. 652–657. DOI: 10.1109/INFOCOMW.2014.6849308.
- Förderer, K., Ahrens, M., Bao, K., Mauser, I. und Schmeck, H. (2018). „Modeling Flexibility Using Artificial Neural Networks“. In: *Energy Informatics* 1.1, S. 21. DOI: 10.1186/s42162-018-0024-4.
- Garcia, F. C. C. und Macabebe, E. Q. B. (2017). „Inferring Appliance Energy Usage from Smart Meters Using Fully Convolutional Encoder Decoder Networks“. In: *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. CIKM '17*. New York, NY, USA: ACM, S. 2075–2078. DOI: 10.1145/3132847.3133146.
- Garcia, F. D. und Jacobs, B. (2011). „Privacy-Friendly Energy-Metering via Homomorphic Encryption“. In: *Security and Trust Management*. Springer, S. 226–238.
- Ghahramani, Z. und Jordan, M. I. (1996). „Factorial Hidden Markov Models“. In: *Advances in Neural Information Processing Systems*, S. 472–478.
- Gisler, C., Ridi, A., Zufferey, D., Khaled, O. A. und Hennebert, J. (2013). „Appliance Consumption Signature Database and Recognition Test Protocols“. In: *2013 8th International Workshop on Systems, Signal Processing and Their Applications (WoSSPA)*, S. 336–341. DOI: 10.1109/WoSSPA.2013.6602387.

- Glorot, X. und Bengio, Y. (2010). „Understanding the Difficulty of Training Deep Feedforward Neural Networks“. In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, S. 249–256.
- Goodfellow, I., Bengio, Y. und Courville, A. (2016). *Deep Learning*. MIT press.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. und Bengio, Y. (2014). „Generative Adversarial Networks“. In: arXiv: 1406.2661 [cs, stat].
- Greveler, U., Justus, B. und Loehr, D. (2012). „Multimedia Content Identification through Smart Meter Power Usage Profiles“. In: *Computers, Privacy and Data Protection* 1, S. 10.
- Gupta, S., Reynolds, M. S. und Patel, S. N. (2010). „ElectriSense: Single-Point Sensing Using EMI for Electrical Event Detection and Classification in the Home“. In: *Proceedings of the 12th ACM International Conference on Ubiquitous Computing*. UbiComp '10. New York, NY, USA: ACM, S. 139–148. DOI: 10.1145/1864349.1864375.
- Hart, G. (1989). „Residential Energy Monitoring and Computerized Surveillance via Utility Power Flows“. In: *IEEE Technology and Society Magazine* 8.2, S. 12–16. DOI: 10.1109/44.31557.
- Hart, G. (1992). „Nonintrusive Appliance Load Monitoring“. In: *Proceedings of the IEEE* 80.12, S. 1870–1891. DOI: 10.1109/5.192069.
- He, K., Stankovic, L., Liao, J. und Stankovic, V. (2018). „Non-Intrusive Load Disaggregation Using Graph Signal Processing“. In: *IEEE Transactions on Smart Grid* 9.3, S. 1739–1747. DOI: 10.1109/TSG.2016.2598872.
- He, K., Zhang, X., Ren, S. und Sun, J. (2015a). „Deep Residual Learning for Image Recognition“. In: arXiv: 1512.03385 [cs].
- He, K., Zhang, X., Ren, S. und Sun, J. (2015b). „Delving Deep into Rectifiers: Surpassing Human-Level Performance on Imagenet Classification“. In: *Proceedings of the IEEE International Conference on Computer Vision*, S. 1026–1034.
- He, K., Zhang, X., Ren, S. und Sun, J. (2016). „Identity Mappings in Deep Residual Networks“. In: Comment: ECCV 2016 camera-ready. arXiv: 1603.05027 [cs].
- He, W. und Chai, Y. (2016). „An Empirical Study on Energy Disaggregation via Deep Learning“. In: *Advances in Intelligent Systems Research* 133.
- Hébrail, G. und Bérard, A. (2012). *Individual Household Electric Power Consumption Data Set*. URL: <https://archive.ics.uci.edu/ml/datasets/Individual+household+electric+power+consumption>.
- Hinton, G. E. und Salakhutdinov, R. R. (2006). „Reducing the Dimensionality of Data with Neural Networks“. In: *Science* 313.5786, S. 504–507. DOI: 10.1126/science.1127647. pmid: 16873662.
- Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I. und Salakhutdinov, R. R. (2012). „Improving Neural Networks by Preventing Co-Adaptation of Feature Detectors“. In: arXiv: 1207.0580 [cs].
- Hochreiter, S. und Schmidhuber, J. (1997). „Long Short-Term Memory“. In: *Neural Comput.* 9.8, S. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

- Hosseini, S. S., Agbossou, K., Kelouwani, S. und Cardenas, A. (2017). „Non-Intrusive Load Monitoring through Home Energy Management Systems: A Comprehensive Review“. In: *Renewable and Sustainable Energy Reviews* 79, S. 1266–1274. DOI: 10.1016/j.rser.2017.05.096.
- Ioffe, S. und Szegedy, C. (2015). „Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift“. In: arXiv: 1502.03167 [cs].
- Iqbal, H. (2018). *HarisIqbal88/PlotNeuralNet v1.0.0*. Version v1.0.0. Zenodo. DOI: 10.5281/ZENODO.2526396.
- Isola, P., Zhu, J.-Y., Zhou, T. und Efros, A. A. (2016). „Image-to-Image Translation with Conditional Adversarial Networks“. In: Website: <https://phillipi.github.io/pix2pix/>. arXiv: 1611.07004 [cs].
- Jawurek, M., Johns, M. und Kerschbaum, F. (2011). „Plug-In Privacy for Smart Metering Billing“. In: *Privacy Enhancing Technologies*. Hrsg. von S. Fischer-Hübner und N. Hopper. Lecture Notes in Computer Science 6794. Springer Berlin Heidelberg, S. 192–210.
- Ji, T. Y., Liu, L., Wang, T. S., Lin, W. B., Li, M. S. und Wu, Q. H. (2019). „Non-Intrusive Load Monitoring Using Additive Factorial Approximate Maximum a Posteriori Based on Iterative Fuzzy μ -Means“. In: *IEEE Transactions on Smart Grid* 10.6, S. 6667–6677. DOI: 10.1109/TSG.2019.2909931.
- Johnson, M. J. und Willsky, A. S. (2013). „Bayesian Nonparametric Hidden Semi-Markov Models“. In: *J. Mach. Learn. Res.* 14.1, S. 673–701.
- Kalogridis, G., Efthymiou, C., Denic, S. Z., Lewis, T. A. und Cepeda, R. (2010). „Privacy for Smart Meters: Towards Undetectable Appliance Load Signatures“. In: *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference On*. IEEE, S. 232–237.
- Kelly, J. und Knottenbelt, W. (2015a). „Neural NILM: Deep Neural Networks Applied to Energy Disaggregation“. In: ACM Press, S. 55–64. DOI: 10.1145/2821650.2821672.
- Kelly, J. und Knottenbelt, W. (2015b). „The UK-DALE Dataset, Domestic Appliance-Level Electricity Demand and Whole-House Demand from Five UK Homes“. In: *Scientific data* 2, S. 150007.
- Kelly, J. und Knottenbelt, W. (2016). „Does Disaggregated Electricity Feedback Reduce Domestic Electricity Consumption? A Systematic Review of the Literature“. In: Comment: Accepted for oral presentation at the 3rd International NILM Workshop, Vancouver, 14-15 May 2016. arXiv: 1605.00962 [cs].
- Kim, H., Marwah, M., Arlitt, M. F., Lyon, G. und Han, J. (2011). „Unsupervised Disaggregation of Low Frequency Power Measurements.“ In: *SDM*. Bd. 11. SIAM, S. 747–758.
- Kingma, D. P. und Ba, J. (2014). „Adam: A Method for Stochastic Optimization“. In: Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015. arXiv: 1412.6980 [cs].
- Kleiminger, W., Beckel, C., Staake, T. und Santini, S. (2013). „Occupancy Detection from Electricity Consumption Data“. In: ACM Press, S. 1–8. DOI: 10.1145/2528282.2528295.

- Kochanneck, S. (2019). „Systemdienstleistungserbringung durch intelligente Gebäude“. 37.06.01; LK 01. Dissertation. Karlsruher Institut für Technologie (KIT). DOI: 10.5445/IR/1000089171.
- Kolter, J. Z., Batra, S. und Ng, A. Y. (2010). „Energy Disaggregation via Discriminative Sparse Coding“. In: *Advances in Neural Information Processing Systems*, S. 1153–1161.
- Kolter, J. Z. und Jaakkola, T. (2012). „Approximate Inference in Additive Factorial Hmms with Application to Energy Disaggregation“. In: *International Conference on Artificial Intelligence and Statistics*, S. 1472–1482.
- Kolter, J. Z. und Johnson, M. J. (2011). „REDD: A Public Data Set for Energy Disaggregation Research“. In: *Proceedings of the 1st KDD Workshop on Data Mining Applications in Sustainability (SustKDD)*. San Diego, CA.
- Kramer, O., Wilken, O., Beenken, P., Hein, A., Hüwel, A., Klingenberg, T., Meinecke, C., Raabe, T. und Sonnenschein, M. (2012). „On Ensemble Classifiers for Nonintrusive Appliance Load Monitoring“. In: *Hybrid Artificial Intelligent Systems*. Springer, S. 322–331.
- Krizhevsky, A., Sutskever, I. und Hinton, G. E. (2012). „ImageNet Classification with Deep Convolutional Neural Networks“. In: *Advances in Neural Information Processing Systems 25*. Hrsg. von F. Pereira, C. J. C. Burges, L. Bottou und K. Q. Weinberger. Curran Associates, Inc., S. 1097–1105.
- Krystalakos, O., Nalmpantis, C. und Vrakas, D. (2018). „Sliding Window Approach for Online Energy Disaggregation Using Artificial Neural Networks“. In: *Proceedings of the 10th Hellenic Conference on Artificial Intelligence*. SETN '18. New York, NY, USA: ACM, 7:1–7:6. DOI: 10.1145/3200947.3201011.
- Kursawe, K., Danezis, G. und Kohlweiss, M. (2011). „Privacy-Friendly Aggregation for the Smart-Grid“. In: *Privacy Enhancing Technologies*. Springer, S. 175–191.
- Lam, H. Y., Fung, G. S. K. und Lee, W. K. (2007). „A Novel Method to Construct Taxonomy Electrical Appliances Based on Load Signatures“. In: *IEEE Transactions on Consumer Electronics* 53.2, S. 653–660. DOI: 10.1109/TCE.2007.381742.
- Landwehr, C. E. und Valdes, A. (2017). „Building Code for Power System Software Security“. In: *Technical Report. IEEE Computer Society*.
- Laughman, C., Lee, K., Cox, R., Shaw, S., Leeb, S., Norford, L. und Armstrong, P. (2003). „Power Signature Analysis“. In: *IEEE Power and Energy Magazine* 1.2, S. 56–63. DOI: 10.1109/MPAE.2003.1192027.
- LeCun, Y. (2017). „Generative Adversarial Networks (GAN)“. Folie 34.
- LeCun, Y., Bottou, L., Orr, G. B. und Müller, K.-R. (2012). „Efficient BackProp“. In: *Neural Networks: Tricks of the Trade*. Hrsg. von G. Montavon, G. B. Orr und K.-R. Müller. Bd. 7700. Berlin, Heidelberg: Springer Berlin Heidelberg, S. 9–48. DOI: 10.1007/978-3-642-35289-8_3.
- Leeb, S. B., Shaw, S. R. und Kirtley, J. L. (1995). „Transient Event Detection in Spectral Envelope Estimates for Nonintrusive Load Monitoring“. In: *IEEE Transactions on Power Delivery* 10.3, S. 1200–1210. DOI: 10.1109/61.400897.
- Lerch, R. (2016). *Elektrische Messtechnik: analoge, digitale und computergestützte Verfahren*. 7., aktualisierte Auflage. Springer-Lehrbuch. Berlin, Heidelberg: Springer Vieweg. 744 S.

- Li, F., Luo, B. und Liu, P. (2010). „Secure Information Aggregation for Smart Grids Using Homomorphic Encryption“. In: *2010 First IEEE International Conference on Smart Grid Communications*. 2010 First IEEE International Conference on Smart Grid Communications, S. 327–332. DOI: 10.1109/SMARTGRID.2010.5622064.
- Li, N., Li, T. und Venkatasubramanian, S. (2007). „T-Closeness: Privacy Beyond k-Anonymity and l-Diversity“. In: *2007 IEEE 23rd International Conference on Data Engineering*. 2007 IEEE 23rd International Conference on Data Engineering, S. 106–115. DOI: 10.1109/ICDE.2007.367856.
- Liang, J., Ng, S. K. K., Kendall, G. und Cheng, J. W. M. (2010). „Load Signature Study—Part I: Basic Concept, Structure, and Methodology“. In: *IEEE Transactions on Power Delivery* 25.2, S. 551–560. DOI: 10.1109/TPWRD.2009.2033799.
- Lisovich, M., Mulligan, D. und Wicker, S. (2010). „Inferring Personal Information from Demand-Response Systems“. In: *IEEE Security Privacy* 8.1, S. 11–20. DOI: 10.1109/MSP.2010.40.
- Maasoumy, M., Sanandaji, B., Poolla, K. und Vincentelli, A. S. (2013). „BERDS - BERkeley EneRgy Disaggregation Data Set“. In: *Proceedings of the Workshop on Big Learning at the Conference on Neural Information Processing Systems (NIPS)*, S. 1–6.
- MaBiS (2011). *Anlage 1 Zum Beschluss BK6-07-002*. Marktregeln für die Durchführung der Bilanzkreisabrechnung Strom (MaBiS) in der konsolidierten Lesefassung vom 28.10.2011.
- Machanavajjhala, A., Gehrke, J., Kifer, D. und Venkatasubramanian, M. (2006). „L-Diversity: Privacy beyond k-Anonymity“. In: *22nd International Conference on Data Engineering (ICDE'06)*. 22nd International Conference on Data Engineering (ICDE'06), S. 24–24. DOI: 10.1109/ICDE.2006.1.
- Makonin, S., Ellert, B., Bajić, I. V. und Popowich, F. (2016). „Electricity, Water, and Natural Gas Consumption of a Residential House in Canada from 2012 to 2014“. In: *Scientific Data* 3, S. 160037. DOI: 10.1038/sdata.2016.37.
- Mao, X., Li, Q., Xie, H., Lau, R. Y. K., Wang, Z. und Paul Smolley, S. (2017). „Least Squares Generative Adversarial Networks“. In: *Proceedings of the IEEE International Conference on Computer Vision*, S. 2794–2802.
- Mauch, L. und Yang, B. (2015). „A New Approach for Supervised Power Disaggregation by Using a Deep Recurrent LSTM Network“. In: *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, S. 63–67. DOI: 10.1109/GlobalSIP.2015.7418157.
- Mausier, I. (2017). „Multi-Modal Building Energy Management“. Dissertation. Karlsruher Institut für Technologie (KIT). DOI: 10.5445/IR/1000070625.
- Mausier, I., Müller, J., Allerdig, F. und Schmeck, H. (2016). „Adaptive Building Energy Management with Multiple Commodities and Flexible Evolutionary Optimization“. In: *Renewable Energy. Optimization Methods in Renewable Energy Systems Design* 87, S. 911–921. DOI: 10.1016/j.renene.2015.09.003.
- McDaniel, P. und McLaughlin, S. (2009). „Security and Privacy Challenges in the Smart Grid“. In: *IEEE Security and Privacy* 7.3, S. 75–77. DOI: 10.1109/MSP.2009.76.
- McLaughlin, S., McDaniel, P. und Aiello, W. (2011). „Protecting Consumer Privacy from Electric Load Monitoring“. In: *ACM Press*, S. 87. DOI: 10.1145/2046707.2046720.

- Metz, L., Poole, B., Pfau, D. und Sohl-Dickstein, J. (2016). „Unrolled Generative Adversarial Networks“. In: arXiv: 1611.02163 [cs, stat].
- Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill Series in Computer Science. New York: McGraw-Hill. 414 S.
- Mittelsdorf, M., Hüwel, A., Klingenberg, T. und Sonnenschein, M. (2013). „Submeter Based Training of Multi-Class Support Vector Machines for Appliance Recognition in Home Electricity Consumption Data.“ In: *SMARTGREENS*, S. 151–158.
- Molina-Markham, A., Shenoy, P., Fu, K., Cecchet, E. und Irwin, D. (2010). „Private Memoirs of a Smart Meter“. In: *Proceedings of the 2nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building*. ACM, S. 61–66.
- Monacchi, A., Egarter, D., Elmenreich, W., D’Alessandro, S. und Tonello, A. (2014). „GREEND: An Energy Consumption Dataset of Households in Italy and Austria“. In: *2014 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, S. 511–516. DOI: 10.1109/SmartGridComm.2014.7007698.
- MsbG (2016). *Messstellenbetriebsgesetz*. vom 29. August 2016 (BGBl. I S. 2034), das durch Artikel 15 des Gesetzes vom 22. Dezember 2016 (BGBl. I S. 3106) geändert worden ist.
- Mühl, T. (2014). *Einführung in die elektrische Messtechnik: Grundlagen, Messverfahren, Anwendungen*. 4., aktualisierte und erw. Aufl. Lehrbuch. Wiesbaden: Springer Vieweg. 300 S.
- Mültin, M., Allerding, F. und Schmeck, H. (2012). „Integration of Electric Vehicles in Smart Homes - an ICT-Based Solution for V2G Scenarios“. In: *2012 IEEE PES Innovative Smart Grid Technologies (ISGT)*. 2012 IEEE PES Innovative Smart Grid Technologies (ISGT), S. 1–8. DOI: 10.1109/ISGT.2012.6175624.
- Murray, D., Stankovic, L. und Stankovic, V. (2017). „An Electrical Load Measurements Dataset of United Kingdom Households from a Two-Year Longitudinal Study“. In: *Scientific Data* 4, S. 160122. DOI: 10.1038/sdata.2016.122.
- Murray, David (2015). „REFIT: Electrical Load Measurements“. In: DOI: 10.15129/31da3ecef902-4e95-a093-e0a9536983c4.
- Nalmpantis, C. und Vrakas, D. (2018). „Machine Learning Approaches for Non-Intrusive Load Monitoring: From Qualitative to Quantitative Comparison“. In: *Artificial Intelligence Review*. DOI: 10.1007/s10462-018-9613-7.
- Norford, L. K. und Leeb, S. 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, S. 51–64. DOI: 10.1016/0378-7788(95)00958-2.
- Nowozin, S., Cseke, B. und Tomioka, R. (2016). „F-GAN: Training Generative Neural Samplers Using Variational Divergence Minimization“. In: arXiv: 1606.00709 [cs, stat].
- Odena, A., Olah, C. und Shlens, J. (2016). „Conditional Image Synthesis With Auxiliary Classifier GANs“. In: arXiv: 1610.09585 [cs, stat].
- Paetz, A., Kaschub, T., Jochem, P. und Fichtner, W. (2013). „Load-Shifting Potentials in Households Including Electric Mobility - A Comparison of User Behaviour with Modelling Results“. In: *2013 10th International Conference on the European Energy Market (EEM)*, S. 1–7. DOI: 10.1109/EEM.2013.6607324.

- Paetz, A.-G., Kaschub, T., Jochem, P. und Fichtner, W. (2011). „Erfahrungen Mit Dynamischen Tarifkonzepten Im Intelligenten Haus“. In: *Tagungsband VDE Kongress*.
- Paillier, P. (1999). „Public-Key Cryptosystems Based on Composite Degree Residuosity Classes“. In: *Advances in Cryptology — EUROCRYPT '99*. Hrsg. von J. Stern. Lecture Notes in Computer Science 1592. Springer Berlin Heidelberg, S. 223–238.
- Parson, O. (2014). „Unsupervised Training Methods for Non-Intrusive Appliance Load Monitoring from Smart Meter Data“. University of Southampton.
- Parson, O., Ghosh, S., Weal, M. und Rogers, A. (2012). „Non-Intrusive Load Monitoring Using Prior Models of General Appliance Types“. In: *AAAI*.
- Parson, O., Ghosh, S., Weal, M. und Rogers, A. (2014). „An Unsupervised Training Method for Non-Intrusive Appliance Load Monitoring“. In: *Artificial Intelligence* 217, S. 1–19. DOI: 10.1016/j.artint.2014.07.010.
- Paulo Marques do Nascimento, P. (2016). „Applications of Deep Learning Techniques on NILM“.
- Pereira, L. und Nunes, N. (2018). „Performance Evaluation in Non-Intrusive Load Monitoring: Datasets, Metrics, and Tools—A Review“. In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 0.0, e1265. DOI: 10.1002/widm.1265.
- Pereira, L., Quintal, F., Gonçalves, R. und Nunes, N. J. (2014). „SustData: A Public Dataset for ICT4S Electric Energy Research“. In: *ICT for Sustainability 2014 (ICT4S-14)*. Atlantis Press. DOI: 10.2991/ict4s-14.2014.44.
- Quinn, E. L. (2009). *Privacy and the New Energy Infrastructure*. SSRN Scholarly Paper ID 1370731. Rochester, NY: Social Science Research Network.
- Radford, A., Metz, L. und Chintala, S. (2015). „Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks“. In: arXiv: 1511.06434 [cs].
- Rashid, H., Singh, P., Stankovic, V. und Stankovic, L. (2019). „Can Non-Intrusive Load Monitoring Be Used for Identifying an Appliance’s Anomalous Behaviour?“ In: *Applied Energy* 238, S. 796–805.
- Rashid, H., Stankovic, V., Stankovic, L. und Singh, P. (2019). „Evaluation of Non-Intrusive Load Monitoring Algorithms for Appliance-Level Anomaly Detection“. In: *2019 IEEE International Conference on Acoustics, Speech, and Signal Processing*. GBR.
- Reinhardt, A., Baumann, P., Burgstahler, D., Hollick, M., Chonov, H., Werner, M. und Steinmetz, R. (2012). „On the Accuracy of Appliance Identification Based on Distributed Load Metering Data“. In: *Sustainable Internet and ICT for Sustainability (SustainIT), 2012*, S. 1–9.
- Reinhardt, A. und Klemenjak, C. (2020). „How Does Load Disaggregation Performance Depend on Data Characteristics?: Insights from a Benchmarking Study“. In: *Proceedings of the Eleventh ACM International Conference on Future Energy Systems*. Virtual Event Australia: ACM, S. 167–177. DOI: 10.1145/3396851.3397691.
- Rial, A. und Danezis, G. (2011). „Privacy-Preserving Smart Metering“. In: *Proceedings of the 10th Annual ACM Workshop on Privacy in the Electronic Society*. ACM, S. 49–60.

- Ribeiro, M., Pereira, L., Quintal, F. und Nunes, N. (2016). „SustDataED: A Public Dataset for Electric Energy Disaggregation Research“. In: *ICT for Sustainability 2016*. Atlantis Press. DOI: 10.2991/ict4s-16.2016.36.
- Rigoll, F. (2017). *Nutzerorientiertes Energiedatenmanagement*. Karlsruhe. DOI: 10.5445/IR/1000068109.
- Ronneberger, O., Fischer, P. und Brox, T. (2015). „U-Net: Convolutional Networks for Biomedical Image Segmentation“. In: Comment: conditionally accepted at MICCAI 2015. arXiv: 1505.04597 [cs].
- Ruder, S. (2016). „An Overview of Gradient Descent Optimization Algorithms“. In: Comment: Added derivations of AdaMax and Nadam. arXiv: 1609.04747 [cs].
- Schwartz, D., Fischhoff, B., Krishnamurti, T. und Sowell, F. (2013). „The Hawthorne Effect and Energy Awareness“. In: *Proceedings of the National Academy of Sciences* 110.38, S. 15242–15246. DOI: 10.1073/pnas.1301687110. pmid: 24003110.
- Shin, C., Joo, S., Yim, J., Lee, H., Moon, T. und Rhee, W. (2018). „Subtask Gated Networks for Non-Intrusive Load Monitoring“. In: arXiv: 1811.06692 [cs, stat].
- Siegl, J. (2010). *Schaltungstechnik - Analog Und Gemischt Analog/Digital*. Springer-Lehrbuch. Berlin, Heidelberg: Springer Berlin Heidelberg. DOI: 10.1007/978-3-642-13304-6.
- Srinivasan, D., Ng, W. S. und Liew, A. C. (2006). „Neural-Network-Based Signature Recognition for Harmonic Source Identification“. In: *IEEE Transactions on Power Delivery* 21.1, S. 398–405. DOI: 10.1109/TPWRD.2005.852370.
- Stankovic, V., Liao, J. und Stankovic, L. (2014). „A Graph-Based Signal Processing Approach for Low-Rate Energy Disaggregation“. In: *2014 IEEE Symposium on Computational Intelligence for Engineering Solutions (CIES)*, S. 81–87. DOI: 10.1109/CIES.2014.7011835.
- Statista (2016). *Haushaltsgeräte - Ausstattung in deutschen Haushalten 2016 | Umfrage*. URL: <https://de.statista.com/statistik/daten/studie/644724/umfrage/umfrage-zur-ausstattung-mit-haushaltsgeraeten-in-deutschen-haushalten/> (besucht am 06.03.2019).
- Statistisches Bundesamt (2018). *Private Haushalte - Ausstattungsgrad mit Haushaltsgeräten 2018 | Statistik*. URL: <https://de.statista.com/statistik/daten/studie/2594/umfrage/ausstattungsgrad-privater-haushalte-mit-haushaltsgeraeten-in-deutschland/> (besucht am 06.03.2019).
- Süddeutsche Zeitung (2011). *Stromverbrauch von Elektrogeräten 2011 | Statistik*. Süddeutsche Zeitung, Nr. 220, 23.09.2011, Seite V2/2. URL: <https://de.statista.com/statistik/daten/studie/201539/umfrage/stromverbrauch-von-elektrogeraeten/> (besucht am 06.03.2019).
- Sutskever, I., Martens, J., Dahl, G. und Hinton, G. (2013). „On the Importance of Initialization and Momentum in Deep Learning“. In: S. 14.
- Sweeney, L. (2002). „K-ANONYMITY: A MODEL FOR PROTECTING PRIVACY“. In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 10.05, S. 557–570. DOI: 10.1142/S0218488502001648.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V. und Rabinovich, A. (2014). „Going Deeper with Convolutions“. In: arXiv: 1409.4842 [cs].

- TR-03109 (2019). *Technische Richtlinie TR-03109. Bundesamt für Sicherheit in der Informationstechnik (BSI)*.
- Ting, K. M. (2017). „Confusion Matrix“. In: *Encyclopedia of Machine Learning and Data Mining*. Hrsg. von C. Sammut und G. I. Webb. Boston, MA: Springer US, S. 260–260.
- Ulyanov, D., Vedaldi, A. und Lempitsky, V. (2016). „Instance Normalization: The Missing Ingredient for Fast Stylization“. In: arXiv: 1607.08022 [cs].
- Uttama Nambi, A. S., Reyes Lua, A. und Prasad, V. R. (2015). „LocED: Location-Aware Energy Disaggregation Framework“. In: *Proceedings of the 2Nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments* (Seoul, South Korea). BuildSys '15. New York, NY, USA: ACM, S. 45–54. DOI: 10.1145/2821650.2821659.
- Valenti, M., Bonfigli, R., Principi, E. und Squartini, a. S. (2018). „Exploiting the Reactive Power in Deep Neural Models for Non-Intrusive Load Monitoring“. In: *2018 International Joint Conference on Neural Networks (IJCNN)*, S. 1–8. DOI: 10.1109/IJCNN.2018.8489271.
- Varodayan, D. und Khisti, A. (2011). „Smart Meter Privacy Using a Rechargeable Battery: Minimizing the Rate of Information Leakage“. In: *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, S. 1932–1935. DOI: 10.1109/ICASSP.2011.5946886.
- Vincent, P., Larochelle, H., Bengio, Y. und Manzagol, P.-A. (2008). „Extracting and Composing Robust Features with Denoising Autoencoders“. In: *Proceedings of the 25th International Conference on Machine Learning* (Helsinki, Finland). ICML '08. New York, NY, USA: ACM, S. 1096–1103. DOI: 10.1145/1390156.1390294.
- DIN 40110-1 (1994). *Wechselstromgrößen; Zweileiter-Stromkreise*. DIN 40110-1:1994-03. Beuth Verlag GmbH. DOI: 10.31030/2572232.
- Xu, H., König, L., Cáliz, D. und Schmeck, H. (2018). „A Generic User Interface for Energy Management in Smart Homes“. In: *Energy Informatics 1.1*, S. 55. DOI: 10.1186/s42162-018-0060-0.
- Yang, W., Li, N., Qi, Y., Qardaji, W., McLaughlin, S. und McDaniel, P. (2012). „Minimizing Private Data Disclosures in the Smart Grid“. In: *Proceedings of the 2012 ACM Conference on Computer and Communications Security*. CCS '12. New York, NY, USA: ACM, S. 415–427. DOI: 10.1145/2382196.2382242.
- Yu, F. und Koltun, V. (2016). „Multi-Scale Context Aggregation by Dilated Convolutions“. In: Comment: Published as a conference paper at ICLR 2016. arXiv: 1511.07122 [cs].
- Yu, H., Jiang, Z., Li, Y., Zhou, J., Wang, K., Cheng, Z. und Gu, Q. (2019). „A Multi-Objective Non-Intrusive Load Monitoring Method Based on Deep Learning“. In: *IOP Conference Series: Materials Science and Engineering* 486, S. 012110. DOI: 10.1088/1757-899X/486/1/012110.
- Zeifman, M. und Roth, K. (2011). „Nonintrusive Appliance Load Monitoring: Review and Outlook“. In: *IEEE Transactions on Consumer Electronics* 57.1, S. 76–84. DOI: 10.1109/TCE.2011.5735484.
- Zhang, C., Zhong, M., Wang, Z., Goddard, N. und Sutton, C. (2016). „Sequence-to-Point Learning with Neural Networks for Nonintrusive Load Monitoring“. In: arXiv: 1612.09106 [cs, stat].

- Zimmermann, J.-P., Evans, M., Griggs, J., King, N., Harding, L., Roberts, P. und Evans, C. (2012). *Household Electricity Survey: A Study of Domestic Electrical Product Usage*. Intertek Testing & Certification Ltd.
- Zoha, A., Gluhak, A., Imran, M. A. und Rajasegarar, S. (2012). „Non-Intrusive Load Monitoring Approaches for Disaggregated Energy Sensing: A Survey“. In: *Sensors* 12.12, S. 16838–16866.

Eidesstattliche Versicherung

Eidesstattliche Versicherung gemäß § 13 Absatz 2 Ziffer 3 der Promotionsordnung des Karlsruher Instituts für Technologie für die KIT-Fakultät für Wirtschaftswissenschaften

1. Bei der eingereichten Dissertation zu dem Thema *Disaggregation von Haushaltsenergiemessdaten mit tiefen neuronalen Netzen* handelt es sich um meine eigenständig erbrachte Leistung.
2. Ich habe nur die angegebenen Quellen und Hilfsmittel benutzt und mich keiner unzulässigen Hilfe Dritter bedient. Insbesondere habe ich wörtlich oder sinngemäß aus anderen Werken übernommene Inhalte als solche kenntlich gemacht.
3. Die Arbeit oder Teile davon habe ich bislang nicht an einer Hochschule des In- oder Auslands als Bestandteil einer Prüfungs- oder Qualifikationsleistung vorgelegt.
4. Die Richtigkeit der vorstehenden Erklärungen bestätige ich.
5. Die Bedeutung der eidesstattlichen Versicherung und die strafrechtlichen Folgen einer unrichtigen oder unvollständigen eidesstattlichen Versicherung sind mir bekannt.

Ich versichere an Eides statt, dass ich nach bestem Wissen die reine Wahrheit erklärt und nichts verschwiegen habe.