DATA MINING APPROACHES FOR OUTLIER DETECTION ANALYSIS

Shahrooz Abghari

Blekinge Institute of Technology
Doctoral Dissertation Series No. 2020:09

Department of Computer Science

Blekinge Institute of Technology
Blekinge Institute of Technology Doctoral Dissertation Series No. 2020:09
ISSN 1653-2090
ISBN 9789172954090

DATA MINING APPROACHES FOR OUTLIER DETECTION ANALYSIS

Shahrooz Abghari

Akademisk avhandling
som för avläggande av Filosofie doktorsexamen vid
Blekinge Tekniska Högskola kommer att offentligen
försvaras i J1630, Karlskrona, 2020-12-01 , kl. 13.00

Handledare:
Professor Niklas Lavesson, Institutionen för datavetenskap

Professor Håkan Grahn, Institutionen för datavetenskap

Professor Veselka Boeva, Institutionen för datavetenskap

Opponent:
Professor Olga Fink, ETH Zurich, Dept. of Civil, Environmental and Geomatic Engineering, CHE

Blekinge Tekniska Högskola
Institutionen för datavetenskap
Karlskrona

Data Mining Approaches for Outlier Detection Analysis

Shahrooz Abghari

Blekinge Institute of Technology Doctoral Dissertation Series
No 2020:09

Data Mining Approaches for Outlier Detection Analysis

Shahrooz Abghari

Doctoral Dissertation in
Computer Science

BLEKINGE TEKNISKA HÖGSKOLA · BTH ·

Department of Computer Science
Blekinge Institute of Technology
SWEDEN

2020 Shahrooz Abghari
Department of Computer Science
Publisher: Blekinge Institute of Technology
SE-371 79 Karlskrona, Sweden
Printed by Exakta Group, Sweden, 2020
ISBN: 978-91-7295-409-0
ISSN: 1653-2090
urn:nbn:se:bth-20454

“The more I learn, the more I realize how much I don’t know.”

Albert Einstein

Abstract

Outlier detection is studied and applied in many domains. Outliers arise due to different reasons such as fraudulent activities, structural defects, health problems, and mechanical issues. The detection of outliers is a challenging task that can reveal system faults, fraud, and save people's lives. Outlier detection techniques are often domain-specific. The main challenge in outlier detection relates to modelling the normal behaviour in order to identify abnormalities. The choice of model is important, i.e., an unsuitable data model can lead to poor results. This requires a good understanding and interpretation of the data, the constraints, and requirements of the domain problem. Outlier detection is largely an unsupervised problem due to unavailability of labeled data and the fact that labeled data is expensive.

In this thesis, we study and apply a combination of both machine learning and data mining techniques to build data-driven and domain-oriented outlier detection models. We focus on three real-world application domains: maritime surveillance, district heating, and online media and sequence datasets. We show the importance of data preprocessing as well as feature selection in building suitable methods for data modelling. We take advantage of both supervised and unsupervised techniques to create hybrid methods.

More specifically, we propose a rule-based anomaly detection system using open data for the maritime surveillance domain. We exploit sequential pattern mining for identifying contextual and collective outliers in online media data. We propose a minimum spanning tree clustering technique for detection of groups of outliers in online media and sequence data. We develop a few higher order mining approaches for identifying manual changes and deviating behaviours in the heating systems at the building level. The proposed approaches are shown to be capable of explaining the underlying properties of the detected outliers. This can facilitate domain experts in narrowing down the scope of analysis and understanding the reasons of such anomalous behaviours. We also investigate the reproducibility of the proposed models in similar application domains.

Preface

Included Papers

This thesis consists of seven papers. In Paper I, the author has been one of the main drivers. While in the next six papers, he has been the main driver. The studies in all papers have been developed and designed under the guidance of the supervisors and domain experts. The formatting of the published papers included in this thesis has been changed to achieve a consistent style.

Paper I Kazemi, S., Abghari, S., Lavesson, N., Johnson, H., & Ryman, P. "Open data for anomaly detection in maritime surveillance". *Expert Systems with Applications*, (40)14 (2013), pp. 5719-5729. DOI: 10.1016/J.ESWA.2013.04.029

Paper II Abghari, S., Boeva, V., Lavesson, N., Grahn, H., Gustafsson, J., & Shaikh, J. "Outlier detection for video session data using sequential pattern mining". In *Association for Computing Machinery's Special Interest Group on Knowledge Discovery and Data Mining: Workshop On Outlier Detection De-constructed*, 2018, London, UK.

Paper III Abghari, S., Boeva, V., Lavesson, N., Grahn, H., Ickin, S., & Gustafsson, J. "A minimum spanning tree clustering approach for outlier detection in event sequences". In *2018 17th IEEE International Conference on Machine Learning and Applications* (pp. 1123-1130). DOI: 10.1109/ICMLA.2018.00182.

Paper IV Abghari, S., Garcia-Martin, E., Johansson, C., Lavesson, N., & Grahn, H. "Trend analysis to automatically identify heat program changes". *Energy Procedia*, 116 (2017), pp. 407-415. DOI: 10.1016/J.EGYPRO.2017.05.088, The paper was presented at the 2016 15th International Symposium on District Heating and Cooling, Seoul, Korea.

Paper V Abghari, S., Boeva, V., Brage, J., & Johansson, C. "District heating substation behaviour modelling for annotating the performance". In *Cellier P., Driessens K. (eds) Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2019. Communications in Computer and Information Science*, vol 1168. Springer, Cham. DOI: 10.1007/978-3-030-43887-6_1

Paper VI Abghari,S., Boeva, V., Brage, J., & Grahn, H. "Multiview clustering analyses for district heating substations". In *2020 9th International Conference on Data Science, Technology and Applications.* (pp. 158-168). DOI: 10.5220/0009780001580168

Paper VII Abghari, S., Boeva, V., Brage, J., & Grahn, H. "Higher order mining approach for analysis of real-world datasets". The paper is an extension of Paper VIII. (Submitted for journal publication)

Other research contributions that are related to this thesis but are not included:

Abstract I Abghari, S., Boeva, V., Lavesson, Gustafsson, J., Shaikh, J., & Grahn, H. "Anomaly detection in video session data". In *2017 5th Swedish Workshop on Data Science.*

Abstract II Abghari, S., Boeva, V., Lavesson, N., Grahn, H., Ickin, S., & Gustafsson, J. "A minimum spanning tree clustering approach for mining sequence datasets". In *2018 6th Swedish Workshop on Data Science.*

Paper VIII Abghari,S., Boeva, V., Brage, J., Johansson, C., Grahn, H., & Lavesson, N. "Higher order mining for monitoring district heating substations". In *2019 6th IEEE International Conference on Data Science and Advanced Analytics* (pp. 382-391). DOI: 10.1109/DSAA.2019.00053

Paper IX Abghari,S., Boeva, V., Brage, J., Johansson, C., Grahn, H., & Lavesson, N. "Monitoring district heating substations via clustering analysis". In *2019 31st Swedish AI Society Workshop.*

Paper X Eghbalian, A., Abghari, S., Boeva, V., Basiri, F. "Multiview data mining approach for behaviour analysis of smart control valve". In *2020 19th IEEE International Conference on Machine Learning and Applications*, December 14-17, 2020, Miami, Florida (Accepted)

Acknowledgements

I would like to thank the people who have supported me to make this thesis possible. First and foremost, I would like to thank my supervisors Professor Niklas Lavesson, Professor Håkan Grahn, and Professor Veselka Boeva for their trust, patience, guidance, and valuable feedback. I appreciate the opportunities you provided for me to learn and grow both professionally and personally. Thanks to all my friends and colleagues who have supported me by having discussions and commenting on my work all these years. In particular, I would like to say thank you to my friend, Dr. Eva García-Martín, for her positive energy, supportive attitude and being someone that I can trust. Thank you Christian Nordahl and Johan Silvander for all the discussions we had that gave me hope and positive energies during times that were tough.

I would like to say thanks to the Swedish Knowledge Foundation for funding my research within the project "Scalable resource-efficient systems for big data analytics" under grant 20140032. Furthermore, I would like to extend my gratitude to Jörgen Gustafsson, the Research Manager at Ericsson Research Stockholm, and his team for their time, support and guidance. I also appreciate Christian Johansson, the CEO of NODA Intelligent Systems AB, and Jens Brage, Head of Research and Innovation at NODA, for providing resources and the opportunity for joint research collaborations.

Last but not least, I would like to say thanks to my family who have always been supportive. I would like to thank my loving girlfriend Amber Gray for her patience, understanding me and always being there.

Karlskrona, Sweden

October 2020

Contents

1 Introduction

Outlier detection is studied and applied in different fields to detect anomalous behaviours. An outlier is a data point which is significantly different from other surrounding data [1–3]. Outliers can happen due to different reasons such as human errors, fraudulent behaviour, mechanical faults, and instrument errors. Regardless of the source of the outliers, their detection can reveal system faults, frauds, and interesting patterns in the data. The detected outliers can assist domain experts in narrowing down the scope of analysis and understanding the root cause(s) of such anomalous behaviours in order to respond to them accordingly.

Almost all outlier detection techniques create a model representing the normal patterns in the data[1] to be able to detect whether a given data point is an outlier or not. There are several factors such as the nature of the input data, availability of the labeled data together with the constraints, and the requirements of the outlier detection problem at hand, that make data modelling challenging and domain specific [1].

This thesis explores data modelling of the outlier detection problem in three different application domains. Each of the studied domains have unique constraints and requirements, which demand validation with different experimental setups and scenarios. Outlier detection techniques are domain-dependent and are usually developed for certain problem formulations. However, similar domains may be able to adopt the same solution with some modifications. We initially investigate the importance of data modelling for outlier detection techniques in surveillance, fault detection, and trend

[1] There are some studies that model abnormalities of the data [4–8]. Some authors referred to this technique as novelty detection. It can be considered as a semi-supervised task since the algorithm is taught the normal class, although it learns to recognize abnormality [1, 2].

analysis. In addition, the reproducibility of the proposed approaches in similar domains are investigated.

The nature of the input data affects the choice of data model, which makes the outlier detection problem data-specific. In this thesis, two different sources of data are used. The majority of data used in our experiments is provided by companies involved in the conducted studies. Such data is referred to as *closed data*, which is not publicly available. The other source, on the other hand, relates to *open data*, which can be freely accessed, used, re-used, and shared by anyone for any purpose. Therefore, the thesis also investigates the application of open data as a complimentary source to closed data for surveillance purposes in the maritime domain.

1.1 Research Problem

This thesis focuses on studying and developing data models for outlier detection problems. Understanding and modelling the available data are the core components of any outlier detection system. The main challenge in outlier detection problems relates to the definition of normal behaviour. In general, defining a normal region that represents every possible normal behaviour is very difficult [1]. Moreover, due to unavailability of the labeled data for training and validating, outlier detection is often categorized as an unsupervised learning problem.

In order to address these challenges and limitations, we design and develop a set of hybrid approaches that use a combination of machine learning and data mining techniques. The proposed approaches are applied to real-world use cases for identifying abnormal activities, deviating behaviours, and faults.

The main research questions we aim to address in this thesis are:

RQ 1. *How accurate and valid are the results of an anomaly detection system that exploits open data as a complement to the available closed data?*

We address this research question in Chapter 7, where we investigate the potential of using open data as a freely accessible source of information for identifying anomalous behaviours in the maritime surveillance domain. Maritime authorities in each country only have overall information of maritime activities in their surveillance area. Exchanging information, between different countries is often a complicated procedure because of the diverse regulation of data protection in each land. Therefore, using data sources that are free from legislative procedures can be a good solution for providing information that belongs to the regions outside the land territory. Furthermore, all the information about maritime activities is not recorded in the authorities' databases or reported to them. On the other hand, there are numerous open data sources consisting of different websites, blogs and social networks that can be useful for observing the hidden aspects of the maritime activities.

RQ 2. *How can contextual dependencies of the data instances be used to facilitate the detection of collective outliers?*

We address this research question in Chapter 8 and 10. Chapter 8 aims to answer this research question through applying sequential pattern mining and clustering analysis for identifying anomalous patterns, in sequence data. In Chapter 10, we investigate an application of trend analysis for identifying manual changes in heating systems at the building level.

RQ 3. *How can group outlier detection be used for identifying deviating behaviour?*

We address this research question in Chapter 9, in which we use a minimum spanning tree to identify a group(s) of outlying patterns in sequence data. Additionally, in Chapter 11, we propose a higher order mining method for modelling district heating substations' behaviour with regards to different performance indicators. In Chapter 12, we propose a multi-view clustering approach for monitoring district heating substations, which includes geographical information for identifying neighbouring substations. In Chapter 13, we propose a higher order mining approach that consists of several different data analysis techniques for monitoring and identifying deviating operational behaviour of the studied phenomenon.

1.2 Outline

The remainder of this thesis is organized as follows:

Chapter 2: Background In this chapter, we explain the necessary background knowledge including variety of machine learning and data mining techniques that are used in the thesis. The section concludes with a discussion on related work performed in different domains to address the outlier detection problem.

Chapter 3: Closed Data vs. Open Data In this chapter, we provide the required terminologies and definitions regarding closed and open data sources.

Chapter 4: Research Methodology This chapter presents the research methodology that is used in this thesis to conduct the studies. In addition, it discusses the validity threats and limitations of the results.

Chapter 5: Results In this chapter, we summarize the results of the seven studies included in the thesis, which are conducted to answer the research questions.

Chapter 6: Conclusions and Future Work This chapter concludes the thesis by summarizing contributions and by discussing possible future directions.

Chapters 7 - 13 are Papers I-VII included in the thesis.

2 Background

2.1 Machine Learning

Machine learning (ML) is a branch of artificial intelligence (AI), which is born from *pattern recognition* and *computational learning theory*. The term *machine learning* is first introduced by Arthur L. Samuel [9] in his famous paper “Some studies in machine learning using the game of checkers” in 1959. He defines ML as the *“field of study that gives computers the ability to learn without being explicitly programmed”*. A more precise definition is proposed by Tom Mitchell in his book “Machine Learning” in 1997 [10]: “*A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E*”. In this definition, T refers to the class of tasks, P the measure of performance to be improved, and E the source of experiences, i.e., the training data. Thus, the primary goal in ML is to allow computer programs to learn automatically by identifying hidden patterns in data without human intervention and to adjust actions accordingly while improving their learning over time.

ML methods can be divided into two main categories based on the availability of labeled training data to a learning algorithm as follows:

- **Supervised Learning Techniques** In the presence of labeled examples, such as input where the target class is known, *supervised learning techniques* is applied. In supervised learning, the goal is to learn a general rule on how to map inputs to outputs. Typical supervised learning tasks are classification and regression. Supervised learning can be divided further into three sub-categories when labeled data is partially available, or when the learning process is restricted to special feedback. These three sub-categories are as follows:

  - *Semi-supervised learning* is used for the same applications as supervised learning. However, both labeled data (in a small portion) and unlabeled data are used for training. The main reason relates to the fact that providing labeled training data is time consuming and requires experts' knowledge which makes the process of data labeling expensive.
  - *Active learning* is focused on learning algorithms that have the ability to select their own training examples from provided datasets and ask from an oracle or a labeling source to label them.
  - *Reinforcement learning* includes learning algorithms that discover which actions can maximize the expected rewards through trial and error over a given period of time [11].

- **Unsupervised Learning Techniques** In the absence of labeled training data, a learning algorithm requires on its own to explore the input data in order to find some structures such as discovering hidden patterns [11]. Clustering analysis is an example of an unsupervised learning technique that is applied in this thesis. Clustering analysis is explained in Section 2.2

2.2 Clustering Analysis

The task of partitioning a set of objects into groups of similar objects such that the objects within each group are similar to each other but dissimilar to the objects in neighboring groups, is referred to as *clustering analysis* [11, 12]. Traditional clustering algorithms can be grouped into five categories, namely partitioning, hierarchical, model-based, density-based, and grid-based methods. Due to the limitations of the traditional clustering algorithms in handling massive amounts of data in applications such as data stream mining, incremental clustering methods are proposed. These clustering methods are capable of analysing data instances one at a time and assigning them to the existing clusters. Detailed surveys of clustering methods can be found in [13–15].

In this thesis, we mainly use the affinity propagation (AP) algorithm [16] for clustering analysis. AP is able to choose the number of clusters from the input data and is suitable for use cases with uneven cluster sizes and specific shapes. In addition, the exemplars (the representative patterns) of the built clustering model created by AP are real data points. In Paper II and Paper IV, we apply k-means [17] algorithm due to its simplicity. Both AP and k-means belong to the partitioning algorithms category.

Consensus Clustering Gionis et al. [18] propose an approach for clustering that is based on the concept of aggregation, in which a number of different clustering solutions are given for some datasets of elements. The objective is to produce a single clustering of the elements that agrees as much as possible with the given clustering solutions. Consensus clustering algorithms deal with problems similar to those treated by clustering aggregation techniques. Namely, such algorithms try to reconcile clustering information about the same data phenomenon coming from different sources [19] or different runs of the same algorithm [20]. In this thesis, we use the consensus clustering algorithm proposed in [19] in order to integrate the clustering solutions produced on the datasets collected for two consecutive time periods in Paper V and Paper VII. This is performed by considering the exemplars of the produced clustering solutions and dividing them into k groups (clusters) according to the degree of their similarity, by applying AP algorithm.

Multi-view Clustering Multi-view datasets consist of multiple data representations or views, where each view may consist of several features [21]. Multi-view learning is a semi-supervised approach with the goal to obtain a better performance by applying the relationships between different views rather than one to facilitate the difficulty of a learning problem [22–24]. Due to availability of inexpensive unlabeled data in many application domains, multi-view unsupervised learning and specifically multi-view clustering (MVC) attracts great attention [21]. The goal of multi-view clustering is to find groups of similar objects based on multiple data representations. In Paper VI, we propose an MVC approach for mining and analysing multi-view network datasets.

2.3 Data Mining

Data mining (DM) refers to the discovery of *models* for data [25] and knowledge extraction from large amounts of data [12]. A *model* can be one of several types. The most important directions in modelling are the followings [25]:

- **Statistical Modelling** The term *data mining* is first used by statisticians. Originally, DM has a *negative* meaning and refers to extracting information that is not supported by the data. Today, statisticians apply DM to construct statistical models to further study the underlying distribution of data.

- **Machine Learning** DM is often used interchangeably with ML. However, ML methods are used to understand the structure of the data. DM, instead, applies different ML algorithms to identify previously unknown patterns and extract insights from data.

- **Computational Approaches to Modelling** Such modelling is looking at DM as an algorithmic problem, i.e., the model of the data can simply explain complex queries about it. *Summarization* and *feature extraction* are two examples of such modelling. The aim of summarization is to provide a clear and approximate synopsis of the data. Regarding feature extraction, we can refer to *frequent itemsets mining* and *sequential pattern mining* that are explained in Section 2.4.

Higher Order Mining (HOM) A form of DM and a sub-field of knowledge discovery that is applied on non-primary, derived data, or patterns to provide human-consumable results is referred to as HOM [26]. Majority of the proposed (hybrid) approaches in this thesis belong to the HOM paradigm, where varieties of ML and DM techniques are combined to provide results and patterns with semantics to facilitate experts in outlier detection and fault inspection in different real-world case studies.

Minimum Spanning Tree (MST) Given an undirected graph $G = (V, E)$, a spanning tree of the graph G is a connected sub-graph with no cycles that includes all vertices. An MST of an edge-weighted graph (G, w), where $G = (V, E)$ is a graph and $w : E \to \mathbb{R}$ is a weight function, is a spanning tree that the sum of the weights of its edges is minimum among all the spanning trees. Prime's [27] and Kruskal's [28] algorithms are two examples of greedy algorithms developed for identifying such a spanning tree. The MST has direct applications in network design such as computer networks and telecommunication networks. It is also used for cluster analysis and outlier detection [29–33]. In this thesis, we apply an MST for identifying groups of outlying patterns in sequence data and district heating data.

Euclidean Minimum Spanning Tree (EMST) The general idea with an EMST is to consider k nearest neighbours of each point for building an MST rather than the entire set of edges in a complete graph [1]. The EMST of a set of n points, $P \subset R^2$, is a subset of the Delaunay Triangulation (DT) of those points. $DT(P)$ is a triangulation in which the circumcircle of every triangle is an empty circle, i.e., there is no point from P in its interior. DT has a number of interesting properties such as maximizing the smallest angle among all triangulations of P. This means, DT avoids narrow triangles, which makes it suitable for terrain modelling and surface analysis [35]. In Paper VI, we apply an EMST for identifying neighbouring substations for monitoring purposes.

2.4 Pattern Mining

Frequent Itemset Mining The application of frequent itemset mining for market-basket analysis is first introduced by Agrawal et al. [36] in 1993. The aim of such analysis is to reveal customers' shopping habits and to find out which sets of products are frequently bought together. The frequent itemset mining can be formulated as follows: let $\mathcal{I} = \{i_1, i_2, ..., i_n\}$ be a set of all items and $\mathcal{T} = \{t_1, t_2, ..., t_j, ..., t_m\}$ a transaction database, where t_j is a set of items that is bought by a customer ($t_j \subseteq \mathcal{I}$). The aim is to find those sets of items that occur frequently in most of the shopping baskets considering s, the user-specified *support threshold.*

The *support* for a *k-itemset* X, which consists of k items from $\mathcal{I}$, is the number of transactions that contain X as a subset, i.e., $_{ST}(X) = |\{t_j | X \subseteq t_j \wedge t_j \in \mathcal{T}\}|$. Note that the support of X can also be defined as the *relative support* which is the ratio of the number of transactions containing X to the total number of transactions in the database $\mathcal{T}$, i.e., $_{RelST}(X) = \frac{_{ST}(X)}{|\mathcal{T}|}$, such X is frequent if and only if its support is equal or greater than s.

Sequential Pattern Mining Originally in frequent itemset mining, the order of items in the itemsets is unimportant. The goal of market-basket analysis is to identify frequent sets of items that are bought together. However, there are some situations in which the order of items inside the itemset is important such as sequence databases. A sequence database consists

[1] The computational complexity of Kruslkal's [28] and Prim's [27] MST algorithms are $\mathcal{O}(E \log V)$ and $\mathcal{O}(E + V \log V)$, respectively, on a graph with V vertices and E edges. The EMST on the other hand, has a computational complexity of $\mathcal{O}(V \log V)$ [34].

of ordered sequences of items listed with or without a concrete notion of time [37]. Sequential pattern mining, the problem of finding interesting frequent ordered patterns, is first introduced in 1995 [38].

Let $\mathcal{I} = \{i_1, i_2, ..., i_n\}$ be a set of all items. A sequence α is defined as $\langle a_1, a_2, ..., a_j, ..., a_m \rangle$, where a_j is an itemset. Each itemset a_j represents a set of items that are happened at the same time. A sequence α is a subsequence of $\beta = \langle b_1, b_2, ..., b_n \rangle$ if and only if there exist integers $1 \leq i_1 < i_2 < ... < i_n \leq m$ and $a_1 \subseteq b_{i_1}, a_2 \subseteq b_{i_2}, ..., a_m \subseteq b_{i_m}$ [38]. Given a sequence database $\mathcal{T} = \{s_1, s_2, ..., s_n\}$, the support for α is the number of sequences in $\mathcal{T}$ that contain α as a subsequence. Consequently, α is a frequent sequential pattern if its support is equal or greater than the user-specified support threshold.

Mining frequent patterns in a large database can lead to generating a huge number of patterns that satisfy the user-specified support threshold. This is due to the fact that if a pattern is frequent, its sub-patterns are also frequent. To mitigate this problem, *closed* and *maximal* frequent pattern mining are proposed [37]. A frequent pattern α is called [37, 39]:

1. a *closed* frequent pattern in the database $\mathcal{T}$ if and only if none of its super-patterns have the same support as α,
2. a *maximal* frequent pattern in the database $\mathcal{T}$ if and only if none of its super-patterns is frequent.

Sequential Pattern Mining Algorithms Since the introduction of frequent itemset mining and the Apriori algorithm [36], several extensions of this algorithm are developed for both frequent itemset mining and sequential pattern mining. In general, there are two main categories of algorithms suitable for frequent pattern mining: 1) *Apriori-based algorithms* and 2) *Pattern-growth algorithms*. Additionally, from a frequent pattern mining point of view, a sequence database can represent the data either in a *horizontal data format* or *vertical data format* [40]. Therefore, based on these two data formats Apriori-based algorithms can be expanded to: 1) horizontal data format algorithms such as AprioriAll [38] and GSP [41] and 2) vertical data format algorithms such as SPADE [42] and SPAM [43]. Apriori-based algorithms generate large sets of candidates and repeatedly scan the database for mining sequential patterns which require a lot of memory [44]. In order to solve this problem, pattern-growth approaches, as extensions of the

FP-growth algorithm [44], are proposed. Pattern-growth algorithms such as FreeSpan [45] and PrefixSpan [46] work in a divide-and-conquer fashion and repeatedly divide the database into a set of smaller *projected databases* and mine them recursively. We use PrefixSpan for pattern mining in Paper II, Paper III, Paper V, and Paper VII.

2.5 Outlier Detection

According to Grubbs [47], an outlier is an observation that deviates significantly from other members of the sample in which it occurs. A similar definition is provided by Barnett and Lewis [48], stating that an outlier observation is the one which appears to be inconsistent with the remainder of that set of data. Hawkins [49] defines an outlier as a distinct observation that is seemed to be generated by a different mechanism. The detection of outliers requires an expert's knowledge for modelling the normal behaviours (patterns) in data.

The problem of finding patterns that are distinct from the majority of the data is called outlier detection. These distinct patterns are often referred to as outliers or anomalies[2]. Outlier detection is related but distinct of noise detection. Noise can appear as *attribute noise* (implicit errors and missing values introduced by the measurement tools), *class noise* (mislabeled instances), or a combination of both. Errors and exceptions that occur during data collection and data preprocessing phases represent noise which should be repaired or eliminated from the data [50]. Outliers, on the other hand, can be considered as interesting and/or unknown patterns hidden in data which may lead to new insights, the discovery of system faults, or fraudulent activities. While some works do not separate noise and outliers [2], others refer to noise and anomalies as weak and strong outliers, respectively [3]. Figure 2.1 illustrates the data spectrum from normal data to noise, and anomalies. In Paper V, Paper VI, and Paper VII, we apply a Hampel filter [51] which is a median absolute deviation based estimation to detect and smooth out extreme values as a result of faults in measurement tools. The filter belongs to the three-sigma rules and robust against noises.

[2] Outliers are also referred to as aberrations, contaminants, deviations, discordant observations, error, exceptions, noise, novelty, peculiarities, or surprises in different domains [1, 2].

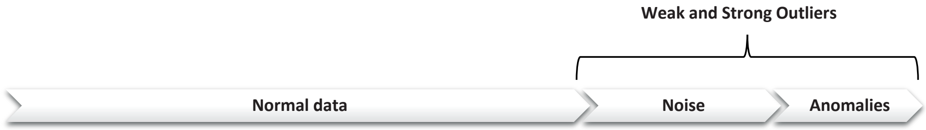


Figure 2.1: The spectrum of data from normal data to outliers (Adopted from [3].) Outlierness score from left to right is increasing.

Outliers can be classified into three categories [1, 12, 52]:

- **Point Outliers** An individual data instance that is distinct from the entirety of the dataset can be considered as a point (global) outlier. This is the simplest type of outlier.
- **Contextual Outliers** A data instance that deviates significantly with respect to a specific context or condition is referred to as a contextual or a conditional outlier. In a contextual outlier, the data instances are evaluated by considering two groups of attributes: 1) *Contextual attributes* define the context or neighborhood for an instance. For example in time series data the notion of time for each instance represents its position 2) *Behavioural attributes* define the non-contextual characteristics of the instance, such as the measured value for each data point in time series data.
- **Collective Outliers** A collection (sequence) of related data instances that deviates significantly from the entire dataset. In a collective outlier, each individual data instance in the sequence may or may not be an outlier by itself, instead their occurrence together make them special.

Outlier detection techniques can be classified into three groups based on the availability of the labeled data [1, 2]: 1) In the absence of prior knowledge of the data, unsupervised learning methods are used to determine outliers. The initial assumption is that normal data represents a significant portion of the data and is distinguishable from noise or error, 2) In the presence of labeled data, both normality and abnormality are modeled. This approach refers to supervised learning, and 3) Define what is normal and only model normality. This approach is known as semi-supervised learning since the algorithm is trained by labeled normal data, however, it is able to detect outliers or abnormalities. Semi-supervised outlier detection techniques are more widely used compared to supervised techniques due to an imbalanced number of normal and abnormal labeled data.

The output of an outlier detection is one of the following two types [1–3]: **Scores:** Outlier detection techniques that belong to this category provide a score quantifying the degree to which each data point is considered as an outlier. Such techniques create a ranked list of outliers, which assist domain experts to analyse the top few outliers or define a domain-specific threshold to select the outliers. **Labels:** This category of techniques assign a binary label to each data point indicating whether it is an outlier or not.

Fault Detection Fault is an abnormal state within the system that may cause a failure or a malfunction. Fault detection is the identification of an unacceptable deviation of at least one feature of the system from the expected or usual behaviour. The main objective of a fault detection and diagnosis (FDD) system is early detection of faults and diagnoses of their causes to reduce the maintenance cost and excessive damage to other parts of the system [53, 54].

FDD methods can fall into *model-based* methods and *data-driven* methods categories. The model-based methods require a priori knowledge of the system and can use either *quantitative* or *qualitative* models. Quantitative models are sets of mathematical relationships mainly based on physical properties, processes or models of the system. Qualitative models, on the other hand, use qualitative knowledge, e.g., including domain expert's experience as a set of rules for identifying proper and faulty operations. Data-driven methods use historical data coming from a system to build models, i.e., they are system-specific. These methods become more and more popular in the recent years. Data-driven methods are easy to develop and do not require explicit knowledge of the system, which makes them suitable for domains with uncertainty [55, 56].

2.6 Related Work

Outlier detection techniques are studied and successfully applied in different domains. There exists a considerable number of studies that provide a comprehensive and structured overview of the state-of-the-art methods and applications for outlier detection [1, 2, 57, 58]. In this thesis, we focus on three different domains, namely maritime surveillance, online media and sequence datasets, and district heating. Outlier detection techniques are domain-specific. This relates mainly to the nature of the data, availability of the labeled data, and type of outliers that should be detected. These factors are determined by the application domain and their identifications by the domain experts can lead to choosing a suitable data modelling.

Maritime Surveillance Domain

Maritime surveillance is the effective understanding of all maritime activities that can impact the security, safety, economy or environment[3]. Maritime transport handles over 80% of the volume of the global trade[4]. Along with the development of the maritime transport system, the threats to maritime security such as illegal fishing and pollution, terrorism, smuggling activities, and illegal immigration are increasing correspondingly.

In recent years, the number of studies that address the use of anomaly detection in the maritime surveillance domain is increasingly growing. The anomaly detection techniques can be divided into two groups, namely *data-driven* and *knowledge-driven* approaches. There are a couple of works that propose knowledge-based anomaly detection systems with different representation techniques and reasoning paradigms such as rule-based, description logic, and case-based reasoning [59–61]. A prototype for a rule-based expert system based on the maritime domain ontologies is developed by Edlund et al. [62]. The proposed prototype can detect some of the anomalies regarding the spatial and kinematic relation between objects such as simple scenarios for hijacking, piloting, and smuggling. Another rule-based prototype is developed by Defence R&D Canada [63, 64]. The aforementioned prototype employs various maritime situational facts about both the kinematic data and the static data in the domain to make a rule-based automated reasoning engine for finding anomalies. One of the popular data-driven anomaly detection approaches is the Bayesian network [65–67]. Johansson and Falkman [66] use the kinematic data for creating the network; however,

[3] Integrating Maritime Surveillance, common information sharing environment, `https://www.ec.europa.eu/maritimeaffairs/policy/integrated_maritime_surveillance/documents/integrating_maritime_surveillance_en.pdf`

[4] United Nations Conference on Trade and Development (UNCTAD), Review of Maritime Transport 2019, `https://unctad.org/en/PublicationsLibrary/rmt2019_en.pdf`

in the work that is performed by Fooladvandi et al. [65] expert's knowledge as well as the kinematic data are used in the detection process. Lane et al. [67] present the detection approaches for five unusual vessel behaviours, where the estimation of the overall threat is performed by using a Bayesian network. Unsupervised learning techniques are widely used for data-driven anomaly detection such as trajectory clustering [68], self organizing map [69] and fuzzy ARTMAP neural network [70]. Some statistical approaches, such as Gaussian mixture model [71], hidden Markov model [72], adaptive kernel density estimator [73] and precise/imprecise state-based anomaly detection [68] are used in this context. The majority of the works that are done in the context of anomaly detection only used transponder data from the automatic identification system (AIS).

There are a number of studies that employ data fusion techniques to fuse data from different sensors in anomaly detection systems [74–77]. In these studies, the surveillance area is restricted to the coastal regions and the combination of data from AIS, synthetic aperture radar, infra-red sensors, video, and other types of radar are used in the fusion process to obtain the vessel tracks. Furthermore, there are some other works that focus on the fusion of both sensor and non-sensor data, e.g., expert's knowledge [65, 78–82]. Riveiro and Falkman [82] introduce a normal model of vessel behaviour based on AIS data by using a self organizing map and a Gaussian mixture model. According to the model, the expert's knowledge about the common characteristic of the maritime traffic is captured as if-then rules and the anomaly detection procedure is supposed to find the deviation from the expected value in the data. Lefebvre and Helleur [80] use radar data with user's knowledge about the vessels of interests. The sensor data is modelled as track and the non-sensor data is modelled as templates. The track-template association is done by defining mathematical models for tracks and using fuzzy membership functions for association possibilities. Mano [81] proposes a prototype for the maritime surveillance system that can collect data from different types of sensors and databases and regroup them for each vessel. Sensors like AIS, high frequency surface wave radar and classical radars and databases such as environmental database, Lloyd's Insurance and TF2000 Vessel database are additionally included in this prototype. By using multi-agent technology an agent is assigned to each vessel and anomalies can be detected by employing a rule-based inference engine. When the combination of anomalies exceeded a threshold, vessel status is reported to the user as an anomaly. Ding et al. [79] propose an architecture of a centralized integrated maritime surveillance system for the Canadian coasts. Sensors and databases included in this architecture are high frequency surface wave radar, automatic dependant surveillance reports, visual reports, information sources, microwave radar, and radar sat. A common data structure is defined for storing data that are collected from different sensors. Andler et al. [78], also describe a conceptual maritime surveillance system that integrate all available information such as databases and sensor systems (AIS, long-range identification and tracking, intelligence reports, registers/databases of vessels, harbours, and crews) to help users to detect and visualize anomalies in the vessel traffic data on a worldwide scale. Furthermore, the authors suggest using open data in addition to other resources in the fusion process.

Online Media Domain and Sequence Datasets

The Internet is transformed almost every aspect of human society by enabling a wide range of applications and services such as online video streaming. Subscribers of such services spend a substantial amount of time online to watch movies and TV shows. This is required online video service providers (OVSPs) to continuously improve their services and equipment to satisfy subscribers' high expectation. According to a study that is performed by Krishnan and Sitaraman [83], a 2-second delay in starting an online video program causes the viewers to start abandoning the video. For each extra second delay beyond that, the viewers' drop-off rate will be increased by 5.8%. Thus, in order for OVSPs to address subscribers' needs it is important to monitor, detect, and resolve any issues or anomalies that can significantly affect the viewers when watching requested video programs. Analysing massive amounts of video sessions for identifying such abnormal behaviours is similar to finding a needle in a haystack.

Barbará et al. [84] propose an intrusion detection system that applies a frequent itemset technique to discover sets of items that are available in most data chunks. Using a clustering algorithm, these items that are considered as attack-free traffic, are divided into different groups based on their similarities. After creating the clusters, an outlier detection technique is applied to all the data points, checking each instance against the set of clusters. Instances that do not belong to any clusters are presumed to be attacks. Recently, Rossi et al. [85] propose an anomaly detection system for the smart grid domain similar to the one that is considered in [84]. The proposed method by Rossi et al. uses frequent itemset mining on different event types collected from smart meters to separate normal and potential anomalous data points. For further evaluation, a clustering technique with SI analysis is applied to detect anomalies.

Hoque et al. [86] develop an anomaly detection system for monitoring daily in-home activities of elderly people called *Holmes*. The proposed system learns a resident's normal behaviour by considering variability of daily activities based on their occurrence time (e.g., day, weekdays, weekends) and applying a context-aware hierarchical clustering algorithm. Moreover, *Holmes* learns temporal relationships between multiple activities with the help of both sequential pattern mining and itemset mining algorithms. New scenarios can be added based on residents' and experts' feedback to increase the accuracy of the system.

There are several clustering algorithms capable of detecting noise and eliminating it from the clustering solution such as DBSCAN [87], CRUE [88], ROCK [89], and SNN [90]. Even though such techniques can be used to detect outliers, the main aim for the clustering algorithm is to perform the partitioning task rather than identifying outliers. This leads to proposing clustering-based techniques that are capable of detecting: 1) single-point outliers such as the application of self organizing maps for clustering normal samples and identifying anomalous samples [91], and expectation maximization [92] for identifying the performance problems in distributed systems or 2) groups of outliers such as the proposed intrusion detection by [93].

The application of an MST is studied by researchers in different fields including cluster analysis and outlier detection [30–33, 94]. A two-phase clustering algorithm is introduced for detecting outliers by Jiang et al. [30]. In the first phase, a modified version of the k-means algorithm is used for partitioning the data. The modified k-means creates $k + i$ clusters, i.e., if a new data point is far enough from all clusters (k, number of clusters defined by the user), it will be considered as a new cluster (the $(k + i)^{th}$ cluster where, $i > 0$). In the second phase, an MST is built where, the tree's nodes represent the center of each cluster and edges show the distance between nodes. In order to detect outliers, the longest edge of the tree is removed. The sub-trees with a few number of clusters and/or smaller clusters are selected as outliers. Wang et al. [32] develop an outlier detection by modifying *k*-means for constructing a spanning tree similar to an MST. The longest edges of the tree are removed to form the clusters. The small clusters are regarded as potential outliers and ranked by calculating a density-based outlying factor.

A spatio-temporal outlier detection for early detection of critical events such as flood through monitoring a set of meteorological stations is introduced in [94]. Using geographical information of the data, a Delaunay triangulation network of the stations is created. The following step limits the connection of nodes to their closest neighbors while preventing far nodes from being linked directly. In the next step, an MST is constructed out of the created graph. In the final step, the outliers are detected by applying two statistical methods to detect exactly one or multiple outliers.

Districts Heating Domain

A district heating (DH) system is a centralized system with the aim of producing space heating and domestic hot water (DHW) for consumers based on their demand within a limited geographic area. A DH system consists of three main parts: 1) production units, 2) distribution network, and 3) consumers. The heat is produced at a *production unit* and circulated through a *distribution network* to reach the consumers. This part of the system is referred to as *primary* side. The consumer unit consists of a heat exchanger, a circulation network, and radiators for space heating in the rooms. This part of the system is called the *secondary* side. The provided heat and DHW produced at the primary side are transferred through a substation into the consumer unit, the secondary side. The substation makes the water temperature and pressure at the primary side suitable for the secondary side.

In the DH domain, energy companies need to address several conflicting goals such as satisfying consumers' heat demand including DHW while minimizing production and distribution costs. In addition, domain experts can consider different features and characteristics of the DH system for their analyses. Such complexity demands heat load forecasting and fault detection and root cause analysis techniques for identification of deviating behaviours and faults. Note that heat load forecasting at the building level can also be used as a fault detection.

Heat load forecasting in a long term can minimize the operational cost and pollution by considering consumers' demand and producing just the necessary amount of heat. However, modelling the heat demand forecasting is a challenging task, since water does not move fast. In some situations, the distribution of heated water can take several hours. Moreover, there are a number of factors that affect the forecast accuracy and need to be considered before any plan for production units can be constructed. Some of these factors include [95, 96]:

1. Weather condition, mainly the outdoor temperature
2. Social behaviour of the consumers
3. Irregular days such as holidays
4. Periodic changes in conditions of heat demand such as seasonal, weekly and day-night

Fumo [97] points out in his review two commonly used techniques for energy demand estimation, namely; forward (classical) and data-driven (inverse) techniques. The first approach describes the behaviour of systems by applying mathematical equations and known inputs to predict the outputs. In contrast, data-driven techniques use ML methods to learn the system's behaviour by building a model with training data in order to make predictions.

Dotzauer [96] introduces a very simple model for forecasting heat demand based on outdoor temperature and social behaviour. The predictions of the model are shown to be comparable with complicated models such as autoregressive moving average model (ARMA). The author concludes that better predictions can be achieved by improving the weather forecasts instead of developing complicated heat demand forecasting models.

In general, different ML methods and techniques are used to predict the heat demand. Some of the most popular prediction models are autoregressive moving average (ARMA) [98], support vector regression (SVR) [99, 100], multiple linear regression (MLR) [101], and artificial neural network (ANN) [102, 103]. In [100], the authors compare four supervised ML methods for building short-term forecasting models. The models are used to predict heat demand for multi-family apartment buildings with different horizon values between 1 to 24 hours ahead. The authors conclude that SVR achieves the best performance followed by MLR in comparison to feed forward neural networks (FFNN), and regression trees methods. Recently, Provatas et al. [104], propose the usage of on-line ML algorithms in combination with decision tree-based ML algorithms for heat load forecasting in a DH system. The authors investigate the impact of two different approaches for heat load aggregation. The results of the study show that the proposed algorithm has a good prediction result. In another study [105], the authors show the application of a context vector (CV) based approach for forecasting energy consumption of single family houses. The proposed method is compared with linear regression, K-nearest neighbors (KNN), and SVR methods. The results of the experiment show that CV performed better in most cases followed by KNN and SVR. The authors conclude that the proposed solution can help DH companies to improve their schedule and reduce operational costs.

There are a number of studies that focus on the application of decision support (DS) systems in domains such as DH and mainly related to advanced energy management [106–111]. In these studies, the main focus is on forecasting and optimization methods that facilitate and support the decision-making processes to increase the energy management quality and bring considerable savings. Furthermore, there are some other works that focused on DH network design [112, 113]. Bordin et al. [112] present a mathematical model to support DH system network planning by selecting an optimal set of new users to be connected to a thermal network that maximizes revenues and minimizes infrastructure and operational costs.

A DH substation involves several components, each a potential source of faults. For example, a fault can consist of a stuck valve, a fouled heat exchanger, less than optimal temperature transmitters, a poorly calibrated control system, and many more [114, 115]. Gadd and Werner [116] classify the possible faults of substations and secondary systems into three categories as follows: 1) Faults resulting in comfort problems such as insufficient heating, or physical issues such as water leakage, 2) Faults with a known cause but unsolved due to cost, and 3) Faults that require advanced fault detection techniques for their discovery, which also includes faults caused by humans, such as unsuitable settings in building operating systems. Undetected faults can lead to underlying problems, which in return can increase the maintenance cost and reduce the consumers' satisfaction.

When it comes to monitoring of a DH network, the domain experts often analyse substations individually or in a group with regard to one specific feature or a combination of features. While this provides useful information for the whole network it does not take into account the location of the substations along the distribution network and their neighbouring substations automatically. In other words, the operational behaviours of the DH substations need to be assessed jointly with surrounding substations within a limited geographical distance. Due to the nature of the data and the fact that different data representations can be used, the process of monitoring and identifying faults and deviating behaviours of the DH system and substations can be treated as a multi-view data analysis problem. Multi-view datasets consist of multiple data representations or views, where each one may contain several features [21]. Due to availability of inexpensive unlabeled data in many application domains, multi-view unsupervised learning and specifically multi-view clustering (MVC) attract great attention [21]. The goal of multi-view clustering is to find groups of similar objects based on multiple data representations. MVC algorithms are proposed based on different frameworks and approaches such as k-means variants [117–119], matrix factorization [120, 121], spectral methods [122, 123] and exemplar-based approaches [124, 125].

Bickel and Scheffers [117] propose extensions to different partitioning and agglomerative MVC algorithm. The study can probably be recognized as one of the earliest works where an extension of k-means algorithm for two-view document clustering is proposed. In another study [118], the authors develop a large-scale MVC algorithm based on k-means with a strategy for weighting views. The proposed method is based on the $\ell_{2,1}$ norm, where the ℓ_1 norm is enforced on data points to reduce the effect of outlier data and the ℓ_2 norm is applied on the features. In a recent study, Jiang et al. [119] propose an extension of k-means with a strategy for weighting both views and features. Each feature within each view is given bi-level weights to express its importance both at the feature level and the view level.

Liu et al. [120] propose an MVC algorithm based on joint non-negative matrix factorization (NMF). The developed algorithm incorporates separate matrix factorizations to achieve similar coefficient matrices and further meaningful and comparable clustering solution across all views. In a recent study, Zong et al. [121] propose an extension of NMF for MVC that is based on manifold regularization. The proposed framework maintains the locally geometrical structure of multi-view data by including consensus manifold and consensus coefficient matrix with multi-manifold regularization.

Kumar and Daumé [122] propose an MVC algorithm for two-view data by combining co-training and spectral clustering. The approach is based on learning the clustering in one view to *label* the data and modify the similarity matrix of the other view. The modification of the similarity matrices are performed using discriminative eigenvectors. Wang et al. [123] propose a variant of spectral MVC method for situations where there are disagreements between data views using Pareto optimization as a means of relaxation of the agreement assumption.

Meng et al. [124] propose an MVC algorithm based on affinity propagation (AP) for scientific journal clustering where the similarity matrices of the two views (text view and citations view) are integrated as a weighted average similarity matrix. In another study, Wang et al. [125] propose a variant of AP where an MVC model consists of two components for measuring 1) the within-view clustering quality and 2) the explicit clustering consistency across different views.

FDD is an active field of research and is studied in different application domains. Isermann [53, 54] provides a general review for FDD. Katipamula and Brambley [55, 56] conduct an extensive review in two parts on fault detection and diagnosis for building systems. Fontes and Pereira [126] propose a fault detection for gas turbines using pattern recognition in multivariate time series. In another study [127], the authors propose a general methodology for identifying faults based on time-series models and statistical process control, where faults can be identified as anomalies in the temporal residual signals obtained from the models using statistical process control charts.

In a recent review, Djenouri et al. [128] focus on the usage of machine learning for smart building applications. The authors classify the existing solutions into two main categories: 1) occupancy monitoring and 2) energy/device-centric. These categories are further divided into a number of sub-categories where the classified solutions in each group are discussed and compared.

Gadd and Werner [116] show that hourly meter readings can be used for detecting faults at DH substations. The authors identify three fault groups:

1) Low average annual temperature difference, 2) Poor substation control, and 3) Unsuitable heat load patterns. The results of the study show that low average annual temperature differences are the most important issues, and that addressing them can improve the efficiency of the DH systems. However, solving unsuitable heat load patterns is probably the easiest and the most cost-effective fault category to be considered. Sandin et al. [129] use probabilistic methods and heuristics for automated detection and ranking of faults in large-scale district energy systems. The authors study a set of methods ranging from limit-checking and basic model to more sophisticated approaches such as regression modelling and clustering analysis on hourly energy metering. Calikus et al. [130] propose an approach for automatically discovering heat load patterns in DH systems. Heat load patterns reflect yearly heat usage in an individual building and their discovery is crucial for effective DH operations and managements. The authors apply k-shape clustering [131] on smart meter data to group buildings with similar heat load profiles. Additionally, the proposed method is shown to be capable of identifying buildings with abnormal heat profiles and unsuitable control strategies.

Xue et al. [132] apply clustering analysis and association rule mining to detect faults in substations *with* and *without* return-water pressure pumps. Clustering analysis is applied in two steps to 1) partition the substations based on monthly historical heat load variations and 2) identify daily heat variation using hourly data. The result of the clustering analysis is used for feature discretization and preparation for association rule mining. The results of the study show that the proposed method can discover useful knowledge to improve the energy performance of the substations. However, for temporal knowledge discovery, advanced DM techniques are required. Capozzoli et al. [133] propose a statistical pattern recognition techniques in combination of artificial neural ensemble network and outlier detection methods to detect real abnormal energy consumption in a cluster of eight smart office buildings. The results of the study show the usefulness of the proposed approach in automatic fault detection and it ability in reducing the number of false alarms. Månsson et al. [115] propose a method based on gradient boosting regression to predict an hourly mass flow of a well performing substation using only a few number of features. The built model is tested by manipulating the well performing substation data to simulate two scenarios: communication problems and a drifting meter fault.

The model prediction performance is evaluated by calculating the hourly residual of the actual and the predicted values on the original and the faulty datasets. Additionally, cumulative sums of residuals using a rolling window that contains residuals from the last 24 hours are calculated. The results of the study show that the proposed model can be used for continued fault detection.

3 Closed Data vs. Open Data

According to the *Open Data Institute* (ODI), data is closed if it can only be accessed by its subject, owner, or holder. This includes data that is only available to an individual, a group of people, or within an organization for specific reasons. *Closed data* should not be shared either for security reasons or as it contains personal information.

The *Open Definition*[1] defines the term *open* as follows: "*Open means **anyone** can **freely access**, **use**, **modify**, **and share** for **any purpose** (subject, at most, to requirements that preserve provenance and openness)*". That is, *open data* can be freely accessed, used, re-used, and shared by anyone for any purpose. More specifically, open data has to be 1) legally open and available under an open license that permits anyone to access, use, and share it, 2) technically open and usable through no more than the cost of reproduction and in a machine-readable format.

Apart from closed data and open data, there is a third category that is referred to as *shared data*. According to ODI, shared data is available to:

- Named people or organizations (Named access)
- Specific groups who meet certain criteria (Group-based access)
- Anyone under terms and conditions that are not *open* (Public access)

Figure 3.1 shows the data spectrum provided by ODI ranges from *closed* to *shared* to *open* data.

The idea behind open data is established for a long time. Open data can be used in a variety of domains and obtained from any resource. The

[1] `https://www.opendefinition.org/`

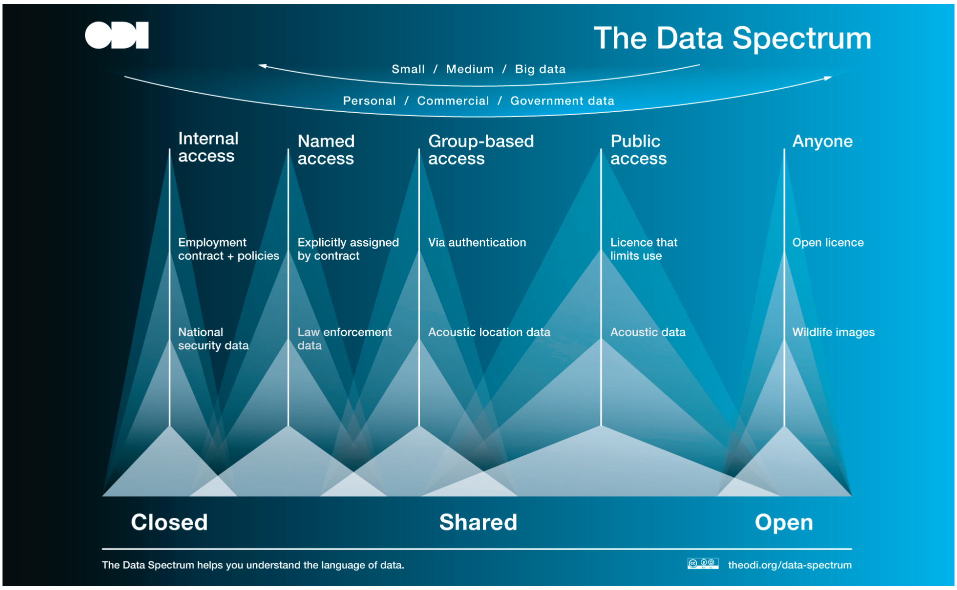


Figure 3.1: The ODI data spectrum (Reproduced from ODI.)

two major sources of open data are scientific communities and governmental sectors. One of the outcomes of the open data movement in science is the availability of large number of scientific online datasets for the public by different organizations. As well as the open data movement in science, governments for over a decade attempt to publish governmental data online and make it publicly accessible, readily available, understandable, and usable [134]. The sharing of governmental data with the public can provide openness and transparency to citizens. It can also improve the degree of participation in societal activities, the efficiency and effectiveness of the government services, and the operations within government and between different governmental organizations or sectors [135].

In Paper I, the application of open data in the maritime surveillance domain is studied. The majority of the studies that are performed in the context of anomaly detection in this domain used sensor data and mainly the automatic identification system (AIS) data to find anomalies in coastal regions. Detection of some suspicious activities such as smuggling requires vessel traffic data beyond the coastal region. Maritime authorities in each country have overall information of maritime activities in their surveillance area. However, exchanging information among different countries is a complicated procedure because of the diverse regulation of data protection in each land. Therefore, using data sources that are free from legislative procedures can be a good solution for providing information that belongs to the regions outside the land territory. Furthermore, all the information about maritime activities is not recorded in the authorities' databases or reported to them. There are numerous open data sources, on the other hand, consisting of different websites, blogs, and social networks that can be useful for observing the hidden aspects of maritime activities.

4 Research Methodology

4.1 Research Method

In this thesis, we design and perform a set of experiments to study the performance of the proposed outlier detection models. An *experiment* is a test or series of tests with some adjustable *factors*, which affect the output. Factors are divided into *controllable factors* such as the choice of the learning algorithm, and *uncontrollable factors* that add undesired effects to the process such as the noise in the data [136]. Figure 4.1 shows the construction of an outlier detection technique.

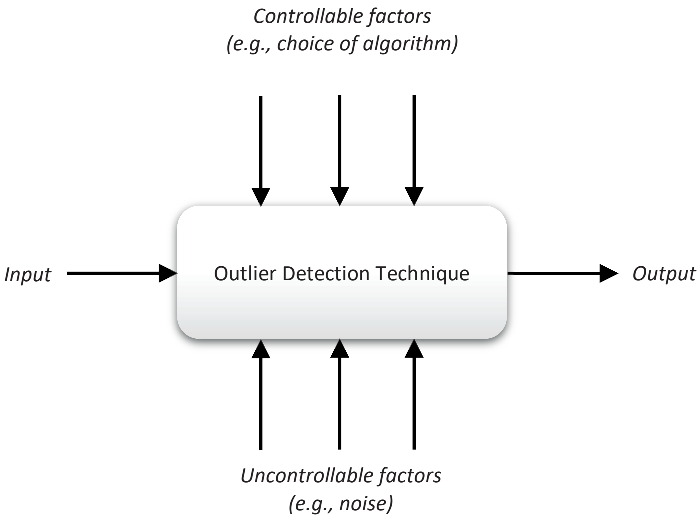


Figure 4.1: Constructing an outlier detection technique (Adopted from [136]).

In each study, we initially formulate our research questions and define the study objectives. Next, we design experimental setups in coordination with the domain experts. After that, data collection, preprocessing, and feature selection are done. The latter is performed under guidance of the domain experts to choose relevant features in model construction. We analyse and evaluate the preprocessed data to choose suitable ML and DM techniques. In this thesis, parameters of the selected algorithms are chosen by running them with different values. In some studies, we run the selected algorithms with their default parameters such as the application of support vector regression (SVR) in Paper IV. Finally, we build a model, evaluate the results, and validate our findings with the domain experts to ensure that the proposed model performs accurately. Figure 4.2 summarizes the experimental design applied in this thesis.

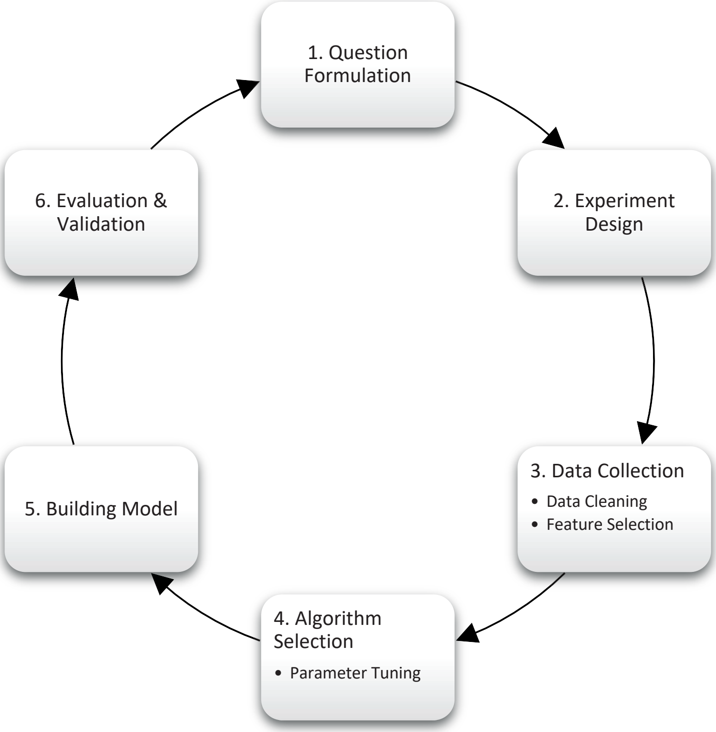


Figure 4.2: Experimental design process of the thesis.

4.2 Datasets

Open Data for Maritime Surveillance In Paper I, we use 12 publicly available datasets to perform the experiment. The collected data contains AIS reports and ports and pilot timetables of the studied surveillance area. This area is restricted to the north of the Baltic Sea and a part of the Gulf of Finland, the regional area between three European countries: Sweden, Finland, and Estonia. These datasets are listed below:

- **AIS data:** We use the data provided by *Marinetraffic* available at "http://www.marinetraffic.com/ais/", to have access to semi real-time AIS information.
- **Pilotage data:** The pilotage schedule related to Stockholm area is collected from the *Swedish Maritime Administration (Sjöfartsverket)*, available at "https://eservices.sjofartsverket.se/lotsinfopublic/lotsning_frames.asp".
- **Ports data:**
  - Ports of *Stockholm*, *Kapellskär*, and *Nynäshamn*: Information regarding vessels in the port and their expected arrival time are available at "http://www.portsofstockholm.com/vessel-calls/".
  - Port of *Norrköping*: Information regarding vessels in the port and their expected arrival time are extracted from "http://www.norrkopingshamn.se/en/ankomstinformation/fartyg-i-hamnen" and "http://www.norrkopingshamn.se/en/ankomstinformation/fartyg-i-hamnen-anlop", respectively.
  - Port of *Helsinki*: Information regarding cargo vessels in the port and their expected arrival time are collected from "https://www.portofhelsinki.fi/en/cargo-traffic-and-ships/arrivals-and-departures-cargo". Information regarding expected passenger vessels' arrival and departure times are available at "https://www.portofhelsinki.fi/en/passengers/arrivals-and-departures". Information about international cruise vessels' arrival time, departure time, and those that previously visited the port are collected at "https://www.portofhelsinki.fi/en/passengers/international-cruise-ships".

  - Port of *Tallinn*: Information regarding vessels in the port and expected cargo vessels' arrival time are available at "`https://www.ts.ee/en/ships-in-port/`" and "`https://www.ts.ee/en/ships-arriving-to-cargo-harbours/`", respectively. Information regarding expected passenger vessels' arrival and departure times are available at "`https://www.ts.ee/en/arrivals/`" and "`https://www.ts.ee/en/departures/`", respectively.

Media Data and Sequence Data In Paper II and Paper III, video session data provided by *Ericsson AB* is used. The data is stored as a transaction dataset. Each row of the dataset contains session ID, video ID, and date and time of an occurred video event together with its type. Video events can be the results of both user's actions, e.g., changing a program and system generated events such as quality related events. After preprocessing of the data, the dataset is transformed into a sequence dataset, i.e., each row represents a video session with its ID, the session's starting and ending times, it's duration, selected video programs, and a sequence of all event types that happened during the session. The data used in Paper II contains 13 unique events. In Paper III, a new data format, which includes 19 events, is used.

Apart from video session data, in Paper III, we use a publicly available dataset containing smart meter data provided by *Elektro Ljubljana*, a power distribution company in Slovenia [137]. The data is stored as a transaction dataset. The data contains device ID, event data (with a short explanation regarding each event), event ID, and event timestamps. The dataset is transformed into a sequence dataset, i.e., each row representing a daily activity of a device, time of recording the first and last event types, and a sequence of all event types that happened during its daily activity.

District Heating Data The data used in Paper IV, Paper V, Paper VI, and Paper VII consists of hourly average measurements of buildings equipped with the *NODA*[1] controller system. The controller is a retrofit smart system for maximizing energy efficiency at the building level. It consists of controlling hardware and a range of sensors, which is added on top of the existing heating system at each building. The measurements

[1] `https://www.noda.se/`

include power consumption, mass flow rate, outdoor temperature, and supply and return water temperatures at both primary and secondary sides.

4.3 Evaluation Measures

In this thesis, different evaluation measures are used for analysis and assessment of the results.

Accuracy In Paper I and Paper IV, the accuracy of the developed systems in identifying anomalous behaviours is calculated as the degree to which the measurements of a quantity describe correctly the exact value of that quantity. In other words, accuracy is the proportion of truly labeled results among the total number of cases examined. Accuracy is calculated as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \tag{4.1}$$

where true positive (TP) and true negative (TN) refer to correctly labeled cases. A false positive (FP) happens when a real negative case is misclassified as positive. A false negative (FN) occurs when a real positive case is labeled as negative [138].

Mean Absolute Error In Paper IV, mean absolute error (MAE) is used as a performance measure to evaluate the accuracy of SVR in terms of predicting the secondary supply water temperature, the water that is used for space heating inside buildings.

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |actual_i - predicted_i| \tag{4.2}$$

In Equation 4.2 the *actual* refers to the measured secondary supply water temperature by the controller system, *predicted* refers to the estimated value of the same feature by the developed system, and n is the total number of predicted instances.

Silhouette Index In Paper II, Silhouette index (SI) [139] is applied as an internal validation measure due to unavailability of the ground truth labels for finding the optimal number of clusters for k-means algorithm. SI evaluates the tightness and separation of each cluster and measures how well an object fits the available clustering. For each i, let $a(i)$ be the average dissimilarity of i to all other objects in the same cluster. Let us now consider $d(i, C)$ as an average dissimilarity of i to all objects of a cluster C. After computing $d(i, C)$ for all clusters, the one with the smallest average dissimilarity is denoted as $b(i)$. Such cluster also refer to *neighboring cluster* of i. The SI score of i, $s(i)$, is obtained by combing $a(i)$ and $b(i)$ as follows:

$$s(i) = \frac{b(i) - a(i)}{max\{a(i), b(i)\}} \tag{4.3}$$

The $s(i)$ has values in a range of $[-1, 1]$. A score close to one implies that the object is well clustered. When $s(i)$ is about zero, this indicates the object is on the decision boundary between two neighboring clusters. The worst situation occurs when $s(i)$ is close to -1. This indicates that the object is misclassified and assigned to the wrong cluster. The average $s(i)$ for all objects i belonging to the same cluster shows how tightly those objects are grouped. The average $s(i)$ for all objects i in the whole dataset judges the quality of the generated clustering solution.

4.4 Similarity Measures

In this thesis, we use different distance measures to calculate the similarity between strings, time series, and clustering solutions.

Jaro-Winkler Distance In Paper I, we use the Jaro-Winkler distance (JW) [140] to match vessels' information collected from different data sources such as AIS data and ports and pilots timetables. The JW is a variation of the *Jaro* metric [141, 142] that is commonly used for name matching in record-linkage [143]. Jaro calculates the number of common characters, c, within the half-length of the longer string and the number of transpositions, t, for two strings s_1 and s_2 as follows:

$$Jaro(s_1, s_2) = \frac{1}{3}\left(\frac{c}{|s_1|} + \frac{c}{|s_2|} + \frac{c - t}{c}\right) \tag{4.4}$$

The JW improves the Jaro metric by assigning extra weight to the common prefix at the beginning of two strings to be compared.

$$JW(s_1, s_2) = Jaro(s_1, s_2) + lp(1 - Jaro(s_1, s_2)), \tag{4.5}$$

where l is the length of common prefix at the beginning of the two strings and p is the prefix assigned weight. The standard value of p is equal to 0.1.

Levenshtein Distance In Paper II, Paper III, Paper V, and Paper VII the similarity between the extracted patterns are assessed with the Levenshtein distance (LD) [144]. The calculated similarity matrices are used for clustering analyses purposes. The LD or edit distance is defined to be the minimum number of edit operations (insertions, deletions, or substitutions) required to transform one string into another. The LD of two strings s and t is computed as follows:

$$LD(i,j) = min \begin{cases} LD(i-1,j) + deletionCost \\ LD(i,j-1) + insertionCost \\ LD(i-1,j-1) + substitutionCost, \end{cases} \tag{4.6}$$

where each operation has unit cost, except the substitution when $s_i = t_j$ that has zero cost.

Dynamic Time Warping Given two time series $Y = (y_1, y_2, ..., y_n)$ and $Y' = (y'_1, y'_2, ..., y'_m)$, the similarity between Y and Y' can be measured using the dynamic time warping (DTW) [145] algorithm. The DTW is proposed for spoken word detection with the focus of eliminating timing differences between two speech patterns. In other words, the DTW identifies an optimal matching between the given sequences by warping the time axis. In order to align the time series Y and Y' of length n and m respectively, a cost matrix, $Q_{n \times m}$ is computed. Each element, q_{ij}, of $Q_{n \times m}$ corresponds to the distance between y_i and y'_j of the two series. Using the cost matrix, the DTW tries to find the best alignment path between these two time series that is leading to minimum overall cost. The best warping path should satisfy a different number of conditions such as monotonicity, continuity, boundary, warping window, and slope constraint.

$$DTW(i,j) = Dist(i,j) + min \begin{cases} DTW(i-1,j) & \#insertion \\ DTW(i,j-1) & \#deletion \\ DTW(i-1,j-1) & \#match, \end{cases} \tag{4.7}$$

where DTW(i, j) is the distance between $Y(y_1, ..., y_i)$ and $Y'(y'_1, ..., y'_j)$ with the best alignment. $Dist(i,j)$ is the distance (often Euclidean distance) between two data points.

In Paper II, we use a FastDTW [146], an approximation of the DTW which has a linear time and space complexity to detect an optimal alignment between categorical data. We set $Dist(i,j)$ in the FastDTW to represent equality, i.e., if $i = j$, $Dist$ is equal to 0, otherwise 1. In Paper V the alignment between time series data is assessed using DTW. In both papers, the calculated similarity matrices are used for clustering analyses purposes.

Clustering Similarities In Paper VI, Paper VII the similarity between two clustering solutions is computed by considering all pairs of members. Given two clustering solutions $C = \{C_1, C_2, \ldots, C_n\}$ and $C' = \{C'_1, C'_2, \ldots, C'_m\}$ of datasets X and X', respectively the similarity, S_w, between C and C' can be assessed as follows:

$$S_w(C,C') = \frac{\sum_{i=1}^{n}(min_{j=1}^{m} w_i.d(c_i,c'_j))}{2} + \frac{\sum_{j=1}^{m}(min_{i=1}^{n} w'_j.d(c_i,c'_j))}{2}, \tag{4.8}$$

where c_i and c'_j are exemplars of the clusters C_i and C'_j, respectively. The weights w_i and w'_j indicate the relative importance of clusters C_i and C'_j compared to other clusters in the clustering solutions C and C', respectively. For example, a weight w_i of a cluster C_i can be calculated as the ratio of its cardinality to the cardinality of the dataset X, i.e., $w_i = |C_i|/|X|$. The S_w has values in a range of [0,1]. Scores of zero imply identical performance while scores close to one show significant dissimilarities.

Adjusted Rand Index The quality of the results of a clustering analysis can be validated by means of *internal* and *external* criteria. Internal criteria evaluate the quality of the clustering solution produced by a clustering algorithm that fits the data in terms of, e.g., compactness and separation by using the inherent information of the data. External criteria on the other hand, can be used for measuring the level of agreements between the results of a clustering algorithm in comparison with ground truth, the results of another clustering algorithm on the same data, or same clustering algorithm but by considering different views.

In Paper VI, we apply a symmetric external validation index, adjusted Rand index (ARI) [147], for assessing the similarity (consensus) between two clustering solutions. These clustering solutions are generated with respect to two different views on the studied DH substations. The ARI is a correction of the Rand index (RI) [148] that measures the similarity between two clustering solutions by considering the level of agreements between the two groups. ARI is computed as follows:

$$ARI = \frac{RI - ExpectedRI}{Max(RI) - ExpectedRI} \tag{4.9}$$

ARI scores are bound between -1 and +1. A score less than or equal to 0 represents random labelling and 1 stands for a perfect match.

4.5 Validity Threats

In this thesis, there are some issues that can be a threat to the validity of the results, which should be considered before applying the proposed approaches in any systems and performing the evaluation and validation.

Construct validity refers to the extent to which the results of a study reflect the theory or the concept behind [149]. The main issue that may be a threat to the construct validity is the design and reliability of the implementation. The results can be affected by the potential faults that may happen in the implementation either because of programming faults or lack of parameter tuning. For decreasing the effect of programming faults, the validity of the implemented techniques are tested different times with both real (in case of availability of the labeled data) and manipulated data that contains anomalies or outliers. There are also some parameters in different studies that are needed to be adjusted properly. In Paper I, in order to match the vessel's names, we use a string matching technique and a *similarity threshold* to determine whether the two names are identical. Regarding the other studies, we can refer to the *user-specified support threshold*, the size of time window for extracting frequent sequential patterns from the data, and the size of time window for comparing a system's behaviour. In all cases the appropriate size for such thresholds or parameters are selected through performing different sets of experiments and/or having discussions with the domain experts.

In addition, data itself can affect the final results. Real-world data, specially sensor data can contain missing values and extreme values (sudden jumps in the measured data) as the results of faults and communication issues in measurement tools. The former requires considering specific setups or conditions for applying imputation methods. The latter needs methods such as the Hampel filter to detect and smooth out extreme values. Moreover, the inaccurate nature of open data that is used in Paper I may have some effect on the results. The open data can have errors due to mistakes of the human operator. It can contain different data formats and can be unavailable for a while. The undesirable effects of the open data can be reduced through applying a common data representation format, preprocessing of the data, and using approximate string matching techniques.

The other validity threat, which can occur while performing the evaluation and validation, targets both the internal and external validities. **Internal validity** ensures that the observed relationship between the treatment and the outcome is due to a causal relationship and it is not because of an uncontrolled factor. **External validity** refers to the ability of generalizing the results of the study to other domains, times or places [149]. The threat to internal and consequently the external validity may occur if the used data in the evaluation process is biased and unrepresentative of the population. In such a situation generalizing the results of the treatments to the whole system is unrealistic. In order to prevent this issue, a proper sampling technique, ensuring a realistic representation of the studied population in both the experiment and the validation is applied.

5 Results

5.1 Open Data for Anomaly Detection

In order to approach the first research question regarding the potential of using open data and its trustworthiness in detecting anomalous behaviours for surveillance purposes, we perform a case study in the maritime domain. In maritime surveillance systems, the majority of the exploited data is obtained from confidential sources. However, there are organizations and communities that provide their maritime related data online and accessible for the public. Therefore, in Paper I, we investigate the potential of open data sources as a complementary resource for anomaly detection in this domain.

We propose a framework for anomaly detection based on the usage of open data sources along with other traditional sources of data. The proposed framework can be generalized to similar application domains due to its modularized design. Based on the proposed framework an anomaly detection system called open data anomaly detection system (ODADS) is developed. The validity of the results is investigated by the domain experts from the Swedish coastguard. The validation results on the defined scenarios (see Section 7.4 for more details) show that the majority of the system's evaluated anomalies (64.47%) are true alarms. Moreover, a potential information gap in the closed data sources is observed during the validation process. That is, for 9.21% of the evaluated anomalies, there are no corresponding data in the coastguard closed data sources.

Despite the high number of true alarms, the number of false alarms (26.32%) is also considerable. This is mainly because the used open data sources suffer from errors due to human operator mistakes, irregular data updates, data update latencies, and incompatible data formats. In order to decrease the false alarms rate in ODADS, the main solution is to integrate open and closed data, which can cover the lack of information or inaccuracy in the open data. In addition, considering a probability for the detected anomalies can decrease the number of false alarms. This can be applied by analysing the history of vessels behaviour as well as the current situation and defining a probability threshold to exclude the anomalies that have a lower score than the threshold.

Conclusion In this section, we have shown that open data has the potential to be used as a complementary resource for anomaly detection in the maritime surveillance domain. We have gathered a collection of 54 open data sources for the maritime surveillance domain. We have proposed a framework for anomaly detection to combine open data and close data sources. We have developed an anomaly detection system based on the proposed framework. The validity of the results has shown that open data can be used for anomaly detection purposes. In addition, we have observed a potential information gap in the closed data sources.

5.2 Individual Outlier Detection

We approach the second research question regarding usage of contextual dependencies of data for identifying outliers in Paper II and Paper IV. In Paper II, we apply sequential pattern mining on video session data to extract and identify sub-patterns that can be related to performance issues at the system level. In Paper IV, we look into an application of regression analysis for identifying manual changes in the heating system at the building level.

Pattern Mining The studied use case in Paper II relates to analysing a sudden increase in the number of video streaming performance events during video sessions. Performance changes in video streams are often reflected by the re-buffering and quality adaptation events. A sudden increase in occurrence of such events for many users can be related to some kind of performance issues at the system level. In order to address this problem, we develop an approach which combines frequent sequential pattern mining (see Section 2.4 for more details) and clustering analysis to detect unexpected or interesting sequences of events (collective outliers).

One of the most important steps of the proposed approach is the Data

Segmentation, where the video sessions are divided into equal-sized segments (time windows). This allows us to include a contextual information while performing the pattern extraction process. The contextual information can provide more insights for the domain experts while assessing the identified outliers.

Due to unavailability of labeled data, our initial assumption is that *high frequent sequential patterns are normal*. Therefore, the extracted patterns are divided by considering their occurrence in one segment or more than one segments into two groups, *low* and *high* frequent sequential patterns respectively. That is, the high frequent patterns are considered to be normal and used for modelling the normal behaviour using clustering analysis. In order to identify outliers, the goodness-of-fit of the low frequent patterns is evaluated against the built model using SI. This leads us to identifying sets of patterns containing repetition of quality related events and/or their combinations with events such as *paused* and *stopped*, which can be interpreted that video sessions contain these patterns are abandoned due to poor quality.

Based on the results of the analysis, we find that since the event types in the studied use case can occur in both *good* and *bad* quality sessions, it is necessary to consider different size of segments for further analysis, which increases the execution time of the purposed approach. This leads us to the idea of group outlier detection that we study in Paper III.

Regression Analysis The studied use case in Paper IV relates to identifying manual changes at the temperature of the heating system. In order to facilitate the domain experts in identifying such changes a hybrid approach is proposed. The approach is capable of predicting the required daily heat based on the outdoor temperature and analysing the energy consumption of each building. The latter is performed to detect operational status of the heating system in terms of being on or off using k-means clustering on a yearly energy consumption of each building. By performing this step, we make sure that the system is only monitored during the period of time that space heating is needed, which provides the required contextual information.

Due to the presence of noise that affects the forecast accuracy, e.g., weather condition, social behaviour, irregular days, and public holidays any potential changes are monitored for some days. That is, only those deviations that last for at least three consecutive days are marked as manual changes.

Conclusion In this section we have shown that considering contextual information of the data can facilitate domain experts in identifying outliers. We mainly have looked into detection of collective outliers using the available contextual information in video session data and DH data.

More specifically, in Paper II we have moved from the notion of point outlier detection to identifying collective outliers where each extracted pattern contains a set of events. In order to provide additional supports for the domain experts a context information such as date-time information and type-of-day are extracted during the Data Segmentation step.

In Paper IV, we have shown that manual changes in a heating system of a building can be identified with respect to the operational status of the system (contextual information) and a set of at least three daily consecutive measurements that do not conform with the heat signature of the building (collective outlier).

5.3 Group Outlier Detection

We approach the third research question regarding group outlier detection through using a minimum spanning tree in Paper III. The paper addresses the problem by proposing a hybrid approach suitable for sequence data such as video session data and smart meters data. Based on the results of this study, we look into more suitable solutions for DH data. In Paper V, we propose a HOM approach for modelling DH substation's behaviour and linking operational behaviour representative profiles with different performance indicators. In Paper VI, we include geographical location of the DH substations as contextual dependencies to provide a more precise analysis by identifying neighbouring substations. In Paper VII, we propose a HOM approach for modelling and monitoring of the behaviour modes of the studied phenomenon for a given time interval. The deviating behaviours are identified as a group of patterns by assessing every two consecutive time intervals.

Minimum Spanning Tree In Paper III and Paper VII, we apply an MST algorithm on top of created clustering models to identify groups of outlying patterns. This is performed by first building a complete weighted graph, where vertices of the graph are the exemplars of a clustering solution and edges of the graph are the dissimilarities between the exemplars. Using the MST algorithm, the aim is to determine a sub-set of edges that connect all the vertices together without any cycles and has the minimum total edge weight. In order to identify outliers, the longest edge(s) of the tree is removed which turns the tree into a forest. Following the definition of outliers that refers to patterns that happen rarely and sufficiently far away from other patterns [1], the distant and the smallest sub-trees are regarded as outliers.

Figure 5.1 (reproduced from Figure 9.1 in Paper III) shows the constructed MSTs on two datasets, smart meter (**Top**) and video session (**Bottom**), the created forest after cutting the longest edges of the MSTs, and the detected sub-trees as outliers. In Figure 5.1 (**Top-right**), sub-trees 1 and 2 are identified as outliers based on their size and distance from majority of the data. Some of the identified patterns as outliers are: {*Wrong phase sequence, Wrong phase sequence*}, {*Under voltage L2, No under voltage L2 anymore*}, and {*Communication error with FLEX meter/Measuring system access error, Meter communication OK with FLEX meter/Measurement System OK, Communication error with FLEX meter/Measuring system access error*}. Fig. 5.1 (**Bottom-right**) shows the constructed forest after cutting the longest edges of the MST, and the top 10 smallest subtrees identified as outliers. Examples of identified patterns as outliers are {*Playback.BufferingStarted, Playback.ScrubbedTo, Playback.Aborted*} and {*Playback.BitrateChanged, Playback.Resumed, Playback.Completed*}.

The results of the experiments show that the extracted patterns from the smart meter data are more comprehensible compared to the video session data. The main reason is the fact that the event types in smart meters are explicitly detailed, explaining the status of the devices. However, in video session data the event types are general, which requires more investigation and the domain experts' knowledge in order to detect video sessions with quality issues. Further analysis of the experts' comments on 18 randomly selected video sessions (9 normal and 9 abnormal) show that assessing the quality of video sessions sometimes can be subjective. As some examples, we can refer to the experts' comments concerning the following three video sessions ($\mathbf{V_1}$ to $\mathbf{V_3}$) out of 6 that are mislabeled as outliers by the proposed approach. $\mathbf{V_1}$: "Short session (15.5 sec), no buffering, good bitrate, hard to

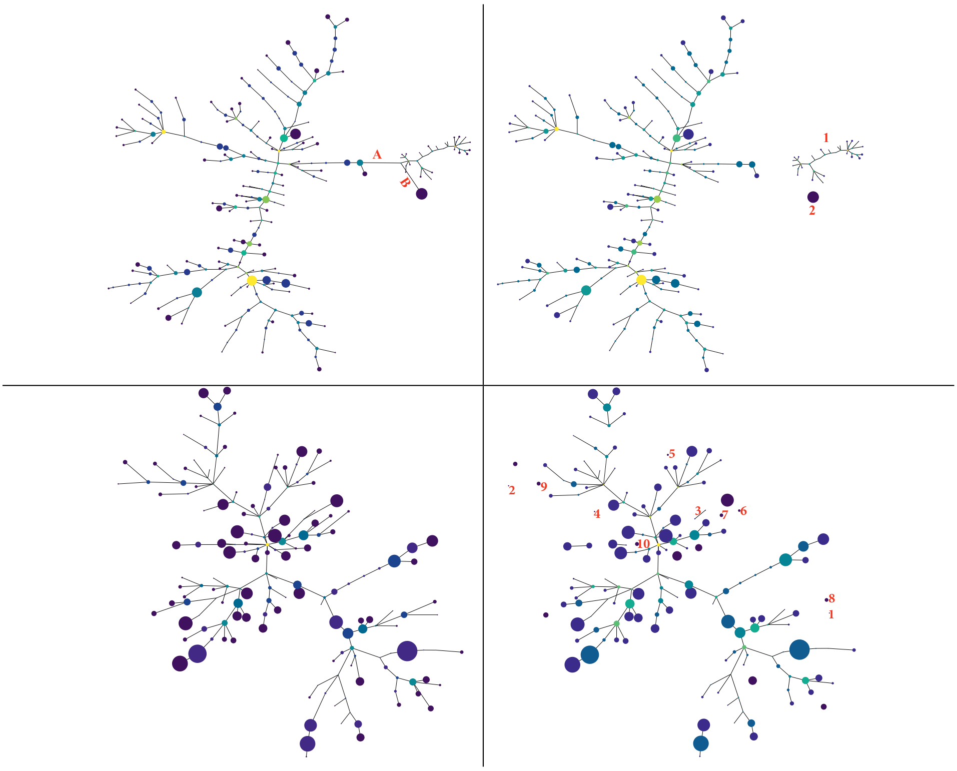


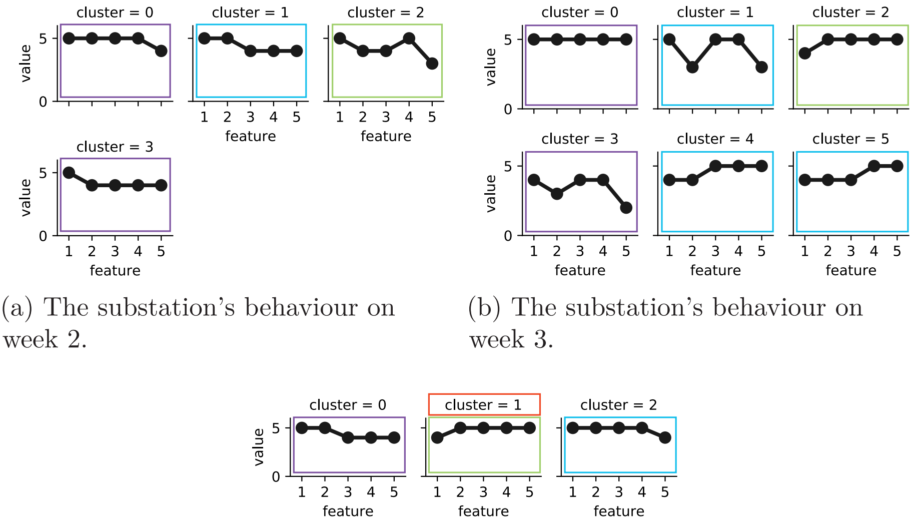
Figure 5.1: **(Top-left)** The constructed MST before removing the longest edges on smart meter sampled dataset 1. Edges A and B represent the longest edges of the tree. **(Top-right)** The transformation of the constructed MST into a forest with 3 sub-trees after the longest edges are removed. The sub-trees 1 and 2 are considered as outliers based on their size. **(Bottom-left)** The constructed MST before removing the longest edges on video session dataset. **(Bottom-right)** The transformation of the constructed MST into a forest with 22 sub-trees after the longest edges are removed. The sub-trees are ranked from smallest to largest based on their size. The top 10 smallest sub-trees are considered as outliers. The size of a node represents the number of smart meters or video sessions that are matched with it. The color of a node shows the degree of the node and is used only for the visualization purposes. The distance between edges range between [0,1].

tell, I tend to OK.” $\mathbf{V_2}$: “No buffering, rather short, 10 sec, bitrate rather good. OK.” $\mathbf{V_3}$: “Some buffering, but below 0.5% of the play duration, played 612 sec, error event at the end of the session with no further explanation, played 10 minutes with good bitrate, I tend to OK.”

In Paper VII, we use an MST for identifying deviating behaviour of the studied phenomenon by monitoring its performance for every two consecutive time intervals due to unavailability of the labeled data. This is performed through the following steps: 1) Building a clustering model for each time interval which represents the operational behaviour of the studied phenomenon for a specific time period. 2) Performing a pairwise comparison of the exemplars of the clustering solutions using Equation 4.8 for measuring, e.g., a discrepancy between observed performances in the two intervals. 3) In case of observing a significant discrepancy (above a domain-specific threshold) further analysis is preformed by integrating the produced clusters into a consensus clustering solution and building an MST on top of it to identify the smallest and distant sub-trees created by removing the longest edge(s) of the MST.

We additionally show that the assessed similarities of the clustering solutions in step (2) can be used for building the phenomenon performance signature profile for the entire studied period. Such profiles can additionally be used for comparing phenomena belonging to the same category, e.g., DH substations belonging to the same head load category such as residential buildings.

In order to provide meaningful patterns representing the behaviour of the studied phenomenon, the continuous features are converted into categorized or nominal features. Figure 5.2 (reproduced from Figure 13.3 in Paper VII) shows the application of the proposed approach for monitoring DH substations. Here, the operational behaviour of a DH substation, B_3, is shown during weeks 2 and 3 in 2017. Each pattern contains five features, where *low* is represented by one and *high* by five, respectively. Figure 5.2a represents four operational behaviour modes of the B_3 substation during week 2 of 2017. In week 3 (see Figure 5.2b) 6 operational behaviour modes are detected. We further analyse the operational behaviour models of weeks 2 and 3 by calculating the dissimilarity between the exemplars of the corresponding clustering solutions. The calculated dissimilarity is above 25% and the average weekly outdoor temperature below 10 °C. Therefore, the proposed method integrates the clustering solutions into a consensus clustering. Figure 5.2c represents the substation's operational behaviour model for the studied two weeks. The model contains 3 clusters where cluster 1 (framed in red) is detected as an outlier (considering its small size, in terms of the number of patterns that it contains, and distance compare



(a) The substation's behaviour on week 2.

(b) The substation's behaviour on week 3.

(c) The consensus clustering integrating the clustering solutions for weeks 2 and 3.

Figure 5.2: B_3 substation's operational behaviour in weeks 2 and 3 in 2017. Each cluster is shown by its exemplar. The colored frames represent the consensus clustering solution, where purple = cluster 0, green = cluster 1, and blue = cluster 2. The exemplars of clusters 0 and 2 are chosen from week 2 and cluster 1 is chosen from week 3. After building an MST on top of the consensus clustering solution, cluster 1 is identified as the deviating behaviour of the substation due to its small size and distance from majority of the data.

to the other two clusters.

Annotation of the Built Models In Paper V, we propose a HOM approach for modelling a DH substation's operational behaviour and linking it with performance indicators. At the modelling step, we use primary side features that are discretized into four categories to build the substation behavioural model by extracting patterns on a weekly basis (mainly to mitigate the social patterns and avoid discovering, e.g., the demand transition between weekdays and weekends). The extracted patterns are used to create weekly behaviour models by clustering them into groups of similar patterns. The built models are further analysed and integrated into an overall substation model for the whole heating season by applying consensus

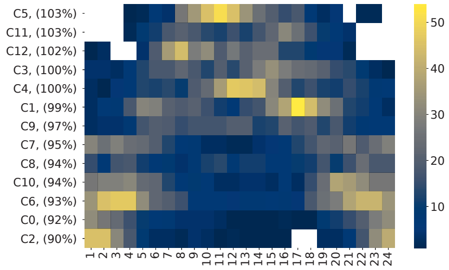


Figure 5.3: Heatmap represents profiles' frequency within 24-hour period for the whole heating season. (x-axis: 24-hour period, y-axis: profiles. Colors show the frequency of each profile.)

clustering. We consider the exemplars of the consensus clustering model as the substation representative operational behaviour profiles. Further, at the annotating step the exemplars are linked with the two performance indicators. These indicators are calculated by using features from both primary and secondary sides.

Figure 5.3 shows an hourly operational behaviour of a DH substation during the heating season. The operational behaviour is modeled with 13 profiles ($C_0 - C_{12}$), where the yellow curve represents profiles with high frequency within a 24-hour period in relation to one performance indicator. We observe that the substation performed on average 92% at early morning (0:00-5:00) and late evening (20:00-23:00). However, for the rest of the day the performance of the substation is closer to and above 100%. The low performance of the substation can be due to social behaviour, which demonstrates low heat demand in the early morning and late evening.

In addition, for each considered performance indicator, the substation's representative behaviour profiles can be summarized into three categories with respect to the associated performance indicator scores: *low*, *medium* and *high*. While low interprets as sub-optimal performance, the other two scores represent satisfactory and optimal performance, respectively.

Euclidean Minimum Spanning Tree In Paper VI, we propose a multi-view clustering analysis approach for mining network datasets with multiple representations. The proposed approach is used for monitoring a DH network and identifying substations with sub-optimal operational behaviour. We initially use geographical location of substations (contextual information) to divide them into groups of similar substations based on their distance and location. This enables us to: 1) group the substations based on their location and distance, 2) build an approximate graph representation of the DH network, and 3) order the substations using information about the DH network structure and the average supply water temperature for a specific period.

In general, the operational behaviour of the DH substations need to be assessed jointly with surrounding substations within a limited geographical distance. This leads us to the application of an EMST for terrain modelling and surface analysis. That is, the built MST using the EMST algorithm is a better fit in comparison to the built MST using the MST algorithm from a complete graph. Figure 5.4 (reproduced from Figure 12.2 in Paper VI) shows the created groups and graph network representation of 70 substations. The substations are partitioned into nine clusters. The substations with distance less than 500 meters from their closest neighbour(s) are grouped together. Clusters 0-3 and 6, each represents a tree, i.e., edges of the tree represent the distance between the substations (the tree nodes) and the remaining four clusters are singletons. The substations' colors represent their received average primary supply temperature, $T_{s,1^{st}}$, (°C) in January 2018.

We further apply two different types of analyses: 1) Step-wise clustering to sequentially consider and analyse substations with respect to a few different views, where at each step a new clustering solution is built on top of the one generated in the previous step with respect to the considered view. 2) Parallel-wise clustering to analyse substations with regards to two different views in parallel. This enables the identification of the relationships between neighbouring substations by organizing them in a bipartite graph and analysing their distribution with respect to the two considered views.

We show that the proposed approach facilitates the visual analysis and inspections of multi-view real-world datasets. For example in the investigated case study, those substations that demonstrate a deviating behaviour from their neighbouring substations can easily be identified for further investigation.

Conclusion In this section we have explored some possible ways for group outlier detection. We have first investigated the application of the MST for

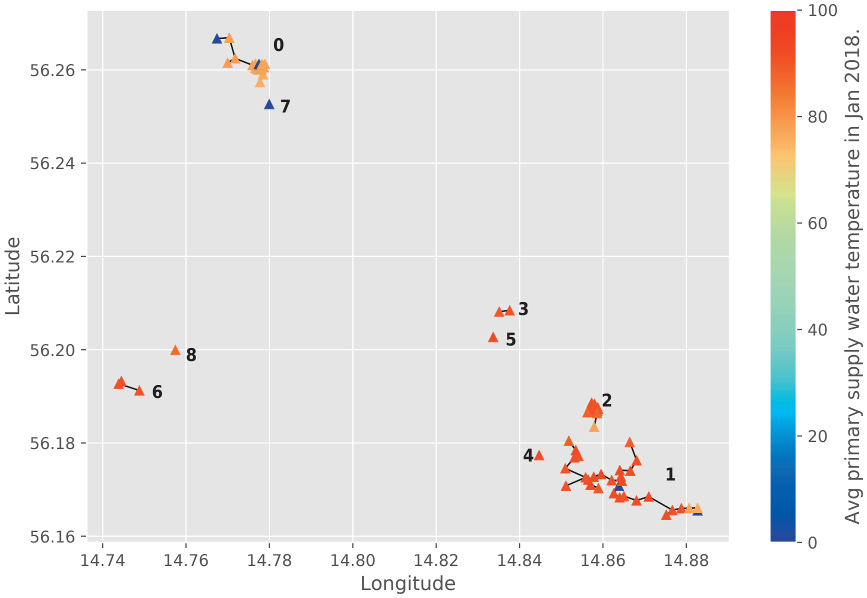


Figure 5.4: 70 substations located in Southern Sweden are grouped into nine clusters using the MST clustering algorithm. The geographical location of the substations is referred to as the Location view (v_0). Substations with distance less than 500 meters from their closest neighbours are grouped together. The color of the substations represents the average primary supply temperature, $T_{s,1^{st}}$, in January 2018, which for most substations is around 87 °C.

detecting outliers in groups. We have additionally shown that annotating the exemplars of the built clustering model with performance indicators can be used for identifying sets of patterns representing deviating behaviours. We have further shown that how the EMST can be applied for providing a more meaningful analysis using the proposed MVC approach.

5.4 Summary

The main research questions of this thesis can be answered with the help of the results explained in sections 5.1-5.3.

RQ 1. *How accurate and valid are the results of an anomaly detection system that exploits open data as a complement to the available closed data?*

We have shown in Paper I that open data has the potential to be used as a complementary resource for anomaly detection in the maritime surveillance domain. The validity of the results have shown that the majority of the detected anomalies by the developed system are true alarms (64.47%). Moreover, a potential information gap in the closed data sources has been observed during the validation process. That is, for 9.21% of the evaluated anomalies, there have been no corresponding data in the closed data sources by the coastguard. The considerable number of false alarms (26.32%) due to human operator mistakes, update latencies, etc. have been suggested the integration of open and closed data sources as the main solution.

In general, exchanging information between different countries is often a complicated procedure because of the diverse regulation of data protection in each land. Therefore, using data sources that are free from legislative procedures can be a good solution for providing information that belongs to the regions outside the land territory.

RQ 2. *How can contextual dependencies of the data instances be used to facilitate the detection of collective outliers?*

We have addressed this research question in Papers II and IV. In Paper II, we have proposed an approach based on a combination of sequential pattern mining and clustering analysis by considering context information such as date-time information and type-of-day (irregular, work, and weekend) for identifying anomalous patterns, in sequence data. In Paper IV, we have proposed an approach for identifying manual changes in the heating system at the building level. The proposed approach combines clustering and regression analyses to predict the required heat demand and identify manual changes using a domain-specific threshold.

RQ 3. *How can group outlier detection be used for identifying deviating behaviour?*

We have initially addressed this research question in Paper III, in which we have applied a minimum spanning tree to identify a group(s) of outlying patterns in sequence data. We have further looked into other solutions for identifying outliers as a group. That is, in Paper V, we have proposed a higher order mining method for modelling district heating substations' behaviour and labeling operational behaviour representative profiles with different performance indicators. In Paper VI, we have considered geographical location of the studied district heating substations to identify neighbouring substations for monitoring purposes. That is, the performance of each substation has only been compared with its neighbours, as they are expected to receive the same water temperature from the district heating network. In Paper VII, we have proposed a monitoring approach consists of several different data analysis techniques such as sequential pattern mining, clustering analysis, consensus clustering, and minimum spanning tree. The proposed approach has been used for modelling, monitoring, and identifying the deviating behaviours of the studied phenomenon for a given time interval. The deviating behaviours has been identified as a group of patterns by assessing every two consecutive time intervals.

The proposed approaches in this thesis can further be categorized from the point of view of data modelling into three categories:

1. **Rule/Signature-based Approach** In Paper I, we present an application of a rule based system for identifying abnormal behaviour in maritime surveillance.
2. **Model/Simulator-based Approach** Paper IV presents a data analysis model that learns the energy consumption of each building and predicts the expected space heating temperature.
3. **Statistical/Data-driven Approach** Papers II, III, V-VII aim to learn the hidden relationships between different feature data without having explicit knowledge of data. In Paper II, we study individual outlier detection, while in Paper III, V-VII group outlier detection is investigated.

6 Conclusions and Future Work

In this thesis, we have explored data analysing and modelling for outlier detection problem. The outlier detection techniques have been designed, developed, and evaluated in three different application domains: maritime surveillance, district heating, and online media and sequence data. The main contributions of this thesis are as follows:

- We have studied and demonstrated the application of open data for identifying anomalous behaviour in the maritime surveillance domain. This has been achieved by i) designing a framework for anomaly detection based on the integration of open data and closed data sources and ii) developing a rule-based system based on the proposed framework.

- We have proposed and studied two hybrid approaches for identifying individual outliers. More specifically, we have combined sequential pattern mining and clustering analysis for identifying outliers in online media and sequence data. In addition, we have taken advantage of clustering analysis together with regression analysis for detecting manual changes in district heating data.

- We have studied and shown how a group of outliers can be identified in two different application domains: online media and sequence and district heating. More specifically, we have proposed two approaches which exploit a minimum spanning tree for identifying groups of outliers. In addition, we have proposed a higher order data mining approach for modelling district heating substations' behaviour and annotating operational behaviour representative profiles with different performance indicators. Furthermore, we have proposed a multi-view clustering approach for mining and analysing multi-view network datasets.

- We have additionally investigated and demonstrated the reproducibility of some of the proposed models for modelling similar outlier detection scenarios in different application domains: online media and sequence data and district heating.

A possible future direction can be towards reproducibility of the proposed approaches in similar scenarios in other application domains. As it is mentioned by Chandola et al. [1], outlier detection techniques are domain-specific, however, same techniques developed in one area can be transferred into other similar domains. Therefore, we have planned to apply some of the approaches proposed in this thesis for other areas in the energy domain such as fault detection for filet of wind turbines and solar arrays. Another possible direction can be extending the proposed approaches by considering users' feedback to improve the monitoring and fault detection via data annotation. We are also interested in creating an integrated monitoring tool to apply different combinations of the proposed approaches for providing more in-depth analyses of the studied phenomena from different perspectives.

Bibliography

[1] V. Chandola, A. Banerjee, and V. Kumar. “Anomaly detection: A survey”. In: *ACM Computing Surveys* 41.3 (2009), p. 15.

[2] V. Hodge and J. Austin. “A survey of outlier detection methodologies”. In: *Artificial Intelligence Review* 22.2 (2004), pp. 85–126.

[3] C. C. Aggarwal. “Outlier analysis”. In: *Data mining.* Springer. 2015, pp. 237–263.

[4] N. Japkowicz, C. Myers, M. Gluck, et al. “A novelty detection approach to classification”. In: *IJCAI.* Vol. 1. 1995, pp. 518–523.

[5] T. Fawcett and F. Provost. “Activity monitoring: Noticing interesting changes in behavior”. In: *Proc. of the fifth ACM SIGKDD Int’l Conf. on Knowledge Discovery and Data Mining.* ACM. 1999, pp. 53–62.

[6] C. Warrender, S. Forrest, and B. Pearlmutter. “Detecting intrusions using system calls: Alternative data models”. In: *Proc. of the 1999 IEEE Symp. on Security and Privacy.* IEEE. 1999, pp. 133–145.

[7] D. Dasgupta and N. S. Majumdar. “Anomaly detection in multidimensional data using negative selection algorithm”. In: *Proc. of the 2002 Congress on Evolutionary Computation.* Vol. 2. IEEE. 2002, pp. 1039–1044.

[8] D. Dasgupta and F. Nino. “A comparison of negative and positive selection algorithms in novel pattern detection”. In: *Int’l Conf. on Systems, Man, and Cybernetics.* Vol. 1. IEEE. 2000, pp. 125–130.

[9] A. L. Samuel. “Some studies in machine learning using the game of checkers”. In: *IBM J. of Research and Development* 3.3 (1959), pp. 210–229.

[10] T. M. Mitchell et al. *Machine learning. 1997.* McGraw-Hill Education, 1997.

[11] P. Flach. *Machine learning: The art and science of algorithms that make sense of data.* Cambridge University Press, 2012.

[12] J. Han, J. Pei, and M. Kamber. *Data mining: Concepts and techniques.* Elsevier, 2011.

[13] R. Xu and D. Wunsch. "Survey of clustering algorithms". In: *IEEE Transactions on Neural Networks* 16.3 (2005), pp. 645–678.

[14] A. Fahad, N. Alshatri, Z. Tari, A. Alamri, I. Khalil, A. Y. Zomaya, S. Foufou, and A. Bouras. "A survey of clustering algorithms for big data: Taxonomy and empirical analysis". In: *IEEE Transactions on Emerging Topics in Computing* 2.3 (2014), pp. 267–279.

[15] M. Angelova. "Clustering techniques for analysis of large datasets". In: *Fundamental Sciences and Applications* (2017), p. 113.

[16] B. J. Frey and D. Dueck. "Clustering by passing messages between data points". In: *Science* 315.5814 (2007), pp. 972–976.

[17] J. MacQueen et al. "Some methods for classification and analysis of multivariate observations". In: *Proc. of the Fifth Berkeley Symp. on Mathematical Statistics and Probability.* Vol. 1. 14. Oakland, CA, USA. 1967, pp. 281–297.

[18] A. Gionis, H. Mannila, and P. Tsaparas. "Clustering Aggregation". In: *ACM Transaction of Knowledge Discovery Data* 1.1 (2007).

[19] V. Boeva, E. Tsiporkova, and E. Kostadinova. "Analysis of Multiple DNA Microarray Datasets". In: *Springer Handbook of Bio-/Neuroinformatics.* Springer Berlin Heidelberg, 2014, pp. 223–234.

[20] A. Goder and V. Filkov. "Consensus Clustering Algorithms: Comparison and Refinement". In: *ALENEX.* 2008, pp. 109–234.

[21] P. Deepak and J.-L. Anna. "Multi-View Clustering". In: *Linking and Mining Heterogeneous and Multi-view Data.* Cham: Springer Int'l Publishing, 2019, pp. 27–53. ISBN: 978-3-030-01872-6. DOI: 10.1007/978-3-030-01872-6_2. URL: https://doi.org/10.1007/978-3-030-01872-6_2.

[22] A. Blum and T. Mitchell. "Combining labeled and unlabeled data with co-training". In: *Proc. of the eleventh annual Conf. on Computational learning theory.* 1998, pp. 92–100.

[23] R. K. Ando and T. Zhang. “Two-view feature generation model for semi-supervised learning”. In: *Proc. of the 24th Int’l Conf. on Machine learning*. ACM. 2007, pp. 25–32.

[24] C. Xu, D. Tao, and C. Xu. “A survey on multi-view learning”. In: *arXiv preprint arXiv:1304.5634* (2013).

[25] J. Leskovec, A. Rajaraman, and J. D. Ullman. *Mining of massive datasets*. Cambridge University Press, 2014.

[26] J. F. Roddick, M. Spiliopoulou, D. Lister, and A. Ceglar. “Higher order mining”. In: *ACM SIGKDD Explorations Newsletter* 10.1 (2008), pp. 5–17.

[27] R. C. Prim. “Shortest connection networks and some generalizations”. In: *Bell System Technical* 36.6 (1957), pp. 1389–1401.

[28] J. B. Kruskal. “On the shortest spanning subtree of a graph and the traveling salesman problem”. In: *Proc. of the American Mathematical Society* 7.1 (1956), pp. 48–50.

[29] C. C. Aggarwal and P. S. Yu. “Outlier detection for high dimensional data”. In: *ACM Sigmod Record*. Vol. 30. 2. ACM. 2001, pp. 37–46.

[30] M.-F. Jiang, S.-S. Tseng, and C.-M. Su. “Two-phase clustering process for outliers detection”. In: *Pattern Recognition Letters* 22.6 (2001), pp. 691–700.

[31] A. C. Müller, S. Nowozin, and C. H. Lampert. “Information theoretic clustering using minimum spanning trees”. In: *Joint DAGM (German Association for Pattern Recognition) and OAGM Symp.* Springer. 2012, pp. 205–215.

[32] X. Wang, X. L. Wang, and D. M. Wilkes. “A minimum spanning tree-inspired clustering-based outlier detection technique”. In: *Ind. Conf. on Data Mining*. Springer. 2012, pp. 209–223.

[33] G.-W. Wang, C.-X. Zhang, and J. Zhuang. “Clustering with Prim’s sequential representation of minimum spanning tree”. In: *Applied Mathematics and Computation* 247 (2014), pp. 521–534.

[34] W. B. March, P. Ram, and A. G. Gray. “Fast euclidean minimum spanning tree: algorithm, analysis, and applications”. In: *Proc. of the 16th ACM SIGKDD Int’l Conf. on Knowledge discovery and data mining*. 2010, pp. 603–612.

[35] J. May. *Multivariate analysis.* Scientific e-Resources, 2018.

[36] R. Agrawal, T. Imieliński, and A. Swami. "Mining association rules between sets of items in large databases". In: *Acm sigmod record.* Vol. 22. 2. 1993, pp. 207–216.

[37] J. Han, H. Cheng, D. Xin, and X. Yan. "Frequent pattern mining: Current status and future directions". In: *Data Mining and Knowledge Discovery* 15.1 (2007), pp. 55–86.

[38] R. Agrawal and R. Srikant. "Mining sequential patterns". In: *Proc. of the 11th Int'l Conf. on Data Engineering.* IEEE. 1995, pp. 3–14.

[39] C. Borgelt. "Frequent item set mining". In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 2.6 (2012), pp. 437–456.

[40] W. Shen, J. Wang, and J. Han. "Sequential pattern mining". In: *Frequent Pattern Mining.* Springer, 2014, pp. 261–282.

[41] R. Srikant and R. Agrawal. "Mining sequential patterns: Generalizations and performance improvements". In: *Advances in Database Technology–EDBT'96* (1996), pp. 1–17.

[42] M. J. Zaki. "SPADE: An efficient algorithm for mining frequent sequences". In: *Machine Learning* 42.1 (2001), pp. 31–60.

[43] J. Ayres, J. Flannick, J. Gehrke, and T. Yiu. "Sequential pattern mining using a bitmap representation". In: *Proc. of the Eighth ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining.* 2002, pp. 429–435.

[44] J. Han, J. Pei, and Y. Yin. "Mining frequent patterns without candidate generation". In: *ACM sigmod record.* Vol. 29. 2. 2000, pp. 1–12.

[45] J. Han, J. Pei, B. Mortazavi-Asl, Q. Chen, U. Dayal, and M.-C. Hsu. "FreeSpan: Frequent pattern-projected sequential pattern mining". In: *Proc. of the Sixth ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining.* 2000, pp. 355–359.

[46] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M. Hsu. "Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth". In: *Proc. of the 17th Int'l Conf. on Data Engineering.* 2001, pp. 215–224.

[47] F. E. Grubbs. “Procedures for detecting outlying observations in samples”. In: *Technometrics* 11.1 (1969), pp. 1–21.

[48] V. Barnett and T. Lewis. *Outliers in statistical data.* Wiley, 1974.

[49] D. M. Hawkins. *Identification of outliers.* Vol. 11. Springer, 1980.

[50] B. Sluban. “Ensemble-based noise and outlier detection”. PhD Dissertation. Jožef Stefan Int’l Postgraduate School, 2014. URL: `http://slais.ijs.si/theses/2014-03-19-Sluban.pdf`.

[51] F. R. Hampel. “A general qualitative definition of robustness”. In: *The Annals of Mathematical Statistics* (1971), pp. 1887–1896.

[52] X. Song, M. Wu, C. Jermaine, and S. Ranka. “Conditional anomaly detection”. In: *IEEE Transactions on Knowledge and Data Engineering* 19.5 (2007), pp. 631–645.

[53] R. Isermann. “Supervision, fault-detection and fault-diagnosis methods—an introduction”. In: *Control engineering practice* 5.5 (1997), pp. 639–652.

[54] R. Isermann. *Fault-diagnosis systems: an introduction from fault detection to fault tolerance.* Springer Science & Business Media, 2006.

[55] S. Katipamula and M. R. Brambley. “Methods for fault detection, diagnostics, and prognostics for building systems-A review, part I”. In: *Hvac&R Research* 11.1 (2005), pp. 3–25.

[56] S. Katipamula and M. R. Brambley. “Methods for fault detection, diagnostics, and prognostics for building systems-A review, part II”. In: *Hvac&R Research* 11.2 (2005), pp. 169–187.

[57] Y. Zhang, N. Meratnia, and P. Havinga. “Outlier detection techniques for wireless sensor networks: A survey”. In: *IEEE Communications Surveys & Tutorials* 12.2 (2010), pp. 159–170.

[58] M. Gupta, J. Gao, C. C. Aggarwal, and J. Han. “Outlier detection for temporal data: A survey”. In: *IEEE Transactions on Knowledge and Data Engineering* 26.9 (2014), pp. 2250–2267.

[59] A. B. Guyard and J. Roy. “Towards case-based reasoning for maritime anomaly detection: A positioning paper”. In: *Proc. of The IASTED Int’l Conf. on Intelligent Systems and Control.* Vol. 665. 006. 2009, p. 1.

[60] M. Nilsson, J. Van Laere, T. Ziemke, and J. Edlund. “Extracting rules from expert operators to support situation awareness in maritime surveillance”. In: *The 11th Int’l Conf. on Information Fusion.* IEEE. 2008, pp. 1–8.

[61] J. Roy and M. Davenport. “Exploitation of maritime domain ontologies for anomaly detection and threat analysis”. In: *Int’l Conf. on Waterside Security.* IEEE. 2010, pp. 1–8.

[62] J. Edlund, M. Grönkvist, A. Lingvall, and E. Sviestins. “Rule-based situation assessment for sea surveillance”. In: *Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications.* Vol. 6242. Int’l Society for Optics and Photonics. 2006, p. 624203.

[63] J. Roy. “Anomaly detection in the maritime domain”. In: *Optics and Photonics in Global Homeland Security IV.* Vol. 6945. Int’l Society for Optics and Photonics. 2008, 69450W.

[64] J. Roy. “Rule-based expert system for maritime anomaly detection”. In: *Sensors, and Command, Control, Communications, and Intelligence Technologies for Homeland Security and Homeland Defense IX.* Vol. 7666. Int’l Society for Optics and Photonics. 2010, 76662N.

[65] F. Fooladvandi, C. Brax, P. Gustavsson, and M. Fredin. “Signature-based activity detection based on Bayesian networks acquired from expert knowledge”. In: *The 12th Int’l Conf. on Information Fusion.* IEEE. 2009, pp. 436–443.

[66] F. Johansson and G. Falkman. “Detection of vessel anomalies-a bayesian network approach”. In: *The Third Int’l Conf. on Intelligent Sensors, Sensor Networks and Information.* IEEE. 2007, pp. 395–400.

[67] R. O. Lane, D. A. Nevell, S. D. Hayward, and T. W. Beaney. “Maritime anomaly detection and threat assessment”. In: *The 13th Conf. on Information Fusion.* IEEE. 2010, pp. 1–8.

[68] A. Dahlbom and L. Niklasson. “Trajectory clustering for coastal surveillance”. In: *The Tenth Int’l Conf. on Information Fusion.* IEEE. 2007, pp. 1–8.

[69] M. Riveiro, G. Falkman, and T. Ziemke. “Improving maritime anomaly detection and situation awareness through interactive visualization”. In: *The 11th Int’l Conf. on Information Fusion.* IEEE. 2008, pp. 1–8.

[70] B. J. Rhodes, N. A. Bomberger, M. Seibert, and A. M. Waxman. “Maritime situation monitoring and awareness using learning mechanisms”. In: *Conf. on Military Communications*. IEEE. 2005, pp. 646–652.

[71] R. Laxhammar. “Anomaly detection for sea surveillance”. In: *The 11th Int’l Conf. on Information Fusion*. IEEE. 2008, pp. 1–8.

[72] M. Andersson and R. Johansson. “Multiple sensor fusion for effective abnormal behaviour detection in counter-piracy operations”. In: *Int’l Conf. on Waterside Security*. IEEE. 2010, pp. 1–7.

[73] B. Ristic, B. La Scala, M. Morelande, and N. Gordon. “Statistical analysis of motion patterns in AIS data: Anomaly detection and motion prediction”. In: *The 11th Int’l Conf. on Information Fusion*. IEEE. 2008, pp. 1–7.

[74] C. Carthel, S. Coraluppi, and P. Grignan. “Multisensor tracking and fusion for maritime surveillance”. In: *The Tenth Int’l Conf. on Information Fusion*. IEEE. 2007, pp. 1–6.

[75] M. Guerriero, P. Willett, S. Coraluppi, and C. Carthel. “Radar/AIS data fusion and SAR tasking for maritime surveillance”. In: *The 11th Int’l Conf. on Information Fusion*. IEEE. 2008, pp. 1–5.

[76] B. J. Rhodes, N. A. Bomberger, M. Seibert, and A. M. Waxman. “SeeCoast: Automated port scene understanding facilitated by normalcy learning”. In: *Conf. on Military Communications*. IEEE. 2006, pp. 1–7.

[77] M. Vespe, M. Sciotti, F. Burro, G. Battistello, and S. Sorge. “Maritime multi-sensor data association based on geographic and navigational knowledge”. In: *Radar Conf.* IEEE. 2008, pp. 1–6.

[78] S. F. Andler, M. Fredin, P. M. Gustavsson, J. van Laere, M. Nilsson, and P. Svenson. “SMARTracIn: A concept for spoof resistant tracking of vessels and detection of adverse intentions”. In: *Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense VIII*. Vol. 7305. Int’l Society for Optics and Photonics. 2009, 73050G.

[79] Z. Ding, G. Kannappan, K. Benameur, T. Kirubarajan, and M. Farooq. “Wide area integrated maritime surveillance: An updated architecture with data fusion”. In: *Proc. of the Sixth Int’l Conf. on Information Fusion*. Vol. 2. 2003, pp. 1324–1333.

[80] E. Lefebvre and C. Helleur. “Automated association of track information from sensor sources with non-sensor information in the context of maritime surveillance”. In: *Proc. of the Seventh Int’l Conf on Information Fusion*. Citeseer. 2004.

[81] J.-P. Mano, J.-P. Georgé, and M.-P. Gleizes. “Adaptive multi-agent system for multi-sensor maritime surveillance”. In: *Advances in Practical Applications of Agents and Multiagent Systems*. Springer, 2010, pp. 285–290.

[82] M. Riveiro and G. Falkman. “Interactive visualization of normal behavioral models and expert rules for maritime anomaly detection”. In: *The Sixth Int’l Conf. on Computer Graphics, Imaging and Visualization*. IEEE. 2009, pp. 459–466.

[83] S. S. Krishnan and R. K. Sitaraman. “Video stream quality impacts viewer behavior: Inferring causality using quasi-experimental designs”. In: *IEEE/ACM Transactions on Networking* 21.6 (2013), pp. 2001–2014.

[84] D. Barbará, Y. Li, J. Couto, J.-L. Lin, and S. Jajodia. “Bootstrapping a data mining intrusion detection system”. In: *Proc. of the 2003 ACM Symp. on Applied computing*. 2003, pp. 421–425.

[85] B. Rossi, S. Chren, B. Buhnova, and T. Pitner. “Anomaly detection in Smart Grid data: An experience report”. In: *Int’l Conf. on Systems, Man, and Cybernetics*. IEEE. 2016, pp. 002313–002318.

[86] E. Hoque, R. F. Dickerson, S. M. Preum, M. Hanson, A. Barth, and J. A. Stankovic. “Holmes: A comprehensive anomaly detection system for daily in-home activities”. In: *Int’l Conf. on Distributed Computing in Sensor Systems*. IEEE. 2015, pp. 40–51.

[87] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, et al. “A density-based algorithm for discovering clusters in large spatial databases with noise”. In: *KDD*. Vol. 96. 34. 1996, pp. 226–231.

[88] S. Guha, R. Rastogi, and K. Shim. “CURE: An efficient clustering algorithm for large databases”. In: *ACM Sigmod Record*. Vol. 27. 2. ACM. 1998, pp. 73–84.

[89] S. Guha, R. Rastogi, and K. Shim. “ROCK: A robust clustering algorithm for categorical attributes”. In: *Proc. of the 15th Int’l Conf. on Data Engineering*. IEEE. 1999, pp. 512–521.

[90] L. Ertöz, M. Steinbach, and V. Kumar. “Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data”. In: *Proc. of the 2003 SIAM Int’l Conf. on Data Mining*. SIAM. 2003, pp. 47–58.

[91] F. A. González and D. Dasgupta. “Anomaly detection using real-valued negative selection”. In: *Genetic Programming and Evolvable Machines* 4.4 (2003), pp. 383–403.

[92] X. Pan, J. Tan, S. Kavulya, R. Gandhi, and P. Narasimhan. “Ganesha: Blackbox diagnosis of mapreduce systems”. In: *ACM SIGMETRICS Performance Evaluation Review* 37.3 (2010), pp. 8–13.

[93] E. Eskin, A. Arnold, M. Prerau, L. Portnoy, and S. Stolfo. “A geometric framework for unsupervised anomaly detection”. In: *Applications of data mining in computer security*. Springer, 2002, pp. 77–101.

[94] E. Cipolla, U. Maniscalco, R. Rizzo, D. Stabile, and F. Vella. “Analysis and visualization of meteorological emergencies”. In: *Ambient Intelligence and Humanized Computing* 8.1 (2017), pp. 57–68.

[95] N. Eriksson. *Predicting demand in districtheating systems: A neural network approach*. 2012.

[96] E. Dotzauer. “Simple model for prediction of loads in district-heating systems”. In: *Applied Energy* 73.3-4 (2002), pp. 277–284.

[97] N. Fumo. “A review on the basics of building energy estimation”. In: *Renewable and Sustainable Energy Reviews* 31 (2014), pp. 53–60.

[98] H. Wiklund. “Short term forecasting on the heat load in a DH-system”. In: *Fernwärme Int’l* 20.5-6 (1991), pp. 286–294.

[99] L. Wu, G. Kaiser, D. Solomon, R. Winter, A. Boulanger, and R. Anderson. “Improving efficiency and reliability of building systems using machine learning and automated online evaluation”. In: *The 11th Conf. on Systems, Applications and Technology*. IEEE. 2012, pp. 1–6.

[100] S. Idowu, S. Saguna, C. Åhlund, and O. Schelén. "Forecasting heat load for smart district heating systems: A machine learning approach". In: *Int'l Conf. on Smart Grid Communications*. IEEE. 2014, pp. 554–559.

[101] T. Catalina, V. Iordache, and B. Caracaleanu. "Multiple regression model for fast prediction of the heating energy demand". In: *Energy and Buildings* 57 (2013), pp. 302–312.

[102] K. Kato, M. Sakawa, K. Ishimaru, S. Ushiro, and T. Shibano. "Heat load prediction through recurrent neural network in district heating and cooling systems". In: *Int'l Conf. on Systems, Man and Cybernetics*. IEEE. 2008, pp. 1401–1406.

[103] M. Sakawa, K. Kato, and S. Ushiro. "Cooling load prediction in a district heating and cooling system through simplified robust filter and multilayered neural network". In: *Applied Artificial Intelligence* 15.7 (2001), pp. 633–643.

[104] S. Provatas. *An online machine learning algorithm for heat load forecasting in district heating systems*. 2014.

[105] S. Rongali, A. R. Choudhury, V. Chandan, and V. Arya. "A context vector regression based approach for demand forecasting in district heating networks". In: *Int'l Conf. on Innovative Smart Grid Technologies Asia*. IEEE. 2015, pp. 1–6.

[106] K. Mařík, Z. Schindler, and P. Stluka. "Decision support tools for advanced energy management". In: *Energy* 33.6 (2008), pp. 858–873.

[107] D. Chinese and A. Meneghetti. "Optimisation models for decision support in the development of biomass-based industrial district-heating networks in Italy". In: *Applied Energy* 82.3 (2005), pp. 228–254.

[108] P. Bardouille and J. Koubsky. "Incorporating sustainable development considerations into energy sector decision-making: Malmö Flintränen district heating facility case study". In: *Energy Policy* 28.10 (2000), pp. 689–711.

[109] S. N. Petrovic and K. B. Karlsson. "Danish heat atlas as a support tool for energy system models". In: *Energy Conversion and Management* 87 (2014), pp. 1063–1076.

[110] A. Meneghetti and G. Nardin. “Enabling industrial symbiosis by a facilities management optimization approach”. In: *J. of Cleaner Production* 35 (2012), pp. 263–273.

[111] E. Brembilla and A. Sciomachen. *Design and verification of a large size district heating network by a DSS.* 1990.

[112] C. Bordin, A. Gordini, and D. Vigo. “An optimization approach for district heating strategic network design”. In: *European J. of Operational Research* 252.1 (2016), pp. 296–307.

[113] A. Sciomachen and R. Sozzi. “The algorithmic structure of a decision support system for a design of a district heating network”. In: *Computers & Operations Research* 17.2 (1990), pp. 221–230.

[114] S. Frederiksen and S. Werner. *District Heating and Cooling.* Studentlitteratur AB, 2013. ISBN: 9789144085302.

[115] S. Månsson, P.-O. J. Kallioniemi, K. Sernhed, and M. Thern. “A machine learning approach to fault detection in district heating substations”. In: *Energy Procedia* 149 (2018), pp. 226–235.

[116] H. Gadd and S. Werner. “Fault detection in district heating substations”. In: *Applied Energy* 157 (2015), pp. 51–59.

[117] S. Bickel and T. Scheffer. “Multi-view clustering.” In: *ICDM.* Vol. 4. 2004, pp. 19–26.

[118] X. Cai, F. Nie, and H. Huang. “Multi-view k-means clustering on big data”. In: *Twenty-Third Int’l Joint Conf. on artificial intelligence.* 2013.

[119] B. Jiang, F. Qiu, and L. Wang. “Multi-view clustering via simultaneous weighting on views and features”. In: *Applied Soft Computing* 47 (2016), pp. 304–315.

[120] J. Liu, C. Wang, J. Gao, and J. Han. “Multi-view clustering via joint nonnegative matrix factorization”. In: *Proc. of the 2013 SIAM Int’l Conf. on Data Mining.* SIAM. 2013, pp. 252–260.

[121] L. Zong, X. Zhang, L. Zhao, H. Yu, and Q. Zhao. “Multi-view clustering via multi-manifold regularized non-negative matrix factorization”. In: *Neural Networks* 88 (2017), pp. 74–89.

[122] A. Kumar and H. Daumé. “A co-training approach for multi-view spectral clustering”. In: *Proc. of the 28th Int’l Conf. on machine learning (ICML-11).* 2011, pp. 393–400.

[123] X. Wang, B. Qian, J. Ye, and I. Davidson. “Multi-objective multi-view spectral clustering via pareto optimization”. In: *Proc. of the 2013 SIAM Int’l Conf. on Data Mining*. SIAM. 2013, pp. 234–242.

[124] X. Meng, X. Liu, Y. Tong, W. Glänzel, and S. Tan. “Multi-view clustering with exemplars for scientific mapping”. In: *Scientometrics* 105.3 (2015), pp. 1527–1552.

[125] C.-D. Wang, J.-H. Lai, and S. Y. Philip. “Multi-view clustering based on belief propagation”. In: *IEEE Transactions on Knowledge and Data Engineering* 28.4 (2015), pp. 1007–1021.

[126] C. H. Fontes and O. Pereira. “Pattern recognition in multivariate time series–A case study applied to fault detection in a gas turbine”. In: *Engineering Applications of Artificial Intelligence* 49 (2016), pp. 10–18.

[127] A. Sánchez-Fernández, F. Baldán, G. Sainz-Palmero, J. Benıtez, and M. Fuente. “Fault detection based on time series modeling and multivariate statistical process control”. In: *Chemometrics and Intelligent Laboratory Systems* 182 (2018), pp. 57–69.

[128] D. Djenouri, R. Laidi, Y. Djenouri, and I. Balasingham. “Machine Learning for Smart Building Applications: Review and Taxonomy”. In: *ACM Computing Surveys* 52.2 (2019), p. 24.

[129] F. Sandin, J. Gustafsson, and J. Delsing. *Fault detection with hourly district energy data: Probabilistic methods and heuristics for automated detection and ranking of anomalies.* Svensk Fjärrvärme, 2013.

[130] E. Calikus, S. Nowaczyk, A. Sant’Anna, H. Gadd, and S. Werner. “A Data-Driven Approach for Discovery of Heat Load Patterns in District Heating”. In: *arXiv preprint arXiv:1901.04863* (2019).

[131] J. Paparrizos and L. Gravano. “k-shape: Efficient and accurate clustering of time series”. In: *Proc. of the 2015 ACM SIGMOD Int’l Conf. on Management of Data.* ACM. 2015, pp. 1855–1870.

[132] P. Xue, Z. Zhou, X. Fang, X. Chen, L. Liu, Y. Liu, and J. Liu. “Fault detection and operation optimization in district heating substations based on data mining techniques”. In: *Applied Energy* 205 (2017), pp. 926–940.

[133] A. Capozzoli, F. Lauro, and I. Khan. “Fault detection analysis using data mining techniques for a cluster of smart office buildings”. In: *Expert Systems with Applications* 42.9 (2015), pp. 4324–4338.

[134] J. Alonso, O. Ambur, M. A. Amutio, O. Azañón, D. Bennett, R. Flagg, D. McAllister, K. Novak, S. Rush, and J. Sheridan. “Improving access to government through better use of the web”. In: *World Wide Web Consortium* (2009).

[135] D. Dietrich, J. Gray, T. McNamara, A. Poikola, R. Pollock, J. Tait, and T. Zijlstra. *Open Data Handbook Open Knowledge Foundation Logo*. 2009. URL: `www.opendatahandbook.org/en/`.

[136] D. C. Montgomery. *Design and analysis of experiments*. John wiley & sons, 2000.

[137] Elektro Ljubljana. *Smart meters recorded events dataset*. 2018. URL: `https://data.edincubator.eu/organization/elektro-ljubljana-podjetje-zadistribucijo-elektricne-energije-d-d`.

[138] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2016.

[139] P. J. Rousseeuw. “Silhouettes: A graphical aid to the interpretation and validation of cluster analysis”. In: *J. of Computational and Applied Mathematics* 20 (1987), pp. 53–65.

[140] W. E. Winkler. “The state of record linkage and current research problems”. In: *Statistical Research Division, US Census Bureau*. Citeseer. 1999.

[141] M. A. Jaro. “Advances in record-linkage methodology as applied to matching the 1985 census of Tampa, Florida”. In: *J. of the American Statistical Association* 84.406 (1989), pp. 414–420.

[142] M. A. Jaro. “Probabilistic linkage of large public health data files”. In: *Statistics in Medicine* 14.5-7 (1995), pp. 491–498.

[143] W. E. Winkler. “Overview of record linkage and current research directions”. In: *Bureau of the Census*. Citeseer. 2006.

[144] V. I. Levenshtein. “Binary codes capable of correcting deletions, insertions, and reversals”. In: *Soviet physics doklady*. Vol. 10. 8. 1966, pp. 707–710.

[145] H. Sakoe and S. Chiba. “Dynamic programming algorithm optimization for spoken word recognition”. In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 26.1 (1978), pp. 43–49.

[146] S. Salvador and P. Chan. “Toward accurate dynamic time warping in linear time and space”. In: *Intelligent Data Analysis* 11.5 (2007), pp. 561–580.

[147] L. Hubert and P. Arabie. “Comparing partitions”. In: *J. of classification* 2.1 (1985), pp. 193–218.

[148] W. M. Rand. “Objective criteria for the evaluation of clustering methods”. In: *J. of the American Statistical association* 66.336 (1971), pp. 846–850.

[149] T. D. Cook, D. T. Campbell, and W. Shadish. *Experimental and quasi-experimental designs for generalized causal inference.* Houghton Mifflin Boston, 2002.

7

Open data for anomaly detection in maritime surveillance

Samira Kazemi, Shahrooz Abghari, Niklas Lavesson, Henric Johnson, Peter Ryman

In: Expert Systems with Applications (40)14 (2013): pp. 5719-5729.

Abstract

Maritime surveillance has received increased attention from a civilian perspective in recent years. Anomaly detection is one of many techniques available for improving the safety and security in this domain. Maritime authorities use confidential data sources for monitoring the maritime activities; however, a paradigm shift on the Internet has created new open sources of data. We investigate the potential of using open data as a complementary resource for anomaly detection in maritime surveillance. We present and evaluate a decision support system based on open data and expert rules for this purpose. We conduct a case study in which experts from the Swedish coastguard participate to conduct a real-world validation of the system. We conclude that the exploitation of open data as a complementary resource is feasible since our results indicate improvements in the efficiency and effectiveness of the existing surveillance systems by increasing the accuracy and covering unseen aspects of maritime activities.

7.1 Introduction

Maritime surveillance is the effective understanding of all maritime activities that could impact the security, safety, economy or environment[1]. Maritime

[1] Integrating Maritime Surveillance, common information sharing environment (cise), `www.ec.europa.eu/maritimeaffairs/policy/integrated_maritime_surveillance/documents/integrating_maritime_surveillance_en.pdf`

transport handles over 80% of the volume of global trade[2]. Along with the development of the maritime transport system, the threats to maritime security such as illegal fishing and pollution, terrorism, smuggling activities and illegal immigration are increasing correspondingly. According to the Department of Homeland Security[3], anomaly detection is one of several techniques available for improving the safety and security in the maritime domain. Furthermore, an efficient maritime surveillance system requires a complete recognized maritime picture, which can be defined as a composite picture of maritime activities over an area of interest [1]. For national maritime sovereignty, this picture should include all activities within the 200 nautical miles wide exclusive economic zone. However, for some purposes such as the detection of illegal vessel transits, the recognized maritime picture could extend beyond this region [2]. Using today's technology, continuous tracking of all maritime activities by a single sensor is insufficient since it cannot monitor everything that happens in the surveillance area. On the other hand, there are large amounts of data in the maritime domain that are gathered from a variety of sensors, databases and information systems. Therefore, by taking advantage of all the available data sources it would be possible to obtain a complete recognized maritime picture. The maritime surveillance systems generally use closed data sources that belong to the surveillance area of each country and are obtained from a variety of sensors and databases that are only accessible by the national authorities (see Section 7.2 For detecting some of the anomalous activities such as smuggling, the maritime data beyond the surveillance area of each country are required. In order to assure security, maritime organizations in different countries need to exchange their privileged data and for this purpose they should deal with the diverse regulations of the data protection in each land. Exchanging data among countries is difficult, time-consuming and in some cases impossible because of the legislative issues. Moreover, there are activities that are neither reported to the maritime organizations, nor recorded in their data sources but these activities can be useful for surveillance purposes. The publicly accessible and reusable data that are free from the legislative issues are referred to as open data. Some of the open data sources may help in revealing previously unknown aspects of

[2] United Nations Conference on Trade and Development (UNCTAD), Review of Maritime Transport 2011, www.unctad.org/en/Docs/rmt2011_en.pdf

[3] National plan to achieve maritime domain awareness for the national strategy for maritime security, www.dhs.gov/xlibrary/assets/HSPD_MDAPlan.pdf

maritime activities. For example, there are different organizations such as ports that publish their vessel traffic data or their facility information online. In addition to the organizations, there are different online communities such as blogs, forums and social networks which provide the possibility of sharing information about maritime events. By exploiting the open data along with other confidential sources of data in the detection process, the anomaly detection can be done more wisely and the results can have more facts of interests for the maritime experts.

7.1.1 Contribution

This article contributes with a deeper understanding of open data as a complementary resource for establishing maritime surveillance operations. It provides a framework for anomaly detection based on the integration of open and closed data sources in the maritime surveillance domain. According to the framework, an anomaly detection system is developed which employs suitable algorithms to implement expert rules for detecting anomalies. Finally, this article contributes with a real-world validation of the developed anomaly detection system. The validation was performed by officers from the Swedish coastguard.

7.1.2 Outline

The remainder of this work is organized as follows: Section 7.2 reviews the background and related work regarding the open data and anomaly detection in the maritime surveillance domain. Section 7.3 and Section 7.4 present the identified open data sources and describe the case study. The framework design and implementation described in Section 7.5 and Section 7.6. Section 7.7 presents the system verification results and the validation results are shown in Section 7.8. Section 7.9 features a detailed discussion about the obtained results. Finally, Section 7.10 concludes the research with a discussion on the possible directions for future work.

7.2 Background

The idea behind open data has been established for a long time. Open data can be used in a variety of domains and can be obtained from any resource. The two major sources of open data are the open data in science and the open data in government. The longstanding concept of open data in science tries to overcome the difficulties in the current system of scientific publishing such as the inability to access data or usage limitation that is applied by the publishers or data providers [3]. Different groups, individuals and organizations are gathered to participate in a movement toward reforming the process of scientific publication [3]. One of the outcomes of the open data movement in science is the online availability of large number of scientific datasets for the public by different organizations. As well as the open data movement in science, governments for over a decade attempt to publish government data online and make them publicly accessible, readily available, understandable and usable [4]. The sharing of government data with the public can provide openness and transparency to citizens. It can also improve the degree of participation in the society activities and the efficiency and effectiveness of the government services and the operations within and between the governments [5].

According to one estimation [6], of all information is open source, 9% is grey information (such as preprints of scientific articles, rumours in business circles, project proposals submitted to a research-funding agency, discussions with well-informed specialists, etc.), 0.9% is secret and 0.1% is non-existent information (i.e. the information you have, but you are not aware of it). Considering the large ratio of the open data sources, there should be a great value in using them in different domains. In the maritime surveillance systems, the majority of the exploited data are obtained from the confidential sources. However, in recent years the new concept of the Web, which takes the network as a platform for information sharing, interoperability and collaboration, has created new sources of data for maritime surveillance. There are organizations and communities that provide their maritime related data online and make them accessible for the public. Therefore, it would be beneficial for the maritime surveillance systems if they can take advantage of the open data to increase the safety and security in their surveillance area.

7.2.1 Terminology

Anomaly detection is widely used in the areas such as video surveillance, network security and military surveillance. Chandola, Banerjee and Kumar [7] define AD as:*The problem of finding patterns in data that do not conform to expected behavior.*

Depending on the domain of study, the non-conforming patterns are called by different names such as anomalies, outliers, exceptions, etc. In the maritime surveillance domain, these non-conforming patterns are referred as anomalies. Defense research and development Canada [8] provides the following definition for the term anomaly in the context of the maritime surveillance domain: *Something peculiar (odd, curious, weird, bizarre, atypical) because it is inconsistent with or deviating from what is usual, normal, or expected, or because it is not conforming to rules, laws or customs.*

The term *open data* refers to the idea of making data freely available to use, reuse or redistribute without any restriction. The open data movement follows the other open movements such as *open access* and *open source*. According to the Open Knowledge Foundation[4], a community based organization that promotes open knowledge (whether it is content, data or information-based), an open work should be available as a whole, with a reasonable reproduction cost, preferably downloading via the Internet without charge and in a convenient and modifiable form. Furthermore, it should be possible to modify and distribute the work without any discrimination against persons, groups, fields or endeavour. In the scope of this study, the open data term refers to the publicly available data that may or may not require free registration.

7.2.2 Related Work

In recent years, the number of studies that address the use of anomaly detection in the maritime surveillance domain is increasingly growing. Anomaly detection techniques are divided into two groups, namely data-driven and knowledge-driven approaches. There are a couple of works that proposed knowledge-based anomaly detection systems with different representation techniques and reasoning paradigms such as rule-based, description logic and case-based reasoning [9–11]. A prototype for a rule-based expert system based on the maritime domain ontologies was developed by Edlund, Gronkvist, Lingvall, and Sviestins [12] that could detect some of the anomalies regarding the spatial and kinematic relation between objects such as simple scenarios for hijacking, piloting and smuggling. Another rule-based prototype was developed by Defence R&D Canada [8, 13]. The aforementioned prototype employed various maritime situational facts about both the

[4] Open definition, `opendefinition.org/okd/`

kinematic and static data in the domain to make a rule-based automated reasoning engine for finding anomalies. One of the popular data-driven anomaly detection approaches is the Bayesian network [14–16]. Johansson and Falkman [15] used the kinematic data for creating the network; however, in the work that was done by Fooladvandi et al. [14] expert's knowledge as well as the kinematic data was used in the detection process. Moreover Lane et al. [16] presented the detection approaches for five unusual vessel behaviors and the estimation of the overall threat was performed by using a Bayesian network. Unsupervised learning techniques have been widely used for data-driven anomaly detection such as Trajectory Clustering [17], self organizing map [18] and fuzzy ARTMAP neural network [19]. Some statistical approaches, such as Gaussian mixture model [20], hidden Markov model [21], adaptive kernel density estimator [22] and precise/imprecise state-based anomaly detection [17] have been used in this context. The majority of the works that have been done in the context of anomaly detection only used transponder data from the Automatic Identification System (AIS).

There are a number of studies that employed data fusion techniques to fuse data from different sensors in anomaly detection systems [23–26]. In these studies, the surveillance area was restricted to the coastal regions and the combination of data from AIS, synthetic aperture radar, infra-red sensors, video and other types of radar was used in the fusion process to obtain the vessel tracks. Furthermore, there are some other works that focused on the fusion of both sensor and non-sensor data [14, 27–31]. For example, Lefebvre and Helleur [29] and Riveiro and Falkman [31] treated the expert's knowledge as the non-sensor data. Riveiro and Falkman [31] introduced a normal model of vessel behavior based on AIS data by using self organizing map and a Gaussian mixture model. According to the model, the expert's knowledge about the common characteristic of the maritime traffic was captured as if – then rules and the anomaly detection procedure was supposed to find the deviation from the expected value in the data. Lefebvre and Helleur [29] used radar data with user's knowledge about the vessels of interests. The sensor data were modelled as track and the non-sensor data were modelled as templates. The track-template association was done by defining mathematical models for tracks and using fuzzy membership functions for association possibilities. Mano [30] proposed a prototype for the maritime surveillance system that could collect data from different types of sensors and databases and regroup them for each vessel. Sensors like AIS, high frequency surface wave radar and classical radars and databases such as environmental database, Lloyd's Insurance and TF2000 Vessel database were included in this prototype. By using multi-agent technology an agent was assigned to each vessel and anomalies could be detected by employing a rule-based inference engine. When the combination of anomalies exceeded a threshold, vessel status was informed to the user as an anomaly. The work presented by Ding et al. [28], proposed the architecture of a centralized integrated maritime surveillance system for the Canadian coasts. Sensors and databases included in this architecture were: high frequency surface wave radar, automatic dependant surveillance reports, visual reports, information sources, microwave radar, and radar sat. A common data structure was defined for storing data that were collected from different sensors. Andler et al. [27], also described a conceptual maritime surveillance system that integrated all available information such as databases and sensor systems (AIS, long-range identification and tracking, intelligence reports, registers/databases of vessels, harbours, and crews) to help user to detect and visualize anomalies in the vessel traffic data in a worldwide scale. Furthermore, the authors suggested using open data in addition to other resources in the fusion process.

In conclusion, the main focus of the studies that have been done in the context of anomaly detection in the maritime surveillance domain was related to using sensors data and mainly the AIS data to find anomalies in the coastal regions. Detection of some suspicious activities such as smuggling requires vessel traffic data beyond the coastal region. Maritime authorities in each country have overall information of maritime activities in their surveillance area. But exchanging information among different countries is a complicated procedure because of the diverse regulation of data protection in each land. Therefore, using data sources that are free from legislative procedures can be a good solution for providing information that belongs to the regions outside the land territory. Furthermore, all the information about maritime activities is not recorded in the authorities' databases or reported to them. On the other hand, there are numerous open data sources consists of different websites, blogs and social networks that can be useful for observing the hidden aspects of maritime activities. Hence, this article will investigate the potential open data sources for maritime activities and exploit them to build an anomaly detection system. The aim of this system is to provide complementary decision support for coastguard operators when they analyze traditional closed data sources.

7.3 Open Data in Maritime Surveillance

To obtain the applicable open data for anomaly detection, the first step is initiated by reviewing the information resources document[5] provided by the International Maritime Organization. This organization is the United Nations' specialized agency with responsibility for the safety and security of shipping and the prevention of marine pollution by vessels. The document introduces 29 governmental and intergovernmental organizations that work in different fields related to the maritime surveillance domain such as maritime safety, prevention of pollution from vessels, liability and insurance issues, shipping information, etc. All these 29 organizations' websites and the links provided by each of them are investigated and a list of online data sources is prepared. The obtained open data sources provide AIS data, information about vessel characteristics, ports, maritime companies, suppliers, weather, etc. Moreover, in the process of finding open data sources, an attempt is made to obtain sources of data that are related to the Baltic region and mostly Sweden by use of the previously observed data sources and also common search engines. This extensive collection of open data sources for the maritime surveillance domain is available for download[6].

7.4 Case Study

The case study is the employed research method in this article. The two important sources of information about maritime anomalies are reports of the workshops that were held in Canada [8] and Sweden [27, 32]. In these two workshops attendees were experts in the maritime domain and a variety of maritime anomalies were identified.

According to the identified anomalies by the two workshops, a list of some potential maritime anomalies that can be detected by use of the available open data sources was prepared in the preliminary work of this

[5] Information resources on maritime security and ISPS code, `www.imo.org/knowledgecentre/informationresourcesoncurrenttopics/maritimesecurityandispscode/documents/information-resources-on-maritime-security-and-isps-code.pdf`

[6] Open maritime-specific data collection, `http://www.bth.se/com/nla.nsf/sidor/resources`.

study. Then, in a meeting with representatives of the Swedish coastguard the types of anomalies that are of high interest for the coastguard operators and the possibility of using open data for anomaly detection were discussed. During the meeting, the prepared list of anomalies was reviewed. Coastguard operators were asked about the possibility of the historical occurrence, and their degree of interest, for each anomaly. As an outcome of the meeting a number of scenarios were created and based on these scenarios, 11 expert rules were defined.

The first scenario refers to the anomalies related to the vessel static information such as name, owner, International Maritime Organization number, dimensions, type and the status (in service or laid up). For example, sailing a vessel with a draught of 22 meters over an area with a 9 meter depth or observing a vessel that should be laid up or changing the name or the owner of a vessel during its voyage indicate the existence of suspicious activities.

The second scenario is related to the prior arrival notification for vessels. Vessels should inform their arrival time to the ports at least 24 h in advance. Each port also provides an online timetable for the incoming vessels. Therefore, any mismatch between the reported AIS data regarding the destination or the arrival time of a vessel and the destination port timetable needs to be checked by the coastguards.

The third scenario is related to ordering pilots. Usually, large vessels because of their size and weight need to be guided by pilots through dangerous and congested waters. Therefore, vessels need to submit their request for a pilot and also inform the destination port. However, in some cases vessels order a pilot without informing the port. Such situations should be investigated.

The case study in this article comprises scenarios two and three, which are conducted in close collaboration with operators from the Swedish coastguard. The aim of the case study is to investigate the potential of open data about maritime activities (vessels, ports, and so on) as a complement to the closed data sources that are already used by the Swedish coastguard. They key questions posed in this study are:

- How can open data complement closed data for anomaly detection in the maritime surveillance domain?

- What is the positive and negative impacts of the open data sources? (that is, what is the increase in true negatives and positives in comparison to the increase in false negatives and positives?)

In the next meeting, the scenarios and the rules are presented to the representatives of the Swedish coastguard and they are asked to comment or suggest new scenarios or rules. By getting the final approval from the coastguard experts, one new rule (rule number 5) is added to the list. Table 7.1 shows the admitted rules by the experts. These identified maritime anomalies can be detected by use of AIS data, vessel traffic timetables in ports and pilots websites and the vessel characteristic data that are available in data sources such as Lloyd's. A capitalized name is provided to each anomaly that can be detected by the rules, and for the remainder of this article the anomalies will be referred to by these names.

Table 7.1: The identified anomalies that can be detected by open data (confirmed by the Swedish Coastguard)

No.	Expert rules	Anomaly
1	If a vessel destination does not exist in the port schedule then anomaly.	VESSEL_NOT_INFORMED_PORT (A1)
2	If a vessel ETA does not match with the port ETA for the vessel then anomaly.	ARRIVAL_TIME_MISMATCHED (A2)
3	If a vessel entered a port without informing the port then anomaly.	VESSEL_ENTERED_PORT_WITHOUT _NOTICE (A3)
4	If a vessel has requested a pilot but has not used the service then anomaly.	VESSEL_NOT_USED_PILOT (A4)
5	If vessel A which normally travels between ports X and Y, suddenly goes to port Z then anomaly.	UNUSUAL_TRIP_PATTERN (A5)
6	If a vessel has not left a port according to the port schedule then anomaly.	VESSEL_NOT_LEFT_PORT (A6)
7	If a vessel exists in a port schedule but it has not entered the port then anomaly.	VESSEL_NOT_ENTERED_PORT (A7)
8	If a vessel does not exist in the port schedule and the vessel has requested a pilot then anomaly.	VESSEL_ORDERED_PILOT_AND _NOT_INFORMED_PORT (A8)
9	If a vessel has moored in a port and has been observed somewhere else then anomaly.	VESSEL_MOORED_IN_PORT (A9)
10	If vessel A has not entered a port according to the port schedule instead vessel B enters the port at the same time slot then anomaly.	WRONG_VESSEL_ENTERED (A10)
11	If a vessel with the laid up status has been observed somewhere else then anomaly.	VESSEL_LAID_UP (A11)

Note. ETA = estimated time of arrival.

7.5 Framework Design

A new maritime surveillance framework and expert-based decision support system is presented in this article. The Open Data Anomaly Detection System (ODADS) is designed for traffic monitoring and detecting anomalies in the maritime domain by using open and closed data sources. Fig. 7.1 depicts the ODADS architecture. The framework is designed to be generalizable to similar applications in other domains; that is, for applications where the objective is to identify anomalous behavior through semi-automatic methods. The proposed framework is designed to provide decision-support based on knowledge-engineering-based or knowledge-discovery-based methods. It is focused on the extraction of information from open data sources. The setup of any implementation of the framework depends largely on the problem at hand. ODADS consists of three core modules:

- Data Collector
- Anomaly Detector and,
- Display Client

The Data Collector module is responsible for collecting open data from the Internet, and for preprocessing and storing the data in the system database. The data can be related to vessel traffic (such as AIS reports, ports and pilots timetables), vessel characteristics, ports equipments and facilities, companies that are involved in maritime activities, news or reports about maritime events and activities available in different social media platforms (such as blogs and social networks), and so on. The Data Store comprises a set of databases that contain data belonging to different types of sensors, authorized databases and open data sources. The data in the Data Store can be fused or integrated before being used in the detection process. When the Data Collector completes its task, the Anomaly Detector becomes available. The Anomaly Detector module analyses the available open and closed data and detects possible anomalies by using both knowledge-driven and data-driven techniques. Different anomaly detection techniques are employed due to the distinct nature of anomalies and the complexity of the environment in the maritime surveillance domain. Previously known anomalies can be detected by knowledge-based techniques but in real-world situations it is desirable that an anomaly detection system can detect previously unseen anomalies as well. One of the potential benefits of using data-driven methods such as machine learning algorithms is the possibility of detecting such unseen anomalies. However, it is difficult to evaluate how well today's data-driven systems manage to detect previously unseen cases. The proposed system in this article is deterministic and completely expert-based but our proposed framework general enough to allow data-driven detection methods in other applications. The Display Client module is the user interface of the system. This module represents the cognitive refinement level (Level 5) of the Joint Directors of Laboratories model. It is argued that the effectiveness of a system can be affected by the way that the system produced information is comprehended by the human user [33]. The cognitive refinement process involves traditional human computer interaction utilities such as geographical display or advanced methods that support functionalities such as cognitive aids, negative reasoning enhancement, focus/defocus of attention and representing uncertainty. While designing the user interface, the six principles of user interface design that are based on the usage-centered design approach are considered. These six principles are: structure, simplicity, visibility, feedback, tolerance, and reuse [34].

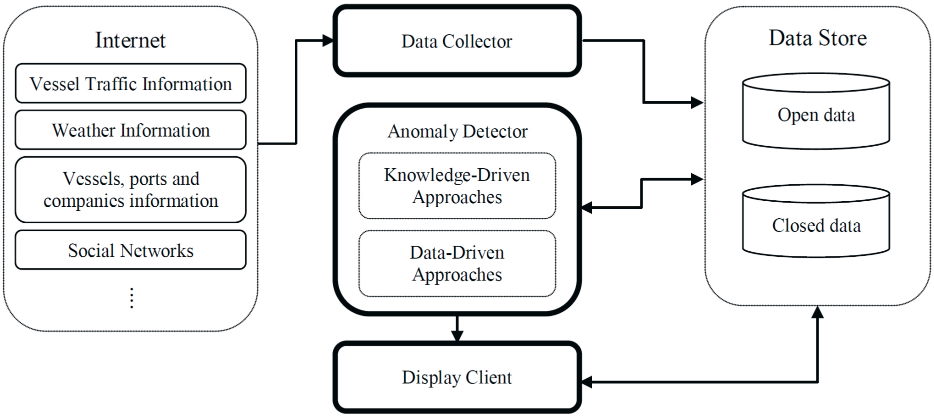


Figure 7.1: The Open Data Anomaly Detection System (ODADS) architecture. The Data Collector module collects data from the Internet and stores them in the database. The Anomaly Detector module detects anomalies by taking advantage of different techniques. The Display Client module displays the detected anomalies to the user and enables system-user interaction.

7.6 Implementation

ODADS is implemented by taking advantage of the identified maritime anomalies and the obtained open data sources. To limit the scope, four types of vessels (passenger, ferry, cargo, and tanker) are considered. Other types of vessels (such as fishing and sailing vessels) are omitted. Secondly, the rule related to the vessel static information is ignored. Furthermore, the WRONG_VESSE_ENTERED anomaly is excluded due to its complexity. Moreover, in further collaboration with the coastguard representatives during the implementation phase, a new type of anomaly is proposed. This anomaly is called UNDER_SURVEILLANCE_VESSEL and occurs when a vessel of interest has any of the A1–A9 anomalies and the vessel exists in the vessels blacklist.

7.6.1 Data Description

The required vessel traffic data can be obtained from AIS reports and ports and pilots timetables. The surveillance area is restricted to the north of the Baltic Sea and a part of the Gulf of Finland, the regional area between three European countries Sweden, Finland and Estonia. Fig. 7.2 shows the surveillance area where the geographic coordinates lie between latitudes 58.49° – 60.24° N and longitude 16.19° – 25.00° E. This region is one of the high-traffic regions in the Baltic Sea and is surrounded by the four highly used ports. The selected ports are: Stockholm group (Stockholm, Kapellskär and Nynäshamn) and Norrköping ports in Sweden, Helsinki port in Finland and Tallinn port in Estonia. Due to inaccessibility to the raw AIS data (a closed source) in the surveillance area, the AIS reports that are provided by the *MarineTraffic.com* website are exploited. These reports consist of both static and dynamic types of data for each vessel during its voyage such as name, type, year built, flag, call sign, maritime mobile service identity, International Maritime Organization identification number, origin, destination, Estimated Time of Arrival, speed (maximum and average), position (longitude and latitude), and heading. The pilot data belong to the Stockholm pilotage area in Sweden. Moreover, a common data representation format for ports and pilots data is defined that contains vessel name, vessel type, origin, destination, company name, vessel status and arrival/departure time. Table A2 in the appendix provides more details about these sources.

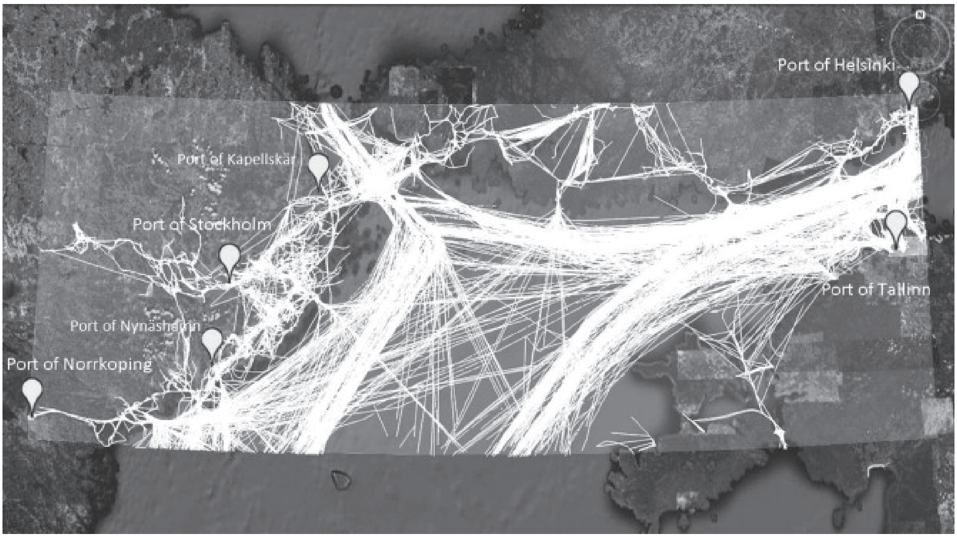


Figure 7.2: The surveillance area and the vessels tracks extracted from AIS datafor 6 days. The area is restricted to the north of the Baltic Sea and part of the Gulf of Finland. Ports from left to right are Norrköping, Nynäshamn, Stockholm, Kapellskär, Helsinki and Tallinn (The image is generated by Google Earth).

7.6.2 Detection Methods

After investigating the nature of the anomalies and the potential techniques, it is determined that except for the UNUSUAL_TRIP_PATTERN anomaly, detection of other anomalies can be done by performing a search in the data for finding the desired match. If the match is not found then the vessel would be marked as an anomaly. Using exact string matching techniques for comparing vessel information from different data sources is inapplicable due to the potential errors that might occur because of different notations or human operator mistakes during data entry. Therefore, a metric should be used for measuring the degree of similarity between two vessels from different sources. After investigating the performance of different string matching techniques on the available data, the *JaroWinkler*[7] metric is chosen. For two strings if the JaroWinkler distance is less than or equal to a predefined threshold then the two strings are considered similar.

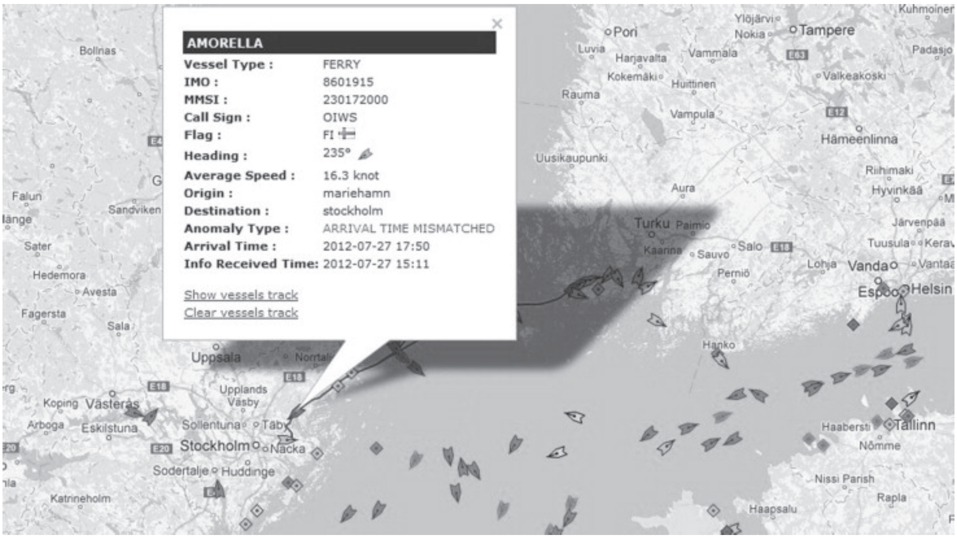
[7] JaroWinkler (the variant of Jaro) measures the number and order of common characters in the two strings and also the number of transposition that needs to change one of the strings to the other.

Detection of the UNUSUAL_TRIP_PATTERN anomaly requires data-driven approaches such as machine learning or statistical techniques. Unlike the other anomalies, detection of this anomaly requires a history of vessel traffic data for training the system. For this reason, data related to six months (September 15, 2011-March 15, 2012) of vessel traffic in the surveillance area are gathered. By monitoring the activities during this period, the system will be able to find the normal pattern of vessels trips in the area of interest. For detecting this anomaly a simple statistical approach is used. A look up table is created and for each vessel the number of times that the vessel travels between two different ports is stored. For each vessel, if the frequency of travelling between its origin and destination is less than a predefined threshold (which is 2 in the current implementation) then the vessel will be reported as an anomaly.

There are situations that multiple anomalies can occur in the same time for a specific vessel. The combinations of anomalies that don't have any features in common are defined as new types of anomalies. For instance, a vessel has not informed its arrival to the destination port and its trip to the port is not common, in such situation a new type of anomaly UNUSUAL_TRIP_PATTERN_AND_VESSEL_NOT_INFORMED_PORT is defined.

7.7 System Verification

To ensure that ODADS works properly, the system is tested manually with both real and manipulated data. The tests are performed during the implementation and also after completing the system. At first, a number of vessels with different types of anomalies are inserted to the real collected data to check whether all types of anomalies can be detected by ODADS. Then, the system is run for a period of time and the detected anomalies are checked manually against the available data to make sure about their correctness. A screenshot of the ODADS visualization of the maritime environment is shown in Fig. 7.3. During the test phase the Anomaly Detector module is updated and some of the detection conditions are narrowed down. The process is repeated until the system can detect all the anomalies correctly. Furthermore, before using ODADS in real-world situations, it is important to figure out to what extent the results of the system are accurate. For this reason, an experiment is conducted to measure the accuracy. Accuracy is



AMORELLA

Vessel Type :	FERRY
IMO :	8601915
MMSI :	230172000
Call Sign :	OIWS
Flag :	FI
Heading :	235°
Average Speed :	16.3 knot
Origin :	mariehamn
Destination :	stockholm
Anomaly Type :	ARRIVAL TIME MISMATCHED
Arrival Time :	2012-07-27 17:50
Info Received Time:	2012-07-27 15:11

Show vessels track
Clear vessels track

Figure 7.3: A screenshot of the ODADS visualization of the maritime environment.

the degree to which the estimates or measurements of a quantity correctly describe the exact value of that quantity. In other words, accuracy is the proportion of true results in the population. To evaluate the system accuracy, the number of True Positives (TP), False Positives (FP), True Negatives (TN) and False Negatives (FN) are needed. Accuracy is calculated by the following formula [35]:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{7.1}$$

The first step in designing the experiment is to identify the population. The population consists of the vessel traffic data in the surveillance area. Since the population is too large and it is impossible to look into all members manually to count the number of TP, FP, TN and FN, a sample should be taken from the population. The sampling frame is the vessel traffic data related to AIS, ports and pilots in the surveillance area, which are provided by ODADS. Due to the high volume of traffic through the surveillance area, it is expected that the majority of anomalies can be observed in one week execution of the system. Therefore, one week of vessel traffic data from April, 2012 is used as the sample frame (Table A3 and Table A4 in the appendix present some information about the vessels traffic and detected anomalies during this week). A two stage random sampling is used in order to have unbiased and independent samples. In the first stage, a simple random sampling without replacement is done for selecting the time slots that ODADS attempts to collect and analyse the data. After selecting the time slots, the corresponding data for each time slot will be selected by a stratified sampling. Three strata are defined according to the type of vessels: ferry and passenger, cargo and tanker vessels. Selection of vessels is also limited to the vessels that are originated from or targeted to the four particular ports. The total number of time slots in the sample frame is 835. This means that on average, ODADS collects data 139 times a day. In the first stage of sampling a random timeslot is selected for each day, which results in 7 time slots for one week. Then, by considering the described limitation in the selection process, the average number of entire data in a selected time slot is about 100 records. Among these records, 30 records are selected by stratification. Almost 73% of the vessels in each time slot are moored. Since the majority of the anomalies are related to the vessels trips, a limitation on the number of moored vessels in the samples is defined. In this way, it can be possible to check more anomalies in the evaluation process. The second stage of sampling is repeated by taking into consideration that the number of moored vessel in the sample cannot exceed from the half of the sample size (in this case, 15).

After carrying out the sampling, all the samples are checked against the primary identified anomalies ($A_1 - A_9$). To compute the number of TP, FP, TN and FN, a confusion matrix is created based on the nine classes of anomalies and the normal class (Table 7.2). According to the matrix, the accuracy of the system is: $17 + 192/17 + 192 + 0 + 1 = 0.99$. The existing FN for the ARRIVAL_TIME_MISMATCHED anomaly is due to the wrong provided AIS data by the vessel or possibly the *MarineTraffic.com* website and also the limitation of the system for considering all conditions. In this case, the vessel arrival time belongs to a couple of days before the current date and for this reason it is ignored by the system. However, it is quite possible to handle such situations if additional sources of AIS data are available.

7.8 System Validity

The validation was made by an officer at the coastguard Headquarters office in Karlskrona, Sweden for four weeks (April 23, 2012–May 18, 2012), at

Table 7.2: Confusion matrix for the nine classes of anomalies and the normal class

						Predicted class						
		A1	A2	A3	A4	A5	A6	A7	A8	A9	Normal	Total
Actual class	A1	6	-	-	-	-	-	-	-	-	-	6
	A2	-	7	-	-	-	-	-	-	-	1	8
	A3	-	-	-	-	-	-	-	-	-	-	0
	A4	-	-	-	-	-	-	-	-	-	-	0
	A5	-	-	-	-	3	-	-	-	-	-	3
	A6	-	-	-	-	-	1	-	-	-	-	1
	A7	-	-	-	-	-	-	-	-	-	-	0
	A8	-	-	-	-	-	-	-	-	-	-	0
	A9	-	-	-	-	-	-	-	-	-	-	0
	Normal	-	-	-	-	-	-	-	-	-	192	192
	Total	6	7	0	0	3	1	0	0	0	193	17

any time during working hours (08:00–17:00). The officers are supposed to evaluate the detected anomalies by checking them against the available data in the systems and data sources that are used during the normal operational activities at the coastguard. They are asked to provide weekly report about their evaluation results in order to decrease the possible malfunctioning of the system and the validation process. A detected anomaly for a vessel is true if it can be confirmed by the available data sources at the coastguard and consequently it is false if the authorized data sources provide any information that declines the detected anomaly. No further assessment is done regarding the classification of the detected anomalies to true and false alarms.

The sea monitoring system that is used by the coastguard officer is called SJöBASIS[8]. SJöBASIS aggregates the maritime data from different systems and agencies with the aim of improving the efficiency of maritime surveillance. In SJöBASIS, he required data that contain vessel position, speed, heading, arrival/departure time and trip, are obtained from the following sources SafeSeaNet[9], SjöC, local AIS and HELCOM AIS[10]. The officer checks the validity of anomalies according to the priority that each anomaly has for him. During the four-week validation period, ODADS is used at the coastguard for 12 working days and in total 76 of the detected anomalies are evaluated. Table 7.3 presents the validation results. Among the evaluated anomalies, there are a number of anomalous vessels

[8] www.kustbevakningen.se/sv/granslos-samverkan/sjoovervakningsuppdraget/samverkan-sjoinformation/

[9] www.emsa.europa.eu/operations/maritime-surveillance/safeseanet.html

[10] www.helcom.fi/BSAP/ActionPlan/en_GB/SegmentSummary/

Table 7.3: Validation results of the Coastguard

Anomaly	Alarms		
	True	False	Not Checked
VESSEL_NOT_INFORMED_PORT	7	3	4
ARRIVAL_TIME_MISMATCHED	19	5	3
VESSEL_NOT_USED_PILOT	2	-	-
UNUSUAL_TRIP_PATTERN	7	6	-
VESSEL_NOT_LEFT_PORT	1	1	-
VESSEL_ORDERED_PILOT_AND_NOT_INFORMED_PORT	2	-	-
VESSEL_MOORED_IN_PORT	-	3	-
UNDER_SURVEILLANCE_VESSEL	1	-	-
UNUSUAL_TRIP_PATTERN_AND_VESSEL_NOT_ INFORMED_PORT	5	1	-
UNUSUAL_TRIP_PATTERN_AND_VESSEL_ ARRIVAL_TIME_ MISMATCHED	4	-	-
UNUSUAL_TRIP_PATTERN_AND_VESSEL_ORDERED_PILOT_ AND_NOT_INFORMED_PORT	-	1	-
VESSEL_ORDERED_PILOT_AND_NOT_INFORMED_PORT_ AND_VESSEL_NOT_USED_PILOT	1	-	-
Total count	49	20	7
Total count (%)	64.47	26.32	9.21

that remain unchecked due to a lack of corresponding data in the coastguard systems. A large number of detected anomalies are related to the ARRIVAL_TIME_MISMATCHED anomaly that in many cases can be due to the inconsistent time formats in different data sources and various settings in the AIS transmitters. In these cases, the detected anomalies by ODADS are correct detections based on the available data but the real-world situation that the data are used represent is considered normal. For example, a vessel that is going from Helsinki to Stockholm is reporting its arrival time according to the Finland local time instead of using the coordinated universal time format. Therefore, this artificial time difference results in that the arrival time reported by the vessel does not match the expected arrival time of the vessel at the destination port, which leads to set the ARRIVAL_TIME_MISMATCHED anomaly for that vessel. In addition to the comparative analysis in the validation process, modus operandi of using an Anomaly Detector that takes advantage of open data is investigated. The coastguard officer uses an analysis tool[11] to analyse the ODADS excel reports and draw conclusions regarding the modus operandi of the system in the emergency situations that can have impact on the maritime surveillance operations. One of the possible analyses of the system reports can be the investigation of vessels with multiple anomalies. Here are some examples of

[11] IBM i2 Analyst's Notebook, `www.i2group.com/us/products/analysis-product-line/ibm-i2-analysts-notebook`

the vessels with multiple anomalies. A cargo vessel has recurring anomalies related to the arrival/departure time, trip and notification to the port. From the types of anomalies that are detected for this vessel, it can be concluded that the vessel behavior points to a higher threat concerning customs and border. For a passenger vessel the following anomalies are often detected: VESSEL_NOT_INFORMED_PORT, ARRIVAL_TIME_MISMATCHED, VESSEL_NOT_ENTERED_PORT and VESSEL_NOT_LEFT_PORT. These anomalies may happen because of inaccurate or wrong provided data by the vessel. The conclusion and modus operandi for this vessel is that the duty officer will contact the vessel to highlight the importance of submitting accurate information. In real-world situations, the occurring anomalies such as VESSEL_NOT_ENTERED_PORT or VESSEL_NOT_LEFT_PORT for a passenger vessel can be related to serious issues such as an accident and it will result in increased scrutiny for that vessel. It is also possible to look into the relation between the types of vessels and the detected anomalies. Such assessment can be used for strategic and risk analysis. For tanker vessels the most common anomaly is VESSEL_NOT_INFORMED_PORT. This anomaly has a high priority for emergency preparedness for accidents involving tankers. Tankers with UNUSUAL_TRIP_PATTERN anomaly are a potential risk to other vessels and can cause accidents. From a risk assessment point of view the combination of this anomaly with the VESSEL_NOT_ENTERED_PORT or VESSEL_NOT_USED_PILOT anomalies can lead to high-risk situations. The most occurring anomaly for ferries and passenger vessels is ARRIVAL_TIME_MISMATCHED. In several cases this anomaly is detected incorrectly because of the wrong reported arrival time; however, this anomaly has great importance for the authorities to plan their operations regarding ferries and passenger vessels arrivals effectively. The most serious anomaly for ferries and passenger vessels is VESSEL_NOT_LEFT_PORT and the authorities should suspect that some form of accident or difficulty is arising regarding to the departure of the vessels. For cargo vessels, the most recurring anomalies are VESSEL_NOT_INFORMED_PORT and ARRIVAL_TIME_MISMATCHED. However, some of the ARRIVAL_TIME_MISMATCHED anomalies are false alarms because of the incorrect data. The VESSEL_NOT_INFORMED _PORT anomaly is important for ports security and safety. The prior notification to the ports is obligatory for vessels, but the fine for breaking this rule is negligible which lets the vessels that are involved in illegal activities such as smuggling disobey this rule. The most serious anomalies for the cargo vessels are the VESSEL_NOT_ENTERED_PORT and UNDER_SURVEILLANCE_VESSEL anomalies.

Furthermore, looking into the most frequent anomalies for different ports will assist the maritime authorities to make their decisions more efficiently. According to the report analysis, the most frequent anomaly in Stockholm and Nynäshamn ports is ARRIVAL_TIME_MISMATCHED, in port of Kapellskär is VESSEL_NOT_INFORMED_PORT and in Norrköping port is UNUSUAL_TRIP_PATTERN. One possible conclusion from the most popular anomalies for the ports is that the port authorities should be informed of the divergence in the traffic flow and the operational management functions in order to plan and allocate resources efficiently. On the other hand, in some cases the anomalies are too commonly occurring for a port because of the inaccurate provided data and they can be disregarded by the officer.

The received feedback from the coastguard representatives during the validation process indicates that ODADS complements the closed sources and assists the human operator in gaining a better understanding of the ongoing maritime activities. The representatives believe that the ODADS results are reliable and the quality of the open data that are used is good and can be used in real-world situations. The functionality, usability, and visualization tools in ODADS provide a simple and intuitive system for coastguard operators. In addition to illustrate the vessel traffic data in a simple, clear, and informative way, ODADS can provide statements about the anomalies and its statistical reports are beneficial when the authorities and freight companies conduct strategic analysis of maritime traffic and risk assessment. Finally, the capability of automatic detection of anomalies based on open data is considered a valuable asset to the coastguard.

7.9 Discussion

Taking advantage of anomaly detection systems will assist authorities to tighten security in the maritime surveillance domain. There are a number of studies which focused on developing anomaly detection systems by using knowledge-driven and data-driven approaches. For instance, Defence research and development Canada [8, 13] developed a rule-based prototype for anomaly detection by exploiting maritime situational facts about both kinematic and static data of the domain. Edlund et al. [12] ddeveloped another prototype for a rule-based expert system to detect the anomalies regarding spatial and kinematic relation between objects. Riveiro and Falkman [31] proposed using a combination of data-driven and knowledge-driven approaches to detect anomalies by use of a normal model of vessel behavior based on AIS data and experts rules. In the majority of studies that addressed anomaly detection, the exploited data for the anomaly detection process were obtained from closed data sources and there is a lack of investigation on using open data sources for anomaly detection in the maritime surveillance domain. Therefore, in this article ODADS is implemented by employing expert rules to investigate the potential open data as a complement to the closed data for anomaly detection in the maritime surveillance domain.

The validity of the system is evaluated in real-world situations by the experts from the Swedish coastguard. Despite the inaccurate nature of open data and by considering the fact that only open data sources are used in the system, the high degree of true alarms (64.47%) in the validation process admits the validity of the system outcomes. Furthermore, there are no corresponding data in the authorized databases for 9.21% of the evaluated anomalies by the coastguard. This fact refers to a potential information gap in the closed data sources. However, the considerable number of false alarms (26.32%) for a surveillance system is still unsatisfactory. The number of false alarms indicates the difference between the accuracy of the system and the validity of the results. Even though the data that are used in ODADS are obtained from relatively trusted data sources such as ports, the false alarms occur mostly because of data inaccuracies. The open data that are exploited by ODADS suffer from these errors due to human operator mistakes, irregular data updates, data update latencies, and incompatible data formats. These are critical issues that unfortunately seem to concern many open data applications. In ODADS, there are situations where a detected anomaly disappears in the next periods of system execution because of the arrival of revised and corrected data. Frequent occurrences of false alarms distract the operator's attention from real anomalies in the surveillance area.

To decrease the false alarms in ODADS, the main solution is to integrate open and closed data, which can cover the lack of information or inaccuracy in the open data. In addition, considering a probability for the detected anomalies can decrease the number of false alarms. This would be possible by analysing the history of vessels behavior as well as the current situation and defining a probability threshold to omit the anomalies that have a lower probability than the threshold. Furthermore, having extra information regarding vessels such as crew and cargo information, can affect the probability of being a real anomaly for a specific vessel. For example, if a vessel has the ARRIVAL_TIME_MISMATCHED anomaly and it has a crew member with a criminal record or a special cargo, then there is a possibility that the vessel is stopped somewhere to exchange something. Therefore, in such situation, the probability of being a true anomaly is high. According to the validation results, the UNUSUAL_TRIP_PATTERN anomaly creates the majority of false alarms. This is due in part to the statistical approach that is used for detecting this anomaly and also the wrong origin and destination information that the vessels provide. The lookup table that is created for storing the frequency of the trips between different places is not updated periodically. While populating the table, the ports timetables are used which can be incomplete. An alternative detection approach can be to use machine learning techniques, which attempt to detect anomalies according to the pattern of movements for individual vessels instead of the reported trip data by the vessels or ports.

The proposed framework is generalizable to similar applications in other domains due to its modularized and general design. This case study has investigated the potential of open data as a complement to closed data for anomaly detection in the maritime surveillance domain. The primary stakeholders in the case study are human operators from the Swedish coastguard. Since the main purpose of the case study was to focus analysis and investigation on two defined scenarios, it is not possible to draw conclusions about the generalizability of the results. Further investigations can shed light on this generalizability by conducting large-scale trials of the implemented expert-based system across additional scenarios and more maritime surveillance areas.

In local areas such as the surveillance area in this article, mainly because of large amount of quality assured data and the limited size of the surveillance area, it is easier for the maritime authorities to track and control the vessels activities. Therefore, the use of open AIS data in this region is not required and it should be prohibited to decrease the negative impacts of open data on the system results. On the other hand, when the vessel information beyond the exclusive economic zone is required, the value of open data becomes more obvious.

7.10 Conclusion and Future Work

This article investigated the potential open data as a complementary resource for anomaly detection in the maritime surveillance domain. A framework for anomaly detection was proposed based on the usage of open data sources along with other traditional sources of data. According to the proposed anomaly detection framework and the algorithms for implementing the expert rules, the Open Data Anomaly Detection System (ODADS) was developed. The validity of the results was investigated by the subject matter experts from the Swedish coastguard. The validation results showed that the majority of the ODADS evaluated anomalies were true alarms. Moreover, a potential information gap in the closed data sources was observed during the validation process. Despite the high number of true alarms, the number of false alarms was also considerable that was mainly because of the inaccurate open data. This article provided insights into the open data as a complement to the common data sources in the MS domain and is concluded that using open data will improve the efficiency of the surveillance systems by increasing the accuracy and covering unseen aspects of maritime activities.

In the future, it is important to investigate how the open data sources in the maritime domain can be used in a global perspective. In this article, the surveillance area was limited to a local area which is fully covered by the authorities' data sources. When the data beyond the exclusive economic zone are needed, it is more valuable to use open data sources. By taking advantage of the subject matter experts' knowledge about maritime surveillance, it would be possible to figure out how the global open data should be exploited for the surveillance purpose. Integration of the open data with maritime confidential data can improve the efficiency of maritime surveillance and should be considered as a further improvement of the system. Another improvement can be considering a probability for each detected anomaly according to the history of the vessels behavior and the current situation. Moreover, further investigation on the other sources of open data such as social data, which is created and shared through social media platforms, and online videos from the ports activities in the high risk regions, will be useful. The data that are used in ODADS are relatively trusted, but in case of using other open data sources in the maritime surveillance domain for anomaly detection, the quality assurance of the data should be investigated. As well as using knowledge-based systems, taking advantage of data-driven approaches such as machine learning techniques can increase the efficiency of the maritime surveillance systems. Finally, the next step for improving the maritime surveillance systems after being equipped with the anomaly detection functionality is to predict the future threats or incoming anomalies based on the analysis of the current situation.

References

[1] E. Lefebvre, M. Simard, and C. Helleur. "Multisource information adaptive fuzzy logic correlator for recognized maritime picture". In: *Int'l Conf. on Artificial Intelligence and Applications Symp.* 1. 2001, pp. 229–234.

[2] A. Ponsford, I. A. D'Souza, and T. Kirubarajan. "Surveillance of the 200 nautical mile EEZ using HFSWR in association with a spaced-based AIS interceptor". In: *Conf. on Technologies for Homeland Security.* IEEE. 2009, pp. 87–92.

[3] J. C. Molloy. "The open knowledge foundation: Open data means better science". In: *PLoS Biology* 9.12 (2011), e1001195.

[4] J. Alonso, O. Ambur, M. A. Amutio, O. Azañón, D. Bennett, R. Flagg, D. McAllister, K. Novak, S. Rush, and J. Sheridan. "Improving access to government through better use of the web". In: *World Wide Web Consortium* (2009).

[5] D. Dietrich, J. Gray, T. McNamara, A. Poikola, R. Pollock, J. Tait, and T. Zijlstra. *Open Data Handbook Open Knowledge Foundation Logo.* 2009. URL: `www.opendatahandbook.org/en/`.

[6] S. Dedijer and N. Jéquier. *Intelligence for economic development: An inquiry into the role of the knowledge industry.* Berg Pub Ltd, 1987.

[7] V. Chandola, A. Banerjee, and V. Kumar. "Anomaly detection: A survey". In: *ACM Computing Surveys* 41.3 (2009), p. 15.

[8] J. Roy. "Anomaly detection in the maritime domain". In: *Optics and Photonics in Global Homeland Security IV.* Vol. 6945. Int'l Society for Optics and Photonics. 2008, 69450W.

[9] A. B. Guyard and J. Roy. "Towards case-based reasoning for maritime anomaly detection: A positioning paper". In: *Proc. of The IASTED Int'l Conf. on Intelligent Systems and Control.* Vol. 665. 006. 2009, p. 1.

[10] M. Nilsson, J. Van Laere, T. Ziemke, and J. Edlund. “Extracting rules from expert operators to support situation awareness in maritime surveillance”. In: *The 11th Int’l Conf. on Information Fusion*. IEEE. 2008, pp. 1–8.

[11] J. Roy and M. Davenport. “Exploitation of maritime domain ontologies for anomaly detection and threat analysis”. In: *Int’l Conf. on Waterside Security*. IEEE. 2010, pp. 1–8.

[12] J. Edlund, M. Grönkvist, A. Lingvall, and E. Sviestins. “Rule-based situation assessment for sea surveillance”. In: *Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications*. Vol. 6242. Int’l Society for Optics and Photonics. 2006, p. 624203.

[13] J. Roy. “Rule-based expert system for maritime anomaly detection”. In: *Sensors, and Command, Control, Communications, and Intelligence Technologies for Homeland Security and Homeland Defense IX*. Vol. 7666. Int’l Society for Optics and Photonics. 2010, 76662N.

[14] F. Fooladvandi, C. Brax, P. Gustavsson, and M. Fredin. “Signature-based activity detection based on Bayesian networks acquired from expert knowledge”. In: *The 12th Int’l Conf. on Information Fusion*. IEEE. 2009, pp. 436–443.

[15] F. Johansson and G. Falkman. “Detection of vessel anomalies-a bayesian network approach”. In: *The Third Int’l Conf. on Intelligent Sensors, Sensor Networks and Information*. IEEE. 2007, pp. 395–400.

[16] R. O. Lane, D. A. Nevell, S. D. Hayward, and T. W. Beaney. “Maritime anomaly detection and threat assessment”. In: *The 13th Conf. on Information Fusion*. IEEE. 2010, pp. 1–8.

[17] A. Dahlbom and L. Niklasson. “Trajectory clustering for coastal surveillance”. In: *The Tenth Int’l Conf. on Information Fusion*. IEEE. 2007, pp. 1–8.

[18] M. Riveiro, G. Falkman, and T. Ziemke. “Improving maritime anomaly detection and situation awareness through interactive visualization”. In: *The 11th Int’l Conf. on Information Fusion*. IEEE. 2008, pp. 1–8.

[19] B. J. Rhodes, N. A. Bomberger, M. Seibert, and A. M. Waxman. “Maritime situation monitoring and awareness using learning mechanisms”. In: *Conf. on Military Communications*. IEEE. 2005, pp. 646–652.

[20] R. Laxhammar. “Anomaly detection for sea surveillance”. In: *The 11th Int’l Conf. on Information Fusion*. IEEE. 2008, pp. 1–8.

[21] M. Andersson and R. Johansson. “Multiple sensor fusion for effective abnormal behaviour detection in counter-piracy operations”. In: *Int’l Conf. on Waterside Security*. IEEE. 2010, pp. 1–7.

[22] B. Ristic, B. La Scala, M. Morelande, and N. Gordon. “Statistical analysis of motion patterns in AIS data: Anomaly detection and motion prediction”. In: *The 11th Int’l Conf. on Information Fusion*. IEEE. 2008, pp. 1–7.

[23] C. Carthel, S. Coraluppi, and P. Grignan. “Multisensor tracking and fusion for maritime surveillance”. In: *The Tenth Int’l Conf. on Information Fusion*. IEEE. 2007, pp. 1–6.

[24] M. Guerriero, P. Willett, S. Coraluppi, and C. Carthel. “Radar/AIS data fusion and SAR tasking for maritime surveillance”. In: *The 11th Int’l Conf. on Information Fusion*. IEEE. 2008, pp. 1–5.

[25] B. J. Rhodes, N. A. Bomberger, M. Seibert, and A. M. Waxman. “SeeCoast: Automated port scene understanding facilitated by normalcy learning”. In: *Conf. on Military Communications*. IEEE. 2006, pp. 1–7.

[26] M. Vespe, M. Sciotti, F. Burro, G. Battistello, and S. Sorge. “Maritime multi-sensor data association based on geographic and navigational knowledge”. In: *Radar Conf.* IEEE. 2008, pp. 1–6.

[27] S. F. Andler, M. Fredin, P. M. Gustavsson, J. van Laere, M. Nilsson, and P. Svenson. “SMARTracIn: A concept for spoof resistant tracking of vessels and detection of adverse intentions”. In: *Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense VIII*. Vol. 7305. Int’l Society for Optics and Photonics. 2009, 73050G.

[28] Z. Ding, G. Kannappan, K. Benameur, T. Kirubarajan, and M. Farooq. “Wide area integrated maritime surveillance: An updated architecture with data fusion”. In: *Proc. of the Sixth Int’l Conf. on Information Fusion.* Vol. 2. 2003, pp. 1324–1333.

[29] E. Lefebvre and C. Helleur. “Automated association of track information from sensor sources with non-sensor information in the context of maritime surveillance”. In: *Proc. of the Seventh Int’l Conf on Information Fusion.* Citeseer. 2004.

[30] J.-P. Mano, J.-P. Georgé, and M.-P. Gleizes. “Adaptive multi-agent system for multi-sensor maritime surveillance”. In: *Advances in Practical Applications of Agents and Multiagent Systems.* Springer, 2010, pp. 285–290.

[31] M. Riveiro and G. Falkman. “Interactive visualization of normal behavioral models and expert rules for maritime anomaly detection”. In: *The Sixth Int’l Conf. on Computer Graphics, Imaging and Visualization.* IEEE. 2009, pp. 459–466.

[32] J. Van Laere and M. Nilsson. “Evaluation of a workshop to capture knowledge from subject matter experts in maritime surveillance”. In: *The 12th Int’l Conf. on Information Fusion.* IEEE. 2009, pp. 171–178.

[33] M. Hall, S. Hall, and T. Tate. “Removing the HCI bottleneck: How the human computer interface (HCI) affects the performance of data fusion systems”. In: *Proc. 2000 Meeting of the MSS, Nat. Symp. Sensor and Data Fusion.* 2000, pp. 89–104.

[34] L. L. Constantine and L. A. Lockwood. *Software for use: A practical guide to the models and methods of usage-centered design.* Pearson Education, 1999.

[35] J. Han, J. Pei, and M. Kamber. *Data mining: Concepts and techniques.* Elsevier, 2011.

Appendix

Table A1. A list of some maritime open data sources available on the Internet

Organization name	Categories	NAR	AR		Provider
			FR	NFR	
European Maritime Safety Agency www.emsa.europa.eu/oil-recovery-vessels/vessel-technical-specifications.html	Maritime safety Prevention of pollution from ships	X			Portugal
ICC Commercial Crime Services www.icc-ccs.org/	Fraud in international trade	X		X	UK
International Association of Classification Societies www.iacs.org.uk/shipdata/default.aspx	Maritime safety Regulation	X		X	UK
International Group of P&I Clubs www.igpandi.org/Home	liability and insurance issues	X	X		UK
World Shipping Register www.world-register.org/	Ships information Ports information Companies information	X			–
Lloyd's Register Ships In Class www.lrshipsinclass.lrfairplay.com/default.aspx	Ships information		X		UK
International Telecommunication Union www.itu.int/ITU-R/index.asp?category=terrestrial&rlink=mars&lang=en	Ships information Addresses of accounting authorities, administrations which notify information Coasts information MMSI assigned to search and rescue aircraft MMSI assigned to AIS Aids to Navigation	X		X	Switzerland

Continued

			AR		
Organization name	Categories	NAR	FR	NFR	Provider
International Maritime Consultancy Specialists in VTS www.maritime-vts.co.uk/	Vessel traffic services and maritime organization	X			UK
Port Directory www.port-directory.com/	Ports information	X			UK
Equasis www.equasis.org/EquasisWeb/public/HomePage?fs=HomePage	Ships information Companies information		X		Portugal
Q88.COM www.q88.com/Home.aspx?c=1	Questionnaire generator Ships information	X		X	USA
InforMare www.informare.it/indexuk.htm	Shipping information	X			Italy
Paris Mou www.parismou.org/	Port State control	X			Netherlands
Tokyo Mou www.tokyo-mou.org/	Port State control	X			Japan
Australian Government Bureau of Meteorology www.bom.gov.au/	Weather, climate and water	X		X	Australia
Finnish Meteorological Institute en.ilmatieteenlaitos.fi/home	Weather, climate and water	X			Finland
Meteo France france.meteofrance.com/	Weather, climate and water	X		X	France
Earth Science Office NASA weather.msfc.nasa.gov/GOES/	Weather, climate and water	X			–

Continued

Organization name	Categories	NAR	AR		Provider
			FR	NFR	
The Weather Channel www.weather.com/	Weather, climate and water	X			–
Ocean Color WEB NASA oceancolor.gsfc.nasa.gov/	Weather, climate and water	X		X	–
Earth European Space Agency earth.esa.int/ers/eeo4.10075/atsr_med.html	Weather, climate and water		X	X	Italy
Weather BBC www.bbc.co.uk/weather/	Weather, climate and water	X			UK
Sailwx.info www.sailwx.info/	Live marin information	X			USA
Ship.gr www.ship.gr/	Ship brokers information Ship suppliers Companies information	X			–
American Bureau of Shipping www.eagle.org/eagleExternalPortalWEB/appmanager/absEagle/absEagleDesktop?_nfpb=true&_pageLabel=abs_eagle_portal_home_page	Classification Societies	X			USA
Det Norske Veritas www.dnv.com/	Classification Societies	X		X	Norway
Bureau Veritas Groups www.bureauveritas.com/wps/wcm/connect/bv_com/Group/Footer/Home/	Classification Societies	X		X	–
China Classifiaction Societies www.ccs.org.cn/en/index.htm	Classification Societies			X	China

Continued

Organization name	Categories	NAR	AR		Provider
			FR	NFR	
HELLENIC REGISTER OF SHIPPING www.hrs.gr/index.htm	Classification Societies	X			Greece
Nippon Kaiji Kyokai www.classnk.or.jp/hp/en/index.aspx	Classification Societies	X			Japan
Vesseltracker.com www.vesseltracker.com/en/VesselArchive.html	AIS Data Ships information	X	X		Germany
Digital Seas www.digital-seas.com/start.html	Ships information AIS Data	X	X		Germany
MarineTraffic.com www.marinetraffic.com/ais/	Ships information AIS Data	X			Greece
Shipspotting.com www.shipspotting.com/	Ships information AIS Data	X	X		–
International Maritime Organization www5.imo.org/SharePoint/mainframe.asp?topic_id=334&offset	Piracy reports	x			UK
Copenhagen Malmö Port www.cmport.com/	Port Authorities	X			Denmark
Port of Gothenburg www.portgot.se/prod/hamnen/ghab/dalis2b.nsf	Port Authorities	X			Sweden
Swedish Maritime Administration (Sjofartsverket) http://www.sjofartsverket.se/sv/	Pilotage Fairway Service Maritime Traffic Information Icebreaking Hydrograpy	X		X	Sweden

Continued

Organization name	Categories	NAR	AR		Provider
			FR	NFR	
	Maritime and Aeronautical Search and Rescue Seamen's Service				
Ports of Stockholm http://www.stockholmshamnar.se/	Port Authorities	X			Sweden
Port of Norrköping http://www.norrkoping-port.se/	Port Authorities	X			Sweden
Port of Helsinki http://www.portofhelsinki.fi	Port Authorities	X			Finland
Port of Tallinn http://www.ts.ee/	Port Authorities	X			Estonia
Genoa Port Authority www.porto.genova.it/index.php/en	Port Authorities	X			Italy
Port of Klaipeda www.portofklaipeda.lt/en.php	Port Authorities	X			Lithuania
Philippine Ports Authority www.ppa.com.ph/	Port Authorities	X			Philippine
Panama Canal Authority www.pancanal.com/eng/index.html	Port Authorities	X			USA
UK P&I CLUB	liability and insurance issues	X	X		UK

Continued

Organization name	Categories	NAR	AR		Provider
			FR	NFR	
www.ukpandi.com/					
The American Club www.american-club.com/	liability and insurance issues	X	X		USA
Steamship Mutual www.simsl.com/	liability and insurance issues	X	X		Bermuda
SKULD www.skuld.com/	liability and insurance issues	X			Norway
North of England P&I Association www.nepia.com/home/	liability and insurance issues	X			UK
The standard club www.standard-club.com/	liability and insurance issues	X		X	UK
Baltic Ports Organization www.bpoports.com/	Port coordinator	X		X	Denmark

Note. NAR = no authorization required; AR= authorization required; FR = free registration; NFR = non-free registration; Dashes indicate undisclosed information.

Table A2. Data sources that are used in the implementation

Website name	Data type
Marinetraffic.com	Real time information based on AIS systems[1]
Swedish Maritime Administration (Sjöfartsverket)	Stockholm pilotage area[2]
Ports of Stockholm, Kapellskär and Nynäshamn	Vessels in port and expected arrival[3]
Port of Norrköping	Vessels in port[4] Expected vessels arrival[5]
Port of Helsinki	Cargo vessels in port[6] Expected cargo vessels arrival[7] Expected passenger vessels departure[8] Expected passenger vessels arrival[9] Passenger vessels have visited the port before[10]
Port of Tallinn	Vessels in port[11] Expected cargo vessels arrival[12] Expected passenger vessels arrival[13] Expected passenger vessels departure[14]

[1] www.marinetraffic.com/ais/
[2] www.sjofartsverket.se/sv/Infrastruktur-amp-Sjotrafik/Lotsning/Lotsinfo/
[3] stockholmshamnar.se/en/Karta/Vessel-calls/
[4] www.norrkoping-port.se/anlop.php?page=snabb_fih&link=110|111
[5] www.norrkoping-port.se/anlop.php?page=snabb_fih_ank&link=110|111
[6] www.portofhelsinki.fi/cargo_traffic/vessels_in_ports
[7] www.portofhelsinki.fi/cargo_traffic/arrival_ships
[8] www.portofhelsinki.fi/passengers/departure_times_and_terminals
[9] www.portofhelsinki.fi/passengers/arrival_times_and_terminals
[10] www.portofhelsinki.fi/passengers/cruise_ships_that_have_visited_the_port
[11] www.ts.ee/?op=ships_in_port&lang=eng
[12] www.ts.ee/?op=cargo_ships_arrivals&lang=eng
[13] www.ts.ee/?op=passenger_ship_arrivals&lang=eng
[14] www.ts.ee/?op=passenger_ship_departures&lang=eng

Table A3. Total number of vessels in the surveillance area during one week of the system execution

	Count (Avg)[1]
Total number of vessels	673.29
Cargo, Tanker, Passenger and Ferry vessels	366.29
Cargo, Tanker, Passenger and Ferry vessels that are originated from or targeted to the specified ports	141.71

[1] The daily average count

Table A4. The average number of detected anomalies during one week of execution

Anomaly	Count (Avg)[1]
ARRIVAL_TIME_MISMATCHED	23.86
VESSEL_NOT_INFORMED_PORT	14.71
VESSEL_NOT_LEFT_PORT	8.29
UNUSUAL_TRIP_PATTERN	3.71
VESSEL_NOT_USED_PILOT	2.00
UNUSUAL_TRIP_PATTERN_AND_VESSEL_ARRIVAL_TIME_MISMATCHED	1.71
VESSEL_MOORED_IN_PORT	1.29
UNUSUAL_TRIP_PATTERN_AND_VESSEL_NOT_INFORMED_PORT	0.57
VESSEL_ENTERED_PORT_WITHOUT_NOTICE	0.29
VESSEL_NOT_ENTERED_PORT	0.29
UNDER_SURVEILLANCE_VESSEL	0.29
VESSEL_ARRIVAL_TIME_MISMATCHED_AND_VESSEL_NOT_USED_PILOT	0.29
VESSEL_ORDERED_PILOT_AND_NOT_INFORMED_PORT	0.14
UNUSUAL_TRIP_PATTERN_AND_VESSEL_NOT_USED_PILOT	0.14
VESSEL_ORDERED_PILOT_AND_NOT_INFORMED_PORT_AND_VESSEL_NOT_USED_PILOT	0.14
UNUSUAL_TRIP_PATTERN_AND_VESSEL_ORDERED_PILOT_AND_NOT_INFORMED_PORT	0.00
UNUSUAL_TRIP_PATTERN_AND_VESSEL_NOT_USED_PILOT_AND_VESSEL_ARRIVAL_TIME_MISMATCHED	0.00
VESSEL_ENTERED_PORT_WITHOUT_NOTICE_AND_NOT_LEFT_PORT_ON_TIME	0.00
VESSEL_NOT_ENTERED_PORT_AND_NOT_USED_PILOT	0.00
UNUSUAL_TRIP_PATTERN_AND_VESSEL_NOT_ENTERED_PORT	0.00
UNUSUAL_TRIP_PATTERN_AND_VESSEL_NOT_ENTERED_PORT_AND_VESSEL_NOT_USED_PILOT	0.00
UNUSUAL_TRIP_PATTERN_AND_VESSEL_ORDERED_PILOT_AND_NOT_INFORMED_PORT_AND_VESSEL_NOT_USED_PILOT	0.00

[1] The daily average count

8

Outlier Detection for Video Session Data Using Sequential Pattern Mining

Shahrooz Abghari, Veselka Boeva, Niklas Lavesson, Håkan Grahn, Jörgen Gustafsson, Junaid Shaikh

In: Association for Computing Machinery's Special Interest Group on Knowledge Discovery and Data Mining: Workshop On Outlier Detection De-constructed, 2018, London, UK.

Abstract

The growth of Internet video and over-the-top transmission techniques has enabled online video service providers to deliver high quality video content to viewers. To maintain and improve the quality of experience, video providers need to detect unexpected issues that can highly affect the viewers' experience. This requires analyzing massive amounts of video session data in order to find unexpected sequences of events. In this paper we combine sequential pattern mining and clustering to discover such event sequences. The proposed approach applies sequential pattern mining to find frequent patterns by considering contextual and collective outliers. In order to distinguish between the normal and abnormal behavior of the system, we initially identify the most frequent patterns. Then a clustering algorithm is applied on the most frequent patterns. The generated clustering model together with Silhouette Index are used for further analysis of less frequent patterns and detection of potential outliers. Our results show that the proposed approach can detect outliers at the system level.

8.1 Introduction

The Internet has transformed almost every aspect of human society by enabling a wide range of applications and services such as online video streaming. Subscribers of such services spend a substantial amount of time online to watch movies and TV shows. This has required online video service providers (OVSPs) to continuously improve their services and equipment to satisfy subscribers' high expectation. According to a study performed by Krishnan and Sitaraman [1], a 2-second delay in starting an online video program causes the viewers to start abandoning the video. For each extra second delay beyond that the viewers' drop-off rate will be increased by 5.8%. Thus, in order for OVSPs to address subscribers' needs it is important to monitor, detect, and resolve any issues or anomalies that can significantly affect the viewers when watching requested video programs. Analyzing massive amounts of video sessions for identifying such abnormal behaviors is like finding a needle in a haystack.

In this study, we use sequential pattern mining in order to analyze video data sequences from an over-the-top video service (a delivery paradigm that uses Internet to deliver video). The video session data has temporal order and contains detailed information regarding which video is requested, what type of device (mobile phone, PC, etc.) is used for watching the video, and the list of occurrences of all event types. The initial assumption with using sequential pattern mining is that frequent patterns can be considered as normal system behavior, while the others can be potential outliers. By applying a clustering method, most frequent patterns can be grouped based on their similarities. Finally, non-most frequent patterns can be evaluated by the created model and their goodness-of-fit identified by applying an internal cluster validation measure such as Silhouette Index [2].

The proposed approach is able to detect outliers by analyzing video event sequences and finding specific patterns that do not commonly occur. Investigating these unexpected patterns can assist online video service providers to identify, diagnose, and resolve possible system level issues. To the best of our knowledge, this is the first study that combines sequential pattern mining and clustering analysis for detecting outliers in online video streaming.

8.2 Background

8.2.1 Frequent Pattern Mining

The application of frequent itemset mining for market-basket analysis was first introduced by Agrawal et al. in 1993 [3]. The aim of such analysis is to reveal the customers' shopping habits and to find out which sets of products are frequently bought together. The frequent itemset mining can be formulated as follows: let $\mathcal{I} = \{i_1, i_2, ..., i_n\}$ be a set of all items and $\mathcal{T} = \{t_1, t_2, ..., t_j, ..., t_m\}$ a transaction database, where t_j is a set of items that has been bought by a customer ($t_j \subseteq \mathcal{I}$). The aim is to find those sets of items that occur frequently in most of the shopping baskets considering s, the user-specified *support threshold*.

The *support* for a *k-itemset* X, which consists of k items from $\mathcal{I}$, is the number of transactions that contain X as a subset, i.e., $_{ST}(X) = |\{t_j | X \subseteq t_j \wedge t_j \in \mathcal{T}\}|$. Note that the support of X can also be defined as the *relative support* which is the ratio of the number of transactions containing X to the total number of transactions in the database $\mathcal{T}$, i.e., $_{RelST}(X) = \frac{_{ST}(X)}{|\mathcal{T}|}$, such X is frequent if and only if its support is equal or greater than s.

Originally in frequent itemset mining, the order of items in the itemsets is unimportant. Looking at the market-basket analysis, the goal is to find frequent sets of items that are bought together. However, there are some situations in which the order of items inside the itemset is important such as sequence databases. A sequence database consists of ordered sequences of items listed with or without a concrete notion of time [4]. Sequential pattern mining, the problem of finding interesting frequent ordered patterns, was first introduced in 1995 [5].

Let $\mathcal{I} = \{i_1, i_2, ..., i_m\}$ be a set of all items. A sequence α defined as $\langle a_1, a_2, ..., a_j, ..., a_m \rangle$, where a_j is an itemset. Each itemset a_j represents a set of items that happened at the same time. A sequence $\alpha = \langle a_1, a_2, ..., a_m \rangle$ is a subsequence of $\beta = \langle b_1, b_2, ..., b_n \rangle$ if and only if there exist integers $1 \leq k_1 < k_2 < ... < k_m \leq n$ and $a_1 \subseteq b_{k_1}, a_2 \subseteq b_{k_2}, ..., a_m \subseteq b_{k_m}$ [5]. Given a sequence database $\mathcal{T} = \{s_1, s_2, ..., s_n\}$, the support for α is the number of sequences in $\mathcal{T}$ that contain α as a subsequence. Consequently, α is a frequent sequential pattern if its support is equal or greater than user-specified support threshold.

Mining frequent patterns in a large database can lead to generating a huge number of patterns that satisfy the user-specified support threshold. This is due to the fact that if a pattern is frequent, its sub-patterns are also frequent. To mitigate this problem, *closed* and *maximal* frequent pattern mining has been proposed [4]. A frequent pattern α is called:

1. a closed frequent pattern in the database $\mathcal{T}$ if and only if none of its

super-patterns have the same support as α,

2. a maximal frequent pattern in the database $\mathcal{T}$ if and only if none of its super-patterns is frequent [4, 6].

8.2.2 Sequential Pattern Mining Algorithms

Since the introduction of frequent itemset mining and the Apriori algorithm [3], several extensions of this algorithm were developed for both frequent itemset mining and sequential pattern mining. In general, there are two main categories of algorithms suitable for frequent pattern mining: 1) *Apriori-based algorithms* and 2) *Pattern-growth algorithms*. Additionally, from a frequent pattern mining point of view, a sequence database can represent the data either in a *horizontal data format* or *vertical data format* [7]. Therefore, based on these two data formats Apriori-based algorithms can expand to *horizontal data format algorithms* such as AprioriAll [5], and GSP [8] and *vertical data format algorithms* such as SPADE [9], and SPAM [10]. Apriori-based algorithms generate large sets of candidates and repeatedly scan the database for mining sequential patterns which require a lot of memory [11]. To solve this problem, pattern-growth approach as an extension of FP-growth algorithm [11] for frequent itemset mining without candidate generation was proposed. Pattern-growth algorithms such as FreeSpan [12], and PrefixSpan [13] work in a divide-and-conquer fashion and repeatedly divide the database into a set of smaller *projected databases* and mine them recursively.

The most popular pattern-growth algorithm is PrefixSpan. Given a sequence database, $\mathcal{T}$, and a user-specified threshold, *min_sup*, PrefixSpan applies a prefix-projection method to mine sequential patterns in $\mathcal{T}$ through 1) scanning the database once to find all frequent items with a length one, 2) dividing search space into a number of subsets according to the extracted frequent items in the previous step, and 3) constructing projected databases that represent each subset of sequential patterns and mining them recursively. This way, only local frequent sequences will be explored to create sequential patterns in each projected database [7, 13].

In 2004, Pei et al. [14] showed that PrefixSpan has the best overall performance compared to GSP and SPADE, and FreeSpan. Therefore, in this study we choose to use PrefixSpan for extracting sequential patterns in video data sessions.

8.3 Related Work

Barbará et al. [15] proposed an intrusion detection system that applies a frequent itemset technique to discover sets of items that are available in most data chunks. Using a clustering algorithm, these items that are considered as attack-free traffic, are divided into different groups based on their similarities. After creating the clusters, an outlier detection technique is applied to all the data points checking each instance against the set of clusters. Instances that do not belong to any clusters are presumed to be attacks. Recently, Rossi et al. [16] proposed an anomaly detection system for the smart grid domain similar to one considered in [15]. The method proposed by Rossi et al. uses frequent itemset mining on different event types collected from smart meters to separate normal and potential anomalous data points. For further evaluation, a clustering technique with Silhouette Index analysis is applied to detect anomalies.

Hoque et al. [17] developed an anomaly detection system for monitoring daily in-home activities of elderly people called *Holmes*. The proposed system learns a resident's normal behavior by considering variability of daily activities based on their occurrence time (e.g., day, weekdays, weekends) and applying a context-aware hierarchical clustering algorithm. Moreover, *Holmes* learns temporal relationships between multiple activities with the help of both sequential pattern mining and itemset mining algorithms. New scenarios can be added based on resident and expert's feedback to increase the accuracy of the system.

8.4 Methods and Technical Solutions

8.4.1 Problem Definition and a Use Case

Outlier detection refers to finding unexpected and abnormal patterns in data. The challenge in detecting outliers comes from the difficulty in defining a normal behavior, which includes the issue of labeling data [18]. Therefore, unsupervised learning methods or a combination of methods such as frequent pattern mining and clustering can be applied to analyze, understand and detect outliers. Finding unexpected patterns in video session data is challenging due to the scarcity of the labeled data.

We investigate a dataset of video sessions, where each video session

Table 8.1: Example of video sessions sorted by *Session ID* and *Date-time*

Session ID	Video ID	Date-time	Event type
1	002	Oct-01-16 22:44	client_roll
1	002	Oct-01-16 22:45	created
1	002	Oct-01-16 22:46	connectivity_changed
1	002	Oct-01-16 22:47	bitrate_switched
1	002	Oct-01-16 22:48	started
1	057	Oct-01-16 22:55	program_changed
1	057	Oct-01-16 23:22	pause
1	057	Oct-01-16 23:48	stopped
2	105	Oct-03-16 17:26	client_roll
2	105	Oct-03-16 17:27	created
2	105	Oct-03-16 17:28	connectivity_changed
2	105	Oct-03-16 17:29	bitrate_switched
2	105	Oct-03-16 17:30	bitrate_switched
2	105	Oct-03-16 17:31	bitrate_switched
2	105	Oct-03-16 17:32	stopped

consists of session ID, video ID, date and time of an occurring video event together with its type. The aim is to use frequent sequential pattern mining on sequences of video events to find unexpected or abnormal patterns of video events. Table 8.1 shows two examples of video sessions. Every session starts with a viewer logging into his/her account (*client_roll*), instantiating the video player (*created*) and ending with *stopped*. We denote an itemset i by $(i_1, i_2, ..., i_j ..., i_n)$, where each i_j is an item. Table 8.2 shows all the available event types that can appear in a video session together with their unique *ID*. A sequence α is an ordered list of itemsets and defined as $\langle a_1, a_2, ..., a_j, ..., a_m \rangle$, where each a_j is an itemset. In our case each itemset, a_j, is a singleton. Table 8.3 shows how the information of Table 8.1 can be summarized as a sequence of events for each viewer. Using the sequential pattern mining, we would like to find frequent sequential patterns in our data, group them into clusters based on their similarities, and then each infrequent sequential pattern can be analyzed and matched to these clusters to find normal and abnormal patterns.

Table 8.2: Event types and their corresponding IDs

Event ID	Event type	Event ID	Event type
1	bitrate_switched	8	paused
2	buffering_started	9	play
3	buffering_stopped	10	program_changed
4	client_roll	11	scrubbed
5	connectivity_changed	12	started
6	created	13	stopped
7	error_occurred		

Table 8.3: Example of video sessions with sequences of events

Session ID	Video ID	Date-time	Event seq
1	002,057	Oct-01-16 22:44	$\langle 4, 6, 5, 1, 12, 10, 8, 13 \rangle$
2	105	Oct-03-16 17:26	$\langle 4, 6, 5, 1, 1, 1, 13 \rangle$

Our use case relates to analyzing a sudden increase in the number of video streaming performance events during video sessions. Performance changes in video streams are often reflected by the re-buffering and quality adaptation events (*buffering_started*, *buffering_stopped*, and *bitrate_switched*). A sudden increase in occurrence of such events can be related to some kind of performance issues at the system level. Considering only the total number of re-buffering and bitrate adaptation events, however, may not be a true indicator of a sudden change in overall performance of the video sessions. It may happen that the number of initiated sessions surge during a certain time interval and that results in an increase in *buffering_started*, *buffering_stopped* and *bitrate_switched* events. This is because every session normally has some buffering and bitrate change events. However, what is more important for the OVSPs is to identify if such event types within sessions increase in number for many concurrent video sessions and for many users approximately at the same time.

8.4.2 Clustering Analysis

Cluster analysis is a process of partitioning a set of objects into groups of similar objects. That is, the objects within each cluster are similar to each other but dissimilar to objects in neighboring clusters [19].

In our experiment, we use two different clustering methods to partition the data, namely *k-means* [20] and *affinity propagation (AP)* [21]. The popular k-means algorithm begins by an initial set of randomly selected centroids. It then iteratively revises this set until the sum of squared errors are minimized. k-means requires the value of k, i.e., the number of clusters, as an input.

Affinity propagation, on the other hand simultaneously considers all data points as potential centroids and exchanges real-valued messages between data points until a good set of centroids and clusters appear. The exchanged messages represent either the suitability of one data point in comparison to others being the centroid (responsibility) or when one data point should choose a new centroid (availability). AP adapts the number of clusters based on the data. In comparison with k-means, the AP algorithm uses actual data points as the cluster's centroids.

8.4.3 Cluster Validation Measures

The cluster validation techniques can be regarded as important aids for interpreting partitioning solutions to find the one that best fits the underlying data. Cluster validation measures can be divided into two major categories: external and internal. External validation measures require the ground truth labels for providing an assessment of clustering quality. In case the ground truth labels are not known, internal validation measures can be used. Internal measures base their analysis on the same information used to create the model itself. In general, internal measures can be used to assess compactness, separation, connectedness, and stability of the clustering results [22]. A detailed overview of different clustering validation measures and their comparison can be found in [23, 24].

In this study we apply *Silhouette Index (SI)* [2] as an internal validation measure due to unavailability of the ground truth labels. SI can be applied to evaluate the tightness and separation of each cluster and it measures how well an object fits the available clustering. For each i, let $a(i)$ be the average dissimilarity of i to all other objects in the same cluster. Let us now consider $d(i, C)$ as an average dissimilarity of i to all objects of a cluster C. After computing $d(i, C)$ for all clusters, the one with the smallest average dissimilarity is denoted as $b(i)$. Such cluster also refer to *neighboring cluster* of i. The Silhouette Index score of i, $s(i)$, is obtained by combining $a(i)$ and $b(i)$ as follows:

$$s(i) = \frac{b(i) - a(i)}{max\{a(i), b(i)\}}$$

The $s(i)$ has values in a range of $[-1, 1]$. A score close to 1 implies that the object is well clustered. When $s(i)$ is about zero, this indicates the object is on the decision boundary between two neighboring clusters. The worst situation occurs when $s(i)$ is close to -1. This indicates that the object is misclassified and assigned to the wrong cluster.

The average $s(i)$ for all objects i belonging to the same cluster shows how tightly those objects are grouped. The average $s(i)$ for all objects i in the whole dataset judges the quality of the generated clustering solution.

8.4.4 Distance Measures

In order to calculate the similarity between the frequent patterns with different lengths, we study two different distance measures, namely *Fast Dynamic Time Warping (FastDTW)* algorithm [25] and *Levenshtein Distance (LD)* [26].

The FastDTW algorithm is able to detect an accurate optimal alignment between two time series and to find the corresponding regions between them. FastDTW reduces the resolution of the time series repeatedly with averaging adjacent pairs of points. Then it takes a minimum-distance warp path at a lower resolution and projects to a higher resolution. The projected warp path is refined and repeatedly projected onto incrementally higher resolutions until a full warp path is found.

The LD, also known as edit distance, is a string similarity metric that measures the minimum number of editing operations (insertion, deletion and substitution) required to change one string into the other.

As mentioned earlier, the video session data has a temporal order, which means that events can only appear in a special set-up. Therefore, FastDTW is chosen for comparison of the patterns. On the other hand, LD as an edit distance considers the elements' alignments of the patterns and the required changes to transform one into other. As an example consider these two patterns, $P_1 : \langle 1, 1, 1, 1, 1 \rangle$ and $P_2 : \langle 1, 1, 1 \rangle$. According to FastDTW, these two patterns are 100% similar since the measure assumes that P_2 is bent. However, LD would show that the similarity between the two patterns are 60% since the insertion of 2 extra 1's are needed to transform P_2 to P_1. From the point of view of time series analysis these two patterns are similar. However, from the video streaming performance point of view, repetition of re-buffering and quality adaptation events may represent performance issues, which in this scenario the result of the LD is more relevant. For this reason, in this study we evaluate the proposed approach with both distance measures (FastDTW and LD).

8.5 Proposed Approach

We combine frequent sequential pattern mining with clustering and Silhouette Index based analysis to detect unexpected patterns in online video data. Our approach can be found similar to Rossi et al.'s proposed method that has been applied for smart grid data in the district heating domain [16]. Both approaches deal with sequences of event types. However, instead of *itemset mining* we use *sequential pattern mining* due to the fact that the temporal order of occurrence of the video events is important.

To analyze video sessions for finding unexpected patterns at the system level the following steps are carried out:
1. *Data segmentation.* The video sessions are first divided into equal-sized segments based on the time period they are instantiated in order to identify sequential patterns. Data segmentation can be performed hourly, daily, and even weekly with different set-ups. For example, daily video sessions can be divided into four 6-hour period segments. Due to availability of daily patterns in the data, similar segments of similar days can be compared. We have conducted some initial experiments of our approach with bigger segment sizes, such as 2-days and weekly. However, additional evaluation and validation of these scenarios are needed to be able to make an informed conclusion about their significance for the approach performance. Therefore, in this paper, we have only considered a daily segment. One segment includes all video sessions that are initiated at the same time period.
2. *Frequent sequential patterns finding.* The PrefixSpan algorithm [13] is used to find frequent sequential patterns in each segment. The extracted patterns are stored in a list corresponding to each segment. These patterns can lead us to find *collective outliers*[1]. Note that we only use sequences of video events as inputs for the algorithm. Moreover, each video session has only one event sequence, such as $\langle 4, 6, 5, 1, 12, 10, 8, 13\rangle$ (see *Session ID 1* in Table 8.3). Those sequential patterns that satisfy the user-specified support will be stored as frequent patterns. In this study, the user-specified support threshold is set to be 0.15, which means any pattern that appears more than $(0.15 * size\_of\_the\_segment)$ times will be considered. The support threshold is tested with different sizes ranging between 0.1 to 0.2. By choosing values close to 0.1 many patterns are extracted which affects

[1] Collective outlier is a collection (sequence) of related data points that deviate significantly from the entire data set. Note that the individual data points in the sequence may or may not be outliers by themselves[18, 19].

the execution time dramatically. On the other hand, choosing values close to 0.2 ends up extracting very few patterns. However, by setting the support threshold to 0.15 we both decrease the execution time and gain a reasonable amount of patterns. Additionally, in order to decrease the computational time of the proposed approach, patterns with lengths less than 3 are omitted.
3. Frequent sequential patterns mapping. The list of extracted frequent sequential patterns is created for each segment in Step 2. Now, for each segment, the following steps will be carried out:

1. Select a pattern, one at a time, from the list of frequent sequential patterns and mark those video sessions that contain the pattern. Note that a video session can be matched with different patterns.

2. Store the date, the pattern(s) and its related length and frequency in the *selected_patterns* list (if the pattern does not match any video sessions its frequency will be set to 0). We can also add additional information here such as whether the pattern happened during working days, weekends or irregular days (e.g., public holidays), and what day-of-week, that can be helpful for finding a *contextual outlier*[2].

3. If not all patterns are selected, go back to 1 and select the next pattern.

After Step 3 the *selected_patterns* list contains the following details: 1) date, 2) pattern, 3) length of the pattern, 4) frequency of the pattern in the segment, 5) date-time information (e.g., day-of-week (Mon = 0, Tue = 2, ..., Sun = 6), and type-of-day (irregular day = 2, workday = 1, and weekend = 0)). Therefore, the *selected_patterns* list can represent one element according to Table 8.3 as [*date:* Oct-01-2016, *sequence:* $\langle 4, 6, 5, 1 \rangle$, *length:* 4, *frequency in segment:* 1, *day-of-week:* 6, *type-of-day:* 0].

4. Most frequent and non-most frequent patterns finding. At this step, we look for those sequential patterns that occurred in more than one segment, i.e., the *Most Frequent Sequential Patterns (MFSPs).* The initial assumption is that frequent patterns that appear in more than one segment can be considered as normal. *Non-most Frequent Sequential Patterns (NMFSPs)* on the other hand can be assumed as potentially unexpected at this stage.

[2] Contextual or conditional outlier is a data point that deviates significantly with respect to a specific context or condition [18, 19].

5. *MFSPs clustering.* The *selected_patterns* list summarizes detailed information regarding all video sessions. Then a clustering algorithm (e.g., k-means) can be used to group MFSPs into clusters. Note that since every video event has an ID (see Table 8.2), a sequential pattern such as $\{$*client_roll*, *created*,*connectivity_changed*, *bitrate_switched*$\}$ can be transformed to $\langle 4, 6, 5, 1 \rangle$.

6. *Analysis of NMFSPs and outlier detection.* The clustering model built in the previous step can be used to analyze the NMFSPs, i.e., by matching each NMFSP into the MFSPs clustering model we can evaluate how well it fits into the model. The goodness-of-fit of a NMFSP can be identified by applying some internal cluster validation measures such as Silhouette Index. That is, those NMFSPs with Silhouette scores, $s(i)$, less than the average $s(i)$ for the whole clustering solution can be defined as outliers. Note that $s(i)$ measures how well an object i, a NMFSP in our case, fits the available clustering and ranges from -1 to 1. The Silhouette score close to 1 implies that the pattern is well clustered. When s_i is about zero, this indicates the NMFSP is on the decision boundary between two neighboring clusters. An $s(i)$ close to -1 indicates that the NMFSP is misclassified and assigned to an erroneous cluster, i.e., such NMFSP can be identified as an outlier.

8.6 Empirical Evaluation

8.6.1 Data Collection

We used two months of data (October-November 2016) for initial evaluation of sequential pattern mining to find unexpected patterns in video sessions. The data is obtained from a large European telecommunication company and contains 202,312 unique video session IDs, 2,213,330 events, 13 event types and 47,938 videos. Table 8.4 summarizes detailed information about the data for each month.

Table 8.4: Summary of the data used in the experiment

	October 2016	November 2016
No. of video session IDs	114,407	87,905
No. of events	1,327,679	885,651
No. of video IDs	26,266	21,672
No. of Event types	13	13

Table 8.5: The results of the experiment

			Affinity Propagation		*k-means*	
			LD	FastDTW	LD	FastDTW
Oct 2016	*No. of MFSPs*	384				
	No. of NMFSPs	60				
	SI		0.149	0.170	**0.182**	**0.203**
	No. of clusters		32	33	22	22
	No. of detected outliers		33	31	**40**	**36**
	No. of days		2 (31)	2 (31)	2 (31)	2 (31)
	No. of matched video sessions / day		143 (4,359)	144 (4,359)	**144** (4,359)	**402** (4,359)
			372 (2,390)	**372** (2,390)	336 (2,390)	336 (2,390)
Nov 2016	*No. of MFSPs*	109				
	No. of NMFSPs	258				
	SI		0.175	0.192	**0.194**	**0.207**
	No. of clusters		14	14	12	12
	No. of detected outliers		120	144	**137**	**160**
	No. of days		1 (30)	1 (30)	1 (30)	1 (30)
	No. of matched video sessions / day		1,068 (3,705)	1,078 (3,705)	1,068 (3,705)	1,078 (3,705)

Note. Numbers inside the parentheses represent the total for both days and video sessions.

8.6.2 Experimental Design

The proposed approach is implemented in Python version 3.6. The Python implementation of PrefixSpan, LD and FastDTW algorithms are fetched from [27–29] respectively. The clustering algorithms are adopted from the scikit-learn module [30]. The implemented code and the experimental results are available at GitHub[3].

In this study, we have investigated the usage of two different distance measures namely, LD and FastDTW together with two clustering methods and sequential pattern mining for detecting outliers. The motivation behind this is due to the fact that these distance measures are able to capture different similarity characteristics between the two compared patterns (see the discussion in Section 4.4 for more details).

We use SI to determine the optimal number of clusters on the set of MFSPs. Namely, we have run k-means algorithm with a different number of clusters. Then we have used the SI as a validity index to identify the best

[3] https://github.com/shahrooz-abghari/Outlier-Detection-for-Video-Session-Data-Using-Sequential-Pattern-Mining

partitioning scheme. Figure 8.1 shows the average Silhouette scores for all k in the range between 2 and 35 using the LD (red color line) and FastDTW (blue color line) measures for data belongs to October 2016. The selected range is based on the number of clusters chosen by AP. We search for a local maximum of each plot that has a sudden change in order to identify the optimal k. The black box in Figure 8.1 shows the selected optimal $k = 22$, which is the same for both measures in October 2016. The optimal k for data belongs to November 2016, is 12. In addition, SI is also applied to analyze the NMFSPs.

8.7 Results and Analysis

The proposed approach is evaluated separately on data collected from October and November 2016. Two different clustering algorithms, AP and k-means together with LD and FastDTW are used for partitioning the MFSPs. The Silhouette Index is used to analyze NMFSPs on both clustering models. The results are presented in Tables 8.5, 8.6 and 8.7. As shown in

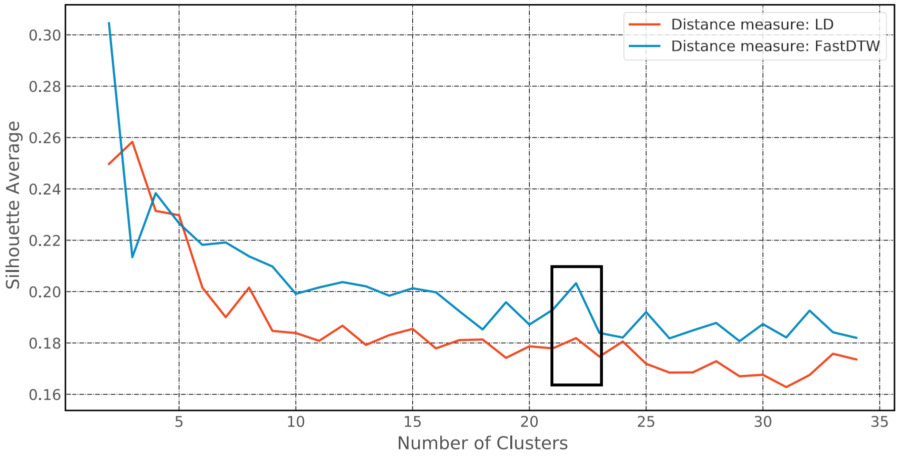


Figure 8.1: Identifying the optimal number of clusters for k-means using Silhouette Index for data belongs to October 2016. The black box shows the selected optimal k for the studied distance measures.

Table 8.5, the number of extracted patterns, both MFSPs and NMFSPs, for October compared to November varies considerably. In October, the daily segments contain higher number of MFSPs, i.e., 384 compared to November which is 109. On the other hand, the number of NMFSPs for November is 258, which is approximately 4 times larger than the extracted patterns for October. This is mainly because the total number of video sessions and the frequency of event types in October is larger compared to November.

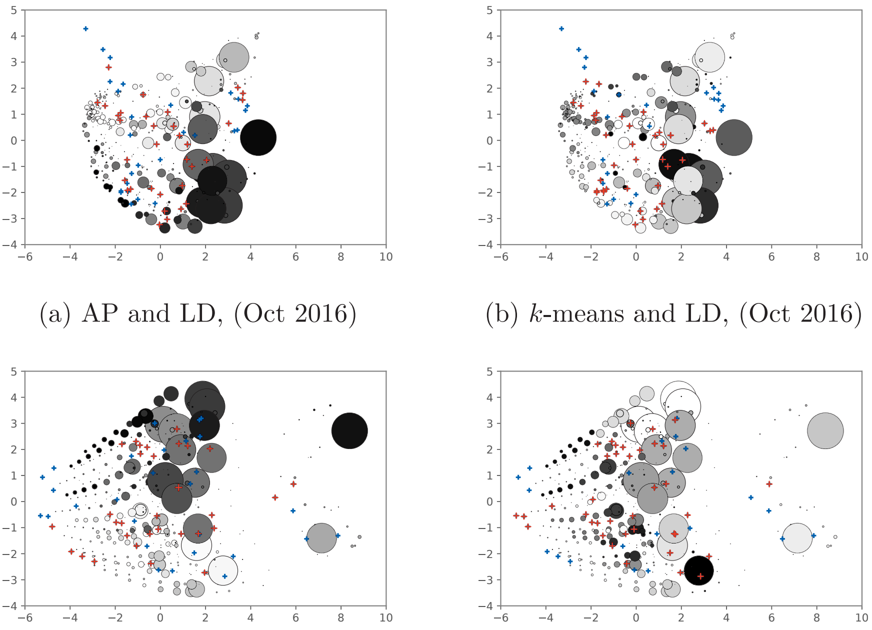
As presented in Table 8.5, during October and November both clustering algorithms detect outliers. In October we identify outliers in two days and in November only in one day. The combination of k-means algorithm with either of LD or FastDTW detected slightly more unexpected patterns compared to AP. The number of video sessions that matched with the detected outliers by the clustering algorithms are quite similar in both months except for the combination of k-means and FastDTW, which hits 402 video sessions in October. This perhaps relates to how each algorithm selects centroids of the clusters and tries to minimize the sum of squared errors. Table 8.5 also presents the number of video sessions that match with detected outliers. Overall, k-means matched more video sessions with the identified outliers during the months of October and November.

Table 8.6: Top 5 most frequent sequential patterns (MFSPs) relate to the daily segment

	Pattern	Oct 2016	Pattern	Nov 2016
MFSP	$\langle \mathbf{6}, \mathbf{4}, \mathbf{5}, \mathbf{1}, \mathbf{12}, \mathbf{8}, \mathbf{13} \rangle$	74,362	$\langle \mathbf{6}, \mathbf{4}, \mathbf{5}, \mathbf{1}, \mathbf{12}, \mathbf{8}, \mathbf{13} \rangle$	59,340
	$\langle \mathbf{6}, \mathbf{4}, \mathbf{5}, \mathbf{1}, \mathbf{12} \rangle$	12,061	$\langle \mathbf{6}, \mathbf{4}, \mathbf{5}, \mathbf{1}, \mathbf{12} \rangle$	19,104
	$\langle 6, 4, 5, 1, 12, 1 \rangle$	6,491	$\langle \mathbf{6}, \mathbf{4}, \mathbf{5}, \mathbf{1}, \mathbf{12}, \mathbf{8} \rangle$	4,449
	$\langle \mathbf{6}, \mathbf{4}, \mathbf{5}, \mathbf{1}, \mathbf{12}, \mathbf{8} \rangle$	5,057	$\langle 6, 5, 1, 12, 8, 13 \rangle$	1,098
	$\langle 6, 4, 5, 1, 12, 1, 8, 13 \rangle$	4,025	$\langle 6, 5, 1, 12 \rangle$	503
Total matched patterns		101,996		84,494

Note. Highlighted patterns represent those that occur in both Oct and Nov 2016.

The results of the top five most frequent sequential patterns for both October and November 2016 are presented in Table 8.6. These patterns relate to the daily segment. These patterns are matched with the majority of the video sessions (101,996 out of 114,407 and 84,494 out of 87,905 matched video sessions for October and November 2016, respectively). Most of these patterns begin with created ($ID = 6$) and client_roll ($ID = 4$) followed by connectivity_changed ($ID = 5$), bitrate_switched ($ID = 1$) and started ($ID = 12$) events. These sequences of the video events are the most common ones. Moreover, three of these sequences contain paused ($ID = 8$)



(a) AP and LD, (Oct 2016)

(b) *k*-means and LD, (Oct 2016)

(c) AP and FastDTW, (Oct 2016)

(d) *k*-means and FastDTW, (Oct 2016)

Figure 8.2: The visualization of the data is preformed by applying *Principal Component Analysis (PCA)* to convert the multi-dimensional dissimilarity matrices into 2-dimensional arrays. Therefore, no labels for axes are given. Each sphere represents one MFSP. The size of a sphere shows the number of video sessions that have been matched with it. Spheres with the same color belong to one cluster. The NMFSPs are shown with the blue pluses (" + ") and those NMFSPs that are identified as outliers are the red pluses (" + ").

and stopped ($ID = 13$), which represent a complete video session that begins with a viewer's login and ends with *stopped*. The bold patterns in Table 8.6 represent those patterns that occur in both months and in the case of MFSPs they cover a high proportion of the video sessions.

Table 8.7 shows the top 5 NMFSPs detected as outliers by clustering algorithms. There are two patterns detected with both AP and *k*-means in October. The first pattern contains bitrate_switched ($ID = 1$), buffering_started ($ID = 2$), followed by two bitrate_switched ($ID = 1$) events. The second pattern contains client_roll ($ID = 4$) followed by connectivity_changed ($ID = 5$), bitrate_switched ($ID = 1$), started

$(ID = 12)$ and program_changed $(ID = 10)$ events. There are three bold patterns, only one detected by AP and two by k-means. The pattern $\langle 1, 12, 10, 8\rangle$, which is detected by k-means is a sub-pattern of $\langle 1, 12, 10, 8, 13\rangle$ detected by AP and they are quite similar. However, this pattern $\langle 1, 2, 1, 1\rangle$ detected by k-means using FastDTW is interesting mostly because it has repetition of bitrate_switched $(ID = 1)$ and matched with 256 video sessions. In general, every video session contains a number of re-buffering and bitrate switched. However, any increase in the quantity of such event types for many viewers can be related to performance issues. This follows the definition of a collective outlier, i.e., an unexpected collection or sequence of related event types (data points) occurring together. Nevertheless, more investigation needs to be performed to find the reason of these issues. For November, both the clustering algorithms detect the same number of patterns.

Table 8.7: Top 5 non-most frequent sequential patterns (NMFSPs) detected as outliers for each month

		Affinity Propagation		*k-means*	
		LD	FastDTW	LD	FastDTW
Oct 2016	$\langle \mathbf{1}, \mathbf{2}, \mathbf{1}, \mathbf{1}\rangle$	-	-	-	256
	$\langle 1, 2, 1, 1, 8, 13\rangle$	101	101	101	101
	$\langle \mathbf{1}, \mathbf{12}, \mathbf{10}, \mathbf{8}\rangle$	-	-	148	148
	$\langle \mathbf{1}, \mathbf{12}, \mathbf{10}, \mathbf{8}, \mathbf{13}\rangle$	136	136	-	-
	$\langle 4, 5, 1, 12, 10\rangle$	152	152	140	140
	Total matched patterns	389	389	389	645
Nov 2016	$\langle 1, 1, 1, 1, 1\rangle$	513	513	513	513
	$\langle 1, 2, 1\rangle$	92	92	92	92
	$\langle 1, 1, 1, 2\rangle$	67	67	67	67
	$\langle 3, 8, 13\rangle$	66	66	66	66
	$\langle 1, 1, 1, 1, 2\rangle$	53	53	53	53
	Total matched patterns	791	791	791	791

Note. ’-’ means unavailable.

The results of applying the proposed approach on data belonging to October are visualized in Figure 8.2. Both AP and k-means detect outliers in two weekdays. In all plots each sphere represents one MFSP. The size of a sphere shows the number of video sessions that have been matched with it. The spheres with the same color belongs to one cluster. The NMFSPs are shown with blue ”$+$” and the detected outliers displayed with red ”$+$”. Principal Component Analysis (PCA) is used to transform the multi-dimensional dissimilarity matrices created by distance measures into

2-dimensional arrays. Plot (a) shows the results of AP using LD measure. As it is shown in Table 5, AP partitioned the MFSPs into 32 clusters. The results of k-means using LD measure is shown in (b). The size of k is set to be 22 and 40 NMFSPs out of 60 are marked as outliers. The plots (c) and (d) present the results of AP (no. of clusters = 33) and k-means (no. of clusters = 22) using FastDTW measure, respectively. Using FastDTW measure, AP and k-means identified 31 and 36 outliers, respectively.

Using LD measure, it appears the partitioned MFSPs especially the bigger spheres, are condensed in clumps, surrounding the NMFSPs. However, with the FastDTW algorithm, MFSPs seem to be stacked vertically and the NMFSPs extend across the 2D space in a horizontal trend. These differences reflect how each distance measure calculates the dissimilarity between two patterns. As we mentioned earlier in Section 4.4, LD is more sensitive when part of one pattern is a sub-pattern of the other with different length. Nevertheless, using different distance measures together with 3D visualization techniques can provide a better understanding of the underlying organization of the data for OVSPs.

8.8 Discussion

Outlier detection approaches can assist the online video service providers to monitor and improve the quality of their services. In general, finding outliers in online video data without having a clear definition of normal behavior is a challenging task. Hybrid approaches combining sequential pattern mining and clustering analysis, as it has been demonstrated in this study, can be useful in detecting unexpected sequences of events. In this study, video session data contains 13 unique event types (see Table 8.2 for more details). These event types are quite general and most of them can appear in both video sessions with *good* and *bad* quality. This makes it hard to draw any conclusions about the detected outliers without experts' validation. However, looking at the ratio of the quality related events can assist us to judge the quality of the video sessions. For example, by being able to identify a sudden increase of re-buffering and bitrate switch events, one may prevent users from having an unsatisfactory experience. The proposed approach has been evaluated with two months of data supplied by a large European telecommunication company. A number of outliers have been identified and matched with the studied use case, which analyzes a sudden increase of the video streaming performance events.

Perhaps it is worth to further study whether the different length of segments (see the discussion in Section 5) can affect the performance of the proposed outlier detection approach. In addition, it will be interesting to take into account the time interval between occurrences of the events in the evaluation set-up.

Furthermore, it is worthwhile to mention that not every pattern created by the sequential pattern mining algorithms can be useful. Although, sequential pattern mining searches for ordered sequences of events that are frequently happening together, some of the sequences might not be matched with any video sessions. The reason is that some of the events of the sequence are not available and pattern matching will not work. The importance of matching the extracted patterns with video sessions is due to the fact that we are trying to identify both the unexpected patterns and those sessions that are affected. Therefore, to ensure that no information will be lost, we plan to further study the combination of sequential pattern mining with frequent itemset mining.

8.9 Conclusion and Future Work

In this study, we have presented a hybrid approach for online video streaming by combining sequential pattern mining and clustering analysis to detect outliers at the system level. In addition, the usage of two different distance measures have been evaluated. In comparison to other studies that often apply statistical analysis to find outliers, we have looked for unexpected patterns that can have impact on the video streaming performance.

By applying this approach, online video service providers can easily monitor suspicious video sessions and capture a better understanding about the viewers' experience.

For future work, we aim to pursue further evaluation and validation of our approach on a variety of datasets by applying alternative clustering analysis techniques, e.g., graph-based clustering approaches and different validation measures. Our future plans also involve integrating additional information into the analysis of non-most frequent sequential patterns supplied by the domain experts.

References

[1] S. S. Krishnan and R. K. Sitaraman. “Video stream quality impacts viewer behavior: Inferring causality using quasi-experimental designs”. In: *IEEE/ACM Transactions on Networking* 21.6 (2013), pp. 2001–2014.

[2] P. J. Rousseeuw. “Silhouettes: A graphical aid to the interpretation and validation of cluster analysis”. In: *J. of Computational and Applied Mathematics* 20 (1987), pp. 53–65.

[3] R. Agrawal, T. Imieliński, and A. Swami. “Mining association rules between sets of items in large databases”. In: *Acm sigmod record.* Vol. 22. 2. 1993, pp. 207–216.

[4] J. Han, H. Cheng, D. Xin, and X. Yan. “Frequent pattern mining: Current status and future directions”. In: *Data Mining and Knowledge Discovery* 15.1 (2007), pp. 55–86.

[5] R. Agrawal and R. Srikant. “Mining sequential patterns”. In: *Proc. of the 11th Int’l Conf. on Data Engineering.* IEEE. 1995, pp. 3–14.

[6] C. Borgelt. “Frequent item set mining”. In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 2.6 (2012), pp. 437–456.

[7] W. Shen, J. Wang, and J. Han. “Sequential pattern mining”. In: *Frequent Pattern Mining.* Springer, 2014, pp. 261–282.

[8] R. Srikant and R. Agrawal. “Mining sequential patterns: Generalizations and performance improvements”. In: *Advances in Database Technology–EDBT’96* (1996), pp. 1–17.

[9] M. J. Zaki. “SPADE: An efficient algorithm for mining frequent sequences”. In: *Machine Learning* 42.1 (2001), pp. 31–60.

[10] J. Ayres, J. Flannick, J. Gehrke, and T. Yiu. “Sequential pattern mining using a bitmap representation”. In: *Proc. of the Eighth ACM SIGKDD Int’l Conf. on Knowledge Discovery and Data Mining.* 2002, pp. 429–435.

[11] J. Han, J. Pei, and Y. Yin. “Mining frequent patterns without candidate generation”. In: *ACM sigmod record.* Vol. 29. 2. 2000, pp. 1–12.

[12] J. Han, J. Pei, B. Mortazavi-Asl, Q. Chen, U. Dayal, and M.-C. Hsu. "FreeSpan: Frequent pattern-projected sequential pattern mining". In: *Proc. of the Sixth ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining.* 2000, pp. 355–359.

[13] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M. Hsu. "Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth". In: *Proc. of the 17th Int'l Conf. on Data Engineering.* 2001, pp. 215–224.

[14] J. Pei, J. Han, B. Mortazavi-Asl, J. Wang, H. Pinto, Q. Chen, U. Dayal, and M.-C. Hsu. "Mining sequential patterns by pattern-growth: The PrefixSpan approach". In: *IEEE Transactions on Knowledge and Data Engineering* 16.11 (2004), pp. 1424–1440.

[15] D. Barbará, Y. Li, J. Couto, J.-L. Lin, and S. Jajodia. "Bootstrapping a data mining intrusion detection system". In: *Proc. of the 2003 ACM Symp. on Applied computing.* 2003, pp. 421–425.

[16] B. Rossi, S. Chren, B. Buhnova, and T. Pitner. "Anomaly detection in Smart Grid data: An experience report". In: *Int'l Conf. on Systems, Man, and Cybernetics.* IEEE. 2016, pp. 002313–002318.

[17] E. Hoque, R. F. Dickerson, S. M. Preum, M. Hanson, A. Barth, and J. A. Stankovic. "Holmes: A comprehensive anomaly detection system for daily in-home activities". In: *Int'l Conf. on Distributed Computing in Sensor Systems.* IEEE. 2015, pp. 40–51.

[18] V. Chandola, A. Banerjee, and V. Kumar. "Anomaly detection: A survey". In: *ACM Computing Surveys* 41.3 (2009), p. 15.

[19] J. Han, J. Pei, and M. Kamber. *Data mining: Concepts and techniques.* Elsevier, 2011.

[20] J. MacQueen et al. "Some methods for classification and analysis of multivariate observations". In: *Proc. of the Fifth Berkeley Symp. on Mathematical Statistics and Probability.* Vol. 1. 14. Oakland, CA, USA. 1967, pp. 281–297.

[21] B. J. Frey and D. Dueck. "Clustering by passing messages between data points". In: *Science* 315.5814 (2007), pp. 972–976.

[22] A. K. Jain and R. C. Dubes. "Algorithms for clustering data". In: Englewood Cliffs, NJ, 1988.

[23] L. Vendramin, R. J. Campello, and E. R. Hruschka. “On the comparison of relative clustering validity criteria”. In: *Proc. of the 2009 SIAM Int’l Conf. on Data Mining*. SIAM. 2009, pp. 733–744.

[24] M. Halkidi, Y. Batistakis, and M. Vazirgiannis. “On clustering validation techniques”. In: *J. of Intelligent Information Systems* 17.2-3 (2001), pp. 107–145.

[25] S. Salvador and P. Chan. “Toward accurate dynamic time warping in linear time and space”. In: *Intelligent Data Analysis* 11.5 (2007), pp. 561–580.

[26] V. I. Levenshtein. “Binary codes capable of correcting deletions, insertions, and reversals”. In: *Soviet physics doklady*. Vol. 10. 8. 1966, pp. 707–710.

[27] C. Gao. “PrefixSpan algorithm source code”. In: (2015). URL: `https://github.com/chuanconggao/PrefixSpan-py`.

[28] B. Jain. “Edit distance algorithm source code”. In: (-). URL: `https://www.geeksforgeeks.org/dynamic-programming-set-5-edit-distance`.

[29] K. Tanida. “FastDTW algorithm source code”. In: (2017). URL: `https://github.com/slaypni/fastdtw`.

[30] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. “Scikit-learn: Machine Learning in Python”. In: *J. of Machine Learning Research* 12 (2011), pp. 2825–2830.

9

A Minimum Spanning Tree Clustering Approach for Outlier Detection in Event Sequences

Shahrooz Abghari, Veselka Boeva, Niklas Lavesson, Håkan Grahn, Selim Ickin, Jörgen Gustafsson

In: 2018 17th IEEE International Conference on Machine Learning and Applications (pp. 1123-1130).
DOI: 10.1109/ICMLA.2018.00182.

Abstract

Outlier detection has been studied in many domains. Outliers arise due to different reasons such as mechanical issues, fraudulent behavior, and human error. In this paper, we propose an unsupervised approach for outlier detection in a sequence dataset. The proposed approach combines sequential pattern mining, cluster analysis, and a minimum spanning tree algorithm in order to identify clusters of outliers. Initially, the sequential pattern mining is used to extract frequent sequential patterns. Next, the extracted patterns are clustered into groups of similar patterns. Finally, the minimum spanning tree algorithm is used to find groups of outliers. The proposed approach has been evaluated on two different real datasets, i.e., smart meter data and video session data. The obtained results have shown that our approach can be applied to narrow down the space of events to a set of potential outliers and facilitate domain experts in further analysis and identification of system level issues.

9.1 Introduction

Outlier detection has been studied and used to detect anomalous behavior in different domains. Outliers refer to data points that are significantly different from the rest of populations. They can happen due to mechanical issues, fraudulent behavior, human error and if they are not identified may lead to uncontrollable situations. Outlier detection refers to the problem of finding unusual patterns in data or unknown behaviors in a system [1, 2].

In this paper, we deal with outlier detection in sequence datasets. A sequence dataset is a collection of sequences of events or elements listed, often with concrete notion of time [3]. Due to importance of the sequential ordering of the events, sequential pattern mining for finding interesting subsequences in sequence datasets was introduced in 1995 [4]. Sequential pattern mining has a broad application in different fields such as bio-informatics [5, 6], marketing [7], network security [8], telecommunication [9, 10], and text mining [11, 12]. Data in these domains is highly dimensional and sparse which makes the identification of outliers more complex [13].

In this study, an outlier is defined as a small cluster (with respect to the number of matched sequences in a sequence dataset) which is significantly different from most of the frequent sequential patterns.

We propose an unsupervised 3-step approach that applies sequential pattern mining, cluster analysis, and a minimum spanning tree (MST) algorithm on a sequence dataset. In the first step, sequential pattern mining is used to extract frequent sequential patterns from data. In the second step, a clustering algorithm is applied on the extracted patterns in order to group the similar patterns together. Partitioning the patterns makes it possible in the third step to identify groups of patterns as outliers rather than detecting outliers individually. Consequently, this can lead to time complexity reduction in the proposed approach. In the third step, similar to [14], a minimum spanning tree is built on the clustering solutions in order to find clusters of outliers. By removing the longest edge(s) of the MST, the tree will be transformed to a forest. The small sub-tree(s) with few number of clusters (nodes) and/or with smaller sized clusters can be identified as outliers. The initial assumption is: the sub-trees with fewer nodes and smaller size contain patterns that happen rarely. Therefore, the clusters in these sub-trees are small, far and different from the clusters in the bigger sub-trees. The process of removing the longest edge(s) of the MST can also be performed by considering a user-specified threshold. The detected clusters of outliers can supply domain experts with a better understanding of the system behavior and facilitate them in the further analysis by mapping the detected patterns to the corresponding sequences. The proposed approach has been evaluated on smart meter data and video session data. The results of the evaluation on video session data has been discussed with the domain experts.

9.2 Background and Related Work

Outlier detection techniques have been studied and successfully applied in different domains. There exists a considerable number of studies that provide a comprehensive, and structured overview of the state-of-the-art methods and applications for outlier detection [1, 2, 15, 16]. This attention shows the importance of outlier detection techniques in different domains and the fact that they are domain-specific.

Outlier detection techniques can be classified into three groups based on the availability of the labeled data [1, 2]: **1)** In the absence of prior knowledge of the data, unsupervised learning methods are used to determine outliers. The initial assumption is that normal data represents a significant portion of the data and is distinguishable from faults or error; **2)** In the presence of labeled data, both normality and abnormality are modeled. This approach refers to supervised learning; **3)** Define what is normal and only model normality. This approach is known as semi-supervised learning since the algorithm is trained by labeled normal data, however, it is able to detect outliers or abnormalities. Semi-supervised outlier detections are more widely used compared to supervised techniques due to an imbalanced number of normal and abnormal labeled data.

The most common unsupervised learning method is cluster analysis, which is used for exploratory data analysis and grouping similar data into clusters. There are several clustering algorithms capable of detecting noise and eliminating it from the clustering solution such as DBSCAN [17], CRUE [18], ROCK [19], and SNN [20]. Even though such techniques can be used to detect outliers, the main aim of the clustering algorithm is to perform the partitioning task rather than identifying outliers.

This led to proposing clustering-based techniques that are capable of detecting: **1)** single-point outliers such as the application of Self Organizing Maps for clustering normal samples and identifying anomalous samples[21], and Expectation Maximization [22] for identifying the performance problems in distributed systems or **2)** groups of outliers such as the intrusion detection proposed by [23].

The application of MST has been studied by researchers in different fields including cluster analysis and outlier detection [14, 24–27]. A two-phase clustering algorithm is introduced for detecting outliers by [14]. In the first phase, a modified version of the k-means algorithm is used for partitioning the data. The modified k-means creates $k+i$ clusters, i.e., if a new data point is far enough from all clusters (k, number of clusters defined by the user), it will be considered as a new cluster (the $(k+i)^{th}$ cluster where, $i > 0$). In the second phase, an MST is built where, the tree's nodes represent the center of each cluster and edges show the distance between nodes. In order to detect outliers, the longest edges of the tree are removed. The sub-trees with a few number of clusters and/or smaller clusters are selected as outliers.

Wang et al. [24] developed an outlier detection by modifying k-means for constructing a spanning tree similar to an MST. The longest edges of the tree are removed to form the clusters. The small clusters are regarded as potential ouliters and ranked by calculating a density-based outlying factor.

A spatio-temporal outlier detection for early detection of critical events such as flood through monitoring a set of meteorological stations is introduced in [27]. Using geographical information of the data, a Delaunay triangulation network of the stations is created. The following step limits the connection of nodes to their closest neighbors while preventing far nodes from being linked directly. In the next step, an MST is constructed out of the created graph. In the final step, the outliers are detected by applying two statistical methods to detect exactly one or multiple outliers.

In this study, we propose a 3-step outlier detection approach that is specifically developed for sequence datasets. In the first step, frequent sequential patterns are extracted. In the second step, the selected patterns are clustered using the affinity propagation (AP) algorithm. In the third step, a minimum spanning tree similar to the study of Jiang et al. [14] is used to identify small groups of clusters as outliers.

9.3 Methods and Techniques

9.3.1 Sequential Pattern Mining

Sequential pattern mining is the process of finding frequently occurring patterns in a sequence dataset. The records of the sequence dataset contain sequences of events that often have chronological order. As examples of sequence data we can refer to customer shopping sequences, biological sequences, and video session events sequences.

Let $\mathcal{I} = \{i_1, i_2, ..., i_m\}$ be a set of all items. A sequence α defined as $\langle a_1, a_2, ..., a_j, ..., a_m \rangle$, where a_j is an itemset. Each itemset a_j is a subset of $\mathcal{I}$ that its items happened at the same time. In this study, each itemset (a_j) is a singleton. A sequence $\alpha = \langle a_1, a_2, ..., a_m \rangle$ is a subsequence of $\beta = \langle b_1, b_2, ..., b_n \rangle$ if and only if there exist integers $1 \leq k_1 < k_2 < ... < k_m \leq n$ and $a_1 \subseteq b_{k_1}, a_2 \subseteq b_{k_2}, ..., a_m \subseteq b_{k_m}$ [4]. Given a sequence dataset $\mathcal{T} = \{s_1, s_2, ..., s_j, ..., s_n\}$, where s_j is a sequence of itemsets, the support for α is the number of sequences that contain α as a subsequence. A sequence α is called a frequent sequential pattern if its support is equal or greater than user-specified support threshold.

To extract frequent sequential patterns the PrefixSpan algorithm [28] is used. PrefixSpan applies a *prefix-projection* method to find sequential patterns. Given a sequence dataset $\mathcal{T}$ and a user-specified threshold, the dataset is first scanned in order to identify all frequent items in sequences. All of these frequent items are considered as length-1 sequential pattern. After that, the search space is divided into a number of subsets based on the extracted prefixes. At last, for each subset a corresponding *projected dataset* is created and mined recursively.

Pei et al. [29] show in their study that PrefixSpan has the best overall performance compared to other sequential pattern mining algorithms such as GSP [30] and SPADE [31]. Therefore, the PrefixSpan algorithm is used for extracting sequential patterns in this study.

9.3.2 Similarity Measure

In order to calculate the similarity between the frequent sequential patterns with different lengths, we use *Levenshtein distance (LD)* metric [32]. The LD, also known as edit distance, is a string similarity metric that measures the minimum number of editing operations (insertion, deletion and substitution) required to change one string into the other. We used normalized LD, i.e., the result of each comparison is normalized by dividing it with the length of the longest pattern. The score ranges between 0 and 1. Score 0 implies 100% similarity between the two patterns and 1 represents no similarity. LD is a simple algorithm capable of measuring the similarity between patterns with different lengths. In this study since the extracted patterns can have different length we choose to use LD as a similarity measure.

9.3.3 Clustering Method

The extracted patterns are clustered by using affinity propagation algorithm [33]. AP works based on the concept of exchanging messages between data points until a good set of exemplars (the most representative of a cluster) and corresponding clustering solution appears. The exchanged messages at each step assists AP to choose the best samples as exemplars and which data points should choose those samples to be their exemplars. AP adapts the number of clusters based on the data. However, the number of clusters can be controlled by *preference* parameter. That is, a high value of the preference will cause AP to form many clusters, while a low value will lead to a small number of clusters. If the preference value is not provided the median similarity or the minimum similarity will be used.

Unlike most clustering algorithms such as k-means that requires the number of clusters as an input, AP is able to estimate the number of clusters based on the data provided. AP can create clusters of different shapes and sizes, and the exemplars (the selected data points) are the representative of the clusters [34]. These characteristics make AP a suitable clustering algorithm for the chosen datasets in this study.

9.3.4 The Proposed Approach

The proposed approach consists of a preprocessing step (*Data segmentation*) and 3 main steps: 1) *Sequential patterns mining*, 2) *Frequent sequential pattern clustering*, and 3) *MST building and outlier detection analysis* as follows:

0) ***Data segmentation.*** The data is first partitioned into equal-sized segments in order to identify sequential patterns. Due to availability of daily patterns in the data, similar segments of similar days can be

compared. In this paper, we have studied a daily segment.

1. ***Sequential patterns mining.*** The first step concerns the extraction of frequent sequential patterns and mapping them with records of a sequence dataset.

    a) ***Frequent sequential patterns finding.*** The PrefixSpan algorithm is used to find frequent sequential patterns from each segment. The extracted patterns can lead us to find sequences of events that based on their occurrence together assumed to be anomalous, also known as *collective* outliers [1]. Those sequential patterns that satisfy the user-specified support threshold will be stored as frequent patterns. Note that in sequential patterns the order of the events is important and by using the sequential pattern mining both the frequency and the order of events in the extracted patterns are taken into account.

    b) ***Frequent sequential patterns mapping.*** In the second step the extracted patterns are mapped with the source they come from and stored in a *selected_patterns* list. This relates to identifying those video sessions that contain the patterns or a device that the patterns are extracted from. The following step can lead us to find additional information about patterns such as pattern frequency and its occurrence time. The latter is also useful for finding a *contextual* or *conditional* outlier, i.e., a data point assumed to be anomalous only in a specific context [1].

2. ***Frequent sequential pattern clustering.*** Using the Levenshtein distance measure, as explained in sub-section *B*, the pairwise similarities between all patterns are calculated and the similarity matrix are constructed. The selected patterns are partitioned by applying affinity propagation on the created similarity matrix. Note, an advantage of using AP is that it can estimate the number of clusters from data.

3. ***MST building and outlier detecting.*** The third step includes two sub-steps concerning the construction of an MST, and the identification of sub-trees with the smallest size as outliers.

a) ***MST building.*** The exemplars of the clusters are used for building a complete weighted graph where vertices of the graph are exemplars of the clusters and edges are the distance between them (traversing weight). Using the MST algorithm, the aim is to determine a sub-set of edges that connect all the vertices together without any cycles that has the minimum total edge weight.

b) ***Outlier detection analysis.***

   i. The longest edge of the tree is removed. Note that there can be more than one edge to cut.

   ii. The constructed MST will be replaced by the created sub-trees, i.e., a single tree becomes a forest. Note that step (ii) can be repeated until the distance between nodes of the sub-trees become less or equal to a user-specified threshold. For example the threshold can be set to 0.5.

   iii. The sub-trees are ranked from smallest to largest based on the number of items they match within the sequence dataset. Following the definition of outliers that refers to patterns that happen rarely and sufficiently far away from other patterns [1], here the smallest sub-trees can be regarded as outliers.

9.4 Experimental Methodology

9.4.1 Datasets

The proposed approach is evaluated on two datasets namely, *smart meter* and *video session*. The first dataset contains smart meters recorded events data provided by Elektro Ljubljana, a power distribution company in Slovenia [35]. The provided data contains 10,739,273 records that logged different events generated by 117,944 smart meters between May and August 2017. Due to the high number of data, we only considered data from May 2017 and sampled 30 devices out of 85,776 without replacement. To decrease the chance of any bias two datasets are created through sampling and the experiment ran on each set separately. This led to selection of 28 unique devices for each dataset, i.e., in total 56 distinct devices and 2 identical devices are sampled. The event sequences generated by the sampled devices for datasets 1 and 2 contain 40 and 44 unique event types respectively. The datasets together contain 36 identical event types. Each of these event types have an informative description and a unique ID that explain the status of a device at a specific time, e.g., ’*Voltage OK L1*’, ’*Power down L2*’ and ’*Power up*’.

Table 9.1 summarizes detailed information about the smart meters dataset. Since daily activity of similar devices are monitored for a period of one month (May 2017), sequential patterns are extracted individually from devices in each daily segment. Therefore, the PrefixSpan user-specified support threshold (Step 1-a of the proposed approach) is set to be 1. Furthermore, to reduce the time complexity, only patterns with length between 2 to 7 are considered. On the other hand, if the interest is to know what kind of issues are mostly common between all devices, frequent patterns can be extracted from each segment and by considering all devices. That is, the PrefixSpan user-specified support threshold can set to be the minimum percentage that extracted patterns should appear in a segment.

Table 9.1: Summary of the smart meter data, May 2017

No. of devices		85,776
No. of recorded event logs		2,265,08
No of Event types		135
	Sampled dataset 1	Sampled dataset 2
No. of selected devices	30 (28)	30 (28)
No. of recorded event logs	28,369	54,555
No. of event types	40 (4)	44 (8)

Note. The distinct number of devices or events in each dataset are listed in the parentheses.

The second dataset contains one month (February 2018) of video session data. The data is obtained from a large European telecommunication company and contains 288,669 unique video session IDs, 4,983,090 events, 19 event types and 23,485 video programs. Examples of the event types in video session data are ’*Playback.Aborted*’, ’*Playback.BitrateChanged*’, and ’*Playback.PlayerReady*’.

Table 9.2 summarizes detailed information regarding this dataset. Since viewers receive a unique video session ID each time they login into their accounts, we extract frequent patterns that are common between all sessions in each daily segment. For this purpose after having some preliminary tests and discussions with the experts the user-specified support threshold is set to be 20%, i.e., the extracted patterns should at least appear in

($length\_of\_segment * 0.2$) of the sessions. Moreover, we assume that the extracted patterns must contain at least 3 event types.

Table 9.2: Summary of the video session data, February 2018

No. of video session IDs	288,669
No. of events	4,983,090
No. of video IDs	23,485
No. of event types	19

In all datasets, each event type represents an item. In the smart meter data, event sequences represent an overall status of each device per day while in the video session data, event sequences contain the quality related events and actions that have been performed by the viewers during the sessions. Moreover, event sequences in both datasets contain itemsets with exactly one event in each, i.e., the itemsets in this study are singletons.

9.4.2 Implementation and Availability

The proposed approach is implemented in Python version 3.6. The Python implementations of PrefixSpan and LD measure are fetched from [36] and [37], respectively. The affinity propagation algorithm is adopted from the scikit-learn module [38]. For constructing, manipulating, and visualizing a minimum spanning tree the NetworkX package is used [39]. The NetworkX package uses Kruskal's algorithm [40] for constructing the MST. The implemented code and the experimental results are available at GitHub[1].

9.5 Results and Discussion

9.5.1 Smart meter dataset

We performed random sampling on the smart meter dataset and created two datasets with 30 devices in each. More details can be found in *Datasets* sub-section and in Table 9.1.

Applying the proposed approach on the first sampled dataset, leads to extracting 6,550 patterns. Using AP the collected patterns are partitioned into 253 clusters. Finally, by building the MST on top of the exemplars of the clustering solution and cutting the longest edge(s) three sub-trees

[1] https://github.com/shahrooz-abghari/MST-Clustering-Approach

are constructed. The two smallest sub-trees are identified as outliers. The patterns in these sub-trees are matched with 8 devices. Examples of detected patterns as outliers in daily operation of smart meters in May 2017 are as follows: {'Adjust time/date (old time/date)', 'Adjust time/date (new time/date)'}, {'Power down L3', 'Power down L1', 'Power down L2', 'Power restored L3', 'Power restored L1', 'Power restored L2'}, and {'Remote communication module OK', 'Communication board access error, PLC or GSM/GPRS module'}.

Table 9.3: The results of the experiment for smart meter data

Dataset 1	No. of extracted patterns	6,550
	No. of patterns detected as outlier	241
	No. of clusters	253
	No. of detected outliers	1 sub-tree with 1 node 1 sub-tree with 39 nodes
	No. of devices with issue	6
Dataset 2	No. of extracted patterns	6,676
	No. of patterns detected as outlier	215
	No. of clusters	211
	No. of detected outliers	2 sub-trees, 1 node each
	No. of devices with issue	10

Note. Numbers inside the parentheses represent unique devices.

For the second sampled dataset, in total 6,676 patterns are identified. The extracted patterns are partitioned in 211 clusters and removing the longest edge(s) of the MST led to identifying 2 sub-trees out of 3 as outliers. Fig. 9.1 (Top) shows the constructed MST, the created forest after cutting the longest edges (edges A and B) of the MST, and the detected sub-trees as outliers. The patterns in these two clusters are matched with 10 devices. Examples of the identified interesting sequential patterns are as follows: {'Wrong phase sequence', 'Wrong phase sequence'}, {'Under voltage L2', 'No under voltage L2 anymore'}, and {'Communication error with FLEX meter/Measuring system access error', 'Meter communication OK with FLEX meter/Measurement System OK', 'Communication error with FLEX meter/Measuring system access error'}. Table 9.3 summarizes the results of the experiment on the smart meter data.

The results of the experiment on the smart meter data showed that the detected patterns as outliers are matched with 8 devices in the sampled dataset 1 and 10 devices in the sampled dataset 2. Fig. 9.2 shows the identified device ids with issues and the number of days they were faulty. Table 9.4 presents the top 5 sequential patterns that identified as outliers

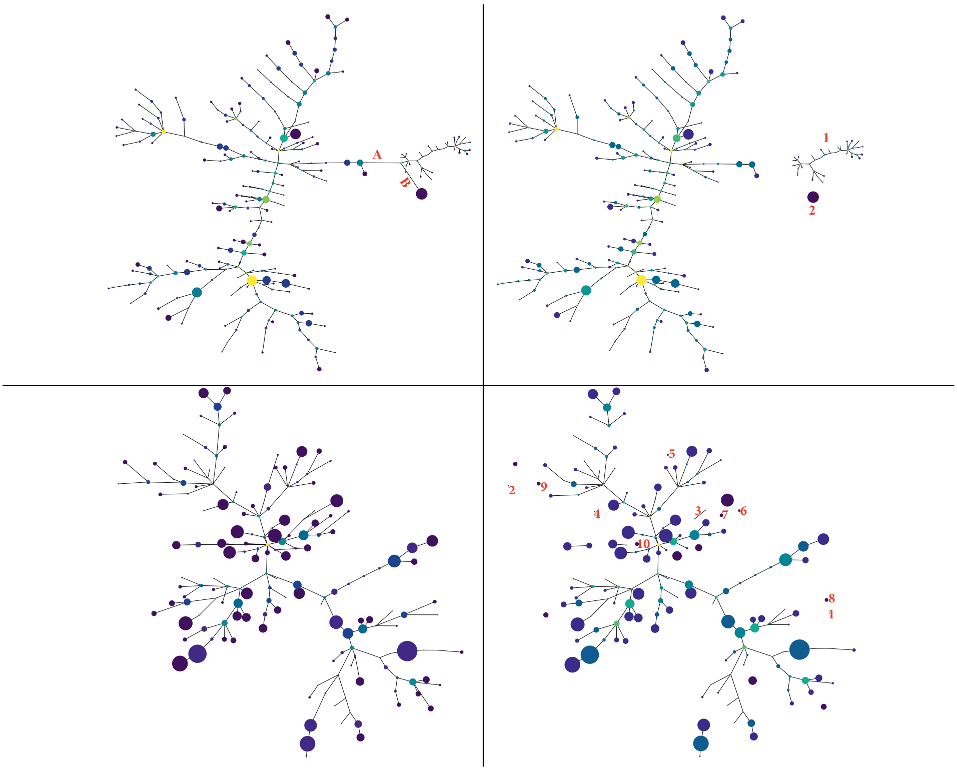
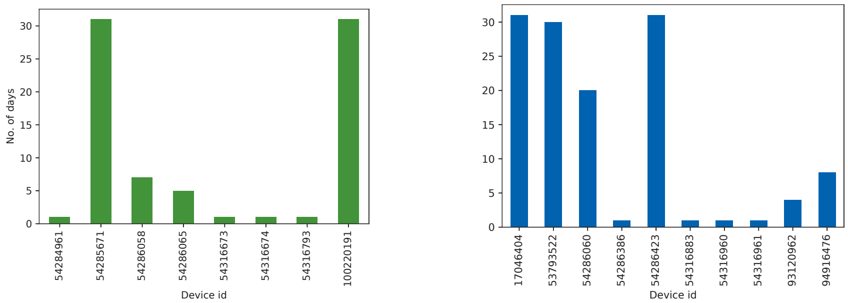


Figure 9.1: **(Top-left)** The constructed MST before removing the longest edges on smart meter sampled dataset 1. Edges A and B represent the longest edges of the tree. **(Top-right)** The transformation of the constructed MST into a forest with 3 sub-trees after the longest edges are removed. The sub-trees 1 and 2 are considered as outliers based on their size. **(Bottom-left)** The constructed MST before removing the longest edges on video session dataset. **(Bottom-right)** The transformation of the constructed MST into a forest with 22 sub-trees after the longest edges are removed. The sub-trees are ranked from smallest to largest based on their size. The top 10 smallest sub-trees are considered as outliers. ***Note.*** The size of a node represents the number of smart meters or video sessions that are matched with it. The color of a node shows the degree of the node and is used only for the visualization purposes. The distance between edges range between [0,1].

for each dataset. Perhaps only some of these patterns represent serious issues and some only relate to miss-configuration such as Adjust time/date. However, since such sequences of patterns occurred rarely in a normal daily activity of the monitored devices they have been detected as outliers. Nevertheless, further analysis and domain experts' opinion are needed to



(a) Dataset 1: 8 devices are identified with issues.

(b) Dataset 2: 10 devices are identified with issues.

Figure 9.2: Identified smart meter with issues for both sampled datasets 1 and 2, May 2017.

determine the severity of the issues raised by these patterns.

9.5.2 Video session dataset

We applied our proposed approach on one month (February 2018) video session data. In total 1,493 sequential patterns are mined. AP partitioned the patterns into 170 clusters and cutting the longest edges of the MST led to 22 sub-trees. We sort the sub-trees based on the number of video sessions they matched with, and choose the top 10 smallest sub-trees as outliers. Fig. 9.1 (Bottom) shows the constructed MST, the created forest after cutting the longest edges of the MST, and the top 10 smallest sub-trees identified as outliers. In total 10,121 video sessions are matched with patterns in these sub-trees. Examples of identified pattern as outlier are {'Playback.BufferingStarted', 'Playback.ScrubbedTo', 'Playback.Aborted'} and {'Playback.BitrateChanged', 'Playback.Resumed', 'Playback.Completed'}. Table 9.5 summarizes the results of the experiment on the video session data.

The results of the experiment on the video session dataset showed on average 3.5% of the sessions in each day contained outliers. Fig. 9.3 shows the total number of video sessions against the number of detected sessions as outliers and their percentage. In days 1, 23, 24, 26, and 28 of February higher number of outliers are detected. In the video session data, it is hard to draw any conclusions regarding the detected patterns

Table 9.4: Top 5 sequential patterns identified as outliers for smart meter sampled datasets, May 2017

	Pattern	Freq. of the pattern
Dataset 1	$\langle 165, 79, 165, 79, 79, 165, 165 \rangle$	27
	$\langle 10, 11, 11, 10, 11, 10, 11 \rangle$	15
	$\langle 20, 22 \rangle$	3
	$\langle 21, 20 \rangle$	3
	$\langle 397, 396, 393, 395, 392, 63, 346 \rangle$	1
Dataset 2	$\langle 20, 20 \rangle$	32
	$\langle 245, 245, 245, 245, 245, 245, 245 \rangle$	31
	$\langle 63, 63, 63, 63, 63, 63, 63 \rangle$	28
	$\langle 21, 21 \rangle$	17
	$\langle 22, 184 \rangle$	17

Note. Event names equivalent to each ID are as follows: **ID** = 10, *Adjust time/date (old time/date)*, **ID** = 11, *Adjust time/date (new time/date)*, **ID** = 20, *Over voltage on L1*, **ID** = 21, *Over voltage on L2*, **ID** = 22, *Over voltage L3*, **ID** = 63, *Wrong phase sequence*, **ID** = 79, *Communication board access error, PLC or GSM/GPRS module*, **ID** = 165, *Remote communication module OK*, **ID** = 184, *No over voltage L3 anymore*, **ID** = 245, *E Meter command error*, **ID** = 346, *Voltage L2 normal*, **ID** = 392, *Power down L1*, **ID** = 393, *Power down L2*, **ID** = 394, *Power down L3*, **ID** = 395, *Power restored L1*, **ID** = 396, *Power restored L2*, **ID** = 397, *Power restored L3.*

Table 9.5: The results of the experiment for video session data

No. of extracted patterns	1,493
No. of patterns detected as outliers	88
No. of clusters	170
No. of sub-trees	22
No. of detected outliers	1 sub-tree with 2 nodes 9 sub-trees, 1 node each
No. of video sessions with issue	10,121

as outliers without experts' validation. Unlike smart meter data that each event shows explicitly the status of a device, the event types in video session data are more general and most events can appear in both sessions with *good* and *bad* quality. However, the ratio of quality related events such as *Playback.BitrateChanged*, $ID = 2$, and *Playback.BufferingStarted*, $ID = 4$ or *Playback.BufferingStopped*, $ID = 5$ in a viewer's session can assist us to judge the quality of the session. A sudden increase in any of these three events in a session and simultaneously for many viewers can represent an issue at the system level.

Table 9.6 shows the top 10 sequential patterns that identified as outliers

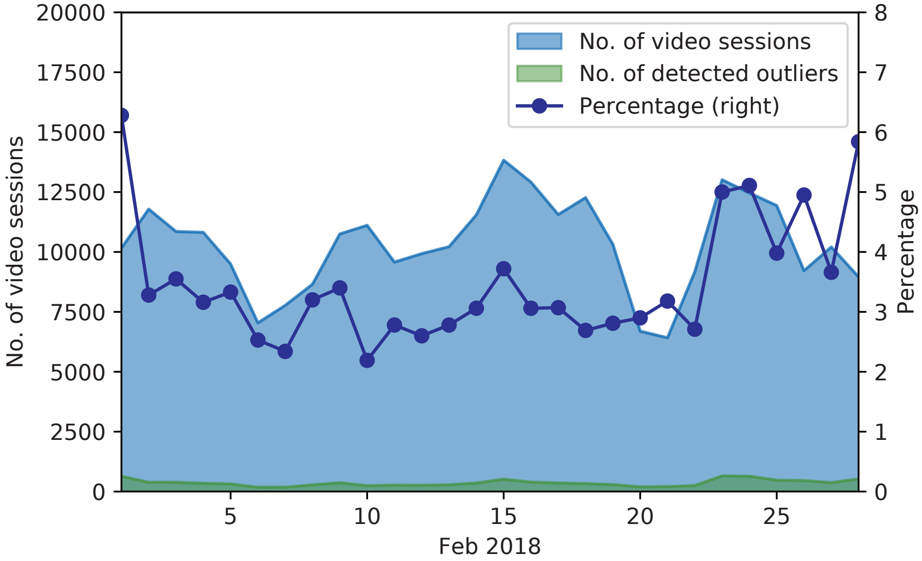


Figure 9.3: Visualization of the total number of video sessions vs. detected outliers for each day of February 2018 together with the percentage of the outliers.

and the number of video sessions that matched with them. Among these ten patterns seven of them (patterns 1-6 and 9) contain the quality related events that are mentioned earlier. On the other hand there are two patterns (7 and 9) that contain an event type *Playback.BufferingEnded*, $ID = 3$. This event often generates when the viewers scrub the video. Scrubbing is an action that helps a viewer to navigate through a video program to watch from a specific section. However, sometimes the viewers have to scrub the video due to frozen screen. Nevertheless, if the ratio of *Playback.BufferingEnded* increases inside a video session and at the same time for many sessions can relate to an issue at the system level. We have discussed the obtained results with domain experts. In order to validate the results, the experts asked us to randomly select 18 video sessions (9 normal and 9 abnormal) from February 1, 2018. The labels of the video sessions were unknown for the expert. The validation shows only 3 video sessions can be considered as true anomalies and the other sessions are normal. This means $12/18 = 67\%$ of the video sessions are labeled correctly by the proposed approach. Further analysis of the experts' comments has revealed that assessing the quality of a video

Table 9.6: Top 10 sequential patterns identified as outliers for video session dataset, February 2018

No.	Pattern	Freq. of the pattern
1	$\langle \mathbf{2}, 16, 6\rangle$	1613
2	$\langle 11, 7, \mathbf{2}, 18\rangle$	838
3	$\langle 11, 17, \mathbf{4}\rangle$	653
4	$\langle 11, 7, \mathbf{2}, 16\rangle$	477
5	$\langle 12, \mathbf{5}, 1\rangle$	459
6	$\langle \mathbf{4}, 17, 1\rangle$	445
7	$\langle 11, 3, 15\rangle$	403
8	$\langle 18, 17, 1\rangle$	401
9	$\langle 11, 3, \mathbf{4}\rangle$	298
10	$\langle 12, 14, 1\rangle$	290

Note. The numbers in bold represent the quality related events. Event names equivalent to each ID are as follows: **ID** = 1, *Playback.Aborted*, **ID** = 2, *Playback.BitrateChanged*, **ID** = 3, *Playback.BufferingEnded*, **ID** = 4, *Playback.BufferingStarted*, **ID** = 5, *Playback.BufferingStopped*, **ID** = 6, *Playback.Completed*, **ID** = 7, *Playback.Created*, **ID** = 11, *Playback.HandshakeStarted*, **ID** = 12, *Playback.InitCompleted*, **ID** = 14, *Playback.PlayReady*, **ID** = 15, *Playback.PlayerReady*, **ID** = 16, *Playback.Resumed*, **ID** = 17, *Playback.ScrubbedTo*, **ID** = 18, *Playback.Started.*

session is not an easy task and sometimes can be subjective. For example, the latter is supported by the experts' comments concerning the following 4 video sessions (V_1 to V_4) out of 6 that are mislabeled as outliers by the proposed approach:

$\mathbf{V_1}$. "Short session (15,5 sec), no buffering, good bitrate, hard to tell, I tend to OK."

$\mathbf{V_2}$. "No buffering, rather short, 10 sec, bitrate rather good. OK."

$\mathbf{V_3}$. "Probably still OK considering the duration of the playback (1055 sec), but some buffering (35 sec in total), with varying bitrate, I tend to OK."

$\mathbf{V_4}$. "Some buffering, but below 0,5% of the play duration, played 612 sec, error event at the end of the session with no further explanation, played 10 minutes with good bitrate, I tend to OK."

The validation of the results has shown that the proposed approach is able to identify video sessions that are significantly different from the majority of the sessions due to occurrence of some specific event types. The identified anomalous sequential patterns can help the domain experts to understand the outlying properties of the detected outliers.

9.6 Conclusion

In this study, we have presented an outlier detection for sequence datasets. Our approach combines sequential pattern mining, clustering and minimum spanning tree to identify outliers. We have shown that the proposed approach can facilitate the domain experts in identification of outliers. Building the minimum spanning tree on top the clustering solution can lead to identifying clusters of outliers. This can reduce the time complexity of the proposed approach. Moreover, in this study we have looked into collective outliers, sequences of events that based on their occurrence together assumed to be anomalous, which may help to find the outlying properties of the detected outliers.

The proposed approach has been applied on two sequence datasets, smart meter data and video session data. Both datasets contain sequences of event types that either shows the operational status of a smart meter or the current action that takes place in a viewer's video session. The results of the experiments on the smart meters data are more comprehensible compared to the video session data. The main reason is the fact that the event types in smart meters are explicitly detailed, explaining the status of the devices. However, in video session data the event types are general which requires more investigation and experts' knowledge in order to detect video sessions with quality issues. The validation of the results on video session data by the domain experts showed that 67% of the labeled sessions by the proposed approach were correct.

For future work, we are going to further evaluate and validate the proposed approach by using different distance measures and clustering algorithms. These two parameters may have a strong correlation and it worths to further study how such correlation can affect the final results.

In this study, the MST is constructed from a complete weighted graph using the exemplars of the clusters. However, according to Cipolla et al.'s study a Delaunay triangulation network can be used to create a simplified graph. The Delaunay triangulation limits the number of connections of a node to its closest neighbors. This makes the constructed MST from this simplified graph a better representative compared to the complete graph. Therefore, we are also interested in testing whether the Delaunay triangulation network can be integrated into our approach.

References

[1] V. Chandola, A. Banerjee, and V. Kumar. "Anomaly detection: A survey". In: *ACM Computing Surveys* 41.3 (2009), p. 15.

[2] V. Hodge and J. Austin. "A survey of outlier detection methodologies". In: *Artificial Intelligence Review* 22.2 (2004), pp. 85–126.

[3] J. Han, H. Cheng, D. Xin, and X. Yan. "Frequent pattern mining: Current status and future directions". In: *Data Mining and Knowledge Discovery* 15.1 (2007), pp. 55–86.

[4] R. Agrawal and R. Srikant. "Mining sequential patterns". In: *Proc. of the 11th Int'l Conf. on Data Engineering.* IEEE. 1995, pp. 3–14.

[5] T. P. Exarchos, C. Papaloukas, C. Lampros, and D. I. Fotiadis. "Mining sequential patterns for protein fold recognition". In: *J. of Biomedical Informatics* 41.1 (2008), pp. 165–179.

[6] J. Guan, D. Liu, and D. A. Bell. "Discovering motifs in DNA sequences". In: *Fundamenta Informaticae* 59.2-3 (2004), pp. 119–134.

[7] Y.-L. Chen, M.-H. Kuo, S.-Y. Wu, and K. Tang. "Discovering recency, frequency, and monetary (RFM) sequential patterns from customers' purchasing data". In: *Electronic Commerce Research and Applications* 8.5 (2009), pp. 241–251.

[8] L.-C. Wuu, C.-H. Hung, and S.-F. Chen. "Building intrusion pattern miner for Snort network intrusion detection system". In: *J. of Systems and Software* 80.10 (2007), pp. 1699–1715.

[9] F. Eichinger, D. D. Nauck, and F. Klawonn. "Sequence mining for customer behaviour predictions in telecommunications". In: *ECML/PKDD 2006 Workshop on Practical Data Mining: Applications, Experiences and Challenges.* 2006, pp. 3–10.

[10] T. H. N. Vu, K. H. Ryu, and N. Park. "A method for predicting future location of mobile user for location-based services system". In: *Computers & Industrial Engineering* 57.1 (2009), pp. 91–105.

[11] S. Jaillet, A. Laurent, and M. Teisseire. “Sequential patterns for text categorization”. In: *Intelligent Data Analysis* 10.3 (2006), pp. 199–214.

[12] P. C. Wong, W. Cowley, H. Foote, E. Jurrus, and J. Thomas. “Visualizing sequential patterns for text mining”. In: *Symp. on Information Visualization.* IEEE. 2000, pp. 105–111.

[13] C. C. Aggarwal and P. S. Yu. “Outlier detection for high dimensional data”. In: *ACM Sigmod Record.* Vol. 30. 2. ACM. 2001, pp. 37–46.

[14] M.-F. Jiang, S.-S. Tseng, and C.-M. Su. “Two-phase clustering process for outliers detection”. In: *Pattern Recognition Letters* 22.6 (2001), pp. 691–700.

[15] Y. Zhang, N. Meratnia, and P. Havinga. “Outlier detection techniques for wireless sensor networks: A survey”. In: *IEEE Communications Surveys & Tutorials* 12.2 (2010), pp. 159–170.

[16] M. Gupta, J. Gao, C. C. Aggarwal, and J. Han. “Outlier detection for temporal data: A survey”. In: *IEEE Transactions on Knowledge and Data Engineering* 26.9 (2014), pp. 2250–2267.

[17] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, et al. “A density-based algorithm for discovering clusters in large spatial databases with noise”. In: *KDD.* Vol. 96. 34. 1996, pp. 226–231.

[18] S. Guha, R. Rastogi, and K. Shim. “CURE: An efficient clustering algorithm for large databases”. In: *ACM Sigmod Record.* Vol. 27. 2. ACM. 1998, pp. 73–84.

[19] S. Guha, R. Rastogi, and K. Shim. “ROCK: A robust clustering algorithm for categorical attributes”. In: *Proc. of the 15th Int’l Conf. on Data Engineering.* IEEE. 1999, pp. 512–521.

[20] L. Ertöz, M. Steinbach, and V. Kumar. “Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data”. In: *Proc. of the 2003 SIAM Int’l Conf. on Data Mining.* SIAM. 2003, pp. 47–58.

[21] F. A. González and D. Dasgupta. “Anomaly detection using real-valued negative selection”. In: *Genetic Programming and Evolvable Machines* 4.4 (2003), pp. 383–403.

[22] X. Pan, J. Tan, S. Kavulya, R. Gandhi, and P. Narasimhan. “Ganesha: Blackbox diagnosis of mapreduce systems”. In: *ACM SIGMETRICS Performance Evaluation Review* 37.3 (2010), pp. 8–13.

[23] E. Eskin, A. Arnold, M. Prerau, L. Portnoy, and S. Stolfo. “A geometric framework for unsupervised anomaly detection”. In: *Applications of data mining in computer security*. Springer, 2002, pp. 77–101.

[24] X. Wang, X. L. Wang, and D. M. Wilkes. “A minimum spanning tree-inspired clustering-based outlier detection technique”. In: *Ind. Conf. on Data Mining*. Springer. 2012, pp. 209–223.

[25] A. C. Müller, S. Nowozin, and C. H. Lampert. “Information theoretic clustering using minimum spanning trees”. In: *Joint DAGM (German Association for Pattern Recognition) and OAGM Symp.* Springer. 2012, pp. 205–215.

[26] G.-W. Wang, C.-X. Zhang, and J. Zhuang. “Clustering with Prim’s sequential representation of minimum spanning tree”. In: *Applied Mathematics and Computation* 247 (2014), pp. 521–534.

[27] E. Cipolla, U. Maniscalco, R. Rizzo, D. Stabile, and F. Vella. “Analysis and visualization of meteorological emergencies”. In: *Ambient Intelligence and Humanized Computing* 8.1 (2017), pp. 57–68.

[28] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M. Hsu. “Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth”. In: *Proc. of the 17th Int’l Conf. on Data Engineering*. 2001, pp. 215–224.

[29] J. Pei, J. Han, B. Mortazavi-Asl, J. Wang, H. Pinto, Q. Chen, U. Dayal, and M.-C. Hsu. “Mining sequential patterns by pattern-growth: The PrefixSpan approach”. In: *IEEE Transactions on Knowledge and Data Engineering* 16.11 (2004), pp. 1424–1440.

[30] R. Srikant and R. Agrawal. “Mining sequential patterns: Generalizations and performance improvements”. In: *Advances in Database Technology–EDBT’96* (1996), pp. 1–17.

[31] M. J. Zaki. “SPADE: An efficient algorithm for mining frequent sequences”. In: *Machine Learning* 42.1 (2001), pp. 31–60.

[32] V. I. Levenshtein. “Binary codes capable of correcting deletions, insertions, and reversals”. In: *Soviet physics doklady*. Vol. 10. 8. 1966, pp. 707–710.

[33] B. J. Frey and D. Dueck. “Clustering by passing messages between data points”. In: *Science* 315.5814 (2007), pp. 972–976.

[34] U. Bodenhofer, A. Kothmeier, and S. Hochreiter. "APCluster: An R package for affinity propagation clustering". In: *Bioinformatics* 27.17 (2011), pp. 2463–2464.

[35] Elektro Ljubljana. *Smart meters recorded events dataset.* 2018. URL: `https://data.edincubator.eu/organization/elektro-ljubljana-podjetje-zadistribucijo-elektricne-energije-d-d`.

[36] C. Gao. "PrefixSpan algorithm source code". In: (2015). URL: `https://github.com/chuanconggao/PrefixSpan-py`.

[37] B. Jain. "Edit distance algorithm source code". In: (-). URL: `https://www.geeksforgeeks.org/dynamic-programming-set-5-edit-distance`.

[38] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. "Scikit-learn: Machine Learning in Python". In: *J. of Machine Learning Research* 12 (2011), pp. 2825–2830.

[39] A. Hagberg, P. Swart, and D. S Chult. *Exploring network structure, dynamics, and function using NetworkX*. Tech. rep. Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.

[40] J. B. Kruskal. "On the shortest spanning subtree of a graph and the traveling salesman problem". In: *Proc. of the American Mathematical Society* 7.1 (1956), pp. 48–50.

10

Trend Analysis to Automatically Identify Heat Program Changes

Shahrooz Abghari, Eva Garcia-Martina, Christian Johansson, Niklas Lavesson, Håkan Grahn

In: Energy Procedia 116 (2017): pp. 407-415.
Also Publised In: The 2016 15th International Symposium on District Heating and Cooling, Seoul, Korea.

Abstract

The aim of this study is to improve the monitoring and controlling of heating systems located at customer buildings through the use of a decision support system. To achieve this, the proposed system applies a two-step classifier to detect manual changes of the temperature of the heating system. We apply data from the Swedish company NODA, active in energy optimization and services for energy efficiency, to train and test the suggested system. The decision support system is evaluated through an experiment and the results are validated by experts at NODA. The results show that the decision support system can detect changes within three days after their occurrence and only by considering daily average measurements.

10.1 Introduction

In the district heating (DH) domain, operators address several conflicting goals, such as satisfying customer demand while minimizing production and distribution costs. To achieve this, one solution is to equip each customer building with a smart system. Such a system should continuously monitor heat usage, predict future demand, exchange information with operators, and perform demand-side management. Moreover, the system needs to automatically learn the energy usage of the building and adopt its behavior accordingly. NODA Intelligent Systems AB[1] , an active company in the DH domain, is developing and providing retrofit smart systems to maximize energy efficiency in buildings. These systems consist of controlling hardware together with a range of sensors, which are added on top of the existing control system.

Self-learning and adaptation are two important features of any smart system. However, these two features make the system sensitive to manual changes in the heating system, forcing the system to re-learn its characteristics. Most commonly this relates to applying changes in the temperature program of the controller e.g. by the owner's building. These changes can lead to use more energy and to add extra charges in the case of increasing the temperature of the system.

Although retrofit solutions such as NODA's smart system can decrease the cost of replacement of the existing control system, their functionality can be affected by the limitation of these existing controllers. Due to this reason, NODA's smart system is unable to detect manual changes online. Hence, NODA's operators need to spend significant efforts to detect the manual changes by analyzing the received information from each building controller. To make this process more efficient, a decision support (DS) system can be used to assist operators. DS systems are computer-based information systems, which aim to facilitate and support the decision-making processes [1]. The major components of DS systems are: 1) the user-interface, 2) the models and main logic, 3) the database, and 4) the DS functionalities and architecture. DS systems are categorized based on their functionalities into: data-driven, knowledge-driven, model-driven, document-driven and communication-driven DS systems [1]. Among these different types, data-driven systems can provide an online support for decision making through applying machine learning (ML) and statistical techniques to analyze large collections of data. Machine learning is a branch of artificial intelligence, which includes the study of algorithms that can learn and improve their knowledge by building models from input data to perform specific tasks. Most common tasks in ML, such as classification and regression modeling, are solved with supervised learning methods. Supervised learning uses labeled data to train models [2]. Suppose we are given data in the form of $(\overrightarrow{x_1}, y_1), (\overrightarrow{x_2}, y_2), ..., (\overrightarrow{x_n}, y_n)$. In each pair or instance $\overrightarrow{x_i}$ (input) denotes a vector, which consists of feature values such as indoor and outdoor tempera-

[1] www.noda.se/en/main

ture, and y_i (output) indicates a label or outcome of the target attribute. The aim is to train a model to predict the label of the target attribute (y_i) of each new instance, e.g. predicting the secondary supply temperature based on the indoor and outdoor temperature. The target attribute in regression modeling is numeric and in classification modeling it is categorical.

In this paper, we propose a data-driven decision support system that uses ML techniques to detect manual changes by predicting the secondary supply temperature based on the outdoor temperature and analyzing the energy consumption of each building. The aim of such a system is to provide complementary decision support for NODA's operators to detect manual changes easily and efficiently. The proposed DS system uses a two-step classifier, a combination of k-means and support vector regression (SVR), to detect manual changes within three days after their occurrence by considering daily average measurements.

10.2 Background and Related Work

A district heating system (DHS) is a centralized system with the aim of producing space heating and hot tap water for consumers based on their demand at a limited geographic area. A DH system consists of three main parts: production units, distribution network, and consumers. The heated water supplied in a production unit circulates through the distribution network and will be available to consumers. The main aim of a DHS is to minimize the cost and pollution by considering consumers' demand and producing just the necessary amount of heat. Hence, being able to predict the heat demand can assist production units to plan better. However, modeling the heat demand forecasting is a challenging task, since water does not move fast. In some situations, the distribution of heated water can take several hours. Moreover, there are a number of factors that affect the forecast accuracy and need to be considered before any plan for production units can be constructed. Some of these factors include [3, 4]:

1. Weather condition, mainly the outdoor temperature
2. Social behavior of the consumers
3. Irregular days such as holidays

4. Periodic changes in conditions of heat demand such as seasonal, weekly and day-night

Fumo [5] pointed out in his review two commonly used techniques for energy demand estimation, namely; forward (classical) and data-driven (inverse) techniques. The first approach describes the behavior of systems by applying mathematical equations and known inputs to predict the outputs. In contrast, data-driven techniques use ML methods to learn the system's behavior by building a model with training data in order to make predictions.

Dotzauer [4] introduced a very simple model for forecasting heat demand based on outdoor temperature and social behavior. He showed that the predictions of his simple model were comparable with complicated models such as autoregressive moving average model (ARMA). The author concluded that better predictions can be achieved by improving the weather forecasts instead of developing complicated heat demand forecasting models.

In general, different ML methods and techniques have been used to predict the heat demand. Some of the most popular prediction models are autoregressive moving average (ARMA) [6], support vector regression (SVR) [7, 8], multiple linear regression (MLR) [9] and artificial neural network (ANN) [10, 11]. In [8], the authors compared four supervised ML methods for building short-term forecasting models. The models are used to predict heat demand for multi-family apartment buildings with different horizon values between 1 to 24 hours ahead. The authors concluded that SVR achieves the best performance followed by MLR in comparison to feed forwards neural network (FFNN), and regression trees methods. Recently, Provatas et al. [12], proposed the usage of on-line ML algorithms in combination with decision tree-based ML algorithms for heat load forecasting in a DH system. The authors investigated the impact of two different approaches for heat load aggregation. The results of the study showed that the proposed algorithm has a good prediction result. In another study [13], the authors showed the application of a context vector (CV) based approach for forecasting energy consumption of single family houses. The proposed method is compared with linear regression, K-nearest neighbors (KNN) and SVR methods. The results of the experiment showed that CV performed better in most cases followed by KNN and SVR. The authors concluded the proposed solution can help DH companies to improve their schedule and reduce operational costs.

There are a number of studies that focused on the application of DS systems in domains such as DH and mainly related to advanced energy management [14–19]. In these studies, the main focus is on forecasting and optimization methods that facilitate and support the decision-making processes to increase the energy management quality and bring considerable savings. Furthermore, there are some other works that focused on DH network design [20, 21]. Bordin et al. [20] presented a mathematical model to support DH system network planning by selecting an optimal set of new users to be connected to a thermal network that maximizes revenues and minimizes infrastructure and operational costs.

In summary, the main focus of the studies that have been done in the context of heat demand forecasting in the DH domain was related to using weather forecast data and mainly the outdoor temperature. In contrast, the aim of the proposed solution in this study is twofold: 1) to provide decision support for operators to detect manual changes efficiently, and 2) to decrease the energy consumption cost and control heat demand by identifying these changes and resolving them at each building.

10.3 Detection of Changes in Trends by Using Regression Methods

In DH, operators try to address several conflicting goals, such as satisfying customer demand while minimizing production and distribution costs. One way to solve this is to use demand side management and data analytics in the customer substations. This can be achieved by a system that continuously predicts the future heat demand, exchanges information with the operator and performs demand side management when the need arises. Such systems can be implemented both in the existing heating controllers as well as in retrofit solutions. One such retrofit system is developed by NODA and it has been used within this study. To make the system efficient, its behavior has to be as automated and self-learning as possible. However, this also makes the system sensitive to manual changes (i.e. changing the temperature) in the heating system, since such changes forces the system to re-learn the characteristics of the heating system.

In order to assist operators to detect these manual changes more efficiently a DS system is implanted to provide decision support for operators. The proposed DS system uses a two-step classifier, *k*-means and SVR, to detect manual changes. To achieve this goal and to avoid generating false alarms in confront with noisy data, changes should be monitored for some days. Hence, in this study only those deviations that last for at least 3 consecutive days would be marked as manual changes. *k*-means, which is the most well-known algorithm for classification task, is used to identify the operational status of the heating system (on or off) by partitioning the consumed energy at each building.

The main reason to perform this task is to decrease the effect of outliers when the heating system is not operating. SVR has been used for both electricity and heat demand forecasting and has been found to be very efficient and accurate [8, 22]. Therefore, SVR is chosen to predict secondary supply temperature based on outdoor temperature and consumed energy for each building. By considering the status of the system and the predicted value of the secondary supply temperature, the DS system can identify manual changes as follows:

IF the absolute difference (actual – predicted) is greater than the threshold FOR 3 consecutive days THEN changes have occurred during these days.

The warning threshold determines the sensitivity of the system to change. This threshold, set to 4.6 °C, was determined empirically after performing some preliminary tests and checking the results with the subject matter experts. Figure 10.1 summarizes the process of automatically identifying the manual changes for each building by the proposed DS system.

10.3.1 Algorithms

The *k*-means algorithm belongs to the group of distance-based clustering methods. It is the best known greedy algorithm for partitioning data into *k* clusters. This popularity is mainly related to *k*-means' simplicity, efficiency, and applied success in partitioning and pattern recognition tasks [2, 23]. It works by reducing the total sum of the squared error over all *k* clusters. *k*-means iterates by generating partitions and assigning data to the closest cluster and computing the centroid from a partition until no further improvement can be achieved [23].

The support vector machine (SVM) algorithm is based on statistical learning theory. SVM is a state-of-the-art algorithm, which belongs to a

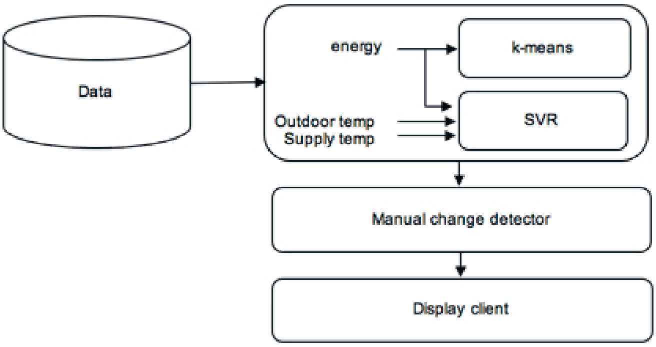


Figure 10.1: The process of automatically identifying the manual changes for each building by the DS system.

group of supervised learning methods that can solve different ML tasks such as classification, pattern recognition and regression [24]. An extended version of SVM for regression tasks is called support vector regression. SVR uses the training data to find the regression line that best fits the data. Using an epsilon-intensive loss function, SVR produces a decision boundary, a subset of training data which is called support vectors (SVs), in order to determine a tube with radius ε fitted to the data. In other words, epsilon defines how well the regression line fits the data by ignoring errors as long as they are less than ε.

10.4 Research Method

10.4.1 Data Collection

The data used in this study consists of daily average measurements from 9 buildings equipped with the NODA controller. The buildings are located in Karlshamn in south Sweden. The collected data was obtained on the period between April 2014 and March 2016. This yields 730 instances per building (one instance per day). However, since data collection instruments, such as sensors, might be faulty, or since data transmission errors can occur [25], some of the measurements were incomplete. Therefore, after performing the data cleaning process the number of instances decreased to approximately 630 per building. Table 10.1 summarizes the information and the way the data is split to train and test set for each building.

Table 10.1: Summary of the data collection for each building

Building ID	Data *(no. of instances)*		
	Train set (Apr 2014 - Mar 2015)	Test set (Apr 2015 - Mar 2016)	Total
A	251	249	500
B	349	365	714
C	357	365	722
D	357	365	722
E	347	347	694
F	270	365	635
G	251	249	500
H	357	332	689
I	362	345	707

Figure 10.2 shows the daily average of the secondary supply temperature of building D with respect to the outdoor temperature for the year 2015 (365 instances). The plot shows that the secondary supply temperature has a strong correlation with the outdoor temperature.

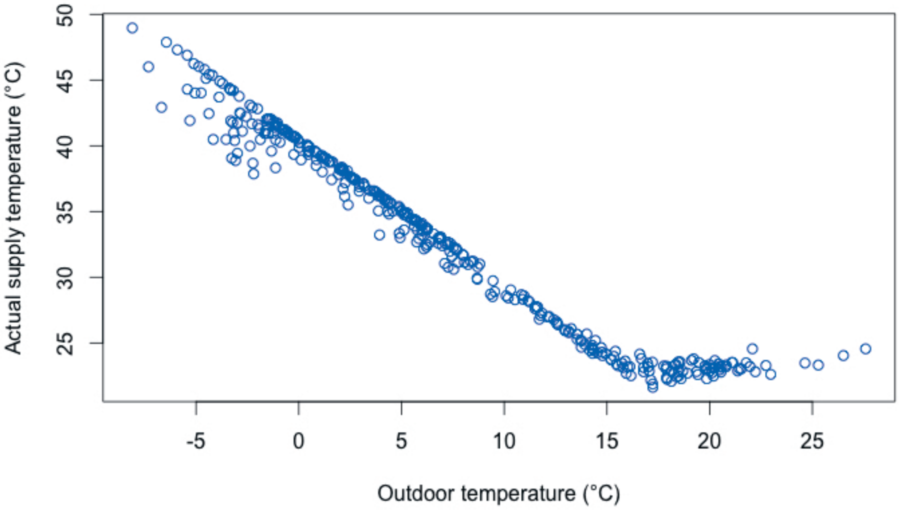


Figure 10.2: Daily average secondary supply temperature of building D with respect to the daily average outdoor temperature for the year 2015 (365 instances).

We used R and RWeka package to conduct the experiment. R is a language and a free software environment for statistical computing with data [26]. R is widely used for visualization and statistical tasks such as linear and non-linear modelling, regression analysis, and statistical tests. RWeka is an R interface to WEKA (Waikato environment for knowledge analysis) [27]. WEKA [28] is a well-known machine learning and data mining workbench written in Java. It contains a wide range of algorithms for different ML tasks such as classification, regression, and clustering. We used RWeka's k-means and SVR implementation with their default parameters.

10.4.1.1 Experimental Design

To detect manual changes in the heating system for each building, the implemented DS system uses a two-step classifier. 10-fold cross validation is used on data from April 2014 to March 2015 to build the model for each building. m-fold cross validation is a standard procedure for a model evaluation in ML. The main idea is to randomly split the dataset into m equal subsets. The model is trained and tested m times. Each time one of the m subsets is used as a test set and the other $m - 1$ subsets are form the training set. The overall performance of the model is computed as the average error across all m runs [29]. The train set is preprocessed and cleaned to make sure that the DS system only learns the normal behavior of the heating system. Additionally, the quality of the model is tested with the data from April 2015 to March 2016.

The performance of the system is evaluated in two ways:

1. using mean absolute error (MAE) as a performance measure to evaluate the accuracy of SVR in terms of predicting the secondary supply temperature.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|actual_i - predicted_i| \tag{10.1}$$

   In equation (10.1), the actual refers to the measured secondary supply temperature by the controller system, predicted refers to the estimated secondary supply temperature by the proposed DS system, and n is the total number of predicted instances.

2. validating the detected changes by subject matter experts at NODA. In this case the accuracy of the system is calculated based on the

number of true positive (TP), true negative (TN), false positive (FP) and false negative (FN) alarms in equation (10.2). The TPs and TNs are correct classifications. A false positive happens when the result is classified incorrectly as a detected change while it is actually not a change. A false negative occurs when an actual change in the system is not detected [25].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{10.2}$$

10.5 Results

The performance of the proposed DS system is evaluated by using the test data (April 2015 – March 2016) and 10-fold cross validation for each building. Furthermore, the identified changes at each building is validated with NODA's experts. Table 10.2 summarizes the performance of the SVR together with the number of manual detected changes at each building. The results show that the DS system detected, in total 60 changes correctly in 3 out of 9 buildings. These changes either related to manual changes or hardware failures. This value represents the number of TP alarms. The majority of the results belonged to the TN category with the value of 2,888. The false positive alarms occurred in 2 buildings and in total contain 34 changes. The main reason for these detected changes are related to a sudden drop in the outdoor temperature, and the fact that the system was not trained

Table 10.2: Mean absolute error and standard deviation for SVR and detected changes for each individual building

Building ID	MAE	Detected changes			
		TP	TN	FP	FN
A	1.64 (±0.016)	26	223	-	-
B	1.70 (±0.008)	-	358	7	-
C	2.40 (±0.006)	8	357	-	-
D	0.72 (±0.002)	-	365	-	-
E	1.07 (±0.003)	-	320	27	-
F	0.73 (±0.004)	-	365	-	-
G	1.64 (±0.016)	26	223	-	-
H	0.49 (±0.002)	-	332	-	-
I	1.02 (±0.004)	-	345	-	-
Total	-	60	2,888	34	0

Note. MAE = mean absolute error, standard deviation appears within the parentheses.

for such a situation. No false negative is detected during the experiment. By considering these values and using the equation (10.2) the accuracy of the system can be computed as follow: $(60 + 2,888)/(60 + 2,888 + 34) = 0.98$.

Figures 10.3 and 10.4 depict the outcome of the system for two different buildings. Figure 3 shows the detected manual changes occurred during 14th until 23rd of January 2016 at the C building. These manual changes are related to the modification of the temperature of the heating system. Figure 4 is related to the D building. This building has no changes, which can be seen since the actual and predicted secondary supply temperature are closely following each other.

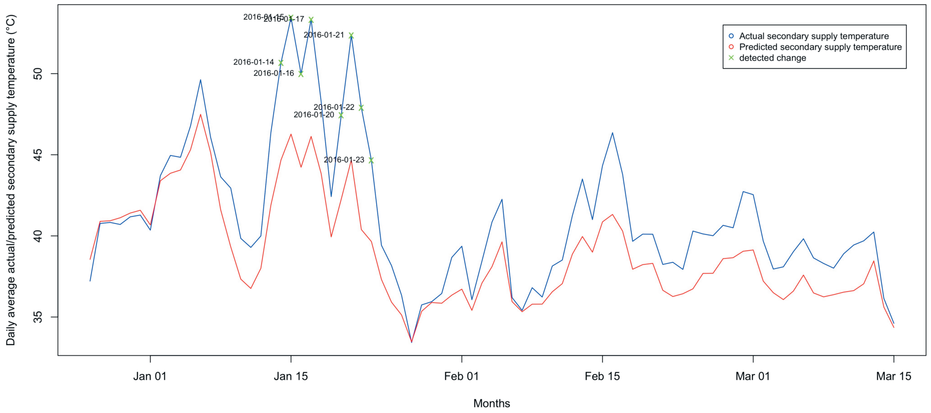


Figure 10.3: Identified manual changes during January 2016 at the C building. The actual secondary supply temperature is showed in blue against the predicted secondary supply temperature in red. The green crosses identify the detected manual changes by the DS system.

10.6 Discussion

The experimental results show that the proposed DS system with a two-step classifier is able to detect manual changes within three days after their occurrence. The accuracy of the system is evaluated by the experts from NODA. The results of the evaluation show that the system has a solid detection ability with an accuracy of 98%. In general, the important aspect

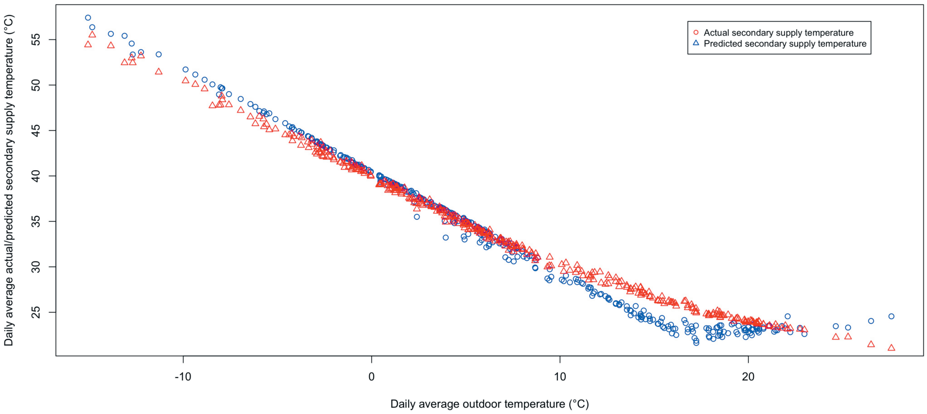


Figure 10.4: The actual and predicted secondary supply temperature related to building D. This building has no changes during April 2015 – March 2016.

of such system is its ability to detect changes correctly and does not miss any changes.

To decrease the false alarms (both FP and FN) in the detection task, the main solution is to train the system with the data that represents the normal behavior of the heating system. Moreover, only those deviations that last at least three days are classified as manual changes. In addition, considering an adjustable threshold instead of a fix one can decrease the number of false alarms. Though the false positive alarms can be quickly determined and dismissed by experts, considerable number of false alarms can be disturbing.

The proposed DS system is generalizable to similar applications such as detection of change in energy demand or detection of faulty equipment based on abnormal behavior of the heating system.

10.7 Conclusion

We propose a decision support system for operators in the district heating domain. Currently, the proposed system is applied to detect manual changes in the heating system at the building level. The decision support system uses a two-step classifier, *k*-means and support vector regression, to identify manual changes within three days after their occurrence and by considering daily average measurements. The performance of the system is evaluated with the real data related to 9 buildings in Sweden. The validity of the results was investigated by the experts at the NODA Intelligent Systems AB. The validation of the results showed that the majority detected changes by the system were true alarms.

Since each building has special characteristics, e.g. its geographical location, used construction materials, and the social behavior of its tenants, having a fixed threshold for all buildings is impractical. Hence, in the future, it is important to investigate how to automatically set the threshold value for each building. Moreover, it is more convenient that operators can have interaction with the DS system by providing feedbacks. Thus, the performance of the system can improve through time.

References

[1] D. J. Power. *Decision support systems: Concepts and resources for managers*. Greenwood Publishing Group, 2002.

[2] P. Flach. *Machine learning: The art and science of algorithms that make sense of data*. Cambridge University Press, 2012.

[3] N. Eriksson. *Predicting demand in districtheating systems: A neural network approach*. 2012.

[4] E. Dotzauer. “Simple model for prediction of loads in district-heating systems”. In: *Applied Energy* 73.3-4 (2002), pp. 277–284.

[5] N. Fumo. “A review on the basics of building energy estimation”. In: *Renewable and Sustainable Energy Reviews* 31 (2014), pp. 53–60.

[6] H. Wiklund. “Short term forecasting on the heat load in a DH-system”. In: *Fernwärme Int’l* 20.5-6 (1991), pp. 286–294.

[7] L. Wu, G. Kaiser, D. Solomon, R. Winter, A. Boulanger, and R. Anderson. “Improving efficiency and reliability of building systems using machine learning and automated online evaluation”. In: *The 11th Conf. on Systems, Applications and Technology*. IEEE. 2012, pp. 1–6.

[8] S. Idowu, S. Saguna, C. Åhlund, and O. Schelén. “Forecasting heat load for smart district heating systems: A machine learning approach”. In: *Int’l Conf. on Smart Grid Communications*. IEEE. 2014, pp. 554–559.

[9] T. Catalina, V. Iordache, and B. Caracaleanu. “Multiple regression model for fast prediction of the heating energy demand”. In: *Energy and Buildings* 57 (2013), pp. 302–312.

[10] K. Kato, M. Sakawa, K. Ishimaru, S. Ushiro, and T. Shibano. “Heat load prediction through recurrent neural network in district heating and cooling systems”. In: *Int’l Conf. on Systems, Man and Cybernetics*. IEEE. 2008, pp. 1401–1406.

[11] M. Sakawa, K. Kato, and S. Ushiro. “Cooling load prediction in a district heating and cooling system through simplified robust filter and multilayered neural network”. In: *Applied Artificial Intelligence* 15.7 (2001), pp. 633–643.

[12] S. Provatas. *An online machine learning algorithm for heat load forecasting in district heating systems*. 2014.

[13] S. Rongali, A. R. Choudhury, V. Chandan, and V. Arya. “A context vector regression based approach for demand forecasting in district heating networks”. In: *Int’l Conf. on Innovative Smart Grid Technologies Asia*. IEEE. 2015, pp. 1–6.

[14] K. Mařík, Z. Schindler, and P. Stluka. “Decision support tools for advanced energy management”. In: *Energy* 33.6 (2008), pp. 858–873.

[15] D. Chinese and A. Meneghetti. “Optimisation models for decision support in the development of biomass-based industrial district-heating networks in Italy”. In: *Applied Energy* 82.3 (2005), pp. 228–254.

[16] P. Bardouille and J. Koubsky. “Incorporating sustainable development considerations into energy sector decision-making: Malmö Flintränen district heating facility case study”. In: *Energy Policy* 28.10 (2000), pp. 689–711.

[17] S. N. Petrovic and K. B. Karlsson. “Danish heat atlas as a support tool for energy system models”. In: *Energy Conversion and Management* 87 (2014), pp. 1063–1076.

[18] A. Meneghetti and G. Nardin. “Enabling industrial symbiosis by a facilities management optimization approach”. In: *J. of Cleaner Production* 35 (2012), pp. 263–273.

[19] E. Brembilla and A. Sciomachen. *Design and verification of a large size district heating network by a DSS*. 1990.

[20] C. Bordin, A. Gordini, and D. Vigo. “An optimization approach for district heating strategic network design”. In: *European J. of Operational Research* 252.1 (2016), pp. 296–307.

[21] A. Sciomachen and R. Sozzi. “The algorithmic structure of a decision support system for a design of a district heating network”. In: *Computers & Operations Research* 17.2 (1990), pp. 221–230.

[22] B.-J. Chen, M.-W. Chang, et al. “Load forecasting using support vector machines: A study on EUNITE competition 2001”. In: *IEEE Transactions on Power Systems* 19.4 (2004), pp. 1821–1830.

[23] A. K. Jain. “Data clustering: 50 years beyond K-means”. In: *Pattern Recognition Letters* 31.8 (2010), pp. 651–666.

[24] V. Vapnik. *Statistical learning theory. 1998*. Wiley, New York, 1998.

[25] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2016.

[26] RDevelopment CORE TEAM, R and others. *R: A language and environment for statistical computing*. 2008.

[27] K. Hornik, C. Buchta, and A. Zeileis. “Open-source machine learning: R meets Weka”. In: *Computational Statistics* 24.2 (2009), pp. 225–232.

[28] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. “The WEKA data mining software: An update”. In: *ACM SIGKDD Explorations Newsletter* 11.1 (2009), pp. 10–18.

[29] R. Kohavi et al. “A study of cross-validation and bootstrap for accuracy estimation and model selection”. In: *IJCAI*. Vol. 14. 2. Montreal, Canada. 1995, pp. 1137–1145.

11

District Heating Substation Behaviour Modelling for Annotating the Performance

Shahrooz Abghari, , Veselka Boeva, Jens Brage, Christian Johansson

In: Cellier P., Driessens K. (eds) Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2019. Communications in Computer and Information Science, vol 1168. Springer, Cham.
DOI: 10.1007/978-3-030-43887-6_1

Abstract

In this ongoing study, we propose a higher order data mining approach for modelling district heating (DH) substations' behaviour and linking operational behaviour representative profiles with different performance indicators. We initially create substation's operational behaviour models by extracting weekly patterns and clustering them into groups of similar patterns. The built models are further analyzed and integrated into an overall substation model by applying consensus clustering. The different operational behaviour profiles represented by the exemplars of the consensus clustering model are then linked to performance indicators. The labelled behaviour profiles are deployed over the whole heating season to derive diverse insights about the substation's performance. The results show that the proposed method can be used for modelling, analyzing and understanding the deviating and sub-optimal DH substation's behaviours.

11.1 Introduction

A district heating (DH) system provides an entire town, or part of it, with heat. The heat is generated in a central boiler and delivered via a distribution pipe network. The provided heat transfers through DH substations from the distribution network into consumers' buildings. This includes providing both space heating for heating seasons and domestic hot water (DHW) for a whole year. The DH system consists of two sides: *primary* and *secondary*. The primary side includes a central boiler, a distribution network (pre-insulated pipes) and consumers' buildings. The secondary side consists of a heat exchanger, a main piping system of the building, and radiators, convectors, or floor heating for the rooms.

The DH substations are made up of different components and each can be a potential source of faults. Faults in substations and the secondary side can be divided into three categories 1) faults resulting in comfort problems such as lack of enough heat, 2) unsolved faults with known cause since their identification are time demanding and costly, and 3) faults that require advanced fault detection systems [1]. Faults in substations do not necessarily result in comfort problems for the consumers, instead in most cases cause sub-optimal behaviour for a long time before they are noticed. Therefore, early detection of faults and deviations can reduce the maintenance cost and help avoid abnormal event progression. Fault detection in DH substations can be performed by monitoring both primary and secondary sides or only primary side.

Gadd and Werner [1] showed that hourly meter readings can be used for detecting faults at DH substations. The authors identified three fault groups: 1) low average annual temperature difference, 2) poor substation control, and 3) unsuitable heat load patterns. The results of the study showed that addressing low average annual temperature differences are the most important issue that can improve efficiency of the DH systems. Nevertheless, unsuitable heat load patterns are probably the easiest and the most cost-effective problem to consider first. In a recent study [2], the authors applied clustering analysis and association rule mining to detect faults in DH substations. In another study, the authors [3] proposed a method based on gradient boosting regression to predict hourly mass flow of a well performing substation. Their built model was tested by manipulating well performed substation data to simulate two different scenarios. Calikus et al. [4] proposed an approach to automatically discover heat load patterns in DH systems. Heat load profiles reflected yearly heat usage in an individual building. Moreover, their discovery is crucial for ensuring effective DH operations and managements.

We propose a higher order mining (HOM) [1] approach for modelling a DH substation's operational behaviour and linking it with two performance indicators. At the modelling step, we use primary side features to build the substation behaviour model by extracting the substation's behaviour patterns on a weekly basis. Heat demand is strongly influenced by social factors, e.g., the need during weekdays versus weekends. However, the social patterns tend to repeat on a weekly basis. Therefore, by considering the time window of a week rather than a day, we can mitigate the social patterns and avoid discovering, e.g., the demand transition between weekdays and weekends. The extracted patterns are used to create weekly behaviour models by clustering them into groups of similar patterns. The built models are further analyzed and integrated into an overall substation model by applying consensus clustering. We consider the exemplars of the consensus clustering model as the substation representative operational behaviour profiles. Further, at the annotating step the exemplars are linked with the two performance indicators. These indicators are calculated by using features from both primary and secondary side data. The annotated behaviour profiles can be deployed over the whole heating season to derive diverse insights about the substation's performance. They can also be used to quantify the performance of incoming heating weeks.

11.2 Methods and Techniques

11.2.1 Sequential pattern mining

Sequential pattern mining is the process of finding frequently occurring patterns in a sequence dataset. The records of the sequence dataset contain sequences of events whose orders are important. We use the PrefixSpan algorithm [6] to extract frequent sequential patterns. PrefixSpan applies a prefix-projection method recursively to find sequential patterns. The prefix-based projection enables PrefixSpan to focus only on prefix sub-sequences and project on their corresponding postfix sub-sequences. This yields less projections which in turn reduces both the length and the number of sequences in the projected datasets.

[1] HOM is a sub-field of knowledge discovery that applies to non-primary, derived data or patterns to provide human-consumable results [5].

11.2.2 Clustering analysis

11.2.2.1 Affinity propagation:

We use the affinity propagation (AP) algorithm [7] for clustering the extracted patterns. AP is based on the concept of *message passing* between data points. Unlike clustering algorithms, such as k-means [8] which requires the number of clusters as an input, AP estimates the optimal number of clusters from the data. In addition, the chosen exemplars are real data points and representative of the clusters.

11.2.2.2 Consensus clustering:

Gionis et al. [9] proposed an approach for clustering based on the concept of aggregation, where a number of different clustering solutions are given on some datasets of elements. The objective is to produce a single clustering solution from those elements that agrees as much as possible with the given clustering solutions. Consensus clustering algorithms deal with similar problems to those treated by clustering aggregation techniques. Such algorithms aim to synthesize clustering information about the same phenomenon coming from different sources [10] or from different runs of the same algorithm [11]. In this study, we use the consensus clustering schema proposed in [10] in order to integrate the clustering solutions produced on the datasets collected on a weekly basis for the heating season. The exemplars of the produced clustering solutions are considered and divided into k clusters according to the degree of their similarity by applying the AP algorithm. Subsequently, clusters whose exemplars belong to the same partition are merged in order to obtain the final consensus clustering.

11.2.3 Distance measure

The similarity between the extracted patterns are assessed with a dynamic programming version of Levenshtein distance (LD) metric [12]. The LD, also known as edit distance, is a string similarity metric that measures the minimum number of editing operations required to transform one string into the other.

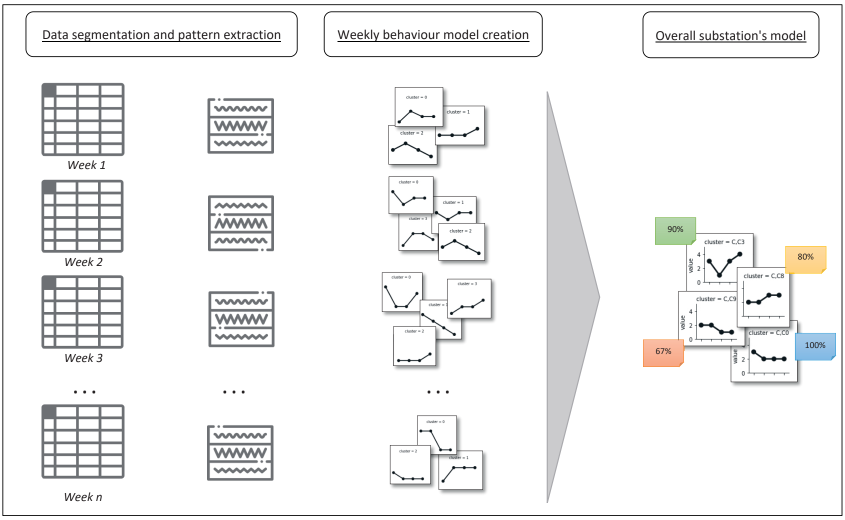


Figure 11.1: Schematic illustration of the proposed approach

11.3 Proposed Method

Our approach has a preprocessing step and two main steps: 1) *Modelling substation's operational behaviour*; 2) *Linking the substation's representative behaviour profiles with performance indicators.* The modelling step consists of three distinctive sub-steps: (i) data segmentation and pattern extraction, (ii) weekly behaviour model creation, and (iii) overall substation's model. The approach is schematically illustrated in Figure 11.1.

Data Preprocessing: In order to prepare data for the modelling step all duplicates are removed and missing values are imputed by averaging the neighbouring values. The first and the last missing values are replaced with the next and the previous available values, respectively.

In addition, extreme values that are often a result of faults in measurement tools are smoothed out by a Hampel filter [13], which is a median absolute deviation (MAD) based estimation. The filter computes the median, MAD, and the standard deviation (SD) over the data in a local window. We apply the filter with the default parameters; the size of the window is seven and the threshold for extreme value detection is three, i.e., 3-neighbours on either side of a sample. The threshold for extreme value detection is three. Therefore, in each window a sample with the distance three times the SD from its local median is considered as an extreme value and is replaced by the local median.

We monitor the operational behaviour of substations based on outdoor temperature and the primary side features of the DH system. Our motivation for this choice relates to the fact that the primary side data is always available while the secondary side data requires specific hardware that might not be available at the consumers' building. After discussions with domain experts, we chose five features that have a strong negative correlation with outdoor temperature. The selected features are: 1) primary return temperature, $T_{r,1^{st}}$, 2) primary temperature difference, $\Delta T_{1^{st}}$, 3) primary energy, $Q_{1^{st}}$, 4) primary mass flow rate, $G_{1^{st}}$, and 5) the substation performance indicator based on the hourly consumed energy divided into the hourly mass flow rate, E_s^{E-F}. The fifth feature represents how many units of energy one substation can provide from the consumed volume flow rate.

Z-score normalization is applied on each feature and for every week's period. The normalization is performed to make it possible to assess and compare a substation's operational behaviours in different weeks.

In order to build the DH substation's operational behaviour model using the HOM paradigm, continuous features are converted into categorized features. All five features together build patterns (sequences of events) that represent the operational behaviour of the substation. In this study, we are interested in contextual outlier detection. The context here is referred to as modelling the DH substation's behaviour, during only the heating season. For this purpose we have applied k-means-based discretization method by setting the size of k to four, similar to the number of seasons in Sweden.

1. **Modelling DH substation's operational behaviour:**

   (i) **Data segmentation and pattern extraction:** We extract the substation's behaviour patterns on a weekly basis. The PrefixSpan algorithm is used to find frequent sequential patterns with the length of five in each week. Those sequential patterns that satisfy the user-specified support are considered as frequent ones. The user-specified support threshold is set to be *one* to capture daily patterns, i.e., any patterns that appear at least once will be considered.

   (ii) **Weekly behaviour model creation:** The extracted patterns from each week are clustered into groups based on their similarities. Since the aim is to build a DH substation behaviour model for the heating season, all exemplars of the clustering models related to the weeks with the average outdoor temperature above 10 °C are filtered out.

(iii) **Overall substation's model:** The weekly behaviour models built at the previous step are further integrated into an overall substation's behaviour model by applying a consensus clustering technique. The exemplars of the consensus clustering solution are considered as representative profiles for the substation's behaviour, i.e., they can be used to further analyze the substation's behaviour and performance for the whole heating season.

2. Linking behaviour profiles with performance indicators: At this step the derived substation's behaviour profiles are linked to performance indicators. In the current study, we annotate behaviour profiles with two performance indicators: *substation effectiveness* and *grädigkeit*. The two indicators are computed by considering features from both the primary and secondary sides.

Substation effectiveness is computed as $E_s^T = \frac{\Delta T_{1^{st}}}{T_{s,1^{st}} - T_{r,2^{nd}}}$ where, $\Delta T_{1^{st}}$ is the difference between primary supply and return temperatures, $T_{s,1^{st}}$ is the primary supply temperature, and $T_{r,2^{nd}}$ is the return temperature at the secondary side. The efficiency of a well-performed substation should be close to 1 in a normal setting. However, due to the affect of DHW generation on the primary return temperature, the E_s^T can be above 1.

Grädigkeit indicator, also known as the least temperature difference [2], represents the difference between primary and secondary return temperatures and it is computed as $\Delta T_{r,(1^{st},2^{nd})} = T_{r,1^{st}} - T_{r,2^{nd}}$. The grädigkeit of a substation can be greater than or equal to zero, though it can go below zero due to usage of DHW. A lower value of grädigkeit implies better performance.

For each considered performance indicator, we partition the substation's representative behaviour profiles into three categories with respect to the associated performance indicator scores: *low*, *medium* and *high*. In that way, we have a group of behaviour profiles that represents the substation's sub-optimal performance and two groups of profiles that are linked with satisfactory and optimal substation's performance, respectively. The labelled

[2] Frederiksen, S., Werner, S.: District heating and cooling, Studentlitteratur Lund (2013)

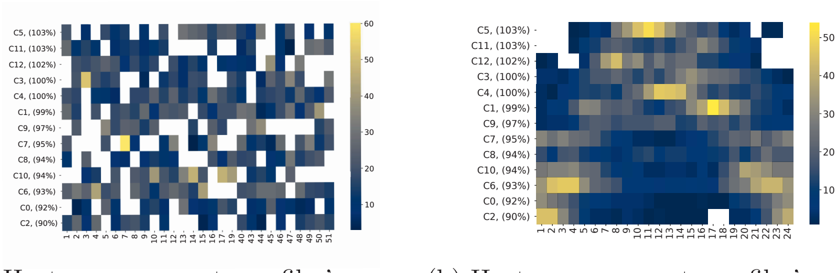
behaviour profiles can be deployed over the whole heating season in order to further analyze and understand the substation's operational behavior and performance. For example, the profiles from the three different categories can be used to interpret the substation's operational behaviours for particular time intervals. In addition, it is possible to backtrack from these higher order representative profiles to the weekly behaviour models and to the hourly patterns.

11.4 Results and Discussion

We studied substations' operational behaviour for ten buildings in 2017. We first modeled each substation's weekly operational behaviours. This was performed by grouping the extracted frequent patterns into clusters of similar patterns. We then stored the exemplars of the built clustering model if the average outdoor temperature of the week was less than or equal to 10 °C. This step is motivated by the fact that we want to model the substation's overall operational performance for the whole heating season. The collected exemplars were integrated into a consensus clustering. At last, the obtained consensus clustering model was linked (annotated) with the selected performance indicators. The extracted profiles with respect to each indicator were used to assess behaviour of the substation on a weekly basis.

For the rest of this section we focus on one specific building, B-21. We identified 13 profiles that model the operational behaviour of the substation for the heating season. The extracted profiles were linked with the two performance indicators, *substation effectiveness* and *grädigkeit*. In order to facilitate further analysis, the profiles were sorted from the highest to the lowest performance separately for each indicator. For example, in case of the substation effectiveness the profiles are within a range from 103% to 90%. Regarding the grädigkeit, the profiles are within a range from -2.15 °C to 5.37 °C.

Figure 11.2 a shows the substation's effectiveness according to the built profiles for each week. As one can notice the heatmap is sparse and only few weeks, e.g., weeks 3, 4, 7, 10, 14, 15, 17, and 18 represent a high number of frequency for some of the profiles. The heatmap is not easy to interpret and it does not provide interesting information about the substation's weekly behaviour.



(a) Heatmap represents profiles' frequency in each week.

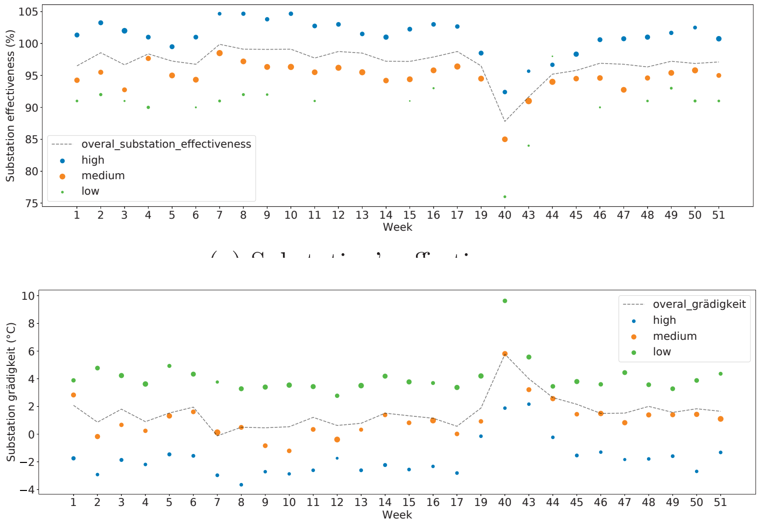
(b) Heatmap represents profiles' frequency in 24-hour period.

Figure 11.2: The deployment of the annotated profiles according to *substation effectiveness* for building B-21 over 2017 heating season.

Figure 11.2 b, on the other hand, provides more information by showing the effectiveness of the same substation at a 24-hour period for the whole heating season. For example, one can recognize a yellowish bell shape. Evidently, the substation performed on average 92% at early morning (0:00-5:00) and late evening (20:00-23:00). However, for the rest of the day the performance of the substation is closer to and above 100%. The low performance of the substation might be due to social behaviour, which demonstrates low heat demand in the early morning and late evening.

As mentioned before, we categorize the extracted profiles with respect to their performance indicator labels (substation effectiveness or grädigkeit) into three categories: low, medium, and high. In the case of substation effectiveness *low* represents efficiency below 90%, *medium* indicates efficiency between 90% to 100%, and high stands for efficiency above 100%. Figure 11.3 a shows the overall effectiveness of the substation over the weeks that space heating was required, based on these three categories. As one can see the orange circles, which represent the medium efficiency of the substation, closely follow the curve showing overall substation's effectiveness. This is also valid for the profiles from the other two categories. For example, all profiles linked to optimal performance (blue circles in Figure 11.3 a) are above the overall substation's effectiveness curve. In Figure 11.3 a, we can also notice that weeks 19 and 40 represent the end and beginning of the heating season, respectively. The low efficiency of the substation in week 40 might be related to the fact the system required sometime to adjust.

Regarding grädigkeit indicator, *low* represents temperature differences above 3 °C, *medium* denotes temperatures between 0 to 3 °C, and high



(a) Substation's effectiveness.

(b) Substation's grädigkeit.

Figure 11.3: The deployment of the annotated profiles according to performance indicators for building B-21 over 2017 heating season.

shows temperature differences equal or below to 0 °C. Figure 11.3 b shows the overall grädigkeit for the studied substation. Similar to Figure 11.3 a, the medium category is closely following the curve that represents the overall substation's grädigkeit. Notice that for grädigkeit indicator the temperature differences close to and below zero show a high efficiency.

11.5 Conclusion and Future Work

We proposed a higher order mining approach for modelling a district heating substation's operational behaviour. The method summarized the substation's behaviour with a series of representative profiles that were linked with two performance indicators. The labelled profiles were deployed over the whole heating season to assess an overall substation's behavior and performance. We applied and studied our method on ten buildings. The initial results showed that the proposed method can be used to analyze and evaluate the operational behaviour of DH substations.

For future work we are interested in studying whether the derived representative behaviour profiles can be used to quantify the performance of incoming heating weeks. In addition we plan to evaluate our approach with other performance indicators.

References

[1] H. Gadd and S. Werner. “Fault detection in district heating substations”. In: *Applied Energy* 157 (2015), pp. 51–59.

[2] P. Xue, Z. Zhou, X. Fang, X. Chen, L. Liu, Y. Liu, and J. Liu. “Fault detection and operation optimization in district heating substations based on data mining techniques”. In: *Applied Energy* 205 (2017), pp. 926–940.

[3] S. Månsson, P.-O. J. Kallioniemi, K. Sernhed, and M. Thern. “A machine learning approach to fault detection in district heating substations”. In: *Energy Procedia* 149 (2018), pp. 226–235.

[4] E. Calikus, S. Nowaczyk, A. Sant’Anna, H. Gadd, and S. Werner. “A Data-Driven Approach for Discovery of Heat Load Patterns in District Heating”. In: *arXiv preprint arXiv:1901.04863* (2019).

[5] J. F. Roddick, M. Spiliopoulou, D. Lister, and A. Ceglar. “Higher order mining”. In: *ACM SIGKDD Explorations Newsletter* 10.1 (2008), pp. 5–17.

[6] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M. Hsu. “Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth”. In: *Proc. of the 17th Int’l Conf. on Data Engineering*. 2001, pp. 215–224.

[7] B. J. Frey and D. Dueck. “Clustering by passing messages between data points”. In: *Science* 315.5814 (2007), pp. 972–976.

[8] J. MacQueen et al. “Some methods for classification and analysis of multivariate observations”. In: *Proc. of the Fifth Berkeley Symp. on Mathematical Statistics and Probability*. Vol. 1. 14. Oakland, CA, USA. 1967, pp. 281–297.

[9] A. Gionis, H. Mannila, and P. Tsaparas. “Clustering Aggregation”. In: *ACM Transaction of Knowledge Discovery Data* 1.1 (2007).

[10] V. Boeva, E. Tsiporkova, and E. Kostadinova. “Analysis of Multiple DNA Microarray Datasets”. In: *Springer Handbook of Bio-/Neuroinformatics*. Springer Berlin Heidelberg, 2014, pp. 223–234.

[11] A. Goder and V. Filkov. "Consensus Clustering Algorithms: Comparison and Refinement". In: *ALENEX*. 2008, pp. 109–234.

[12] V. I. Levenshtein. "Binary codes capable of correcting deletions, insertions, and reversals". In: *Soviet physics doklady*. Vol. 10. 8. 1966, pp. 707–710.

[13] F. R. Hampel. "A general qualitative definition of robustness". In: *The Annals of Mathematical Statistics* (1971), pp. 1887–1896.

12

Multi-view Clustering Analyses for District Heating Substations

Shahrooz Abghari, , Veselka Boeva, Jens Brage, Håkan Grahn

In: 2020 9th International Conference on Data Science, Technology and Applications. (pp. 158-168).
DOI: 10.5220/0009780001580168

Abstract

In this study, we propose a multi-view clustering approach for mining and analysing multi-view network datasets. The proposed approach is applied and evaluated on a real-world scenario for monitoring and analysing district heating (DH) network conditions and identifying substations with sub-optimal behaviour. Initially, geographical locations of the substations are used to build an approximate graph representation of the DH network. Two different analyses can further be applied in this context: step-wise and parallel-wise multi-view clustering. The step-wise analysis is meant to sequentially consider and analyse substations with respect to a few different views. At each step, a new clustering solution is built on top of the one generated by the previously considered view, which organizes the substations in a hierarchical structure that can be used for multi-view comparisons. The parallel-wise analysis on the other hand, provides the opportunity to analyse substations with regards to two different views in parallel. Such analysis is aimed to represent and identify the relationships between substations by organizing them in a bipartite graph and analysing the substations' distribution with respect to each view. The proposed data analysis and visualization approach arms domain experts with means for analysing DH network performance. In addition, it will facilitate the identification of substations with deviating operational behaviour based on comparative analysis with their closely located neighbours.

12.1 Introduction

District heating (DH) systems utilize hot water and heat produced at a *production unit* for a number of *consumer units*, i.e., buildings, in a limited geographical area through a *distribution network*. This part of the system is referred to as the *primary* side. The consumer unit itself consists of a heat exchanger, a circulation network, and radiators for the rooms, which are considered as the secondary side. The primary and secondary sides are connected together through a substation, which is responsible for adjusting the pressure and the temperature of the supply water suitable for the consumer unit.

In the DH domain, energy companies need to address several conflicting goals such as satisfying consumers' heat demand including domestic hot water (DHW) while minimizing production and distribution costs. Such complexity demands fault detection and root cause analysis techniques for identification of deviating behaviours and faults. Undetected faults can lead to underlying problems, which in return can increase the maintenance cost and reduce the consumers' satisfaction. When it comes to monitoring of a DH network there are different features and characteristics that one needs to consider. Domain experts often analyse substations individually or in a group with regard to one specific feature or a combination of features. While this provides useful information for the whole network it does not take into account the location of the substations along the distribution network and their neighbouring substations automatically. In other words, the operational behaviours of the DH substations need to be assessed jointly with surrounding substations within a limited geographical distance. Due to the nature of the data and the fact that different data representations can be used, the process of monitoring and identifying faults and deviating behaviours of the DH system and substations can be treated as a multi-view data analysis problem.

Multi-view datasets consist of multiple data representations or views, where each one may contain several features [1]. Multi-view learning is a semi-supervised approach with the goal to obtain better performance by applying the relationship between different views rather than one to facilitate the difficulty of a learning problem [2–4]. Due to availability of inexpensive unlabeled data in many application domains, multi-view unsupervised learning and specifically multi-view clustering (MVC) attract great attention [1]. The goal of multi-view clustering is to find groups of similar objects based on multiple data representations.

We propose a multi-view clustering analysis approach for mining network datasets with multiple representations. The proposed approach is used for monitoring a DH network and identifying DH substations with sub-optimal operational behavior. We initially use geographical location of substations to divide them into groups of similar substations based on their distance and location. In that way, we are able to: 1) group the substations (network nodes) based on their location and distance, 2) build an approximate graph representation of the DH network, and 3) order the substations using information about the DH network structure and the average supply water temperature for a specific period. Two different types of analyses can then be applied in this scenario: i) step-wise clustering to sequentially consider and analyse substations with respect to a few different views; ii) parallel-wise clustering to analyse substations with regards to two different views in parallel.

12.2 Related Work

MVC clustering algorithms have been proposed based on different frameworks and approaches such as k-means variants [5–7], matrix factorization [8, 9], spectral methods [10, 11] and exemplar-based approaches [12, 13].

Bickel and Scheffers [5] proposed extensions to different partitioning and agglomerative MVC algorithm. That study can probably be recognized as one of the earliest works where an extension of k-means algorithm for two-view document clustering is proposed. In another study [6], the authors developed a large-scale MVC algorithm based on k-means with a strategy for weighting views. The proposed method is based on the $\ell_{2,1}$ norm, where the ℓ_1 norm is enforced on data points to reduce the effect of outlier data and the ℓ_2 norm is applied on the features. In a recent study, Jiang et al. [7] proposed an extension of k-means with a strategy for weighting both views and features. Each feature within each view is given bi-level weights to express its importance both at the feature level and the view level.

Liu et al. [8] proposed an MVC algorithm based on joint non-negative matrix factorization (NMF). The developed algorithm incorporates separate matrix factorizations to achieve similar coefficient matrices and further meaningful and comparable clustering solution across all views. In a recent study, Zong et al. [9] proposed an extension of NMF for MVC that is based on manifold regularization. The proposed framework maintains the locally geometrical structure of multi-view data by including consensus manifold and consensus coefficient matrix with multi-manifold regularization.

Kumar and Daumé [10] proposed an MVC algorithm for two-view data by combining co-training and spectral clustering. The approach is based on learning the clustering in one view to *label* the data and modify the similarity matrix of the other view. The modification of the similarity matrices are performed using discriminative eigenvectors. Wang et al. [11] proposed a variant of spectral MVC method for situations where there are disagreements between data views using Pareto optimization as a means of relaxation of the agreement assumption.

Meng et al. [12] proposed an MVC algorithm based on affinity propagation (AP) for scientific journal clustering where the similarity matrices of the two views (text view and citations view) are integrated as a weighted average similarity matrix. In another study, Wang et al. [13] proposed a variant of AP where an MVC model consists of two components for measuring 1) the within-view clustering quality and 2) the explicit clustering consistency across different views.

Fault detection and diagnosis (FDD) is an active field of research and has been studied in different application domains. Isermann [14, 15] provided a general review for FDD. Katipamula and Brambley [16, 17] conducted an extensive review in two parts on fault detection and diagnosis for building systems. Xue et al. [18] applied clustering analysis and association rule mining to detect faults in substations. Sandin et al. [19] used probabilistic methods and heuristics for automated detection and ranking of faults in large-scale district energy systems. Calikus et al. [20] proposed an approach for automatically 1) discovering heat load patterns in DH systems and 2) identifying buildings with abnormal heat profiles and unsuitable control strategies.

In contrast to the above mentioned methods, this study proposes a multi-view data analysis approach that can be applied for monitoring, evaluating and visualizing the operational behaviour of DH substations. The geographical location data is used as a backbone of the analysis and the operational performance of the substations is further assessed in conjunction with their neighbours.

12.3 Problem Formalization

We have a network with N nodes, e.g., a DH network linking a set of substations located in some geographical region. Assume that each network node, substation, i ($i = 1, 2, \ldots, N$) is monitored under n different conditions (i.e., the measurements of n different features are collected) for a given time period, e.g., m days. Each monitored condition j ($j = 1, 2, \ldots, n$) contains the measured levels of the corresponding feature for a period of m days in t different time points. This leads to a set of n time series data matrices D_j ($j = 1, 2, \ldots, n$), one per feature, for each network node.

This multi-view data context can additionally be complicated in the case of a real-world scenario such as one related to a DH network. For example, the operational behaviour of the substations varies during heating and non-heating seasons which requires separate analysis. Therefore, for each substation two datasets are usually collected and available for further analysis and comparison. Notice that in this study, we are only interested in the operational behaviour of the DH substations during heating season due to the importance of space heating.

The main challenge in the above multi-view context is how to use all available measurements about the substations' operational behaviour and performance for better understanding and improved maintenance of the DH network. Exploiting the whole potential of these real-world datasets is not trivial and it requires suitable data analysis techniques to prevent information loss.

12.4 Methods

12.4.1 Clustering Analysis

In this study, we are interested in identifying homogeneous groups of substations by considering their locations and additionally analysing them with respect to different views (features). Due to unavailability of the labeled data, clustering analysis is applied to explore hidden structures within the data. We apply two clustering algorithms as follows:

1.) *Minimum Spanning Tree Clustering*: We use VanderPlas' [21] Python implantation of the minimum spanning tree (MST) clustering algorithm for grouping substations based on their geographical location. The algorithm is based on constructing an approximate Euclidean minimum spanning tree (EMST), which considers only k nearest neighbours of each point for building the minimum spanning tree rather than the entire set of edges in a complete graph.

2.) *Affinity Propagation*: We use the affinity propagation (AP) algorithm [22] for clustering the time series based on their similarities. AP works based on the concept of *message passing* between data points to first identify a suitable set of exemplars and then to choose which data points should pick which exemplars. One of the advantages of AP, unlike other clustering algorithms, such as k-means [23] which requires the number of clusters as an input, is that it estimates the optimal number of clusters from the data. In addition, the chosen exemplars, the representative of the clusters, are real data points which makes AP a suitable clustering algorithm for this study.

12.4.2 Similarity Measures

We use different similarity measures, 1) to check the similarity between daily time series profiles of each feature, 2) to perform pairwise comparison between exemplars of clustering solutions of different substations, and 3) to compute a similarity between two clustering solutions by considering all pairs of members. These similarity measures are as follows:

1. *Dynamic Time Warping*: Given two time series $Y = (y_1, y_2, ..., y_n)$ and $Y' = (y'_1, y'_2, ..., y'_m)$, the similarity between Y and Y' can be measured using the dynamic time warping (DTW) algorithm. DTW is proposed by Sakoe and Chiba [24] for spoken word detection with the focus of eliminating timing differences between two speech patterns. In other words, DTW identifies an optimal matching between the given sequences by warping the time axis. In order to align the time series Y and Y' of length n and m respectively, a cost matrix, $Q_{n\times m}$ is computed. Each element, q_{ij}, of $Q_{n\times m}$ corresponds to the distance (often Euclidean) between y_i and y'_j of the two series. Using the cost matrix, the DTW tries to find the best alignment path between these two time series that is leading to minimum overall cost. The best warping path should satisfy a different number of conditions such as monotonicity, continuity, boundary, warping window, and slope constraint.

2. *Clustering Solution Similarities*: Given two clustering solutions $C = \{C_1, C_2, \ldots, C_n\}$ and $C' = \{C'_1, C'_2, \ldots, C'_m\}$ of datasets X and X', respectively, the similarity, C_{S_w}, between C and C' can be assessed as follows [25]:

$$C_{S_w}(C, C') = \frac{\sum_{i=1}^{n}(min_{j=1}^{m} w_i.d(c_i, c'_j))}{2} + \frac{\sum_{j=1}^{m}(min_{i=1}^{n} w'_j.d(c_i, c'_j))}{2}, \tag{12.1}$$

where c_i and c'_j are exemplars of the clustering solution C_i and C'_j, respectively. The weights w_i and w'_j indicate the relative importance of clusters C_i and C'_j compared to other clusters in the clustering solution C and C', respectively. For example, a weight w_i of a cluster C_i can be calculated as the ratio of its cardinality with respect to the size of X, i.e., $w_i = |C_i|/|X|$. The C_{S_w} has values in a range of [0,1]. Scores equal to zero imply identical performance while scores close to one show significant dissimilarities.

3. *Adjusted Rand Index*: The quality of the results of a clustering analysis can be validated by means of *internal* and *external* criteria. Internal criteria evaluate the quality of the clustering solution produced by a clustering algorithm that fits the data in terms of, e.g., compactness and separation by using the inherent information of the data. External criteria on the other hand, can be used for measuring the level of agreements between the results of a clustering algorithm in comparison with ground truth, the results of another clustering algorithm on the same data, or same clustering algorithm but by considering different views.

In this study, we apply a symmetric external validation index for assessing the similarity (consensus) between two clustering results generated on the studied DH substations with respect to two different views. The adjusted Rand index (ARI) [26] is a correction of the Rand index (RI) [27] that measures the similarity between two clustering solutions by considering the level of agreements between the two groups. ARI is computed as follows:

$$ARI = \frac{RI - ExpectedRI}{Max(RI) - ExpectedRI} \tag{12.2}$$

ARI scores are bound between -1 and +1. A score less than or equal to 0 represents random labelling and 1 stands for perfect match.

12.5 Proposed Approach

Geographical locations of N substations are initially used for building an approximate graph representation of the DH network. We refer to the geographical location of the substations as the Location view (v_0). This is performed by applying the MST clustering algorithm described in Section 12.4. The aim is to connect substations based on their distance by building a minimum spanning tree and removing edges of the tree with regard to a cut-off threshold. Therefore, each cluster is represented by a tree that can be interpreted as a representation of the DH network structure. In order to provide additional support for the domain experts, the graph representation can in turn be used as a backbone for additional information about the DH network, e.g., average yearly values and different forms of ranking.

On the foundation of the created grouping of the substations we can perform further analysis by focusing on a specific feature or subset of features and evaluate the substations' operational behaviours in each single location-based cluster. We study and evaluate the following two scenarios:

1. *Step-wise multi-view clustering* (SW-MVC), we can apply clustering analysis on substations that have been grouped together at the previous step with respect to a set of features, i.e., the substations can be grouped by considering one feature at a time. This scenario can be used when the domain experts are interested in grouping similar substations based on their performance with respect to one feature and then finding similar substations in each group by using another feature and so on. Figure 12.1 shows how the results of this analysis can be visualized based on the location of the substations and two features.

As an example, consider two substations s_i and s_j in the cluster C_{00} from v_0, where the similarity of the two substations can be analysed in terms of their operational behaviour, first based on Feature 1 (v_1) and then Feature 2 (v_2). Here two scenarios can occur, either s_i and s_j are grouped together with respect to v_1, since they performed similarly, or they are assigned into different groups. In case of the first scenario, after applying the second step of the analysis (i.e., using v_2) if s_i and s_j are in the same group this shows that the operational behaviour of the two substations are similar with respect to v_1 and v_2. Otherwise, the two substations are only similar with regards to v_1 and dissimilar with regards to v_2. In case of the second scenario, the

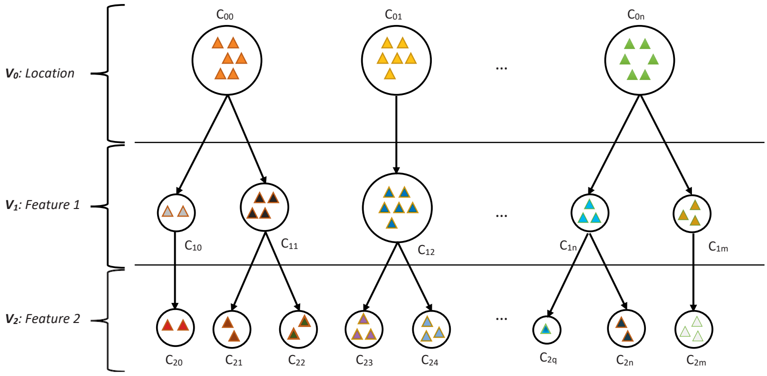


Figure 12.1: SW-MVC analysis, each view represents the clustering analysis based on one feature. Every analysing step is based on the results obtained on the previously considered view. Triangles represent substations.

substations are dissimilar with respect to both views. Nevertheless, in all cases the domain expert might be interested to further analyse groups of substations with a smaller size.

2. *Parallel-wise multi-view clustering* (PW-MVC), in this scenario a group of substations can be studied by considering different features in parallel. For example, the substations can be clustered separately with respect to two different features (or subsets of features). The produced clustering solution can further be compared and analysed to find out whether similar substations, that have been grouped together based on one feature, are still in the same group with respect to the other feature. In addition, one can use a *bipartite graph*, to present and visualize the relationships between a clustering solution based on one view and a clustering solution produced on the other view. This will provide domain experts more information by supplying them with deeper insights about substations' operational behaviours in different groups with regards to two different views. For further analysis, one can label clusters of each clustering solution with performance indicators [1] and rank them from the highest to the lowest performance.

The results of this pairwise comparison can be used in conjunction with the SW-MVC analysis to provide a better understanding of operational

[1] The operational performance of a DH substation can be evaluated with respect to different indicators, which are usually computed based on the quantitative relation between the substation's inputs and outputs.

behaviour for each individual substation and the group as a whole. Our initial assumption is that using the SW-MVC analysis, one can construct a hierarchical graph-model of a heating network for the area of study. Those substations that are located in the same cluster are assumed to share similar characteristics. While the PW-MVC analysis focus is on identifying a similar group of substations that are in the intersection of the two views.

12.6 Experiments and Evaluation

12.6.1 Dataset

The data used in this study is provided by an energy company. The data consists of hourly average measurements from 70 substations located in Southern Sweden during 2015 to 2018. The dataset contains eight features both from primary and secondary sides of the DH network. The primary side data is always available. The secondary side data on the other hand, requires specific hardware to be extracted. Therefore, in this study we mainly focus on primary side data to analyse the operational behaviour of the DH substations.

Apart from these features there are two performance indicators that are computed using both sides of the DH network. The first indicator is called the *least temperature difference* [28] which represents the difference between primary and secondary return temperatures. The least temperature difference of a substation can be greater than or equal to zero, though it can go below zero due to usage of DHW. A lower value of this indicator implies better performance. The second indicator is referred to as *substation effectiveness.* It is the ratio of the difference between primary supply and return temperatures to the difference between primary supply temperature and the secondary return temperature. The efficiency of a well-performing substation should be close to one in a normal setting. However, due to the affect of DHW generation on the primary return temperature, it can represent values above one. Table 12.1 shows the dataset features and the performance indicators.

Figure 12.2 shows the groups and graph network representation produced by applying MST clustering on the above mentioned 70 substations. The substations are partitioned into nine clusters by applying the MST clustering algorithm while the cut-off parameter is set to 500 meters. That is substations with distance less than 500 meters from their closest neighbour(s) are grouped

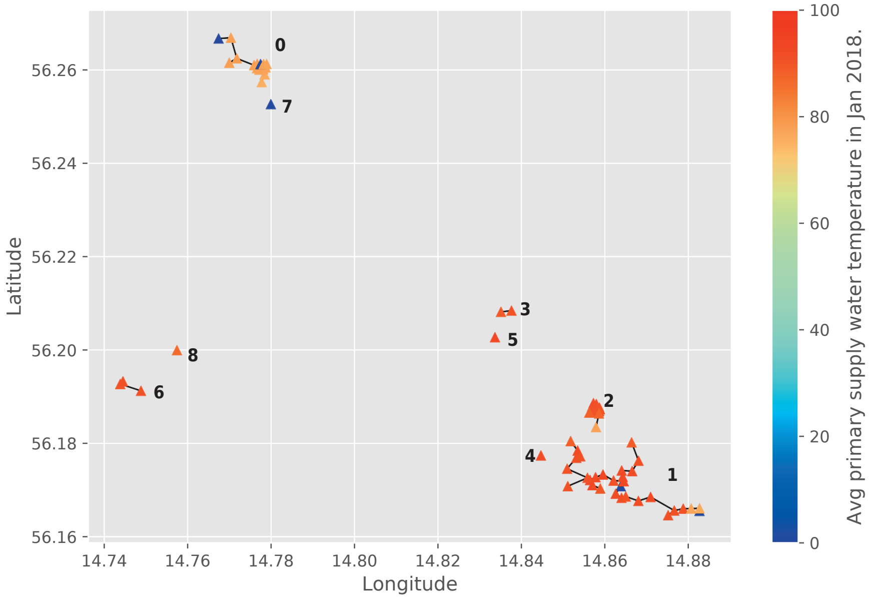


Figure 12.2: 70 substations located in Southern Sweden are grouped into nine clusters using the MST clustering algorithm. The geographical location of the substations is referred to as the Location view (v_0). Substations with distance less than 500 meters from their closest neighbours are grouped together. The color of the substations represents the average $T_{s,1^{st}}$ in January 2018, which for most substations is around 87 °C.

together. Five clusters represent as a tree, i.e., edges of the tree represent the distance between the substations (the tree nodes) and the remaining four clusters are singletons. The substations' colors represent their received average $T_{s,1^{st}}$ (°C) in January 2018.

Table 12.1: Features included in the dataset

No.	Feature	Notation	Unit
1	T_o	Outdoor temperature	°C
2	$T_{s,1^{st}}$	Primary supply temperature	°C
3	$T_{r,1^{st}}$	Primary return temperature	°C
4	$\Delta T_{1^{st}}$	Primary delta temperature	°C
5	$G_{1^{st}}$	Primary mass flow rate	l/h
6	$Q_{1^{st}}$	Primary heat	kW
7	$T_{s,2^{nd}}$	Secondary supply temperature	°C
8	$T_{r,2^{nd}}$	Secondary return temperature	°C

In order to make the data ready for the experiment, first the duplicates are removed. Then we focused on extreme values which can appear as a result of faults in measurement tools. We apply a Hampel filter [29] which is a median absolute deviation (MAD) based estimation to detect and smooth out such extreme values. The filter is used with the default parameters, i.e., the size of the window is set to be seven and the threshold for extreme value detection is set to be three.

In the studied context, we have hourly measurements data. This gives one time series every 24-hours and in total 365 time series per year. Time series with less than 24 measurement values are excluded. Since we are expecting different behaviours from a DH substation during heating and non-heating seasons, the time series are divided into two groups with respect to the outdoor temperature (T_o). That is, if the outdoor temperature is above a certain threshold, $T_{o_{threshold}}$, the DH substation behaviour can be categorized into the non-heating season otherwise to the heating season. This threshold in Sweden can be set to be $T_{o_{threshold}} = 10$ °C. In order to assess each DH substation's operational behaviours during heating season and in comparison with other substations, the extracted time series are scaled with z-score normalization. That is, each time series is scaled to have a mean of zero and a standard deviation of one. Notice that in the considered context the general shape of the time series, rather than their amplitude, is important. Now for every category, the time series related to one specific feature, i, can be compared in terms of similarity with respect to a distance measure $d(y_i, y'_j)$, where d in this study is DTW. This leads to a similarity matrix, SM_i. In the next step, SM_i is fed to a clustering algorithm. Here we aim to group time series based on their similarities into a number of clusters. Considering each feature as one view, we can analyse the operational behaviour of a set of DH substations by using the explained evaluation scenarios in Section 12.5.

12.6.2 Implementation and Availability

The proposed approach is implemented in Python version 3.6. The affinity propagation and the adjusted Rand index are adopted from the scikit-learn module [30] and the MST clustering algorithm is fetched from [21]. The alignments between time series are identified using dtwalign's package [2]. The

[2] https://github.com/statefb/dtwalign

implemented code and the experimental results are available at GitHub[3].

12.7 Results and Discussion

The initial view in our analyses is always the outcome of the MST clustering, i.e., 70 substations in the studied area are grouped into nine clusters based on their distances. We set the cut-off parameter to be 500 meters which means any edges greater than 500 meters are removed from the MST. The first three clusters (0, 1, and 2) include 15, 32, and 14 substations, respectively (in total 61 of 70 substations). The remaining substations are grouped into 6 clusters as follows: cluster 3 contains 2 substations, clusters 4, 5, 7, and 8 are singletons and cluster 6 has 3 substations.

In the remainder of this study we only consider and discuss the results produced on clusters 0, 1, and 2, since the majority of the substations are distributed in these clusters. Each analysis can be performed based on different combinations of the features in Table 12.1. However, due to the page limit, we only report the results of the analyses with respect to $T_{r,1^{st}}$ and $\Delta T_{1^{st}}$ (the difference between $T_{s,1^{st}}$ and $T_{r,1^{st}}$).

12.7.1 SW-MVC Analysis

Table 12.2 shows the results of SW-MVC analysis for 61 substations throughout the heating seasons from 2015 to 2018. For each MST cluster (v_0), initially substations are grouped based on $T_{r,1^{st}}$ (v_1) and then for each created subgroup the clustering analysis is performed using $\Delta T_{1^{st}}$ (v_2). The information in Table 12.2 can be used in three different ways: column-wise, row-wise, or both. In the column-wise case, one can see how the substations in each MST cluster are grouped based on the other two views (i.e., v_1 and v_2) in different years. The row-wise analysis shows how the substations in each MST cluster are grouped step-wise based on first v_1 and second v_2. For example, the domain experts might be interested in performing further analysis when the grouped substations in v_1 are split into more subgroups based on v_2. Numbers in bold in Table 12.2 represent the number of substations that are grouped into different clusters based on v_2 as opposed to v_1. By considering both cases one can track the transition of the operational behaviour of substations throughout the years.

[3] https://github.com/shahrooz-abghari/MVC-DH-Monitoring

Table 12.2: SW-MVC analysis based on $T_{r,1^{st}}$ and $\Delta T_{1^{st}}$ from 2015 to 2018.

Year	$v_0 : MST$	$v_1 : T_{r,1^{st}}$		$v_2 : \Delta T_{1^{st}}$			
	Label	#substations	Label	0	1	2	Total
2015	0	5	0	5			5
		10	1	**1**	**9**		10
	1	32	0	**11**	**12**	**9**	32
	2	5	0	**3**	**2**		5
		9	1	**6**	**3**		9
2016	0	3	0	3			3
		5	1	**4**	**1**		5
		7	2	7			7
	1	15	0	**9**	**6**		15
		13	1	**7**	**6**		13
		4	2	4			4
	2	9	0	9			9
		5	1	5			5
2017	0	8	0	8			8
		1	1	1			1
		5	2	5			5
		1	3	1			1
	1	11	0	**8**	**3**		11
		4	1	**1**	**3**		4
		16	2	**8**	**8**		16
		1	3	1			1
	2	2	0	2			2
		9	1	9			9
		3	2	3			3
2018	0	1	0	1			1
		14	1	**10**	**4**		14
	1	12	0	12			12
		5	1	5			5
		8	2	**5**	**3**		8
		7	3	**6**	**1**		7
	2	4	0	**1**	**3**		4
		4	1	4			4
		1	2	1			1
		5	3	5			5

*Note. Number of substations that are grouped into different clusters based on v_2 as opposed to v_1 are shown in **bold**.*

Figure 12.3 depicts the SW-MVC analysis by considering $T_{r,1^{st}}$ as the first view (squares) and $\Delta T_{1^{st}}$ as the second view (circles) in the period from 2015 to 2018. Figure 12.4 represents the SW-MVC analysis specifically for the MST cluster with label 0 during a period covering 2017 and 2018. Notice, the color of squares shows $T_{r,1^{st}}$ while the colored circles represent $\Delta T_{1^{st}}$. These two features can be used as an assessment indicator for the operational behaviour of the substations. Technically, it is desired in a well performed substation that the $T_{r,1^{st}}$ has a lower value in comparison to the $T_{s,1^{st}}$. In other words, a greater delta means that the substation is making more efficient use of the supplied heat for space heating.

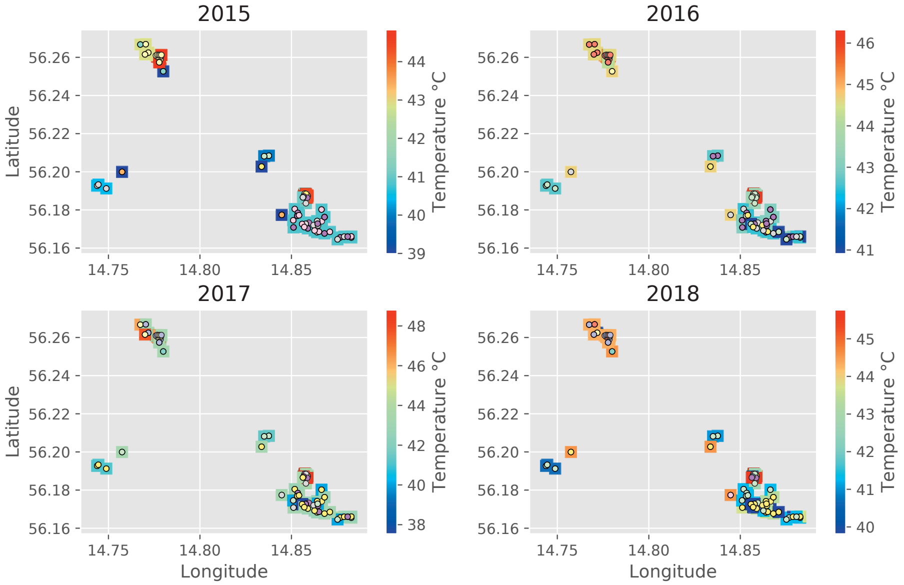


Figure 12.3: The results of SW-MVC analysis for the whole studied area contains 70 substations. Squares represent clusters of substations based on the first view, $T_{r,1^{st}}$, and circles represent groups of substations with respect to the second view, $\Delta T_{1^{st}}$. Note that the substations with similar colors in different MST clusters are not related.

Table 12.3 provides the statistics, the average values of the actual measurements and their standard deviations, regarding the DH substations that are discussed in Figure 12.4. As one can see in 2017, substations are grouped into 4 clusters, where clusters 0 (green squares) and 2 (orange squares) contains the majority of substations, 8 and 5, respectively. The two other clusters include only one substation each (red squares). The average $T_{r,1^{st}}$ for cluster 0 is approximately 43 °C and for cluster 2 is around 46 °C. Clusters 1 and 3 both show the average $T_{r,1^{st}}$ of 48 °C. All the grouped substations in the previous step stayed together based on the $\Delta T_{1^{st}}$, i.e., no new cluster is created. The cluster with 8 substations (grey circles inside green squares) represents the $\Delta T_{1^{st}}$ of 34.79 °C, while the cluster with 5 substations (yellow circles inside orange squares) shows the $\Delta T_{1^{st}}$ of 33.26 °C. The other two clusters (yellow circles inside red squares) show the same value, 34.25 °C for the $\Delta T_{1^{st}}$.

In 2018, the same number of substations, 15, are grouped into only

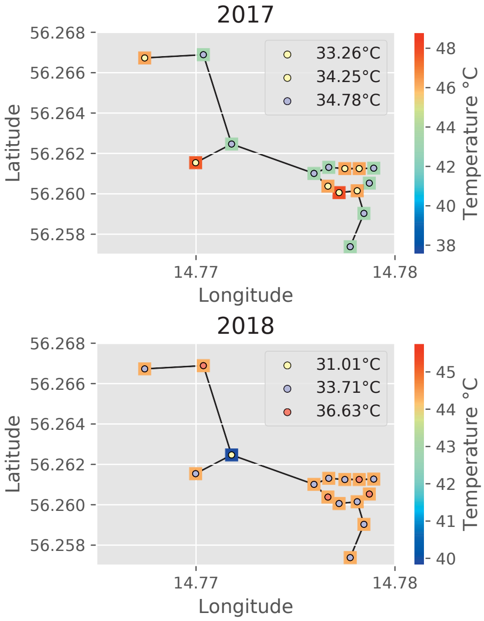


Figure 12.4: The results of SW-MVC analysis for the MST cluster with label 0 and 15 substations in 2017 (**top**) and 2018 (**bottom**). Colored squares represent groups of substations based on $v_1 : T_{r,1^{st}}$ and colored circles represent groups of substations based on $v_2 : \Delta T_{1^{st}}$.

Table 12.3: SW-MVC analysis for the MST cluster with label 0, 2017-2018

Year	$v_0 : MST$	$v_1 : T_{r,1^{st}}$				$v_2 : \Delta T_{1^{st}}$			
	Label	Label	#substations	Avg(°C)	SD(°C)	Label	#substations	Avg(°C)	SD(°C)
2017	0	0	8	43.31	5.39	0	8	34.78	4.13
		1	1	47.88	-	0	1	34.25	-
		2	5	46.23	3.35	0	5	33.26	3.01
		3	1	47.61	-	0	1	34.25	-
2018	0	0	1	39.91	-	0	1	31.01	-
		1	14	44.31	4.66	0	10	33.71	4.11
						1	4	36.63	2.10

Note. Avg: average, SD: standard deviation

two clusters with an average $T_{r,1^{st}}$ of approximately 40 °C for cluster 0 (purple square) and 44 °C for cluster 1 (orange squares). A majority of the substations, 13 out of 14, are grouped in cluster 1. This cluster is further divided into two clusters, one with 10 DH substations (grey circles inside orange squares) and the average $\Delta T_{1^{st}}$ of approximately 34 °C and the other with 4 substations (orange circles inside orange squares) and the average

$\Delta T_{1^{st}}$ of approximately 37 °C. Cluster 0 represents one substation (yellow circle inside purple square) with the average $\Delta T_{1^{st}}$ of 31.01 °C. In both years there are substations that show slightly different operational behaviour in comparison to their neighbouring substations, e.g., the red substations in 2017 and the purple substation in 2018. The domain experts can investigate the reasons why these substations performed differently in comparison to the majority of substations. In addition, it is important to mention that the order in which views are used for the SW-MVC analysis affects the results, which can be decided based on the domain expert's preferences.

12.7.2 PW-MVC Analysis

The aim of this analysis is to group the substations based on two different views (i.e., v_1 and v_2) and compare the results of the clustering solution to find out which substations are similar based on both views. Such analysis can provide useful information for the domain experts while it applies for a period of time, e.g., different years, where the transition of the operational behaviour of substations can be monitored. Table 12.4 shows the distribution of the studied substations based on PW-MVC analysis throughout the heating season in the period from 2015 to 2018. Figure 12.5 depicts the computed

Table 12.4: PW-MVC analysis is performed based on $T_{r,1^{st}}$ and $\Delta T_{1^{st}}$ separately from 2015 to 2018.

Year	$v_0 : MST$		$v_1 : T_{r,1^{st}}$				$v_2 : \Delta T_{1^{st}}$		
	Label	#substation	0	1	2	3	0	1	2
2015	0	15	5	10			5	10	
	1	32	32				11	12	9
	2	14	5	9			6	5	3
Total		70	51	19			31	27	12
2016	0	15	3	5	7		6	9	
	1	32	15	13	4		32		
	2	14	9	5			10	4	
Total		70	36	23	11		57	13	
2017	0	15	8	1	5	1	15		
	1	32	11	4	16	1	10	22	
	2	14	2	9	3		10	4	
Total		70	30	14	24	2	44	26	
2018	0	15	1	14			9	6	
	1	32	12	5	8	7	26	6	
	2	14	4	4	1	5	4	10	
Total		70	26	23	9	12	48	22	

ARI scores for the clustering solution based on $T_{r,1^{st}}$ and $\Delta T_{1^{st}}$ of each

MST cluster in the period from 2015 to 2018. As one can see, the ARI scores of the first three clusters are absolutely dissimilar in 2015, 2016, and 2018. However, cluster 2 with 14 substations shows the ARI score of 0.71 in 2017, which means the majority of DH substations in this cluster performed similarly. Other clusters, 3 to 8 represent the adjusted rand index of 1 for all the years.

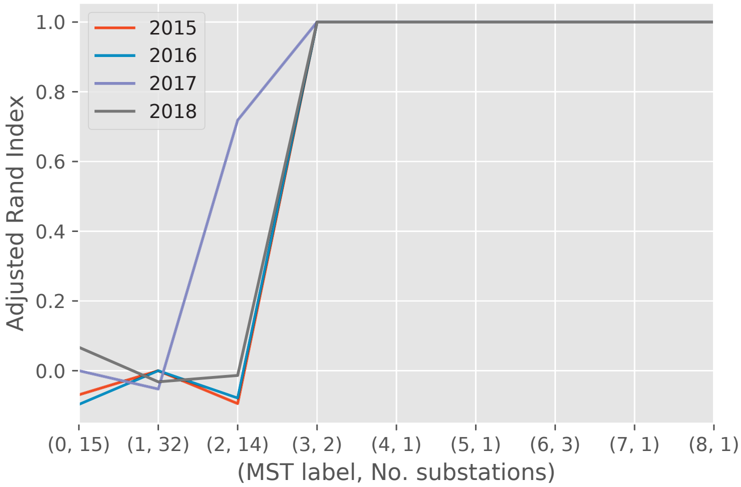


Figure 12.5: PW-MVC analysis, the computed ARI scores for $T_{r,1^{st}}$ and $\Delta T_{1^{st}}$ throughout 2015 to 2018.

Figure 12.6 shows the PW-MVC analysis for MST cluster with label 1 with respect to $T_{r,1^{st}}$ and $\Delta T_{1^{st}}$ in the period from 2015 to 2018. The substations with similar cluster labels with regard to both features are shown in red. The insight provided by Figure 12.6 can be used for analysing the difference between operational behaviour of the red substations against the greys within one specific year. In addition, the transition of the substations from one color group to another can be tracked and further analysed.

12.8 Conclusions

We have proposed a multi-view clustering approach for analysing datasets that consist of different data representations. The proposed approach has been applied for monitoring and analysing operational behaviour of district heating substations. We have initially used the substations' geographical

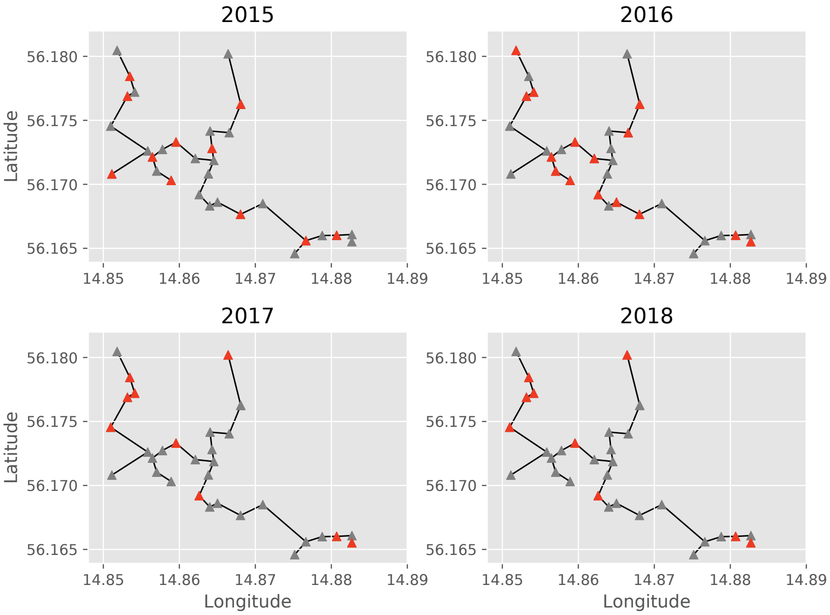


Figure 12.6: The results of PW-MVC analysis for the MST cluster with label 1 and 32 substations. Substations in red are those that are grouped with the same label according to the both views, namely $T_{r,1^{st}}$ and $\Delta T_{1^{st}}$ from 2015 to 2018.

information to build an approximate graph representation of the DH network. This graph structure has been used as a backbone for further analysis of the network performance.

In the above context, we have proposed and discussed two different types of analysis: 1) step-wise multi-view clustering that sequentially considers and analyses the operational behaviour of the DH substations with respect to different views and organizes the substations into a hierarchical structure. That is, at each step a new clustering solution is built on top of the one generated in the previous step with respect to the considered view. 2) parallel-wise multi-view clustering that analyses substations with regards to two different views in side by side. This enables the identification of the relationships between neighbouring substations by organizing them in a bipartite graph and analysing their distribution with respect to the two considered views. The proposed data analysis approach facilitates the visual analysis and inspections of multi-view real-world datasets such as ones related to the DH networks. For example, the proposed approach provides the opportunity to consider the DH substations in close relation with their neighbours. That is, those substations that demonstrate a deviating behavior from their neighbouring substations can easily be identified for further investigation.

For future work, we are interested in expanding our approach by adding a third scenario where the clustering solution is the outcome of integration of different views. We believe that the proposed approach provides a verity of analysis techniques to supply the domain experts with a complete picture about the DH network operations. In addition, the proposed approach can facilitate the identification of substations with deviating behaviours and suggest initiation of further inspections by domain experts.

References

[1] P. Deepak and J.-L. Anna. "Multi-View Clustering". In: *Linking and Mining Heterogeneous and Multi-view Data*. Cham: Springer Int'l Publishing, 2019, pp. 27–53. ISBN: 978-3-030-01872-6. DOI: 10.1007/978-3-030-01872-6_2. URL: https://doi.org/10.1007/978-3-030-01872-6_2.

[2] A. Blum and T. Mitchell. "Combining labeled and unlabeled data with co-training". In: *Proc. of the eleventh annual Conf. on Computational learning theory*. 1998, pp. 92–100.

[3] R. K. Ando and T. Zhang. "Two-view feature generation model for semi-supervised learning". In: *Proc. of the 24th Int'l Conf. on Machine learning*. ACM. 2007, pp. 25–32.

[4] C. Xu, D. Tao, and C. Xu. "A survey on multi-view learning". In: *arXiv preprint arXiv:1304.5634* (2013).

[5] S. Bickel and T. Scheffer. "Multi-view clustering." In: *ICDM*. Vol. 4. 2004, pp. 19–26.

[6] X. Cai, F. Nie, and H. Huang. "Multi-view k-means clustering on big data". In: *Twenty-Third Int'l Joint Conf. on artificial intelligence*. 2013.

[7] B. Jiang, F. Qiu, and L. Wang. "Multi-view clustering via simultaneous weighting on views and features". In: *Applied Soft Computing* 47 (2016), pp. 304–315.

[8] J. Liu, C. Wang, J. Gao, and J. Han. “Multi-view clustering via joint nonnegative matrix factorization”. In: *Proc. of the 2013 SIAM Int’l Conf. on Data Mining.* SIAM. 2013, pp. 252–260.

[9] L. Zong, X. Zhang, L. Zhao, H. Yu, and Q. Zhao. “Multi-view clustering via multi-manifold regularized non-negative matrix factorization”. In: *Neural Networks* 88 (2017), pp. 74–89.

[10] A. Kumar and H. Daumé. “A co-training approach for multi-view spectral clustering”. In: *Proc. of the 28th Int’l Conf. on machine learning (ICML-11).* 2011, pp. 393–400.

[11] X. Wang, B. Qian, J. Ye, and I. Davidson. “Multi-objective multi-view spectral clustering via pareto optimization”. In: *Proc. of the 2013 SIAM Int’l Conf. on Data Mining.* SIAM. 2013, pp. 234–242.

[12] X. Meng, X. Liu, Y. Tong, W. Glänzel, and S. Tan. “Multi-view clustering with exemplars for scientific mapping”. In: *Scientometrics* 105.3 (2015), pp. 1527–1552.

[13] C.-D. Wang, J.-H. Lai, and S. Y. Philip. “Multi-view clustering based on belief propagation”. In: *IEEE Transactions on Knowledge and Data Engineering* 28.4 (2015), pp. 1007–1021.

[14] R. Isermann. “Supervision, fault-detection and fault-diagnosis methods—an introduction”. In: *Control engineering practice* 5.5 (1997), pp. 639–652.

[15] R. Isermann. *Fault-diagnosis systems: an introduction from fault detection to fault tolerance.* Springer Science & Business Media, 2006.

[16] S. Katipamula and M. R. Brambley. “Methods for fault detection, diagnostics, and prognostics for building systems-A review, part I”. In: *Hvac&R Research* 11.1 (2005), pp. 3–25.

[17] S. Katipamula and M. R. Brambley. “Methods for fault detection, diagnostics, and prognostics for building systems-A review, part II”. In: *Hvac&R Research* 11.2 (2005), pp. 169–187.

[18] P. Xue, Z. Zhou, X. Fang, X. Chen, L. Liu, Y. Liu, and J. Liu. “Fault detection and operation optimization in district heating substations based on data mining techniques”. In: *Applied Energy* 205 (2017), pp. 926–940.

[19] F. Sandin, J. Gustafsson, and J. Delsing. *Fault detection with hourly district energy data: Probabilistic methods and heuristics for automated detection and ranking of anomalies.* Svensk Fjärrvärme, 2013.

[20] E. Calikus, S. Nowaczyk, A. Sant'Anna, H. Gadd, and S. Werner. "A Data-Driven Approach for Discovery of Heat Load Patterns in District Heating". In: *arXiv preprint arXiv:1901.04863* (2019).

[21] J. VanderPlas. "mst_clustering: Clustering via Euclidean Minimum Spanning Trees." In: *J. Open Source Software* 1.1 (2016), p. 12.

[22] B. J. Frey and D. Dueck. "Clustering by passing messages between data points". In: *Science* 315.5814 (2007), pp. 972–976.

[23] J. MacQueen et al. "Some methods for classification and analysis of multivariate observations". In: *Proc. of the Fifth Berkeley Symp. on Mathematical Statistics and Probability.* Vol. 1. 14. Oakland, CA, USA. 1967, pp. 281–297.

[24] H. Sakoe and S. Chiba. "Dynamic programming algorithm optimization for spoken word recognition". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 26.1 (1978), pp. 43–49.

[25] S. Abghari, V. Boeva, J. Brage, C. Johansson, H. Grahn, and N. Lavesson. "Higher Order Mining for Monitoring District Heating Substations". In: *2019 IEEE Int'l Conf. on Data Science and Advanced Analytics (DSAA).* IEEE. 2019, pp. 382–391.

[26] L. Hubert and P. Arabie. "Comparing partitions". In: *J. of classification* 2.1 (1985), pp. 193–218.

[27] W. M. Rand. "Objective criteria for the evaluation of clustering methods". In: *J. of the American Statistical association* 66.336 (1971), pp. 846–850.

[28] S. Frederiksen and S. Werner. *District Heating and Cooling.* Studentlitteratur AB, 2013. isbn: 9789144085302.

[29] F. R. Hampel. "A general qualitative definition of robustness". In: *The Annals of Mathematical Statistics* (1971), pp. 1887–1896.

[30] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. "Scikit-learn: Machine Learning in Python". In: *J. of Machine Learning Research* 12 (2011), pp. 2825–2830.

13

Higher Order Mining Approach for Analysis of Real-world Datasets

Shahrooz Abghari, Veselka Boeva, Jens Brage, Håkan Grahn,
In: Submitted for journal publication.

Abstract

In this study, we propose a higher order mining approach that can be used for analysis of real-world datasets. The approach can be used for monitoring and identifying deviating operational behaviour of the studied phenomenon in the absence of prior knowledge about data. The proposed approach consists of several different data analysis techniques such as sequential pattern mining, clustering analysis, consensus clustering and minimum spanning tree (MST). Initially, clustering analysis is performed on the extracted patterns to model the behaviour modes of the studied phenomenon for a given time interval. The built clustering models corresponding to every two consecutive time intervals can further be assessed for mining changes in the monitored behaviour. In case some significant difference is observed, further analysis is performed by integrating the built models into a consensus clustering and applying an MST for identifying deviating behaviours. The validity and potential of the proposed approach has been demonstrated on a real-world dataset originating from a network of district heating (DH) substations. The obtained results show that our approach is capable of detecting deviating and sub-optimal behaviours of the DH substations.

13.1 Introduction

Fault is an abnormal state within the system that may cause a failure or a malfunction. Fault detection is the identification of an unacceptable deviation of at least one feature of the system from the expected or usual behaviour [1]. The fault detection problem has been studied in different domains and it belongs to a more general category, outlier (anomaly) detection. There are several factors such as the nature of data, availability of labeled data, constraints and requirements of the outlier detection problem that make it domain specific [2–6]. Outlier detection techniques can be classified into three groups based on availability of the labeled data, namely supervised methods, unsupervised methods, and semi-supervised methods, where the aim is to learn what is (ab)normal and only model (ab)normality [2, 3, 6]. In the absence of prior knowledge of data, which is often the case for more real-world datasets due to, e.g., expensiveness of the data labeling process by domain experts, the initial assumption is that normal data represents a significant portion of data.

Isermann [1, 7] provided a general review for fault detection and diagnosis (FDD). The main objective of an FDD system is early detection of faults and diagnosis of their causes to reduce the maintenance cost and excessive damage to other parts of the system. Katipamula and Brambley [8, 9] conducted an extensive review in two parts on FDD for building systems. The authors classified FDD methods based on the availability of a priori knowledge for formulating the diagnostics and highlighted their advantages and disadvantages.

According to this classification, FDD methods can fall into two categories: model-based methods and data-driven methods. The model-based methods require a priori knowledge of the system and can use either quantitative or qualitative models. Quantitative models are sets of mathematical relationships mainly based on physical properties, processes or models of the system. Qualitative models, on the other hand, use qualitative knowledge, e.g., including domain expert's experience as a set of rules for identifying proper and faulty operations. Data-driven methods use historical data coming from a system to build models, i.e., they are system-specific. These methods became more and more popular in the recent years. Data-driven methods are easy to develop and do not require explicit knowledge of the system, which makes them suitable for domains with uncertainty.

In this study, we propose a data-driven approach that also relies on domain experts' qualitative knowledge for setting user specific thresholds for identifying deviating behaviours. We show that the proposed approach is capable of revealing patterns representing deviating behaviours of the studied phenomenon. Such deviating behaviours can arise due to different reasons, in the case of district heating (DH) systems, they might be related to faulty equipment, sudden change of outdoor temperature, and/or social behaviour of tenants. Regardless of the reasons behind these deviating behaviours, fault detection systems should be able to provide understandable reports for domain experts. The proposed approach in this study, supplies the domain experts with patterns that are suitable for human inspection. In addition, the results are visualized from different points of view to provide supplementary information for the experts. The contributions of the proposed approach can be summarized into following three steps:

1. Build a data model of the studied phenomenon, which presents its behavioural modes for a given time interval.
2. Monitor the phenomenon's behaviour by comparing the models corresponding to two consecutive time intervals.
3. Analyse and identify deviating behaviours by integration analysis of the built models using a domain specific threshold.

This paper extends our previous work [10], where the data cleaning and preparation are improved by utilizing more suitable data preprocessing methods for time series data. We apply a k-Nearest neighbours based approach for missing data imputation, which takes into consideration the structure of the data. The seasonality of the time series are adjusted using the differencing method. In addition, the raw time series (sequences of numbers) are discretized into string series using the symbolic aggregate approximation method. Furthermore, we have conducted more extensive evaluations of the proposed approach for the chosen case study by considering a larger number of substations coming from different heat load categories. In addition, a more general explanation of the proposed approach is provided to demonstrate its applicability for similar problems in other domains.

13.2 Related work

The validity and potential of the proposed approach has been demonstrated in a use case from DH domain. Therefore in this section we mainly review literature related to outlier and fault detection approaches applied to DH and smart buildings domains.

Fontes and Pereira [11] proposed a fault detection for gas turbines using pattern recognition in multivariate time series. In another study [12], the authors proposed a general methodology for identifying faults based on time-series models and statistical process control, where faults can be identified as anomalies in the temporal residual signals obtained from the models using statistical process control charts.

In a recent review, Djenouri et al. [13] focused on the usage of machine learning for smart building applications. The authors classified the existing solutions into two main categories: 1) occupancy monitoring and 2) energy/device-centric. These categories are further divided into a number of sub-categories where the classified solutions in each group are discussed and compared.

Gadd and Werner [14] showed that hourly meter readings can be used for detecting faults at DH substations. The authors identified three fault groups: 1) low average annual temperature difference, 2) poor substation control, and 3) unsuitable heat load patterns. The results of the study showed that low average annual temperature differences are the most important issues, and that addressing them can improve the efficiency of the DH systems. However, solving unsuitable heat load patterns is probably the easiest and the most cost-effective fault category to be considered.

Xue et al. [15] applied clustering analysis and association rule mining to detect faults in substations *with* and *without* return-water pressure pumps. Clustering analysis was applied in two steps 1) to partition the substations based on monthly historical heat load variations and 2) to identify daily heat variation using hourly data. The result of the clustering analysis was used for feature discretization and preparation for association rule mining. The results of the study showed the method can discover useful knowledge to improve the energy performance of the substations. However, for temporal knowledge discovery, advanced data mining techniques are required.

Capozzoli et al. [16] proposed a statistical pattern recognition techniques in combination of artificial neural ensemble network and outlier detection methods to detect real abnormal energy consumption in a cluster of eight smart office buildings. The results of the study show the usefulness of the proposed approach in automatic fault detection and it ability in reducing the number of false alarms. Månsson et al. [17] proposed a method based on gradient boosting regression to predict an hourly mass flow of a well performing substation using only a few number of features. The built model is tested by manipulating the well performing substation data to simulate two scenarios: communication problems and a drifting meter fault. The model prediction performance is evaluated by calculating the hourly residual of the actual and the predicted values on original and faulty datasets. Additionally, cumulative sums of residuals using a rolling window that contains residuals from the last 24 hours were calculated. The results of the study showed that the proposed model can be used for continued fault detection.

Calikus et al. [18] proposed an approach for automatically discovering heat load patterns in DH systems. Heat load patterns reflect yearly heat usage in an individual building and their discovery is crucial for effective DH operations and managements. The authors applied k-shape clustering [19] on smart meter data to group buildings with similar heat load profiles. Additionally, the proposed method was shown to be capable of identifying buildings with abnormal heat profiles and unsuitable control strategies.

Sandin et al. [20] used probabilistic methods and heuristics for automated detection and ranking of faults in large-scale district energy systems. The authors studied a set of methods ranging from limit-checking and basic model to more sophisticated approaches such as regression modelling and clustering analysis on hourly energy metering.

In this study, we use a combination of data analysis techniques for modelling, monitoring, and analyzing operational behaviours of a studied phenomenon. We propose a higher order mining (HOM) approach to facilitate domain experts in identifying deviating behaviours and potential faults of the phenomenon under study. HOM is a sub-field of knowledge discovery that is applied on non-primary, derived data, or patterns to provide human-consumable results [21]. We first apply sequential pattern mining on raw data and extract patterns based on a user defined time interval. Then, the behavioural model is built by performing clustering analysis for each time interval. We further analyze and assess the similarity of the built behavioural models for every two consecutive time intervals. In case of observing any significant discrepancy (given a domain specific threshold) the built clustering models are integrated into a consensus clustering model. We additionally construct a minimum spanning tree (MST) on each consensus clustering model, considering the exemplars of the clustering solution as nodes and the distance between them as edges of the tree. Note that an

MST is a spanning tree that covers all the nodes with the least traversing cost. In order to identify deviating behaviours, we cut the longest edge(s) of the MST, which turns the tree into a forest. Any small and distant trees can be interpreted as outliers, which can be further analysed by the domain expert.

13.3 Methods and Techniques

13.3.1 Sequential Pattern Mining

Sequential pattern mining is the process of finding frequently occurring patterns in a sequence dataset. The records of the sequence dataset contain sequences of events that often have chronological order. In this study, we apply the PrefixSpan algorithm [22] to extract frequent sequential patterns. PrefixSpan applies a prefix-projection method to find sequential patterns. Given a sequence dataset, minimum and maximum lengths of patterns, and a user-specified threshold, the dataset is first scanned in order to identify all frequent items with the length one in sequences. Using a divide and conquer fashion the search space is divided into a number of subsets based on the extracted prefixes. Finally, for each subset a corresponding projected dataset is created and mined recursively.

13.3.2 Clustering Analysis

13.3.2.1 Affinity Propagation

We use the affinity propagation (AP) algorithm [23] for clustering the extracted patterns. AP is based on the concept of exchanging messages between data points. The exchanged messages at each step assist AP to choose the best samples as exemplars (representatives of clusters) and which data points should choose those data points as their immediate exemplars. Unlike most clustering algorithms, such as k-means [24] which requires the number of clusters as an input, AP estimates the optimal number of clusters from the provided data or similarity matrix and the chosen exemplars are real data points. These characteristics make AP a suitable clustering algorithm for this study.

13.3.2.2 Consensus Clustering

Gionis et al. [25] proposed an approach for clustering that is based on the concept of aggregation. They are interested in a problem in which a number of different clustering solutions are given on some datasets of elements. The objective is to produce a single clustering of the elements that agrees as much as possible with the given clustering solutions. Consensus clustering algorithms deal with similar problems to those treated by clustering aggregation techniques. Namely, such algorithms try to reconcile clustering information about the same data phenomenon coming from different sources [26] or from different runs of the same algorithm [27]. In this study, we use the consensus clustering algorithm proposed in [26] in order to integrate the clustering solutions produced on the datasets collected for two consecutive weeks. We consider the exemplars (the representative patterns) of the produced clustering solutions. These exemplars are then divided into k groups (clusters) according to the degree of their similarity by applying the AP algorithm. Subsequently, the clusters whose exemplars belong to the same partition are merged in order to obtain the final consensus clustering.

13.3.3 Time Series Discretization

We apply symbolic aggregate approximation (SAX) [28] as a discretization method to transform raw time series (sequences of numbers) into symbolic strings. SAX first applies piecewise aggregate approximation (PAA) to transform a time series $Y = (y_1, y_2, \ldots, y_n)$ of length n into a PAA representation $\bar{Y} = (\bar{y}_1, \bar{y}_2, \ldots, \bar{y}_m)$ of length $m \leqslant n$, where $\bar{y}_i$ is computed as follows:

$$\bar{y}_i = \frac{m}{n} \sum_{j=\frac{n}{m}(i-1)+1}^{\frac{n}{m}i} y_i \tag{13.1}$$

That is, the time series Y is first divided into m equally sized windows and then for each window the mean value is computed. The final product of these values is the vector $\bar{Y}$ where it is the data-reduced (PAA) representation of Y. The PAA representation is then descritized into a string $\hat{Y} = (\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_m)$ using alphabet A of size a, where a is any integer greater than 2.

13.3.4 Similarity Measures

We use two similarity measures 1) to perform pairwise comparison between extracted patterns and 2) to compute a similarity between two clustering solutions by considering all pairs of members.

13.3.4.1 Levenshtein Distance

The similarity between the extracted patterns are assessed with Levenshtein distance (LD) metric [29]. The LD, also known as edit distance, is a string similarity metric that measures the minimum number of editing operations (insertion, deletion and substitution) required to transform one string into the other. We use the normalized LD where score *zero* implies 100% similarity between the two patterns and *one* represents no similarity. LD is a simple algorithm capable of measuring the similarity between patterns with different lengths. Although in the presented case study the extracted patterns have similar lengths, for some scenarios it might be suitable to perform the pattern extraction with different lengths. Therefore, we choose to use LD which can provide more flexibility when patterns with different lengths are required.

13.3.4.2 Clustering Solution Similarities

Given two clustering solutions $C = \{C_1, C_2, \ldots, C_n\}$ and $C' = \{C'_1, C'_2, \ldots, C'_m\}$ of datasets X and X', respectively the similarity, S_w, between C and C' can be assessed as follows:

$$\begin{aligned} S_w(C, C') = \frac{\sum_{i=1}^{n}(min_{j=1}^{m} w_i.d(c_i, c'_j))}{2} + \\ \frac{\sum_{j=1}^{m}(min_{i=1}^{n} w'_j.d(c_i, c'_j))}{2}, \end{aligned} \tag{13.2}$$

where c_i and c'_j are exemplars of the clustering solutions C_i and C'_j, respectively. The weights w_i and w'_j indicate the relative importance of clusters C_i and C'_j compared to other clusters in the clustering solutions C and C', respectively. For example, a weight w_i of a cluster C_i can be calculated as the ratio of its cardinality to the cardinality of the dataset X, i.e., $w_i = |C_i|/|X|$. The S_w has values in a range of [0,1]. Scores of zero imply identical performance while scores close to one show significant dissimilarities.

13.3.5 Minimum Spanning Tree

Given an undirected and connected graph $G = (V, E)$, a spanning tree of the graph G is a connected sub-graph with no cycles that include all vertices. A minimum spanning tree (MST) of an edge-weighted graph (G, w), where $G = (V, E)$ is a graph and $w : E \to \mathbb{R}$ is a weight function, is a spanning tree where the sum of the weights of its edges is minimum among all the spanning trees. MSTs have been studied and applied in different fields including cluster analysis and outlier detection [30–34]. In this study we apply an MST on top of the created consensus clustering solution to further analyse the deviating substations' behaviours. We use Kruskal's algorithm [35] for building the MST. Kruskal's algorithm follows a greedy approach, i.e., at each iteration it chooses an edge which has least weight and adds it to the growing spanning tree. The algorithm first sorts the edges of (G, w) in an increasing order with respect to their weights. Then, it starts adding edges in sorted order and only those that do not form a cycle in the MST.

13.4 Proposed Method

We propose a higher order mining approach for modelling, monitoring, and analyzing real-world data phenomena. The proposed approach uses a combination of different data analysis techniques such as sequential pattern mining, clustering analysis, consensus clustering, and the MST algorithm. Note that the last three data mining techniques fall into the HOM paradigm, i.e., they are not applied on raw data, but on derived patterns and built models. The HOM paradigm brings new potential and perspective for knowledge discovery by generating human-understandable results and simplifying the comparative analysis among the studied phenomena. Thus, in the proposed approach the available data are initially partitioned across the time axis based on a user specified time interval. This allows for conventional data mining within each time interval and for HOM over the patterns extracted from the time intervals covering the studied period. This facilitates not only revealing similarities and interesting differences among the studied entities but also contributes to a more tractable process for the whole monitoring period.

The proposed approach consists of three main steps: 1) building a data model of the studied phenomenon that represents its behavioural modes for a given time interval, 2) monitoring the phenomenon's behaviour by comparing the models corresponding to two consecutive time intervals, and 3) analysing and identifying deviating behaviours. For the rest of this section, each step and its sub-steps will be described.

13.4.1 Building Behavioural Model

This step consists of two sub-steps. Initially, the data collected for a given time interval is analysed and frequent sequential patterns are extracted using the PrefixSpan algorithm discussed in Section 13.3.1. In order to build a model representing the phenomenon behavioural modes for the considered time interval, the extracted patterns are further analysed and partitioned into a number of groups. This is performed by applying the Affinity Propagation algorithm (see Section 13.3.2.1). The similarity between the patterns are calculated using LD. The built clustering model, at each time interval, represents the operational behaviour of the studied phenomenon for a specific time period. Note that the exemplars of each cluster serves as the representative operational behaviour for that cluster, i.e., every clustering solution can express the behaviour of the studied phenomenon for a specific time interval with the a set of exemplars.

13.4.2 Monitoring of Behaviour

The goal of this step is to make use of built behavioural models in the previous step to monitor the phenomenon behaviour. As we mentioned earlier each time interval is represented by a behavioural model. These models together can be used for monitoring purposes. Considering the fact that most real-world datasets contain unlabeled data, the similarity between the detected behaviours can be analyzed and assessed with the neighbouring time intervals, i.e., every two consecutive intervals. This is done through pairwise comparison of the exemplars of the clustering solutions using equation 4.8. The assessed similarities can be used for measuring, e.g., a discrepancy between observed performances in the two intervals. When the discrepancy is significant (above a domain specific threshold) further analysis (see Section 13.4.3) is preformed by integrating the produced clusters into a *consensus clustering* solution, i.e., AP is applied only on the exemplars of the two clustering solutions. In addition, it is important to mention that a significant discrepancy can be interpreted as deviating behaviour, e.g., newly observed patterns in the time interval $t + 1$ are absolutely different from the ones in the time interval t.

13.4.3 Data Analysis

The created consensus clustering solution is meant to group the observed behavioural patterns (the clusters' exemplars) in the two consecutive time intervals based on their similarities. The observed number of clusters can be considered as an indication of how much the extracted patterns in the two compared time intervals are similar. In order to identify deviating behaviours, first an MST is built, by applying Kruskal's algorithm, on top of each consensus clustering solution, where the exemplars are tree nodes and the distances between them represent the tree edges. The longest edge(s) of the built MST can lead us to groups of patterns with distinct behaviour. Therefore, in the next step the longest edge(s) of the built MST is removed. This turns the tree into a forest where the smallest and distant trees created by the cut can be interpreted as outliers. Nevertheless, the identified outliers need be further analyzed by the domain experts. In addition, the assessed similarities in the previous step, all together, can be used for building up a performance signature profile of the studied phenomenon for the given time period. In addition, such performance profiles can be applied for comparing different phenomena belonging to the same category.

13.5 Real-world Case Study

District Heating (DH) is an energy service based on circulating heated fluid from available heat sources such as natural geothermal, combustible renewable, and excess heat from industrial processes to customers [36]. A DH system provides heat and domestic hot water (DHW) for a number of *consumer units* (buildings) in a limited geographical area. The heat is produced at a *production unit* and circulated through a *distribution network* to reach the consumers. This part of the system is referred to as *primary* side. The consumer unit consists of a heat exchanger, a circulation network, and radiators for space heating in the rooms. This part of the system is called *secondary* side. The provided heat and DHW produced at the primary side transfer through a substation into the consumer unit, i.e., the secondary side. The substation makes the water temperature and pressure at the primary side suitable for the secondary side. A DH substation involves several components, each a potential source of faults. For example, a fault can consist of a stuck valve, a fouled heat exchanger, less than optimal temperature transmitters, a poorly calibrated control system, and many more [17, 36]. Gadd and Werner [14] classify the possible faults of substations and secondary systems into three categories as follows: 1) faults resulting in comfort problems such as insufficient heating, or physical issues such as water leakage, 2) faults with a known cause but unsolved due to cost, and 3) faults that require advanced fault detection techniques for their discovery, which also includes faults caused by humans, such as unsuitable settings in building operating systems.

Substations are designed to meet heat demands despite possible faults degrading their performance. For example, poor heat transfer can to some extent be compensated by increasing the flow through the substation, meeting the heat demand at a higher cost for the energy company operating the network. In addition, ownership of the substation (often part of the building) and the subsequent high cost of customer interaction makes such situations difficult to address in a traditional manner. Consequently, early detection and classification of faults and other deviations from preferred behaviour can be used to reduce the overall cost, for example, by lowering maintenance costs, by reducing the need for the energy company to compensate poor performance with an increased flow, and by streamlining customer interactions. Finally, higher performing substations makes it possible to lower the system's overall temperature, which in turn makes it possible to use a greater amount of heat from renewable and other low value energy sources such as excess heat from subway stations.

13.6 Experimental Design

13.6.1 Dataset

The data used in this study is provided by an energy company located in Southern Sweden. The dataset consists of hourly average measurements from 82 buildings equipped with the company's smart system. The collected data was obtained during February 2014 until December 2018. This means 43,800 instances per building (24 instances per day). However, since most of the buildings have a high percentage of missing days and hourly missing values in the time span of 2014 to 2015, we limit our analysis to 47 buildings (that conform the largest DH network in the studied dataset) for the period covering the recent three years (2016, 2017, and 2018). The building are divided into four categories: 1) company (C), 2) residential (R), 3) mixture of both residential and company (R-C), and 4) school (S). Table 13.1 summarizes the number of buildings in each category.

Table 13.1

Building type	count
Company (C)	9
Residential (R)	14
Residential and company (R-C)	20
School (S)	4
Total	47

Since we are monitoring the operational behaviour of the substations based on outdoor temperature, 5 out of 10 features that have a strong negative correlation with the outdoor temperature are selected. These features are 1) Secondary temperature difference ($\Delta T_{2^{nd}}$), 2) Primary temperature difference ($\Delta T_{1^{st}}$), 3) Primary power ($Q_{1^{st}}$), 4) Primary mass flow rate ($G_{1^{st}}$), and 5) Substation effectiveness (E_s^T). The substation effectiveness is calculated by considering features from both primary and secondary sides as follows:

$$E_s^T = \frac{\Delta T_{1^{st}}}{T_{s,1^{st}} - T_{r,2^{nd}}}, \tag{13.3}$$

where $\Delta T_{1^{st}}$ is the difference between primary supply and return temperatures, $T_{s,1^{st}}$ is the primary supply temperature, and $T_{r,2^{nd}}$ is the return temperature at the secondary side. The substation effectiveness has a value within a range 0.0 and 1.0. The efficiency of a well-performed substation should be close to 1.0 in a normal setting. However, due to the affect of the domestic hot water generation on the primary return temperature, the E_s^T can be higher than 1.0. Table 13.2 shows all features included in the dataset. The features, 4-6, 9, and 10 in bold font are selected.

In this study we only assess the substations' behaviour while space heating is needed. Figure 13.1 shows the yearly seasonality of outdoor temperature recorded for one building in two consecutive years. As one can see the average outdoor temperature in 2017 during January - April and November - December is below 10 °C. The heating season in 2018 mainly includes January - April and October - December.

Table 13.2: Features included in the dataset

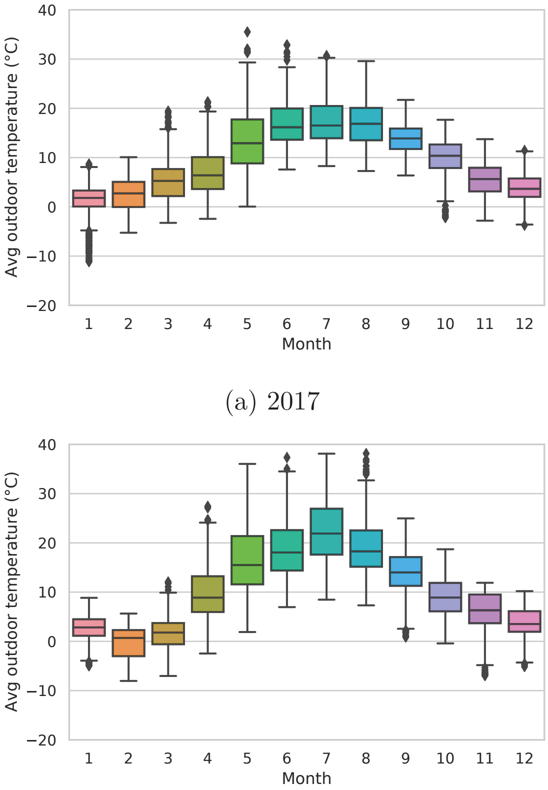
No.	Feature	Notation	Unit/Format
1	T_o	Outdoor temperature	°C
2	$T_{s,1^{st}}$	Primary supply temperature	°C
3	$T_{r,1^{st}}$	Primary return temperature	°C
4	$\mathbf{\Delta T_{1st}}$	**Primary temperature difference**	**°C**
5	$\mathbf{Q_{1st}}$	**Primary power**	**kW**
6	$\mathbf{G_{1st}}$	**Primary mass flow rate**	**l/h**
7	$T_{s,2^{nd}}$	Secondary supply temperature	°C
8	$T_{r,2^{nd}}$	Secondary return temperature	°C
9	$\mathbf{\Delta T_{2nd}}$	**Secondary temperature difference**	**°C**
10	$\mathbf{E_s^T}$	**Substation effectiveness**	**%**

Note. Features in bold font are selected due to their strong correlation with outdoor temperature.

13.6.2 Data Preprocessing

Missing values can occur due to different reasons such as connection problems of measuring instruments, e.g., energy meters. Since we are interested in identifying deviating operational behaviour on a weekly basis, daily time series with more than 25% missing values (or six missing hours within a 24-hour period) are discarded. There are different imputation methods such as mean substitution, hot-deck imputation [37], regression analysis, and multiple imputation [38]. We apply a k-Nearest neighbours (kNN) based approach [39] to impute the missing values. That is, for each feature and year, the missing hours are imputed using the values of the five nearest neighbours (days) and the same hours (we let the number of neighbours, k, to be five which is a default value for the used library). The method identifies the nearest neighbours with the help of Euclidean distance and the missing values for each hour are weighted by distance to each neighbour.

Faults in measurement tools can appear as extreme values or sudden jumps in the measured data. We use a Hampel filter [40] which is a median absolute deviation (MAD) based estimation to detect and smooth out such extreme values. The filter computes the median, MAD, and the standard deviation (SD) over the data in a local window. We apply the filter with default parameters, i.e, the size of the window is considered to be seven which yields 3-neighbours on each side of a sample and the threshold for extreme value detection is set to be three. Therefore, in each window a sample with the distance three times the SD from its local median is considered as an extreme value and is replaced by the local median.



(a) 2017

(b) 2018

Figure 13.1: Yearly seasonality of outdoor temperature for a building in **a)** 2017 and **b)** 2018.

Time series data may contain seasonality, a repeating pattern within a fixed time period. The process of identifying and removing a time series seasonal effects is called seasonal adjustment. Seasonal effects can mask other interesting characteristics of the data. Seasonal adjustment helps better reveal any interesting components and allows better data analysis. In our case study, the data contains yearly seasonality. We apply differencing to adjust the seasonality. That is, each observation is subtracted by the value from previous year. We consider three years of data (2016, 2017, and 2018) where this gives us two years of data after seasonality adjustment (the first year of data is skipped to adjust the seasonality). Additionally, since 2016 is a leap year, February 29 is excluded while differencing.

As it was mentioned earlier, the proposed approach partitions the available data across the time axis on a weekly basis in order to extract patterns within each week. Therefore, it is necessary to convert the continuous features to categorized or nominal features, i.e., *data discretization* must be conducted. We apply SAX to transform the time series into symbolic representation. In order to have a meaningful comparison between time series with different offsets and amplitudes, the time series need to be normalized, i.e., to have mean of zero and standard deviation of one. Therefore, each yearly time series (feature vector) are normalized with z-score normalization. This step is performed automatically while applying SAX. We consider five categories (alphabet size) for the SAX transformation process as follows: *low*, *low_medium*, *medium*, *medium_high*, and *high*. Nevertheless, the alphabet size can be adjusted based on the available data. In the previous study [10], we considered four categories, i.e., the same as the number of season periods. However, due to the risk of losing information the fifth category *medium* is added.

13.6.3 Data Segmentation and Pattern Extraction

The size of the time window (partition) for pattern extraction is important for further analysis. The proper partition length leads us to monitor operational behaviour of the substations rather than the residents' behaviour. Therefore, after performing some preliminary tests and having discussions with domain experts, the time window is set to be a week. The PrefixSpan algorithm is used for identifying frequent sequential patterns with a desired length. Any patterns that satisfy the user-specified threshold are considered as frequent. For the studied problem, the user-specified threshold is set to be one, i.e., any patterns that appear at least once will be considered. In addition, due to the importance of the selected features, the desired length of pattern is set to be equal to number of features, which is five.

13.6.4 Affinity Propagation Parameters Tuning

AP has a number of parameters. In this study we adjust two of these parameters, namely *affinity* and *damping*. The *affinity* parameter is set to be *pre-computed* since the algorithm is fed with a similarity matrix. The *damping* factor can be regarded as a slowly converging learning rate to avoid numerical oscillations. It is within a range 0.5 to 1.0. We always apply AP with damping factor equal to 0.5, in case the convergence does not occur the damping factor will be increased by 0.05 units and AP will be rerun with a new damping factor until convergence occurs.

13.6.5 Implementation and Availability

The proposed approach is implemented in Python version 3.6. The Python implementations of the PrefixSpan algorithm and the edit distance are fetched from [22] and [29], respectively. The AP algorithm and the k-means-based discretization are adopted from the scikit-learn module [41]. For constructing and manipulating an MST the NetworkX package is used [42]. The package uses Kruskal's algorithm for constructing the MST. The implemented code and the experimental results are available at GitHub[1].

13.7 Results and Discussion

We have studied substations' operational behaviour of 47 buildings during a period of two years (2017 and 2018). For each building, we first model the substation's weekly operational behaviour. This is performed by grouping the extracted frequent patterns into clusters of similar patterns. In order to monitor the substation's performance, we analyze and assess the similarity between substation's behaviours for every two consecutive weeks. When the bi-weekly comparison shows more than 25% (a user-specified threshold) difference and if the average temperature be less than or equal to 10 °C, further analysis is conducted by integrating the produced clustering solutions into a consensus clustering. The obtained consensus clustering solution is used for building an MST, where the exemplars are tree nodes and the distances between them represent the tree edges. In order to identify unusual behaviours, the longest edge(s) of the MST is removed. The smallest subtrees created by the cut are interpreted as faults or deviations.

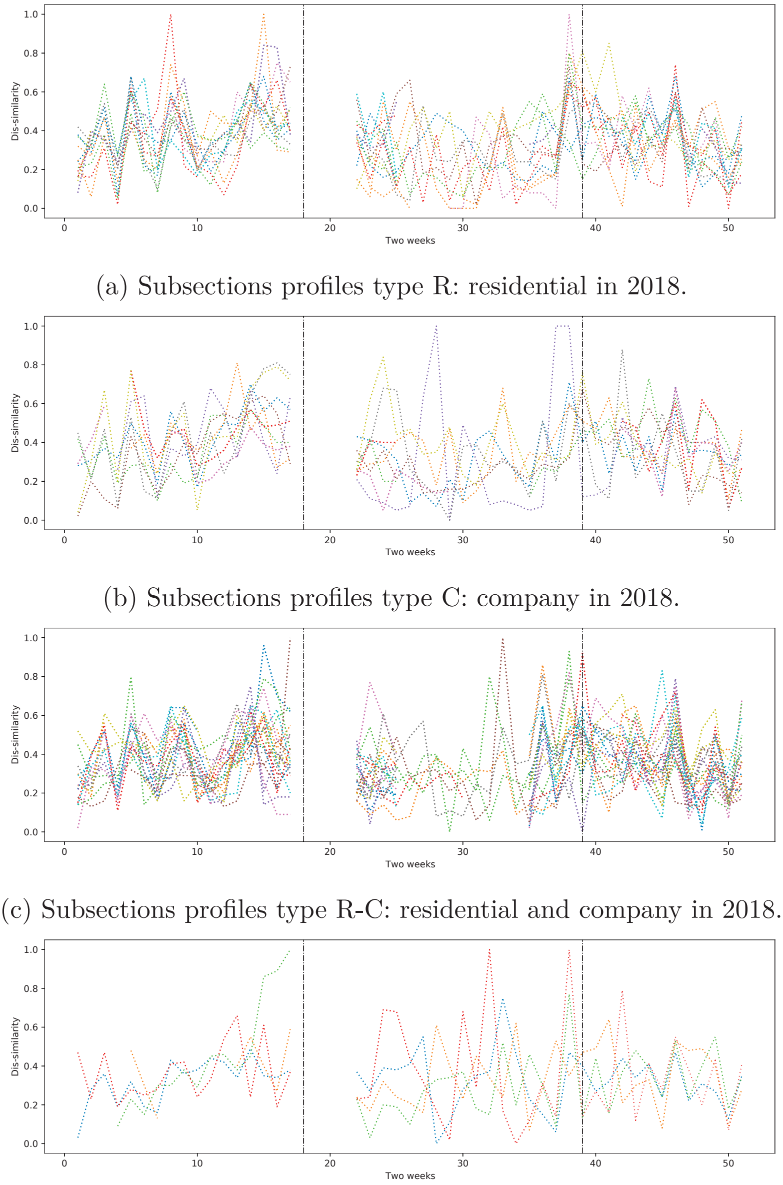
13.7.1 Substations bi-weekly performance signature

As we mentioned earlier in Section 13.4.3, the assessed similarities of a substation's operational behaviour can be used for building the substation performance signature profile for the entire studied period. Additionally, such profiles can be used for comparing the substations belonging to the same heat load category. We studied four types of buildings: C, R, R-C, and

[1] https://github.com/shahrooz-abghari/HOM-Real-World-Datasets

S (see Table 13.1 for more information). Figure 13.2a shows the signature profiles of 14 residential buildings in 2018. The area between the two vertical dashed lines represents the non-heating season in all figures. Figure 13.2b depicts the substations' performance signature profiles of nine company buildings for the same year. Figure 13.2c contains the largest number of buildings belonging to the R-C category, i.e., 20 in total. Figure 13.2d represents the signature profiles of four schools. All the studied buildings are located in the same city. As one can see, Figures 13.2a-d contain signatures that are quite similar in the period of week 1 until week 18 (January 1 - May 6, 2018) and week 45 to week 46 (November 5-18, 2018).

Although, the expectation was to observe similar performance signatures from the buildings that are in the same category, there are substations showing quite different behaviours. The main reasons can be related to the difference between average outdoor temperature within two weeks in different areas of the city. Our further analysis shows that buildings of the same type and close together tend to have similar performance signature profiles during heating seasons. In addition, social behaviour of people, special holidays, and/or faulty substations and equipment have high impact on substations' performance. It is also the case that buildings of same category behave differently mostly due to installation issues, unsuitable configurations, or different brand of equipment. Nevertheless, this requires further analysis by domain experts.



(a) Subsections profiles type R: residential in 2018.

(b) Subsections profiles type C: company in 2018.

(c) Subsections profiles type R-C: residential and company in 2018.

(d) Subsections profiles type S: school in 2018

Figure 13.2: Substations profiles based on buildings types. Due to missing values some of the bi-weekly comparisons are missing. The area between two vertical dashed lines in each plot represents the non-heating season.

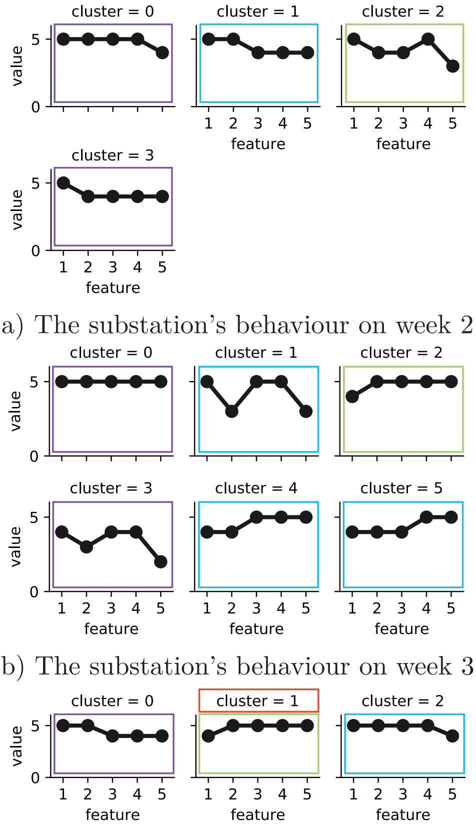
13.7.2 Modelling Substation Operational Behaviour

Weekly operational behaviour of a substation can be modeled by clustering the extracted patterns based on their similarities into groups. Using the AP algorithm, each cluster can be recognized by its exemplar, a representative pattern of the whole group. Each cluster models the substation's operational behaviour for some hours up to a couple of days, based on its frequency. The number of clusters in each clustering solution can be interpreted as different operational modes of the substation for the studied week. High number of clusters may due to the same reasons as ones discussed in the foregoing section such as the difference between outdoor temperature during days and nights. The extracted patterns contain five features. Each feature can belong to one of five available categories: *low*, *low_medium*, *medium*, *medium_high*, and *high*.

Figure 13.3 shows the operational behaviour of B_3 substation during weeks 2 and 3 in 2017, where *low* is represented by one and *high* by five, respectively. Note that each category is the result of differencing to adjust yearly seasonality for each feature.

Figure 13.3a represents 4 operational behaviour modes of B_3 substation during the second week of 2017. Cluster 2 covers 57 hours, the most number of hours among the others, while cluster 1 covers only 26 hours, which is the least number of hours. In week 3 (see Figure 13.3b) 6 operational behaviour modes are detected. Cluster 3 in this week, similar to cluster 2 in week 2, covers 57 hours. Cluster 4 with the least number of hours covers only 11 hours.

We further analyze the operational behaviour models of weeks 2 and 3 by calculating the similarity between the exemplars of the corresponding clustering solutions. The calculated dissimilarity is above 25% and the average weekly outdoor temperature below 10 °C. Therefore, the proposed method integrates the clustering solutions into consensus clustering. Figure 13.3c represents the substation's operational behaviour model for the studied two weeks. The model contains 3 clusters. In order to detect deviating behaviour as explained previously in Section 13.4.3, first an MST is built on top of the consensus clustering solution. Next, the longest edge(s) of the tree is removed and sub-tree(s) with the smallest size and far from majority of data can be marked as deviating behaviour. In Figure 13.3c, cluster 1 (framed in red) is detected as an outlier.



(a) The substation’s behaviour on week 2.

(b) The substation’s behaviour on week 3.

(c) The consensus clustering integrating the clustering solutions for weeks 2 and 3.

Figure 13.3: The B_3’s substation operational behaviour in weeks 2 and 3 in 2017. Each cluster is shown by its exemplar. The colored frames represent the consensus clustering solution, where purple = cluster 0, green = cluster 1, and blue = cluster 2. The exemplars of clusters 0 and 2 are chosen from week 2 and cluster 1 is chosen from week 3. After building an MST on top of the consensus clustering solution, cluster 1 is identified as the deviating behaviour of the substation due to its small size and distance from majority of the data.

Tables 13.3 and 13.4 show the distribution of each weekly cluster together with the number of days and hours that they cover across the consensus clustering solution respectively. As one can see, in Table 13.3, the detected operational behaviours in week 2 are divided into four groups. While week 3 contains six categories of different operational behaviours. In week 2, the majority of the behaviours appear across the whole week except cluster 1, which covers only 6 days. In week 3, on the other hand, clusters 1 and 2 each contain operational behaviours observed within 4 days. Clusters 4 and 5 each cover 5 days and the last two remaining clusters cover 6 days each. Considering the consensus clustering solution, cluster 1 contains the least number of days, 11, while clusters 0 and 2 each include 12 and 14 days. Note that, weekly clustering solutions can have daily overlap, however, they cover different hours.

Table 13.3: Number of identified daily deviating behaviours in weeks 2 and 3 for B_3 in 2017

	Consensus cluster (CC)		
Weekly cluster	CC0	CC1	CC2
W2,C0			7
W2,C1	6		
W2,C2		7	
W2,C3			7
W3,C0			6
W3,C1	4		
W3,C2		4	
W3,C3			6
W3,C4	5		
W3,C5	5		
Total days	12	11	14

Table 13.4 shows the number of hours covered by each weekly cluster and the total hours for each bi-weekly cluster (consensus cluster). As one can see, consensus cluster 2 contains the most number of hours, 168. Cluster 0 and 1 cover 88 and 80 hours, respectively, within weeks 2 and 3. As it was mentioned earlier, by cutting the longest edge(s) of the built MST on top of consensus clustering solution the smallest and distant cluster can be considered as deviating behaviour, i.e., consensus cluster 1. This cluster appears in 11 days (Table 3, consensus cluster CC1) and in total 80 times (Table 13.4, consensus cluster CC1) out of 336 (*24 hours* × *14 days*). The data collected for these particular days can be further analyzed by domain experts to get better insight and understanding of the identified deviating behaviour.

In general, an increase or decrease in the number of observed clusters in one week in comparison to its neighbouring week, can be interpreted as an indication of deviating behaviours. This can occur for different reasons such as sudden drop in outdoor temperature. Therefore, in order not to take

Table 13.4: Identified hourly deviating behaviours in weeks 2 and 3 for B_3 in 2017.

	Consensus cluster (CC)		
Weekly cluster	CC0	CC1	CC2
W2,C0			37
W2,C1	26		
W2,C2		57	
W2,C3			48
W3,C0			26
W3,C1	30		
W3,C2		23	
W3,C3			57
W3,C4	11		
W3,C5	21		
Total hours	88	80	168

into account every single change as deviating behaviour, we consider using a performance measure called overflow. The measure expresses a substation's performance in terms volume flow per unit of energy flow. In the district heating domain overflow of a well-performed substation is expected to be 20 $\frac{l}{kWh}$. Therefore, by computing a substation's weekly overflow, any bi-weekly detected deviations in conjunction with weekly overflow above 20 can be flagged as real changes in the substation.

Table 13.5 represents substations with bi-weekly deviating behaviours on an hourly basis and overflow more than 20 in 2017 and 2018. In total substations that belong to the *Residential* category have the most number of deviating behaviours in both years, i.e., four substations with 95 and three substations with a total 86 detected deviating behaviours in 2017 and 2018 respectively. In the *Residential-Company* category there is only one substation which in both years contains considerable number of deviating behaviours, i.e., 88 and 64. The *Company* category contains four substations with 47 and five substations with 38 identified deviating behaviours in 2017 and 2018 respectively. For the *Schools* category, all four substations contain in total 11 detected deviating behaviours in 2017, while in 2018 only one of these substations contain one deviating behaviour. In Table 13.5 those substations that appear in both years are shown in bold.

Figure 13.4 summarizes the statistics of Table 13.5 on a daily basis for the four categories in 2017 and 2018. The plot can help the domain expert in identifying categories with the most number of daily deviating behaviours for a specific time period. For example, substations belong to *Company* and

Table 13.5: Substations with bi-weekly deviating behaviours on an hourly basis and overflow more than 20 in 2017 and 2018.

Year	Substation	C	R	R-C	S
2017	B_L	2			
	B_3			**88**	
	E_D_32_A		**14**		
	G_3		6		
	M_S		**14**		
	O_S				6
	P_S				1
	R_4	**4**			
	S_1		**61**		
	S_10	**40**			
	S_2A_C	1			
	S			1	
	V_S				**3**
	Total	47	95	88	11
Year	Substation	C	R	R-C	S
2018	**B_3**			**64**	
	C_T	2			
	E_D_32_A		**2**		
	F_12_18	21			
	M_S		**46**		
	R_4	**1**			
	S_		**38**		
	S_10	**9**			
	S_2D-F	5			
	V_S				**1**
	Total	38	86	64	1

Note. Substations in bold font are those that had deviating behaviours in both 2017 and 2018. C: company, R: residential, R-C: residential and company, S: school.

Residential categories had the most number of deviating behaviours during weeks 42 to 45 in 2017.

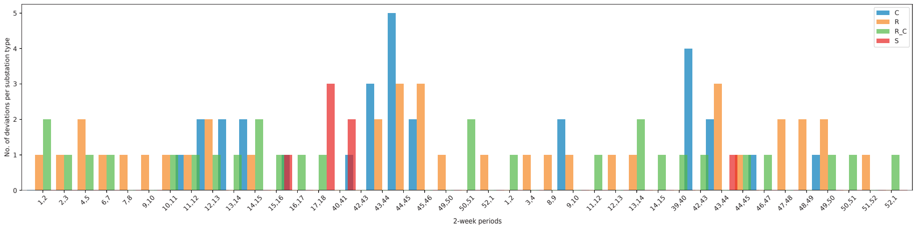


Figure 13.4: Number of substations with bi-weekly deviating behaviours on a daily basis and their category during heating season in 2017 and 2018.

13.7.3 Patterns representative of deviating behaviours

Extracted patterns can provide meaningful information for the domain experts, i.e., each pattern represents the status of the five selected features at a specific time period. Note that each category shows the status of a feature at time t in comparison to its value in time $365 - t$. As it was mentioned in Section 13.6.2, the yearly seasonality of the data is adjusted by the differencing method. Table 13.6 shows the top 10 weekly patterns detected as deviating behaviours in 2017 and 2018. These patterns are exemplars (representative) of the bi-weekly consensus clustering solutions. As one can see in all patterns except pattern number 7, features 3 and 4 (shown underlined in table) primary heat and primary mass flow rate, respectively, hold similar values.

Table 13.6: Top 10 weekly patterns detected as outliers in 2017 and 2018 in total

No.	Pattern	Count
1	high,high,high,high,high	80
2	high,high,high,high,medium_high	63
3	medium_high,medium,medium_high,medium_high,medium	39
4	medium,medium,medium,medium,medium_high	39
5	low,high,low_medium,low_medium,high	21
6	medium,medium_high,medium,medium,high	15
7	medium_high,low_medium,medium,medium_high,low_medium	13
8	medium_high,medium_high,medium,medium,medium	13
9	low_medium,low_medium,medium,medium,medium	11
10	medium,medium,low,low,medium	7

Table 13.7 shows top weekly patterns based on four categories of the substations. The majority of patterns have only occurred for specific types of substations, except pattern "medium, medium, medium, medium, medium_high", which is observed for types R-C and S in 2017 (row 6 and 8) and R in 2018 (row 2). Patterns belonging to *Residential-Company* and *Residential* categories are in total the most frequent in count. In addition, some of these patterns re-occurred in both 2017 and 2018 for the same category of substations, as shown in bold in table.

13.7.4 Substation Performance

Substation efficiency, E_s^T, can be used as an indicator to assess a substation's operational behaviour throughout the entire year. Figure 13.5 depicts the detected deviations for two substations belonging to the *Residential* category

Table 13.7: Top weekly patterns detected as outliers with respect to substation types in 2017 and 2018

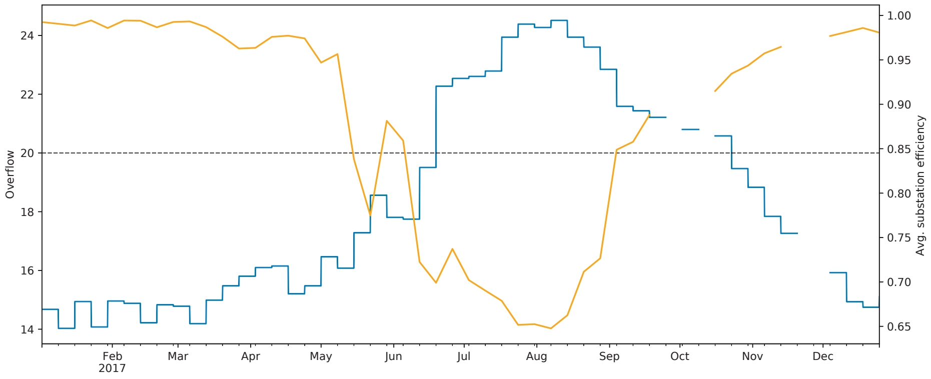
Year	No.	Type	Pattern	Count
2017	1	R-C	**high,high,high,high,medium__high**	**57**
	2	R	**high,high,high,high,high**	**55**
	3	C	medium__high,low__medium,medium,medium__high,low__medium	13
	4	R	low__medium,low__medium,medium,medium,medium	11
	5	C	medium__high,medium__high,medium,medium,medium	10
	6	R-C	**medium,medium,medium,medium,medium__high**	**8**
	7	C	medium,medium,low,low,medium	7
	8	S	**medium,medium,medium,medium,medium__high**	**6**
	9	R-C	medium,high,low__medium,low__medium,high	5
	10	R-C	medium,low__medium,low__medium,low__medium,low__medium	5
2018	1	R-C	medium__high,medium,medium__high,medium__high,medium	39
	2	R	**medium,medium,medium,medium,medium__high**	**25**
	3	R	**high,high,high,high,high**	**24**
	4	C	low,high,low__medium,low__medium,high	21
	5	R	medium,medium__high,medium,medium,high	15
	6	R-C	**high,high,high,high,medium__high**	**6**
	7	R	high,medium,high,high,medium	5
	8	C	low,low,medium__high,high,low__medium	5
	9	C	high,low__medium,medium,medium__high,low__medium	4
	10	R-C	medium__high,high,high,medium,high	4
	11	R-C	medium__high,medium__high,medium__high,medium__high,high	4

Note. Patterns in bold font occurred in both years.

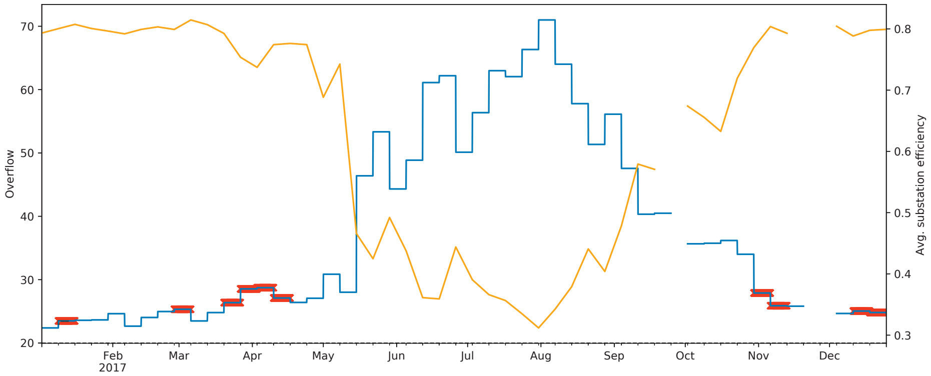
using their average efficiency and overflow for year 2017. Figure 13.5a, represents a well-performed substation. Notice that the substation's efficiency on average is around 98%. In addition the weekly substation's overflows for the whole heating season (January-May and November-December) are below 20. Figure 13.5b, on the other hand, shows substation with sub-optimal performance during 2017. As one can see the substation's efficiency on average is around 80% during heating season. Moreover, the weekly overflows for the whole year are above 20. In addition, the proposed approach identified deviating behaviours in 10 weeks which are marked with red.

Figure 13.6 represents the performance of the same substations based on weekly overflow and outdoor temperature. Something that is noticeable in Figure 13.6b is that whenever the outdoor temperature has a sudden change the purposed approach observes that as deviating behaviour.

Notice that in this study we only consider the smallest sub-tree(s), after cutting the longest edge(s) of an MST, as outlier(s). Nevertheless, one can consider sorting the sub-trees based on their size from smallest to the largest for further analysis. Alternatively, by defining a domain specific threshold any edges with a distance greater than the threshold can be removed and



(a) Well performed substation: overflow (blue line) below 20 during heating season vs. substation effectiveness (orange line)



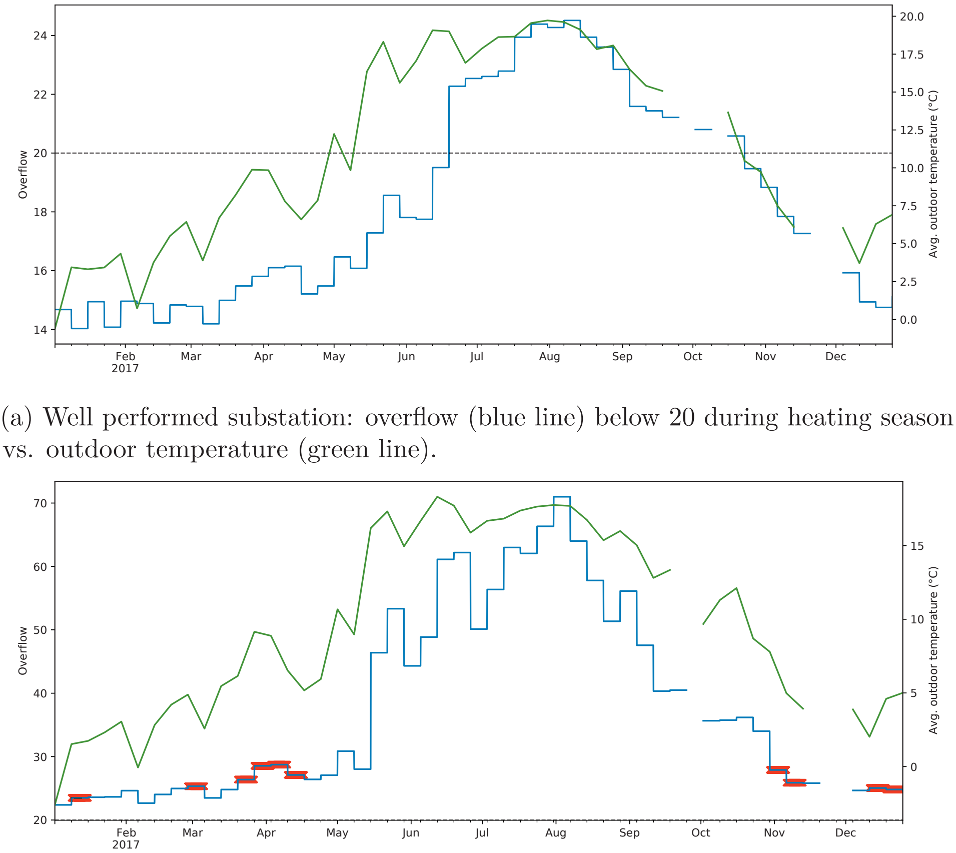
(b) Sub-optimal performed substation: overflow (blue line) above 20 during heating season vs. substation effectiveness (orange line). The red marks represent identified deviating behaviours in 10 weeks.

Figure 13.5: Example of well and sub-optimal performed substations in 2017. Overflow vs. Substation's efficiency. Due to missing values some of the weeks are missing.

further analysis can be performed on smaller sub-trees.

13.8 Conclusion and Future Work

We have proposed a higher order data mining approach for analyzing real-world datasets. The proposed approach combines different data analysis techniques for 1) building a behavioural model of the studied phenomenon,



(a) Well performed substation: overflow (blue line) below 20 during heating season vs. outdoor temperature (green line).

(b) Sub-optimal performed substation: overflow (blue line) above 20 during heating season vs. outdoor temperature (green line). The red marks represent identified deviating behaviours in 10 weeks.

Figure 13.6: Example of well and sub-optimal performed substations in 2017. Overflow vs. outdoor temperature. Due to missing values some of the weeks are missing.

2) using the built model for monitoring the phenomenon's behaviour, and 3) finally identifying and further analyzing deviating behavioural patterns. At each step, adequate information is supplied which can facilitate the domain experts in decision making and inspection.

The approach has been demonstrated and evaluated on a case study from the district heating (DH) domain. For this purpose, we have used data collected for a three-year period of 47 operational substations belonging to four different heat load categories (see Table 13.1). The results have shown that the method is able to identify and analyze deviating and sub-optimal behaviours of the DH substations. In addition, the proposed approach provides different techniques for monitoring and data analysis, which can facilitate domain experts to better understand and interpret the DH substations' operational behaviour and performance.

For future work we aim to pursue further analysis and evaluation of the proposed approach on similar scenarios in different application domains, e.g., fleets of wind turbines. In addition, we want to extend the proposed approach with means for root-cause analysis and diagnosis of detected deviations.

Author Contributions: conceptualization, S.A. and V.B.; methodology, S.A. and V.B.; software, S.A.; validation, S.A and J.B.; formal analysis, S.A.; investigation, S.A., J.B.; data curation, S.A.; writing–original draft preparation, S.A.; writing–review and editing, S.A., V.B., J.B., and H.G.; visualization, S.A.; funding acquisition, H.G.

Funding: This work is part of the research project "*Scalable resource-efficient systems for big data analytics*“ funded by the Knowledge Foundation (grant: 20140032) in Sweden.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Abbreviations: The following abbreviations are used in this manuscript:

AP	Affinity Propagation
C	Company
DH	District Heating
DHW	Domestic Hot Water
FDD	Fault Detection and Diagnosis
HOM	Higher Order Mining
LD	Levenshtein Distance
MAD	Median Absolute Deviation
MST	Minimum Spanning Tree
PAA	Piecewise Aggregate Approximation
R	Residential
R-C	Residential and Company
S	School
SAX	Symbolic Aggregate Approximation
SD	Standard Deviation

References

[1] R. Isermann. *Fault-diagnosis systems: an introduction from fault detection to fault tolerance.* Springer Science & Business Media, 2006.

[2] V. Hodge and J. Austin. "A survey of outlier detection methodologies". In: *Artificial Intelligence Review* 22.2 (2004), pp. 85–126.

[3] V. Chandola, A. Banerjee, and V. Kumar. "Anomaly detection: A survey". In: *ACM Computing Surveys* 41.3 (2009), p. 15.

[4] Y. Zhang, N. Meratnia, and P. Havinga. "Outlier detection techniques for wireless sensor networks: A survey". In: *IEEE Communications Surveys & Tutorials* 12.2 (2010), pp. 159–170.

[5] M. Gupta, J. Gao, C. C. Aggarwal, and J. Han. "Outlier detection for temporal data: A survey". In: *IEEE Transactions on Knowledge and Data Engineering* 26.9 (2014), pp. 2250–2267.

[6] C. C. Aggarwal. "Outlier analysis". In: *Data mining.* Springer. 2015, pp. 237–263.

[7] R. Isermann. "Supervision, fault-detection and fault-diagnosis methods—an introduction". In: *Control engineering practice* 5.5 (1997), pp. 639–652.

[8] S. Katipamula and M. R. Brambley. "Methods for fault detection, diagnostics, and prognostics for building systems-A review, part I". In: *Hvac&R Research* 11.1 (2005), pp. 3–25.

[9] S. Katipamula and M. R. Brambley. "Methods for fault detection, diagnostics, and prognostics for building systems-A review, part II". In: *Hvac&R Research* 11.2 (2005), pp. 169–187.

[10] S. Abghari, V. Boeva, J. Brage, C. Johansson, H. Grahn, and N. Lavesson. "Higher Order Mining for Monitoring District Heating Substations". In: *2019 IEEE Int'l Conf. on Data Science and Advanced Analytics (DSAA).* IEEE. 2019, pp. 382–391.

[11] C. H. Fontes and O. Pereira. "Pattern recognition in multivariate time series–A case study applied to fault detection in a gas turbine". In: *Engineering Applications of Artificial Intelligence* 49 (2016), pp. 10–18.

[12] A. Sánchez-Fernández, F. Baldán, G. Sainz-Palmero, J. Benıtez, and M. Fuente. “Fault detection based on time series modeling and multivariate statistical process control”. In: *Chemometrics and Intelligent Laboratory Systems* 182 (2018), pp. 57–69.

[13] D. Djenouri, R. Laidi, Y. Djenouri, and I. Balasingham. “Machine Learning for Smart Building Applications: Review and Taxonomy”. In: *ACM Computing Surveys* 52.2 (2019), p. 24.

[14] H. Gadd and S. Werner. “Fault detection in district heating substations”. In: *Applied Energy* 157 (2015), pp. 51–59.

[15] P. Xue, Z. Zhou, X. Fang, X. Chen, L. Liu, Y. Liu, and J. Liu. “Fault detection and operation optimization in district heating substations based on data mining techniques”. In: *Applied Energy* 205 (2017), pp. 926–940.

[16] A. Capozzoli, F. Lauro, and I. Khan. “Fault detection analysis using data mining techniques for a cluster of smart office buildings”. In: *Expert Systems with Applications* 42.9 (2015), pp. 4324–4338.

[17] S. Månsson, P.-O. J. Kallioniemi, K. Sernhed, and M. Thern. “A machine learning approach to fault detection in district heating substations”. In: *Energy Procedia* 149 (2018), pp. 226–235.

[18] E. Calikus, S. Nowaczyk, A. Sant’Anna, H. Gadd, and S. Werner. “A Data-Driven Approach for Discovery of Heat Load Patterns in District Heating”. In: *arXiv preprint arXiv:1901.04863* (2019).

[19] J. Paparrizos and L. Gravano. “k-shape: Efficient and accurate clustering of time series”. In: *Proc. of the 2015 ACM SIGMOD Int’l Conf. on Management of Data*. ACM. 2015, pp. 1855–1870.

[20] F. Sandin, J. Gustafsson, and J. Delsing. *Fault detection with hourly district energy data: Probabilistic methods and heuristics for automated detection and ranking of anomalies*. Svensk Fjärrvärme, 2013.

[21] J. F. Roddick, M. Spiliopoulou, D. Lister, and A. Ceglar. “Higher order mining”. In: *ACM SIGKDD Explorations Newsletter* 10.1 (2008), pp. 5–17.

[22] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M. Hsu. “Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth”. In: *Proc. of the 17th Int’l Conf. on Data Engineering*. 2001, pp. 215–224.

[23] B. J. Frey and D. Dueck. “Clustering by passing messages between data points”. In: *Science* 315.5814 (2007), pp. 972–976.

[24] J. MacQueen et al. “Some methods for classification and analysis of multivariate observations”. In: *Proc. of the Fifth Berkeley Symp. on Mathematical Statistics and Probability*. Vol. 1. 14. Oakland, CA, USA. 1967, pp. 281–297.

[25] A. Gionis, H. Mannila, and P. Tsaparas. “Clustering Aggregation”. In: *ACM Transaction of Knowledge Discovery Data* 1.1 (2007).

[26] V. Boeva, E. Tsiporkova, and E. Kostadinova. “Analysis of Multiple DNA Microarray Datasets”. In: *Springer Handbook of Bio-/Neuroinformatics*. Springer Berlin Heidelberg, 2014, pp. 223–234.

[27] A. Goder and V. Filkov. “Consensus Clustering Algorithms: Comparison and Refinement”. In: *ALENEX*. 2008, pp. 109–234.

[28] J. Lin, E. Keogh, L. Wei, and S. Lonardi. “Experiencing SAX: a novel symbolic representation of time series”. In: *Data Mining and knowledge discovery* 15.2 (2007), pp. 107–144.

[29] V. I. Levenshtein. “Binary codes capable of correcting deletions, insertions, and reversals”. In: *Soviet physics doklady*. Vol. 10. 8. 1966, pp. 707–710.

[30] C. C. Aggarwal and P. S. Yu. “Outlier detection for high dimensional data”. In: *ACM Sigmod Record*. Vol. 30. 2. ACM. 2001, pp. 37–46.

[31] M.-F. Jiang, S.-S. Tseng, and C.-M. Su. “Two-phase clustering process for outliers detection”. In: *Pattern Recognition Letters* 22.6 (2001), pp. 691–700.

[32] A. C. Müller, S. Nowozin, and C. H. Lampert. “Information theoretic clustering using minimum spanning trees”. In: *Joint DAGM (German Association for Pattern Recognition) and OAGM Symp.* Springer. 2012, pp. 205–215.

[33] X. Wang, X. L. Wang, and D. M. Wilkes. “A minimum spanning tree-inspired clustering-based outlier detection technique”. In: *Ind. Conf. on Data Mining*. Springer. 2012, pp. 209–223.

[34] G.-W. Wang, C.-X. Zhang, and J. Zhuang. “Clustering with Prim’s sequential representation of minimum spanning tree”. In: *Applied Mathematics and Computation* 247 (2014), pp. 521–534.

[35] J. B. Kruskal. “On the shortest spanning subtree of a graph and the traveling salesman problem”. In: *Proc. of the American Mathematical Society* 7.1 (1956), pp. 48–50.

[36] S. Frederiksen and S. Werner. *District Heating and Cooling.* Studentlitteratur AB, 2013. ISBN: 9789144085302.

[37] B. L. Ford. “An overview of hot-deck procedures”. In: *Incomplete data in sample surveys* 2.Part IV (1983), pp. 185–207.

[38] D. B. Rubin. “Multiple imputations in sample surveys-a phenomenological Bayesian approach to nonresponse”. In: *Proc. of the survey research methods section of the American Statistical Association.* Vol. 1. American Statistical Association. 1978, pp. 20–34.

[39] O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani, D. Botstein, and R. B. Altman. “Missing value estimation methods for DNA microarrays”. In: *Bioinformatics* 17.6 (2001), pp. 520–525.

[40] F. R. Hampel. “A general qualitative definition of robustness”. In: *The Annals of Mathematical Statistics* (1971), pp. 1887–1896.

[41] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. “Scikit-learn: Machine Learning in Python”. In: *J. of Machine Learning Research* 12 (2011), pp. 2825–2830.

[42] A. Hagberg, P. Swart, and D. S Chult. *Exploring network structure, dynamics, and function using NetworkX.* Tech. rep. Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.

ABSTRACT

Outlier detection is studied and applied in many domains. Outliers arise due to different reasons such as fraudulent activities, structural defects, health problems, and mechanical issues. The detection of outliers is a challenging task that can reveal system faults, fraud, and save people's lives. Outlier detection techniques are often domain-specific. The main challenge in outlier detection relates to modelling the normal behaviour in order to identify abnormalities. The choice of model is important, i.e., an unsuitable data model can lead to poor results. This requires a good understanding and interpretation of the data, the constraints, and requirements of the domain problem. Outlier detection is largely an unsupervised problem due to unavailability of labeled data and the fact that labeled data is expensive.

In this thesis, we study and apply a combination of both machine learning and data mining techniques to build data-driven and domain-oriented outlier detection models. We focus on three real-world application domains: maritime surveillance, district heating, and online media and sequence datasets. We show the importance of data preprocessing as well as feature selection in building suitable methods for data modelling. We take advantage of both supervised and unsupervised techniques to create hybrid methods.

More specifically, we propose a rule-based anomaly detection system using open data for the maritime surveillance domain. We exploit sequential pattern mining for identifying contextual and collective outliers in online media data. We propose a minimum spanning tree clustering technique for detection of groups of outliers in online media and sequence data. We develop a few higher order mining approaches for identifying manual changes and deviating behaviours in the heating systems at the building level. The proposed approaches are shown to be capable of explaining the underlying properties of the detected outliers. This can facilitate domain experts in narrowing down the scope of analysis and understanding the reasons of such anomalous behaviours. We also investigate the reproducibility of the proposed models in similar application domains.

BLEKINGE TEKNISKA HÖGSKOLA • BTH •

ISSN: 1653-2090
ISBN: 978-91-7295-409-0

2020:09