BA 201

Bachelor's Thesis

Classification of time series data of building automation systems based on machine learning

Klassifizierung von Zeitreihendaten aus Gebäudeautomationssystemen
unter Anwendung maschinellen Lernens

Aachen, April 2019

Vincent Evenschor
Matriculation number: 322811

Supervisors:
Florian Stinner, M.Sc.
Univ.-Prof. Dr.-Ing. Dirk Müller

The present work was submitted to the:
E.ON Energy Research Center | ERC
Institute for Energy Efficient Buildings and Indoor Climate | EBC
Mathieustrasse 10, 52074 Aachen, Germany

Abstract

Optimized building automation systems are highly promising to contribute for significant energy savings related to the building sector. In practice, most operate inefficiently though, since their analysis and fault detection is difficult, as there exists no standardized scheme for the structure and denomination of data points in building automation systems yet. However, growing numbers of deployed sensors and actuators are encouraging to integrate machine learning for knowledge discovery by using only the produced data itself.

So far most research was focused on unsupervised clustering algorithms or supervised classification of univariate time series. Motivated by the task of separating concrete core activation from other energetic subsystems, a multivariate time series classification approach was chosen to assign sets of data points to classes of energy systems that appear in heating, ventilation and air conditioning by applying state-of-the-art deep learning models.

In use of flow and return temperatures, as well as volume flow that were retrieved from the monitoring database of the E.ON Energy Center, located in Aachen, Germany, multivariate time series datasets were created and preprocessed. The best trained models achieve accuracies beyond 95% on the test data in classifying concrete core activations and façade ventilation unit's heating and cooling water systems, being able to even distinguish between systems of the same class on energy distribution and utilization level. Further, from an application on datasets with less strict to almost no applied preprocessing, advice for the filtering of raw data is derived in order to allow for good classification performance. Altogether the results indicate great potential to gain contextual knowledge about data points using the applied multivariate classification techniques, while for generalization purposes further research is necessary.

Zusammenfassung

Optimierte Gebäudeautomationssysteme sind vielversprechend, um zu signifikanten Energieeinsparungen im Gebäudebereich beizutragen. Jedoch arbeiten die meisten in der Praxis nicht effizient, da ihre Analyse und Fehlererkennung schwierig sind und es kein standardisiertes Schema für die Struktur und die Benennung von Datenpunkten in Gebäudeautomationssystemen gibt. Eine steigende Anzahl eingesetzter Sensoren und Aktoren ermutigt jedoch dazu maschinelles Lernen zur Wissensgewinnung zu integrieren, indem sie nur die erzeugten Daten selbst verwenden.

Bisher hat sich die Forschung hauptsächlich auf unüberwachte Clustering-Algorithmen oder die überwachte Klassifizierung univariater Zeitreihen konzentriert. Motiviert durch die Aufgabe technische Anlagen voneinander unterscheiden zu können, wurde ein Ansatz auf Basis neuer Deep-Learning-Modelle zur multivariaten Zeitreihenklassifizierung gewählt, um Datenpunkte verschiedenen Klassen von Energiesystemen zuzuordnen, die in der Heizungs-, Lüftungs- und Klimatechnik auftreten.

Unter Verwendung von Vor- und Rücklauftemperaturen sowie Volumenströmen, entnommen aus der Monitoring-Datenbank des E.ON Energy Centers in Aachen, wurden Datensätze multivariate Zeitreihen erstellt und vorbereitet. Die am besten trainierten Modelle erreichen auf den Testdaten Genauigkeiten von mehr als 95% bei der Klassifizierung von Betonkernaktivierungen, Heiz- und Kühlwassersystemen von Fassadenlüftungsgeräten und können dabei sogar zwischen Systemen derselben Klasse auf Energieverteilungs- und Nutzungsebene unterscheiden. Darüber hinaus werden im Zuge einer Anwendung auf Datensätze, an die weniger strenge Anforderungen in Bezug auf die Datenvorbereitung gesetzt wurden, Ratschläge zur Filterung von Rohdaten abgeleitet, die gute Klassifizierungsresultate ermöglichen. Insgesamt deuten die Ergebnisse auf ein großes Potenzial hin mit Hilfe der angewandten Klassifizierung kontextuelles Wissen über Datenpunkte zu gewinnen. Zur Bestimmung einer universelleren Anwendbarkeit sind jedoch weitere Untersuchungen zwingend erforderlich.

Contents

Glossary

Abbreviations

Symbol	**Description**
ALSTM-FCN	Attention long short-term memory fully convolutional network
BA	Building automation
BAS	Building automation system
BN	Batch normalization
CCA	Concrete core activation
Conv1D	One dimensional convolutional layer
FCN	Fully convolutional network
FN	False negative
FP	False positive
FVU	Façade ventilation unit
HVAC	Heating, ventilation and air conditioning
LSTM	Long short-term memory
LSTM-FCN	Long short-term memory fully convolutional network
MALSTM-FCN	Multivariate attention long short-term memory fully convolutional network
MLSTM-FCN	Multivariate long short-term memory fully convolutional network
RNN	Recurrent neural network
ReLU	Rectified linear unit
RR	Reference room
TN	True negative
TP	True positive

Symbols

Symbol	Description
T	Temperature
$\dot{V}$	Volume flow

Indices

Symbol	Description
c	Cooling water
dist	Distribution level
h	Heating water
in	Flow
out	Return
rr	Reference room

List of Figures

List of Tables

1 Introduction

In order to accomplish the climate and energy targets of the European Union of reducing greenhouse gas emissions by at least 40% below 1990 levels by 2030 [Amanatidis, 2019], the role of energy savings in the building sector that account for over 40% of total primary energy consumption cannot be underestimated [Cao et al., 2016]. In this regard, building automation systems (BAS) are considered to have potential for energy savings [Waide et al., 2014]. Especially modern buildings in commercial and public use nowadays possess BAS and represent powerful levers of energy reduction [Fan et al., 2015].

However, as shown in a survey by Fütterer et al. [2017b], in practice, most building automation systems operate inefficiently due to faults in planning, installation, commissioning and deficient maintenance. In addition, the description of sensors and actuators in buildings is not yet standardized and highly individual, since different companies establish different naming schemes, thus making analysis and optimization at a later date more difficult [Stinner et al., 2018]. Crucial and therefore a prerequisite for efficient control is the correct labeling of all data points. In difference to other disciplines, in building automation (BA) the term data point defines an information carrier that continuously provides data describing a current state and as such it can supply sensor data or internal systemic information. In this thesis the term will be used in accordance with this definition. As numbers of installed sensors and data points respectively are steadily increasing, detecting faults is a growing challenge [Chakraborty et al., 2018]. Doing this manually is an extraordinary time-consuming and costly process. Fan et al. [2015] state that data generated from building automation systems is often underused, on the other hand BAS data is typically of poor quality what constitutes an issue that requires powerful data mining methods [Zucker et al., 2015].

Recently, engineers are tackling that problem by incorporating machine learning into the process [Alfred, 2016]. Continuously improving algorithms show promise of serious advancements, especially in terms of prediction, control optimization, fault detection and prevention [Fan et al., 2015]. In this regard, Pritoni et al. [2015] emphasize the importance to gain contextual information of sensor data. This thesis exactly aims at the latter by applying a cutting edge machine learning method from the field of multivariate time series classification to assign sets of data points to particular energetic subsystems and investigates whether that algorithm can improve the analysis of data generated in building automation systems.

1.1 Related Work

In building energy systems the application of both supervised and unsupervised machine learning is currently a major research area [Szilagyi and Wira, 2018]. With respect to knowledge discovery from sensor data, supervised techniques are used to train classification models based on previously labeled time series and then make predictions about the label of unlabeled time series. On the contrary, unsupervised techniques are primarily used in clustering time series data by learning from the data itself and thereby find hidden patterns and similarities [Bell, 2015].

Because of this, Fan et al. [2015] recommend unsupervised clustering as the first means to discover underlying data structures and proposed a framework of its application on BAS data. Bode et al. [2019] showed that unsupervised machine learning can enhance the performance of classifiers by dividing data points that are easily distinguishable. The potential of classifiers that predict labels of single data points was demonstrated before [Gao et al., 2015; Fütterer et al., 2017a]. Recently, Stinner et al. [2019] showed that supervised learning algorithms can be supported by combining real with simulated data and made a standardized dataset in high resolution publicly available that meets the requirements for research in this area. However, research so far has mostly been limited to the classification of univariate time series. Nonetheless these studies verified that machine learning algorithms have the power to significantly improve the capacity to automatically analyze building automation systems and in future can support the process of data analysis and e.g. data point labeling in particular.

1.2 Motivation

The original purpose of this thesis was to train a model that could identify concrete core activation (CCA) by separating these from other energetic subsystems that appear in heating, ventilation and air conditioning (HVAC) using only BAS data. Since thermally activated building structures have characteristic long thermal response times of up to 19 h, depending predominantly on the concrete thickness [Ning et al., 2017], an approach by estimating a system's specific thermal response time first and based on the result classify a system was thought to be constructive. However, considering that data from building automation systems usually provides flow and return temperatures rather than a radiant's actual surface temperature, this turned out to be difficult.

A different approach was motivated by advancements in the field of multivariate time series classification. Recent research has demonstrated that deep neural networks achieve state-of-the-art performances on this task. Karim et al. [2018b] extended their existing univariate classification model to multivariate time series [Karim et al., 2018a] and thus could outperformed most previous state-of-the-art models. The proposed models basically consist of a fully convolutional network (FCN) [Wang et al., 2016] augmented with a long short-term memory recurrent neural network (LSTM RNN) [Hochreiter and Schmidhuber, 1997]. While their first model already produced dramatically better results on many univariate time series benchmarks from the UCR database [Dau et al., 2018], their augmented version also improved the results on a collection of different multivariate time series benchmarks [Pei et al., 2018; Schäfer and Leser, 2017; Dua and Graff, 2017].

So far, BAS research was limited to the classification of univariate time series [Gao et al., 2015; Fütterer et al., 2017a]. The previous mentioned work did not only prove the capacity of deep neural networks when dealing with multivariate time series, but also provided an easy applicable classification algorithm that does not require heavy data preprocessing. For these reasons another approach based on the research by Karim et al. [2018b] was chosen. The objective of this thesis was to merely use water flow and return temperatures, an integral part of most monitoring systems, as time series inputs to classify particular energetic subsystems in HVAC. In extension, also water volume flow should be included to yield better results by increasing the information content of the time series. In doing so the intermediate step of estimating the thermal response time first could be skipped, with the benefit of classifying other types of subsystems with the same method.

1.3 Dataset

All data used was collected at the E.ON Energy Research Center, located in Aachen, Germany. It is a multi-functional building with a variety of different users, containing small and large scale offices, laboratories, as well as seminar and conference rooms. Base loads for heating and cooling are distributed by concrete core activation due to its high thermal capacity. Peak loads in heating and cooling of offices are covered by façade ventilation units (FVU), those of seminar and conference rooms by displacement ventilation with conditioned air, provided by a sorption-supported air handling unit. The thermal comfort of the laboratories is regulated by active chilled beams. In addition, on energy distribution level a distinction is made between different zones of supply hydraulic networks for FVU and CCA. On energy utilization level, ten reference rooms (RR) have been chosen to gather a representative picture. An advanced monitoring, control and interface system, implemented by Fütterer et al. [2013], logs values of over 9000 installed data points and saves them in an event-based database. [Fütterer and Constantin, 2012]

Within the scope of this thesis, data of CCA and FVU was used. On energy distribution level, this resulted in collected data from four concrete core activation supply systems ($\mathrm{CCA_{dist}}$) and two façade ventilation unit supply systems, for each hot and cold water ($\mathrm{FVU_{h,dist}}$ and $\mathrm{FVU_{c,dist}}$). The ten reference rooms provided data from seven concrete core activation systems ($\mathrm{CCA_{rr}}$), five façade ventilation unit systems for each heating and cooling water ($\mathrm{FVU_{h,rr}}$ and $\mathrm{FVU_{c,rr}}$). Active chilled beams were neglected since only one room was equipped with the necessary measurement devices.

For these 25 systems, flow and return temperatures (T_{in}, T_{out}), as well as volume flow ($\dot{V}$) were retrieved from the database that were saved during the period from June 1, 2014 to May 31 , 2018 and prepared as multivariate time series of the length one day. A data granularity with a resolution of 5 min was chosen to represent the data detailed in small time steps. The time series were prepared and preprocessed as described in Section 2.2. For training and testing of the classification models, time series with data point values that did not change for more than six hours were discarded in order to provide a clean dataset by minimizing the share of defect sensor values and at the same time ensure that the training data represents the characteristic dynamical thermal behaviors of the considered systems. A detailed view on the resulting 13448 multivariate time series consisting of only flow and return temperatures, as well of 3272 extended with volume flow is shown in Table 1.1. The extension with volume flow results in a smaller sample size due to an increasing probability of discarding a time series with each added data point. In terms of labeling the time series, it was differentiated between two evaluations. On the one hand, a simplification was made by aggregating CCA_{dist} and CCA_{rr}, $FVU_{h,dist}$ and $FVU_{h,rr}$, as well as $FVU_{c,dist}$ and $FVU_{c,rr}$, resulting

System	$\mathbf{T_{in}}$, $\mathbf{T_{out}}$			$\mathbf{T_{in}}$, $\mathbf{T_{out}}$, $\mathbf{\dot{V}}$		
	CCA	FVU_c	FVU_h	CCA	FVU_c	FVU_h
$CCA_{dist,south}$	132	-	-	22	-	-
$CCA_{dist,north}$	113	-	-	6	-	-
$CCA_{dist,interior}$	130	-	-	33	-	-
$CCA_{dist,conference}$	164	-	-	27	-	-
$FVU_{dist,east}$	-	975	942	-	388	255
$FVU_{dist,west}$	-	1090	977	-	519	310
$\mathbf{\Sigma_{dist}}$	**539**	**2065**	**1919**	**88**	**907**	**565**
RR_1	197	-	-	-	-	-
RR_2	623	620	555	84	117	129
RR_3	712	588	278	112	109	91
RR_4	708	609	389	106	106	131
RR_5	573	575	265	95	96	99
RR_6	652	596	380	117	101	102
RR_7	605	-	-	117	-	-
$\mathbf{\Sigma_{rr}}$	**4070**	**2988**	**1867**	**631**	**529**	**552**
$\mathbf{\Sigma_{dist} + \Sigma_{rr}}$	**4609**	**5053**	**3786**	**719**	**1436**	**1097**
Total samples:		**13448**			**3272**	

Table 1.1: Resulting samples divided into distribution and utilization level

in three classes. On the other hand, distribution and utilization level were separated, resulting in six different classes. It was expected that an aggregation might increase the overall accuracy to predict the right label of a class, whereas the distinction should test the classifier's ability to distinguish even between very similar behaving classes. Considering volume flow as an additional variable was also expected to improve the classification results for both labeling approaches as more information about the behavior of the systems is provided. The datasets were shuffled and split into 70% training and 30% testing data, followed by a training of all models for 250 epochs with a batch size of 128 on a 12 x 2.0 GHz high performance computing unit with 32 GB RAM provided by the RWTH Aachen University. The corresponding training, testing and evaluation process with an additional application on unfiltered BAS data is described in detail in Section 2.2, its results in Section 3, interpretations in Section 4 and a summary with implications of the approach in Section 5.

2 Method

In the beginning of this section, the fundamental concept of the long short-term memory fully convolutional network (LSTM-FCN) and its modifications is briefly introduced by giving a basic understanding of their components. Subsequently, the data preparation, classifier training and evaluation process is presented in Section **??**.

2.1 Theoretical Framework

The following time series classification models subdivide into the basic long short-term memory fully convolutional network (LSTM-FCN), attention long short-term memory fully convolutional network (ALSTM-FCN) [Karim et al., 2018a] and their augmentations MLSTM-FCN and MALSTM-FCN for enhanced performance on multivariate time series [Karim et al., 2018b]. In these models a fully convolutional block is extended by an LSTM block, as shown in Figure 2.1.

Fully convolutional networks (FCN) were first introduced by Long et al. [2014] and are implemented in the form proposed by Wang et al. [2016] that consists of three stacked one dimensional

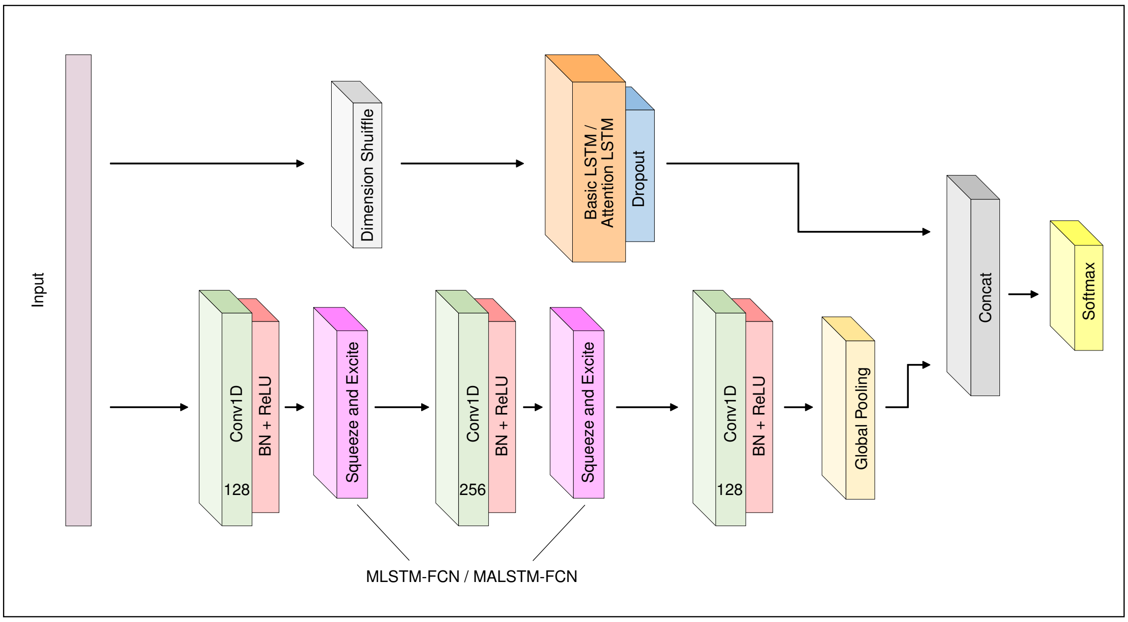


Figure 2.1: Network Architecture of all model variants derived from the LSTM-FCN, adapted from Karim et al. [2018b]

convolutional blocks (Conv1D) with filter sizes 128, 256 and 128, each accompanied by batch normalization (BN) and followed by a rectified linear unit (ReLU) activation function. In the end of the block global average pooling is applied. One dimensional convolutional layers, or temporal convolutions respectively, serve as feature extraction modules and are generally used to process time series signals [Lea et al., 2016]. Batch normalization helps to accelerate the training of deep networks by reducing internal covariate shift, an effect that refers to the fact that a layer's input distribution changes with each weight update and on this way complicates the training process [Ioffe and Szegedy, 2015]. At the same time it acts as a regularizer and thus helps to prevent overfitting. Global average pooling finally takes the average of a feature map that results from a convolution, and thereby can be interpreted to provide the network's confidence for the features of the last convolutional layer [Lin et al., 2013]. For better performance on multivariate time series, Karim et al. [2018b] added a squeeze and excite block to the first two convolutional layers that adaptavely weighs the feature maps and helps to model their interdependencies [Hu et al., 2017].

In parallel operates the LSTM block that Karim et al. [2018a] used to improve the performance of the FCN. Long short-term memory recurrent neural networks were introduced by Hochreiter and Schmidhuber [1997] in order to improve the capacity of usual Recurrent Neural Networks to deal with the vanishing gradient problem and learn long-term dependencies in sequential data. Basic RNN cells hold a hidden vector that is passed from one cell to the next and updated with each time step. That way information can only kept short-term [Pascanu et al., 2013]. LSTM RNN address that problem by having a cell state that serves as a memory and is passed in addition. The LSTM cell looks closely at what information should enter and exit the inner cell. The LSTM has the ability to remove or add information to the cell state, carefully regulated by structures called gates. The input gate controls the extent to which Information can enter, the forget gate the extent to which Information remains and the output gate to which extent information is passed to the next LSTM cell. These additional gates provide a way of optionally letting information through [Graves, 2012]. While LSTM do well in learning temporal dependencies they can still have difficulties with long-term dependencies. For that reason Karim et al. [2018a] added an attention mechanism to the LSTM RNN that is often used in neural translation of texts. It enables the LSTM to focus on more relevant parts of the input and have a more global look on the time series [Bahdanau et al., 2014]. The LSTM RNN, ALSTM RNN respectively, is then followed by a dropout layer. Dropout aims to prevent the network from overfitting by randomly ignoring a fixed amount of neurons during each training step and thus establishes better generalizing neural connections [Srivastava et al., 2014].

While the fully convolutional block gets a multivariate time series with n time steps and m variables, a dimension shuffle is applied on the time series before it enters the LSTM block. The dimension shuffle transposes the input signal to a time series of m time steps and n variables [Karim et al., 2018b]. This is the key for an improvement of fully convolutional block. When no dimension shuffle was applied the network tended to overfit rapidly, with dimension shuffle the LSTM block could enhance the performance of the FCN block [Karim et al., 2019]. At the same time it reduces the complexity of the model and speeds up the training time. After passing the time series through the LSTM block and the FCN block, the resulting feature vectors are concatenated before the softmax function provides a probabilistic distribution for the input over all classes [Bishop, 2009].

2.2 Model Implementation and Evaluation Scheme

Time series creation and preprocessing, model training and evaluation are realized as described in the following parts of this section. All parts of data analysis were implemented in the Python programming language. In terms of data preprocessing, the libraries Numpy and Pandas were used [Jones et al., 2001]. To implement the classification model, Keras with TensorFlow backend [Chollet, 2015] was used, particularly with the aid of the github repository by Karim et al. [2018c] who provided the full source code of their work. In terms of dataset shuffling and splitting, evaluation metrics and visualization purposes, Scikit-learn [Pedregosa et al., 2011], Matplotlib [Hunter, 2007] and Seaborn [Michael Waskom et al., 2018] were used.

The key steps of time series preprocessing, model estimation and application are presented in Figure 2.2. First all relevant data points are combined to multivariate timeseries corresponding to their belonging energetic subsystems whose length and step duration was defined beforehand. In the same step outliers are identified and adjusted with the Hampel Filter [Pearson et al., 2016]. Afterwards time series with data point values that do not change for a previously defined timeout threshold are discarded. At the same moment each time series is given a unique label according to its originating energy system and the chosen labeling approach. Then all samples are shuffled and randomly split into a fixed training and testing data ratio.

In the following, each of the used models, LSTM-FCN, ALSTM-FCN, MLSTM-FCN and MALSTM-

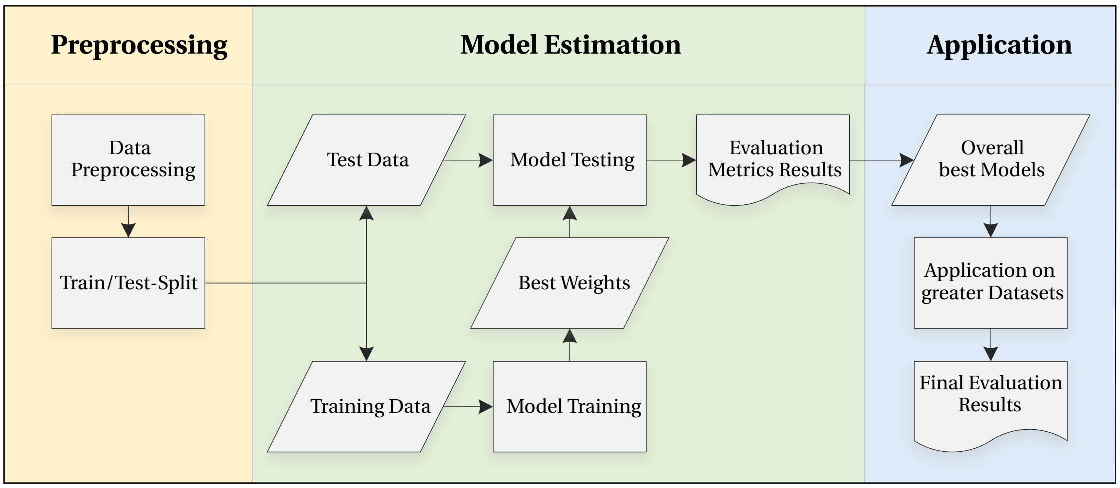


Figure 2.2: Framework for preprocessing of the data, model estimation and application

FCN, are trained independently on each datasets for a previously determined number of epochs and batch size. In neural network training one epoch means that all training samples are passed once through the whole network. Correspondingly, the batch size defines after how many samples the neural network's weights are adjusted. For weight optimization the Adam optimizer is used [Kingma and Ba, 2014]. After each epoch the model's validation accuracy using the testing data is calculated and the latest model weights that generalize best are saved. The validation accuracy is the percentage of correctly predicted labels, the sum of true positives (TP) and true negatives (TN), divided by the total number of predictions made, the sum of true positives, false positives (FP), true negatives and false negatives (FN). [Sammut and Webb, 2017]

$$\mathrm{Accuracy} = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{FP}+\mathrm{TN}+\mathrm{FN}} \tag{2.1}$$

After training has ended, each model is evaluated. Validation accuracy and for each class precision, recall and its harmonic mean, the F_1-score are calculated on the testing dataset. Precision refers to the ratio of correctly predicted positive labels to the total amount of positive predictions, whereas recall refers to the ratio of correctly predicted positive labels to the total amount of samples of that label. In addition, confusion matrices are generated to visualize the performance. [Sammut and Webb, 2017]

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{2.2}$$

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{2.3}$$

$$\mathrm{F}_1 = 2\cdot\frac{\mathrm{Precision}\cdot\mathrm{Recall}}{\mathrm{Precision}+\mathrm{Recall}} \tag{2.4}$$

Since validation accuracy can still be high on imbalanced datasets when less presented classes have a low recall, it is misleading to interpret the overall performance of a model when all classes are equally important. However, there is no single metric that can distinguish all the strengths and weaknesses of a classifier [Lever et al., 2016]. As the F_1-score takes into account both precision and recall, while giving more weight to the lower one, its arithmetic mean over all classes is chosen as the criterion to determine the best model on each dataset. For further evaluation, based on the highest F_1-score, the best models are applied on datasets that provide less filtered time series by varying the temporal limit threshold of the discarding filter. Thereby, the trained models will be also tested on datasets with growing amounts of defect sensor data and afterwards evaluated with the same metrics.

3 Results

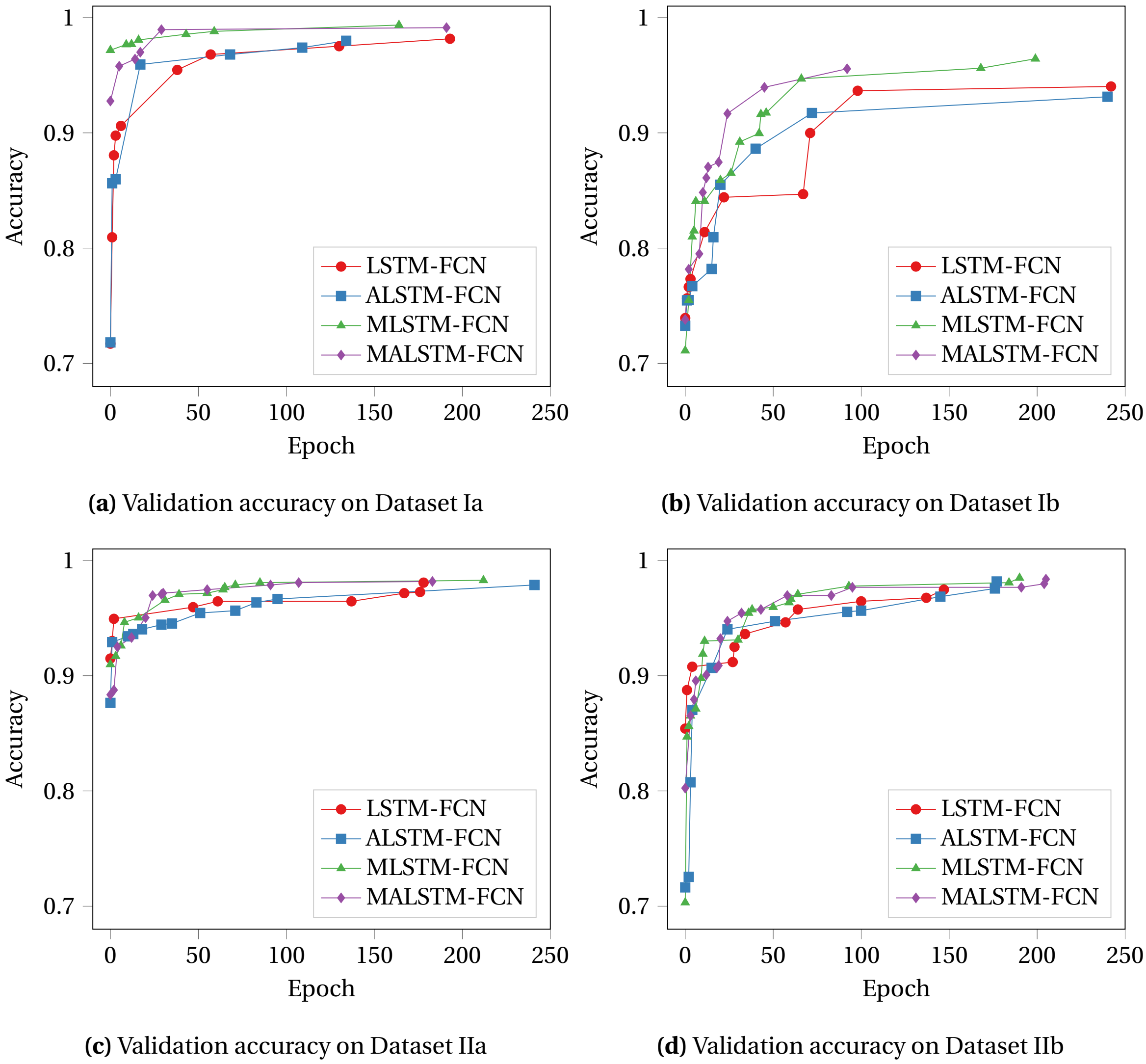
The following section will provide the outcome of the training, evaluation and application procedure presented in Section 2.2 that was applied on the datasets introduced in Section 1.3. Two multivariate time series datasets, with samples that each covered a period of one day, created one time with flow and return temperature and another time with volume flow as an additional, extending variable and two different labeling approaches resulted in four different datasets for model training and evaluation (see Table 3.1). With four model variants to be evaluated on each dataset, accordingly 16 models were trained and evaluated afterwards.

Dataset	Data points	No. of variables	Classes	Time series length	Train-Test Split
Ia	T_{in}, T_{out}	2	3	288, 5 min steps	70%-30% split
Ib	T_{in}, T_{out}	2	6	288, 5 min steps	70%-30% split
IIa	T_{in}, T_{out}, $\dot{V}$	3	3	288, 5 min steps	70%-30% split
IIb	T_{in}, T_{out}, $\dot{V}$	3	6	288, 5 min steps	70%-30% split

Table 3.1: Properties of all datasets used for model training and validation

3.1 Model Training

As shown in Figure 3.1, all trained models quickly reached a validation accuracy beyond 90%. With exception of the models trained on dataset Ib this happened early within the first 10 epochs. For the computation time per epoch one the following times were identified: LSTM-FCN and ALSTM-FCN needed on average 55 seconds on dataset Ia and Ib and 16 seconds on dataset IIa and IIb. The more complex MLSTM-FCN and MALSTM-FCN on the contrary needed on average 65 seconds on Dataset Ia and Ib and 19 seconds on Dataset IIa and IIb. Moreover it can be noticed that in general on all datasets the fastest winnings in validation accuracy could be made by MLSTM-FCN and MALSTM-FCN, almost always laying higher than LSTM-FCN and ALSTM-FCN. Moreover, apart from few exceptions, training for more than 100 epochs seemed to have only very little effect on generating better models on datasets Ia and IIa, whereas for models trained on datasets Ib and IIb there is a still positive slope visible, indicating better models if trained for more than 250 epochs.



(a) Validation accuracy on Dataset Ia

(b) Validation accuracy on Dataset Ib

(c) Validation accuracy on Dataset IIa

(d) Validation accuracy on Dataset IIb

Figure 3.1: Progress on validation accuracy for each model during training

3.2 Model Estimation

Subsequently to the training, all models were tested based on the found best generalizing model weights. All models trained on Dataset Ia, IIa and IIb reached validation accuracies and mean F_1-scores of more than 95%. Except MLSTM-FCN, models trained on Dataset Ib performed clearly worse. Figure 3.2 illustrates explicitly the performances of all models in terms of these metrics. Models trained on Dataset Ia achieved on average a validation accuracy of 98.66%, those trained on Dataset Ib 94.79%. On Dataset IIa and IIb similar high values were identified with 98.10% and 98,13% respectively. In general the models optimized for multivariate time series slightly outperformed the basic variants. MLSTM-FCN achieved the highest accuracies in all cases with 99.36%

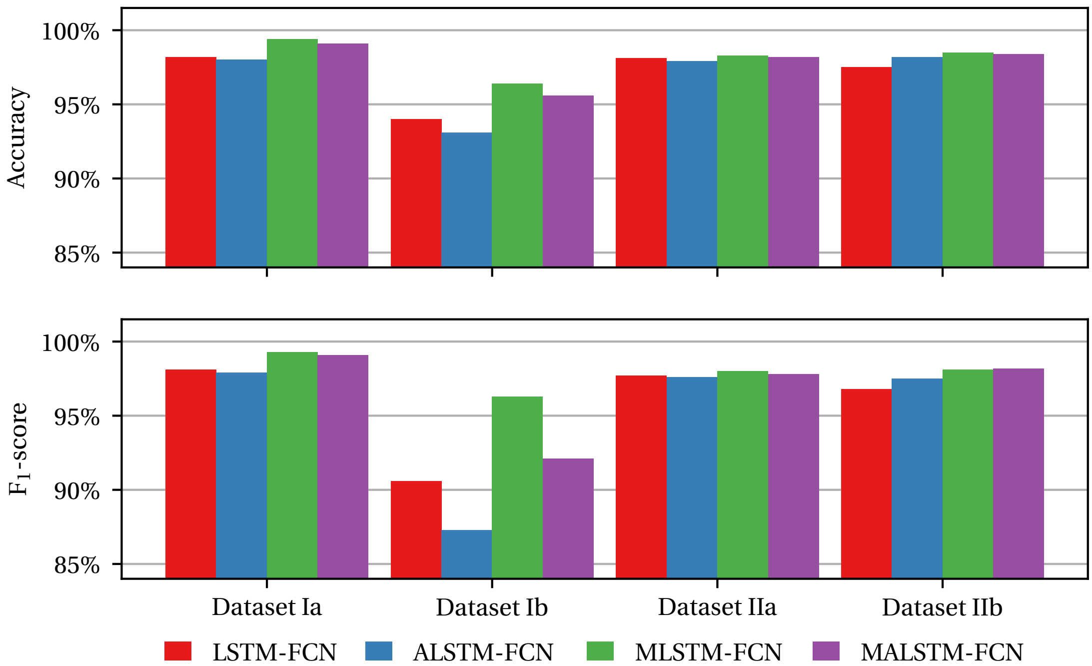
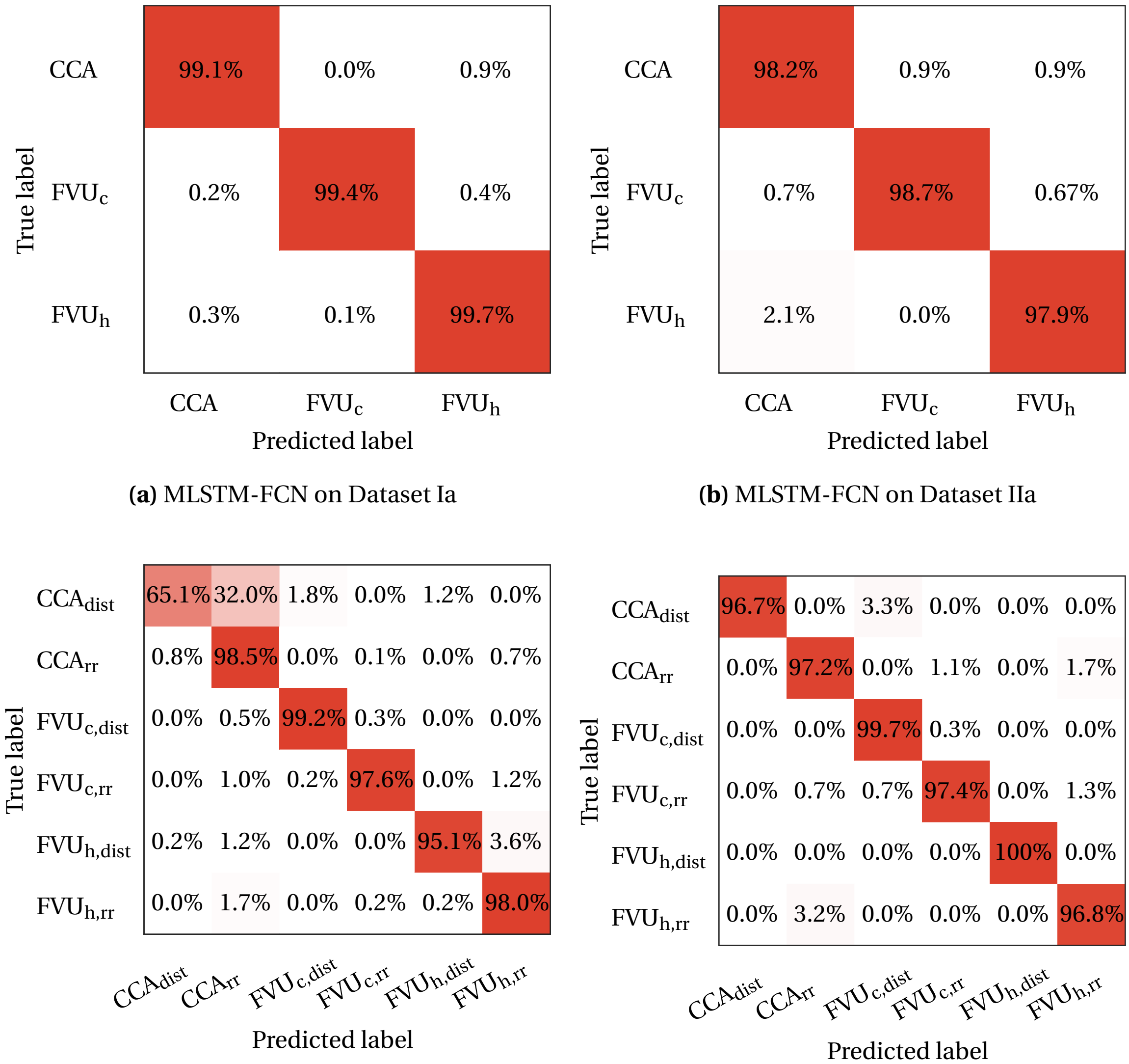


Figure 3.2: Validation accuracy and mean F_1-score per class of all models in comparison

on Dataset Ia, 96.43% on Dataset Ib, 98.28% on Dataset IIa and 98.48% on Dataset IIb. Whereas including volume flow to extend the time series slightly worsened the performance in classification without differentiation between systems from distributional and utilization level, it could improve the results in classification with differentiation.

In view of the mean F_1-scores per class, it becomes clear which models performanced better in consideration of the class imbalanced datasets. On Dataset Ia the models yielded on average a score of 98,61%, on Dataset Ib 91,59%, on Dataset IIa 96,44% and on Dataset IIb a mean of 97,65%. While the MLSTM-FCN still reached the highest values on Dataset Ia, Ib and IIa, the trained MALSTM-FCN model was slightly better on Dataset IIb. It is also striking to see that except from MLSTM-FCN, the discrepance between accuracy and mean F_1-score on Dataset Ib is greatest, indicating worse values of recall and precision for the less represented classes CCA_{dist} and CCA_{rr}. Accordingly, based on the calculated mean F_1-scores, the MLSTM-FCN trained on Dataset Ia with a value of 99.3%, the MLSTM-FCN trained on Dataset Ib with 96.3%, the MLSTM-FCN trained on Dataset IIa and the MALSTM-FCN trained on Dataset IIb are chosen as the best models.

For better visualization of the chosen best model's performances, Figure 3.3 shows the classification results of the normalized confusion matrices. A comparison between the best models on Dataset Ia and IIa illustrates the slightly better performance of the MLSTM-FCN trained on merely



(a) MLSTM-FCN on Dataset Ia

(b) MLSTM-FCN on Dataset IIa

(c) MLSTM-FCN on Dataset Ib

(d) MALSTM-FCN on Dataset IIb

Figure 3.3: Normalized confusion matrices of each dataset's best model

flow and return temperatures, as it yielded besides a higher accuracy also lower precision and recall scores for all classes. In case of the classification task with a distinction between distribution and utilization level, it can be noticed that including volume flow yielded better results, because in the first instance the recall of class CCA$_{dist}$ and the precision of CCA$_{rr}$ were dramatically lower. The trained MLSTM-FCN model on Dataset Ib marked a precision of 98.5% but only a recall of 65.0% for class CCA$_{dist}$, since 32% of its samples were falsely classified as CCA$_{rr}$. Class CCA$_{rr}$ on the contrary had a precision of 73.0%, while almost all samples of that class were correctly identified as such. In summary, the model tended to rather classify CCA systems as CCA$_{rr}$. The inclusion of volume flow as an additional time series variable allowed the trained MALSTM-FCN model on

Dataset IIb to increase both recall of CCA_{dist} and precision of CCA_{rr}. At the same time precision and recall for all other classes improved or changed only slightly, leading to a higher F_1-score in this case.

3.3 Model Application

At last, the determined best models were applied each on extended datasets to test their capacities on less preprocessed data (Figure 3.4). Originating from the training datasets, the time series discarding threshold was incrementally increased in 3h intervals, meaning that samples with longer periods of idle sensor values were added. By these increases the time series became less dynamic and the overall probability to include samples with defect sensor data was enhanced. Starting with 13448 samples at a 6 h threshold for datasets consisting of T_{in} and T_{out} and 3272 for T_{in}, T_{out} and $\dot{V}$, the total sample size grew steadily to 27071 at 21 h threshold, respectively 15622. A threshold of 24h meant that all data was considered as time series were of the length one day, causing a gap from the 21 h to 24 h mark. An increase until 21 h had only little worsening effect on the initial accuracies of MLSTM-FCNs trained on Dataset Ia and IIa. To that point MLSTM-FCN Ia could yield a higher accuracy with 98.2% compared with 97.1% of the MLSTM-FCN IIa. With no filtering applied the accuracy fell almost 5%, both reaching 93.0%.

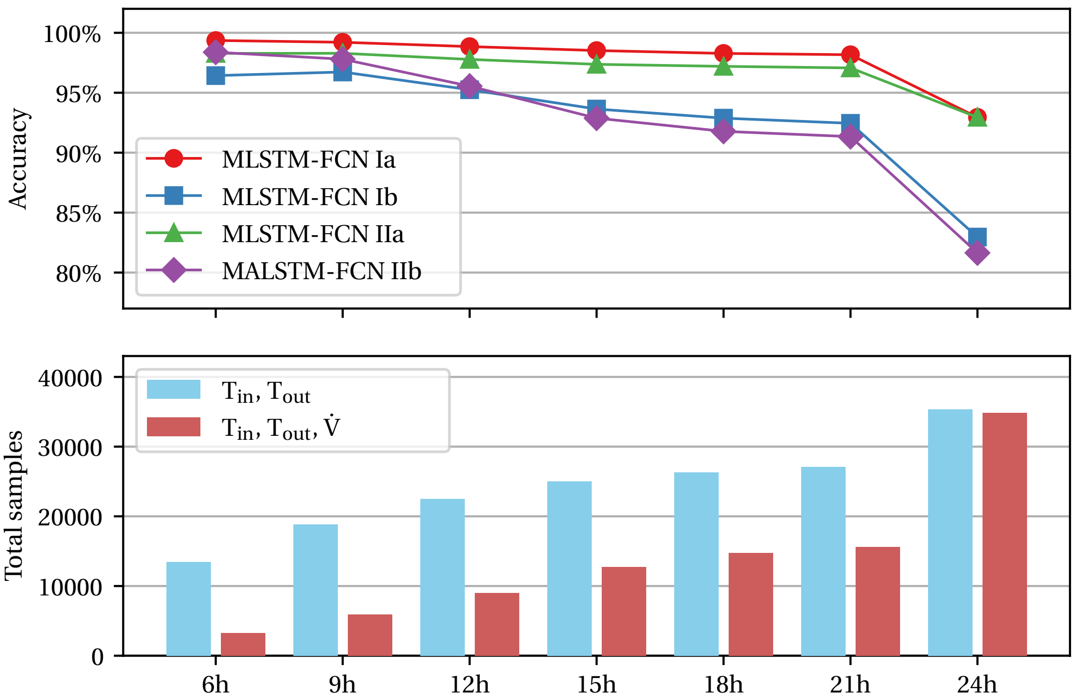
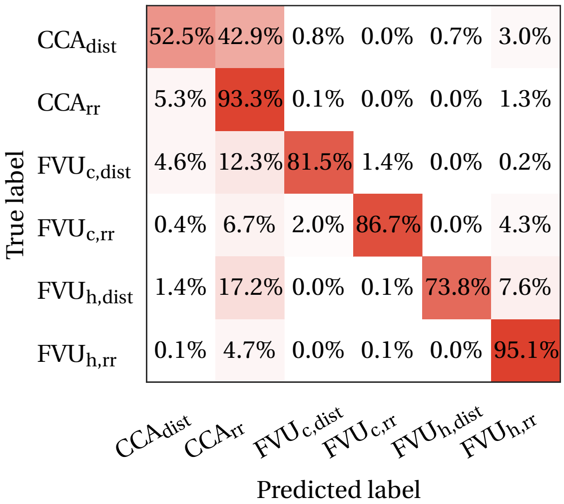
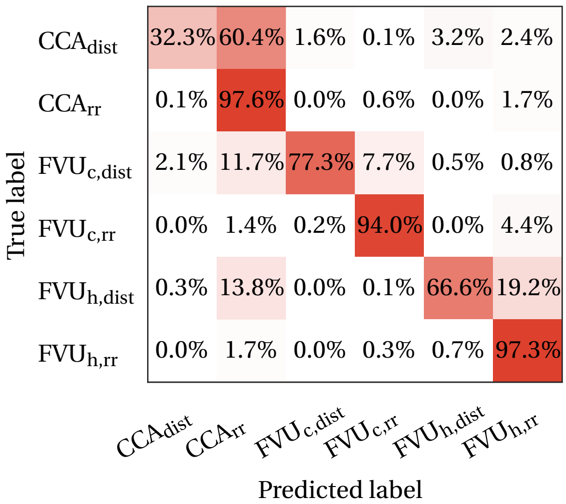


Figure 3.4: Performances of the trained models on datasets with varying discarding threshold



(a) MLSTM-FCN Ib



(b) MALSTM-FCN IIb

Figure 3.5: Best model's normalized confusion matrices with distinction between utilization and distribution level when applied on all data

In case of classification where a distinction between distribution and utilization level was made, the MALSTM-FCN trained on Dataset IIb fell behind the MLSTM-FCN trained on Dataset Ib first at the 12h mark. From 91.0% at 21h to finally 81.2% at 24h, respectively 92.5% to 83.0%, accuracies for both models further crashed. In general it can be noticed that in these cases an extension of the data had a strong worsening effect. For these cases Figure 3.5 contains the new confusion matrices. When all data was considered, classes from distribution and utilization level are less distinguishable. In both cases though, precision for classes FVU$_{c,dist}$, FVU$_{c,rr}$ and FVU$_{h,dist}$, as well as recall of CCA$_{rr}$ and FVU$_{h,rr}$ remained on a high level, while overall a tendency to an increased probability to classify a system on its utilization level could be recognized.

4 Discussion

In this section, a closer look in terms of differences in performance of the investigated models will be taken. Based on this, advice for better classification results and prerequisites for an application in analysis of building automation systems data will be derived.

4.1 Interpretation of the Results

Regarding the model validation results, an aggregation of labels from systems on distribution and utilization level yielded clearly better results on accuracy and F_1-score when flow and return temperatures were considered, as previously expected. In the case of time series extended by volume flow, there was no noticeable change in performance between both approaches. This indicates that with volume flow a more complete image of the energetic system's thermal behavior on the different levels was given to the neural network classification model since a distinction did not worsen the results. On the other side a better performance in classification for the aggregated labeling approach could not be noticed. It is even remarkable that the performance slightly declined. A possible explanation could be the fact a difference in the dataset size. While in the extended case, time series with dynamically changing flow and return temperatures were discarded when volume flow sensor data standed still, time series of these temperatures would not be neglected in the case when merely flow and return temperatures were considered. However, dynamically changing temperatures suggest an existing water flow and thus possibly provided more samples that describe the thermal behavior of a system. The sample size, being four times higher in this case, might overcompensate the lack of information by not including volume flow. In this regard, having equally large datasets could lead to opposite results as especially deep learning benefits from more training samples [Sun et al., 2017].

Taking a closer at the application results, some preprocessing requirements can be deduced based on an examination of the misclassification events. In short, almost all false decisions of the classifiers occurred when there was either no volume flow measured or sensor values were constant up to the duration of a whole time series sample. In the case involving volume flow, in 63.7% of in total 6402 misclassified samples, there was no water flowing for the duration of the whole time series. Constant flow and return temperatures account for 31.2% of the missclassifications. With regard to defect sensors, especially volume flow sensors seemed to be prone for malfunctioning, sometimes standing still several days, in extreme cases even up to more than a week. As volume flow is a physical quantity that can change very quickly, deficiencies of volume flow sensors seem to be a factor that more likely accounts for wrong classifications. With regard to time series composed of only flow and return temperatures, stationary or idle temperature levels lead to an enhanced misclassifications rate. Here, in 32.52% of the total 6023 misclassified samples, temperature sensors seemed to malfunction, since there was no change in their values measured over the duration of the whole time series. Time series whose values are in between a 1 K temperature band contribute for 51.3% of false decisions. An examination of the increase taking volume flow into account, suggested that these events occurred mostly due to no or little measured volume flow. In consideration of these observations, besides detection of defect sensor data, future data preprocessing for multivariate time series classification of energy systems should predominantly aim to filter time series where no volume flow is measured, even when in the end only temperatures are used as variables. This seems to be a prerequisite for good classification results, as the time series can only describe characteristic thermal behavior when water is flowing. For the datasets used for training and validation, these prerequisites are supposed to have been considered implicitly, but not optimally, by using a threshold value of a maximum of six hours that a data point is allowed to be constant.

5 Conclusion

In this thesis, it was investigated how a state-of-the-art neural network classification technique can be used to identify concrete core activation and other energetic subsystems using BAS data. The results even proved to distinguish between supply systems on distribution level and reference rooms on utilization level. An application on unfiltered BAS data further suggested some advice for efficient data preprocessing in order to achieve high accuracies on the task of classification when flow and return temperatures, as well as volume flow are used as variables. It was found, that an application of the used multivariate time series classifiers is promising to yield contextual information about data points, information that is hardly obtained from univariate time series analysis.

5.1 Possible Future Applications

Multivariate time series classification can be expected to have a variety of possible applications in the analysis of BAS data. It could help to reconstruct a building energy systems topology from distance, using merely collected sensor data to identify its components. In terms of fault detection in the denomination of data points, it could be used to improve the mutual validation of time series data, data point labels and topology plans of plants. Apart from identifying different energy systems, multivariate time series classification could be applied to receive information about a particular system's changing operating status that could be integrated in control strategies. However, as the used data in the scope of this thesis came from one building alone, an application that involves other buildings requires extensive further research.

5.2 Directions for Further Research

For generalization purposes, the impact of including data from other building automation systems has still to be investigated. Differences in building structures, compositions of components and control strategies are expected to complicate the classification and probably increase the demands for data preprocessing or data quantity. Moreover building automation systems possess by far more different types of energetic subsystems than the few considered in this thesis. Including these and their effect on the performance should also be addressed. While an extension involving the mentioned aspects makes classification more difficult, derived from the findings in Section 4.1, a primarily focus to filter time series with no measured volume flow in the preprocessing step is expected to be able to improve the performance on this task.

Another point to bear in mind is that it was initially known which data points belonged together. Analyzing BAS from only the data itself, powerful clustering techniques have to be applied in order to find temporal dependencies between data points and thus find these connections. Both machine learning methods in interaction is another topic for future research.

Bibliography

Rayner Alfred. The rise of machine learning for big data analytics. In *2016 2nd International Conference on Science in Information Technology (ICSITech)*, page 1. IEEE, 2016. ISBN 978-1-5090-1721-8. doi: 10.1109/ICSITech.2016.7852593.

Georgios Amanatidis. European policies on climate and energy towards 2020, 2030 and 2050, 2019. URL http://www.europarl.europa.eu/RegData/etudes/BRIE/2019/631047/IPOL_BRI(2019)631047_EN.pdf.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural Machine Translation by Jointly Learning to Align and Translate, 2014. URL http://arxiv.org/pdf/1409.0473v7.

Jason Bell. *Machine learning: Hands-on for developers and technical professionals*. Wiley, Indianapolis, Indiana, 2015. ISBN 9781118889060. doi: 10.1002/9781119183464. URL http://onlinelibrary.wiley.com/book/10.1002/9781119183464.

Christopher M. Bishop. *Pattern recognition and machine learning*. Information science and statistics. Springer, New York, NY, corrected at 8th printing 2009 edition, 2009. ISBN 978-0387-31073-2.

Gerrit Bode, Thomas Schreiber, Marc Baranski, and Dirk Müller. A time series clustering approach for Building Automation and Control Systems. *Applied Energy*, 238:1337–1345, 2019. ISSN 03062619. doi: 10.1016/j.apenergy.2019.01.196.

Xiaodong Cao, Xilei Dai, and Junjie Liu. Building energy-consumption status worldwide and the state-of-the-art technologies for zero-energy buildings during the past decade. *Energy and Buildings*, 128:198–213, 2016. ISSN 03787788. doi: 10.1016/j.enbuild.2016.06.089.

Tusher Chakraborty, Akshay Uttama Nambi, Ranveer Chandra, Rahul Sharma, Manohar Swaminathan, and Zerina Kapetanovic. Sensor Identification and Fault Detection in IoT Systems. In Gowri Sankar Ramachandran and Bhaskar Krishnamachari, editors, *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems - SenSys '18*, pages 375–376, New York, New York, USA, 2018. ACM Press. ISBN 9781450359528. doi: 10.1145/3274783.3275190.

Francois Chollet. Keras, 2015. URL https://keras.io.

Hoang Anh Dau, Eamonn Keogh, Kaveh Kamgar, Chin-Chia Michael Yeh, and Yan Zhu. The UCR Time Series Classification Archive, 2018. URL https://www.cs.ucr.edu/~eamonn/time_series_data_2018/.

Dheeru Dua and Casey Graff. UCI Machine Learning Repository, 2017. URL http://archive.ics.uci.edu/ml.

Cheng Fan, Fu Xiao, and Chengchu Yan. A framework for knowledge discovery in massive building automation data and its application in building diagnostics. *Automation in Construction*, 50: 81–90, 2015. ISSN 09265805. doi: 10.1016/j.autcon.2014.12.006.

Johannes Fütterer and Ana Constantin. Energy Concept for the E.ON ERC Main Building, 2012. URL http://publications.rwth-aachen.de/record/443118/files/5046.pdf.

Johannes Fütterer, Ana Constantin, Martin Schmidt, Rita Streblow, Dirk Muller, and Elias Kosmatopoulos. A multifunctional demonstration bench for advanced control research in buildings—Monitoring, control, and interface system. In *IECON 2013*, pages 5696–5701, Piscataway, NJ, 2013. IEEE. ISBN 978-1-4799-0224-8. doi: 10.1109/IECON.2013.6700068.

Johannes Fütterer, Maksymilian Kochanski, and Dirk Müller. Application of selected supervised learning methods for time series classification in Building Automation and Control Systems. *Energy Procedia*, 122:943–948, 2017a. ISSN 18766102. doi: 10.1016/j.egypro.2017.07.428.

Johannes Peter Fütterer, Thomas Peter Schild, and Dirk Müller. Building Automation Systems in Practice, 2017b.

Jingkun Gao, Joern Ploennigs, and Mario Berges. A Data-driven Meta-data Inference Framework for Building Automation Systems. In David Culler, Yuvraj Agarwal, and Rahul Mangharam, editors, *BuildSys'15*, pages 23–32, New York, New York, 2015. The Association for Computing Machinery. ISBN 9781450339810. doi: 10.1145/2821650.2821670.

Alex Graves. *Supervised Sequence Labelling with Recurrent Neural Networks*, volume 385. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. ISBN 978-3-642-24796-5. doi: 10.1007/978-3-642-24797-2.

Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8): 1735–1780, 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735.

Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu. Squeeze-and-Excitation Networks, 2017. URL http://arxiv.org/pdf/1709.01507v3.

John D. Hunter. Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering*, 9 (3):90–95, 2007. ISSN 1521-9615. doi: 10.1109/MCSE.2007.55.

Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, 2015. URL http://arxiv.org/pdf/1502.03167v3.

Eric Jones, Travis Oliphant, and Pearu Peterson. SciPy: Open source scientific tools for Python, 2001. URL http://www.scipy.org/.

Fazle Karim, Somshubra Majumdar, Houshang Darabi, and Shun Chen. LSTM Fully Convolutional Networks for Time Series Classification. *IEEE Access*, 6:1662–1669, 2018a. doi: 10.1109/ACCESS.2017.2779939. URL http://arxiv.org/pdf/1709.05206v1.

Fazle Karim, Somshubra Majumdar, Houshang Darabi, and Samuel Harford. Multivariate LSTM-FCNs for Time Series Classification, 2018b. URL http://arxiv.org/pdf/1801.04503v1.

Fazle Karim, Somshubra Majumdar, Houshang Darabi, and Samuel Harford. Multivariate LSTM-FCNs for Time Series Classification, 2018c. URL https://github.com/titu1994/MLSTM-FCN.

Fazle Karim, Somshubra Majumdar, and Houshang Darabi. Insights into LSTM Fully Convolutional Networks for Time Series Classification, 2019. URL http://arxiv.org/pdf/1902.10756v2.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization, 2014. URL http://arxiv.org/pdf/1412.6980v9.

Colin Lea, René Vidal, Austin Reiter, and Gregory D. Hager. Temporal Convolutional Networks: A Unified Approach to Action Segmentation. In Gang Hua and Hervé Jégou, editors, *Computer Vision – ECCV 2016 Workshops*, volume 9915 of *Lecture Notes in Computer Science*, pages 47–54. Springer International Publishing, Cham, 2016. ISBN 978-3-319-49408-1. doi: 10.1007/978-3-319-49409-8{\textunderscore}7.

Jake Lever, Martin Krzywinski, and Naomi Altman. Classification evaluation. *Nature Methods*, 13 (8):603–604, 2016. ISSN 1548-7091. doi: 10.1038/nmeth.3945.

Min Lin, Qiang Chen, and Shuicheng Yan. Network In Network, 2013. URL http://arxiv.org/pdf/1312.4400v3.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation, 2014. URL http://arxiv.org/pdf/1411.4038v2.

Michael Waskom, Olga Botvinnik, Drew O'Kane, Paul Hobson, Joel Ostblom, Saulius Lukauskas, David C Gemperline, Tom Augspurger, Yaroslav Halchenko, John B. Cole, Jordi Warmenhoven, Julian de Ruiter, Cameron Pye, Stephan Hoyer, Jake Vanderplas, Santi Villalba, Gero Kunter, Eric Quintero, Pete Bachant, Marcel Martin, Kyle Meyer, Alistair Miles, Yoav Ram, Thomas Brunner, Tal Yarkoni, Mike Lee Williams, Constantine Evans, Clark Fitzgerald, Brian, and Adel Qalieh. mwaskom/seaborn: v0.9.0 (July 2018), 2018.

Baisong Ning, Stefano Schiavon, and Fred S. Bauman. A novel classification scheme for design and control of radiant system based on thermal response time. *Energy and Buildings*, 137:38–45, 2017. ISSN 03787788. doi: 10.1016/j.enbuild.2016.12.013.

Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. How to Construct Deep Recurrent Neural Networks, 2013. URL http://arxiv.org/pdf/1312.6026v5.

Ronald K. Pearson, Yrjö Neuvo, Jaakko Astola, and Moncef Gabbouj. Generalized Hampel Filters. *EURASIP Journal on Advances in Signal Processing*, 2016(1):115, 2016. doi: 10.1186/s13634-016-0383-6.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Andreas Müller, Joel Nothman, Gilles Louppe, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 2011. URL http://arxiv.org/pdf/1201.0490v4.

Wenjie Pei, Hamdi Dibeklioglu, David M. J. Tax, and Laurens van der Maaten. Multivariate Time-Series Classification Using the Hidden-Unit Logistic Model. *IEEE transactions on neural networks and learning systems*, 29(4):920–931, 2018. doi: 10.1109/TNNLS.2017.2651018.

Marco Pritoni, Arka A. Bhattacharya, David Culler, and Mark Modera. Short Paper: A Method for Discovering Functional Relationships Between Air Handling Units and Variable-Air-Volume Boxes From Sensor Data. In David Culler, Yuvraj Agarwal, and Rahul Mangharam, editors, *BuildSys'15*, pages 133–136, New York, New York, 2015. The Association for Computing Machinery. ISBN 9781450339810. doi: 10.1145/2821650.2821677.

Claude Sammut and Geoffrey I. Webb. *Encyclopedia of Machine Learning and Data Mining*. Springer US, Boston, MA, 2017. ISBN 978-1-4899-7685-7. doi: 10.1007/978-1-4899-7687-1.

Patrick Schäfer and Ulf Leser. Multivariate Time Series Classification with WEASEL+MUSE, 2017. URL http://arxiv.org/pdf/1711.11343v4.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

Florian Stinner, Alina Kornas, Marc Baranski, and Dirk Müller. Structuring building monitoring and automation system data. In REHVA, editor, *The REHVA European HVAC Journal - August 2018*, REHVA Journal, pages 10–15. 2018. URL https://www.rehva.eu/fileadmin/user_upload/10-15_RJ1804_WEB.pdf.

Florian Stinner, Yingying Yang, Thomas Schreiber, Bode,Gerrit Baranski, Marc, and Dirk Müller. Generating generic data sets for machine learning applications in building services using standardized time series data. In *Proceedings of the 36th International Symposium on Automation and Robotics in Construction (ISARC)*, Proceedings of the International Symposium on Automation and Robotics in Construction (IAARC). International Association for Automation and Robotics in Construction (IAARC), 2019.

Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting Unreasonable Effectiveness of Data in Deep Learning Era. In *2017 IEEE International Conference on Computer*

Vision, pages 843–852, Piscataway, NJ, 2017. IEEE. ISBN 978-1-5386-1032-9. doi: 10.1109/ICCV.2017.97.

Ioan Szilagyi and Patrice Wira. An intelligent system for smart buildings using machine learning and semantic technologies: A hybrid data-knowledge approach. In *2018 IEEE Industrial Cyber-Physical Systems (ICPS)*, pages 20–25. IEEE, 2018. ISBN 978-1-5386-6531-2. doi: 10.1109/ICPHYS.2018.8387631.

Paul Waide, Jim Ure, Nadia Karagianni, Graham Smith, and Bill Bordass. The scope for energy and CO2 savings in the EU through the use of building automation technology, 2014.

Zhiguang Wang, Weizhong Yan, and Tim Oates. Time series classification from scratch with deep neural networks: A strong baseline, 2016. URL http://arxiv.org/pdf/1611.06455v4.

Gerhard Zucker, Usman Habib, Max Blöchle, Florian Judex, and Thomas Leber. Sanitation and Analysis of Operation Data in Energy Systems. *Energies*, 8(11):12776–12794, 2015. doi: 10.3390/en81112337.

Declaration of Originality

I hereby declare that this thesis and the work reported herein was composed by and originated entirely from me. Information derived from the published and unpublished work of others has been acknowledged in the text and references are given in the list of sources. This thesis has not been submitted as exam work in neither the same nor a similar form. I agree that this thesis may be stored in the institutes library and database. This work may also be copied for internal use.

Aachen, Wednesday 17th April, 2019

Vincent Evenschor