Master Thesis

Utrecht University

Detecting outages in the Dutch medium voltage electrical grid on the basis of telemetry signals

---

Patrick van Eijk

January 2019

Master Thesis

Submitted in partial fulfillment of the requirements for the degree Master of Science in the Subject of Artificial Intelligence

Detecting outages in the Dutch medium voltage electrical grid on the basis of telemetry signals

Patrick van Eijk

January 2019

1. Examiner dr. ing. habil. Georg M. Krempl
Department of Information and Computing Sciences
Utrecht University

2. Examiner prof. dr. Arno P.J.M. Siebes
Department of Information and Computing Sciences
Utrecht University

Utrecht University

1. Supervisor MSc Justus van de Sande
Asset Information Management
Stedin

2. Supervisor BSc Bart Bikker
Network Operation Centre
Stedin

Patrick van Eijk
Detecting outages in the Dutch medium voltage electrical grid on the basis of telemetry signals
January 2019
Reviewers: dr. ing. habil. Georg M. Krempl & prof. dr. Arno P.J.M. Siebes
Supervisors: MSc Justus van de Sande & BSc Bart Bikker

Utrecht University
Algorithmic Data Analysis Group
Faculty of Science
Department of Information and Computing Sciences
Princetonplein 5
3584 CC Utrecht

Stedin
Asset Information Management
Blaak 8
3011 TA Rotterdam

Abstract

A worldwide change in power generation is inevitable, which requires a transition from fossil fuels to only clean energy. This transition requires a reliable electricity network. Stedin is a regional distribution system operator for gas and electricity that facilitates this transition. The Network Operation Centre (NOC) monitors and manages the network ensuring the continuous energy supply. Some parts of the gas and electricity network have telemetry and are able to send signals to the NOC. Alarm management is a big part of the operators' job. Problematic patterns need to be recognized so that outages can be detected and fixed promptly.

The goal of this thesis is to automatically detect outages from the signal data using machine learning technology. A combination of topology, signal and outage data is used. This research focuses specifically on the distribution part of the medium voltage network of Stedin. A combination of association rule learning and graph theory was used to find outage related patterns. These patterns are then used to build a classifier that detects outages according to the incoming signals.

Results indicate that this is a promising approach for alarm management using machine learning. Because the topology is incorporated, the model also identifies the affected medium voltage ring. A process that can take up to 5 minutes is done instantly using this approach. The method can be implemented in the NOC of Stedin, but could potentially also be applied to other alarm management systems.

Acknowledgement

This master thesis would not have been possible without some people. This section will acknowledge those people that helped making this project possible.

First of all, special thanks goes out to my supervisor from the University Utrecht, Georg Krempl, for being my supervisor when no one else would. Most teachers declined my request because they were too busy or had no faith in me. I lacked knowledge of data mining, but Georg Krempl was the only one who did not see this as an issue. Without him, I could not have continued this project and therefore many thanks to him.

Secondly, special thanks goes out to my supervisor from Stedin, Justus van de Sande, for helping me and guiding me through the process. He ensured I could always continue with my work. His criticism helped me improve my research and this thesis. Beside the fact that he was my supervisor, I would like to see him as a friend with whom I could drink a beer and have a laugh. Justus gave me a warm and friendly work environment, which made the past 8 months very pleasant.

Thirdly, special thanks goes out to my client and supervisor from Stedin, Bart Bikker, for being the specialist who could help me verify patterns, clusters and rules. My knowledge of physics has faded since high school, but Bart always helped me understand the processes related to the electrical grid. The small physics lectures were very interesting and helpful for understanding the problem.

Fourthly, special thanks goes out to Pieter Minnaar from Stedin, for always being there when I needed help with the data. Very little was documented about the data and Pieter was the person to go to. Whenever I needed something, he would drop whatever he was doing to assist me in my project.

Lastly, special thanks goes out to my parents, who both work at Stedin and established contact between Justus and me preceding this project. I did not think Stedin had a project related Artificial Intelligence and rejected the initial idea. My parents asked around anyway and I would have never found this project without them.

Contents

1 Introduction

The energy transition is near and a worldwide change in power generation is inevitable. The use of fossil fuels will mostly disappear and only clean energy will be used. With the Paris Agreement[90], the transition is well under way. This results in a higher demand for a reliable electricity network. The network needs to be able to cope with this change. Stedin is one of the companies that is responsible for maintaining the Dutch electrical grid and ensuring it is reliable. It therefore plays a big role in facilitating the energy transition.

The company Stedin is a regional distribution system operator for gas and electricity. Many signals from the gas and electricity network arrive at Stedin every day, which contains a lot of information on their own or combined. These signals have been divided into four categories, namely gas, low voltage, medium voltage and high voltage. Some of these signals are labelled as alarms and those arrive at the Network Operation Centre (NOC) in Stedin. This project will focus on the medium voltage part of the alarm management. The medium voltage network can be recognized by the so-called Ring Main Units (RMUs) that can be seen in the streets of cities. An example of such a Ring Main Unit is shown in Figure 1.1. Inside these buildings lies the connection between the medium and low voltage network.

Fig. 1.1: A Ring Main Unit of Stedin in a street somewhere in Rotterdam.

In the early days of alarm management at Stedin, there were only a few assets in the medium voltage electricity network that had telemetry and could send signals.

With the improving technology of the recent years, more and more assets in the medium voltage network have telemetry and are able to send signals. This results in a rapid increase in the amount of alarms. The classification of the (series of) alarms in the Network Operation Centre takes place largely on the basis of the experience of the NOC operators on duty. The task of alarm management has become more complex over the years. Instead of having multiple operators monitoring the alarms that arrive at the NOC, it should be possible to address this problem of alarm management more efficiently. The assignment is to investigate whether it is possible to automatically detect and locate outages with machine learning technology. The used approach could possibly be applied to many alarm management systems and is not restricted to the electricity sector.

1.1 Stedin

In today's everyday life, it is self-evident that everyone has a constant supply of electricity. It is unimaginable to live without it. Electricity is needed in almost every job, at home and in the streets. All in all, electricity needs to be available 24 hours a day, all year long. Stedin is one of the companies in the Netherlands that ensures that this need is met. As mentioned earlier, Stedin is a regional distribution system operator. It operates and maintains the regional network for electricity and gas to ensure the connection between the supplier and the consumer. The national transmission system operators, TenneT for electricity and Gasunie for gas, maintain the stability of the respective electricity and gas networks. They ensure that the amount of electricity or gas that leaves the network is equal to the amount that enters the network. The service area of Stedin for electricity and gas is shown in Figure 1.2. The electricity that is supplied by TenneT is always under high voltage, this is then converted to medium voltage. Stedin operates mostly on the medium and low voltage part of the electrical grid.

Stedin has more than 3,000 employees and over 2 million clients. The vision of the company is to "focus on core tasks for future network management with excellent service to customers"[17]. To accomplish this, Stedin has three major spearheads:

- **Better grid management**
- **Facilitating the energy transition**
- **Sustainable business operations**

Fig. 1.2: Service area of Stedin for electricity (yellow) and gas (black)[62].

The second spearhead is where this project fits in. By providing a potential improvement to the alarm management system, outages in the network will be resolved more quickly and maybe even prevented. This results in more reliable networks and less down-time.

1.2 Project

Stedin receives around 20,000 signals every day from the medium voltage network, with peaks of 100,000 signals a day. While many of those signals arrive at the Network Operation Centre as alarms, there are only around 1,000 outages every year. This means that most of these alarms are false alarms because of for example maintenance, tests or other causes. The NOC operators need to monitor and filter all these alarms and find the problematic alarms, which can be a complicated task. The process of reacting to an outage is visualized in Figure 1.3.

The figure shows several steps which lead to fixing the problem. One part is detecting problematic patterns and another part is locating the problem. To give an indication, it takes around 5 minutes between an alarm and the acting of the operator. It is suspected that it should be possible to do this more efficiently using machine learning. The following main research question arises:

1. ***Is it possible to filter out the vast majority of signals and detect the outages?***

To answer this question, two sub-questions have been formulated that need to be answered.

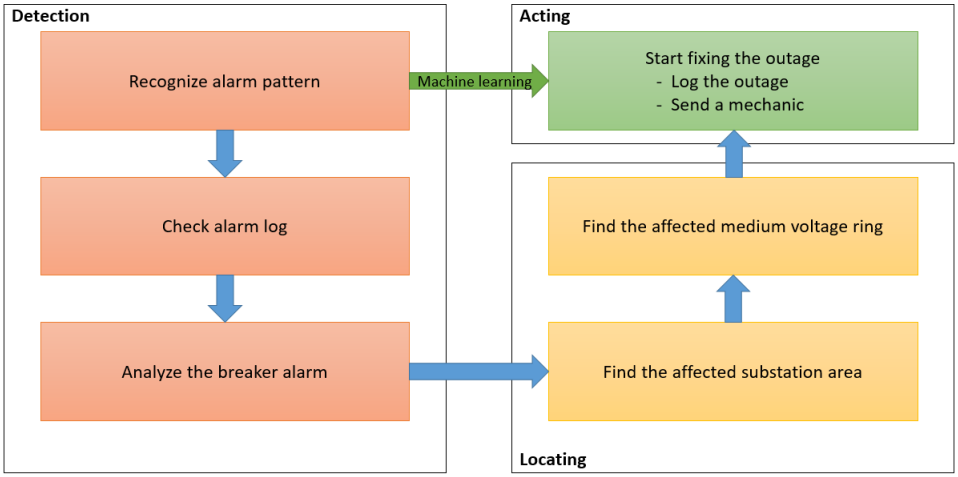

Fig. 1.3: The process of an operator reacting to an outage in the Network Operation Centre of Stedin. The green arrow represent the potential machine learning technique that could help save time.

(i) What percentage of true positives is the machine learning technique able to detect?

(ii) What percentage of signals is the machine learning technique able to filter out without missing additional true positives?

When the operator notices something is wrong, it is not directly clear where the outage occurred. For instance, when a signal suggests there is a short circuit in the network, the operator still needs to find the affected substation area and thereafter find the affected medium voltage ring. In short, the source of the problem has to be located approximately. This can be a very intensive job, because the operator has to look at the topology of the network and discard the areas that are certainly not the problem. Of the average 5 minutes, this part of the process consumes the most time. This leads to the second main research question:

2. ***How can you make a machine learning system that can locate the source of the problem?***

Again, this question can be dissected into two sub questions.

(i) To what extend is the machine learning system able to locate the correct source of the problem?

(ii) How much time is saved by using machine learning?

The first research question applies to the 'Detection' part of the process shown in Figure 1.3, while the second research question applies to the 'Locating' part of the process. If results show that it is possible to detect and locate the outages, then a lot of time can be saved on these parts of the process.

The research questions mentioned in this section will be answered throughout the thesis. The conclusion will revisit all the research questions and answer them. At the end of the project, Stedin wants to further improve its network with the results of this research. As mentioned in the previous section, the improvement of the alarm management will contribute to the spearhead of facilitating the energy transition. Although this is not a research question, in the interest of Stedin a final question arises.

3. ***In what way can the applied method help Stedin ensure more efficient alarm management?***

1.3 Outline

The outline of this thesis is as follows. First, the related literature will be discussed in Chapter 2. After this, the used methods will be described in Chapter 3. The following Chapter 4 will discuss the experimental setup and the used data. The results of the experiments will be presented in Chapter 5, together with a discussion. Last but not least, a conclusion will be given in Chapter 6.

Literature study

2

This chapter will discuss the relevant literature that was found. The first section focusses on the Dutch electrical grid. Literature on alarm management will be discussed in the second section. Graph theory will briefly be discussed in the third section, which is relevant because the Dutch electrical grid can be seen as one big graph. Lastly, this chapter will conclude with literature on association rule learning, which is the machine learning technique that is used for this project.

2.1 Electrical grid

Everything discussed in this section is according to the Dutch electrical grid. The information is acquired from the book from the company Phase to Phase[89]. The part of the electrical grid that is operated by Stedin, is structured and operated according to this information.

2.1.1 Structure of the grid

The electrical grid in the Netherlands is divided into four different voltage levels, which can be seen in Figure 2.1.

- *High voltage* is shown in red with voltages ranging from 110 kV to 380 kV. This part of the Dutch electrical grid is managed by the national transmission system operator TenneT. These networks have a transport function.
- *Intermediate voltage* is shown in green with voltages ranging from 25 kV to 50 kV. These networks have a transport function.
- *Medium voltage* is shown in black with voltages ranging from 10 kV to 20 kV. Parts of these networks can have a transport function or a distribution function.
- *Low voltage* is shown in blue with voltages of 0.4 kV. These networks have a distribution function.

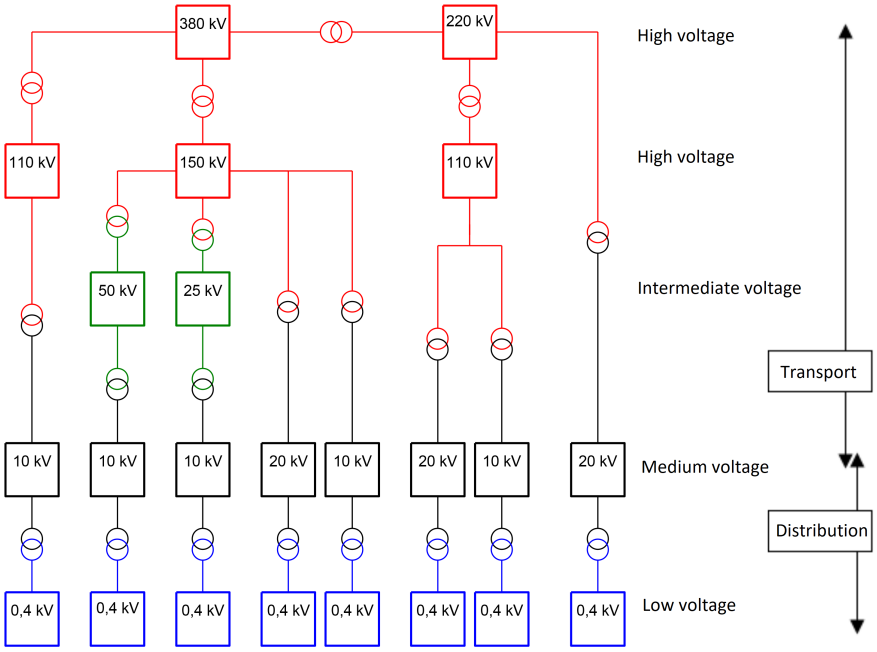

Fig. 2.1: The different voltage levels of the electrical grid in the Netherlands[89].

The intermediate, medium and low voltage levels are managed by regional distribution system operators, like Stedin. Some small parts of the high voltage network is also managed by Stedin, but this is mostly managed by TenneT. Transformers provide the connection between different voltage levels, which are depicted as two intersecting circles. As mentioned in the introduction, this project focuses on the medium voltage part of the network, thus with voltages between 10 kV and 20 kV. To summarize, the part from the connection to high or intermediate voltage network until the connection to low voltage network is the relevant part.

To give a clearer picture of how this is medium voltage level is build up, three different stations will be discussed. There are more stations that can occur, but those are very similar or irrelevant. The connection to the high or intermediate voltage network lies in a substation, which can be seen in Figure 2.2. In a substation, a cable with high or intermediate voltage enters and the electricity is converted to medium voltage by a transformer. This electricity is transferred to a so-called busbar, which is depicted as a thick vertical line in the figure. Medium voltage cables can branch from the busbar. These can have a transport function, as depicted in the figure, or a distribution function.

Switching stations are placed between the transport and distribution part of the medium voltage networks to connect the two parts. An example of this is also

shown in Figure 2.2. A switching station also has a busbar where the electricity gets distributed over the outgoing cables.

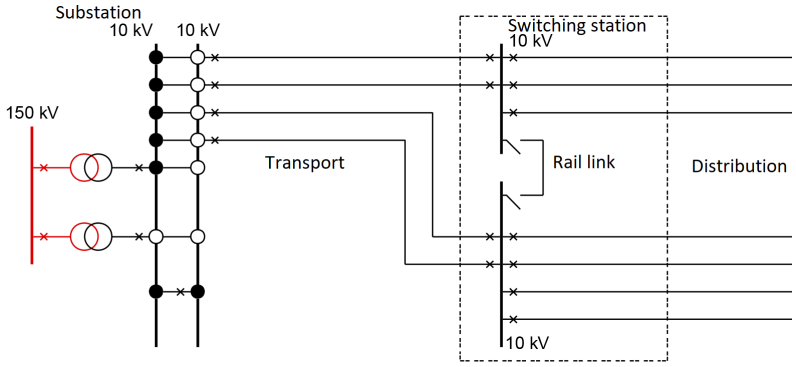

Fig. 2.2: Schematic example of a substation and a switching station[89].

As mentioned earlier, in a Ring Main Unit lies the connection between the medium voltage network to the low voltage network. Figure 2.3 shows example of the schematic of a Ring Main Unit as shown in Figure 1.1. It can be seen in the schematic that there is also a busbar system, with a transformer that converts medium voltage electricity into low voltage.

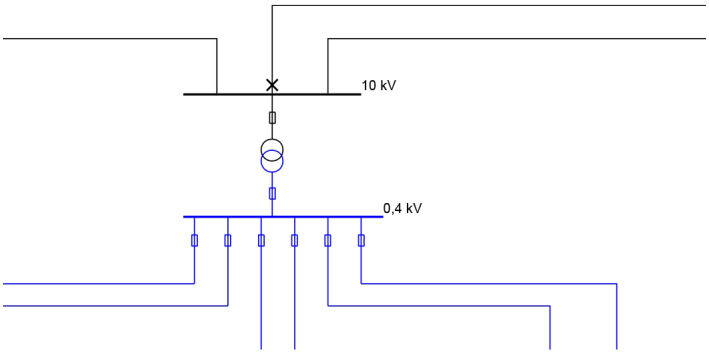

Fig. 2.3: Schematic example of a Ring Main Unit[89].

2.1.2 Quality and availability

In the current everyday life it is self-evident to have electricity available throughout the day. This means that distribution system operators need to ensure this quality and availability. When there is a short circuit or some other problem somewhere in the electrical grid, this needs to be resolved quickly. In the medium voltage network, an average of 1,000 affiliates is affected by a short circuit. When managing the quality and availability, a distribution system operator will consider single outages.

There are four different gradations that are defined for resolving an outage in the network.

- *Single outage reserve without energy interruption*: When there is a single outage in the network, this will not result in the interruption of power supply. There are multiple parallel paths to the source of the power supply, which means that the reserve is directly available.
- *Single outage reserve with energy interruption*: When there is a single outage in the network, this will result in the interruption of power supply. The power supply can be restored through switching. The reserve is stand-by and can be switched on by operators in the Network Operation Centre.
- *No outage reserve, restoration by use of aggregate*: When there is a single outage in the network, this will result in the interruption of power supply. The power supply can only be restored by using an aggregate, because there is no reserve.
- *No outage reserve*: When there is a single outage in the network, this will result in the interruption of power supply. There no reserve and there is no possibility to connect an aggregate, thus the power supply can not be restored.

As mentioned earlier, an outage in the medium voltage network may affect a lot of affiliates. Therefore there is always the possibility to restore the power supply. In the transport part of the medium voltage network, there is the possibility for outage reserve without energy interruption because of the high demands.

In the distribution part of the medium voltage network, there is outage reserve with energy interruption. These distribution networks are often structured as rings, but the ring is interrupted somewhere with an opened switched. If there is an outage somewhere in the medium voltage ring, the operators can close a switch to restore the power supply. In some rare cases there is no outage reserve in which case an aggregate can be used to restore the power supply.

2.1.3 Network structures

The electrical network can be structured in different ways. This section will briefly discuss the different structures, the structures that are used by the medium voltage network and the explanation why the medium voltage network is structured the way it is. A few definitions will be discussed first, the visualization of these definitions can be seen in Figure 2.4.

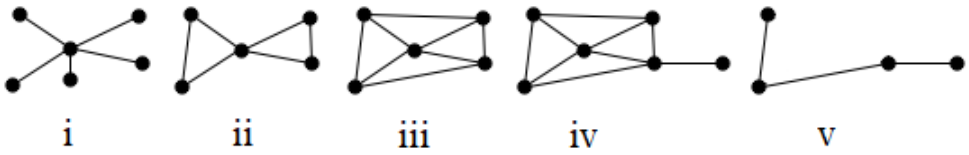

Fig. 2.4: Different structures that occur in the electrical grid[89].

(i) *Radial network* is a structure where the receiving point is connected to the source of the power supply through one link. There is no possibility for single outage reserve.

(ii) *Ring network* is a structure where the receiving point is connected to the source of the power supply through two links. There is possibility for single fault reserve without energy interruption, because there are multiple parallel paths to the power supply source. In the case where the ring network is practised radially, there is a net opening in the network. There is the possibility for single outage reserve with energy interruption, by closing the net opening.

(iii) *Mesh network* is a structure where the receiving point is connected to the source of the power supply through more than two links. There is possibility for single fault reserve without energy interruption, because there are multiple parallel paths to the power supply source. In the case where the ring network is practised radially, there are multiple net openings in the network. There is the possibility for single fault reserve with energy interruption, by closing the net opening.

(iv) *Branch* is a radial part coming from a ring or mesh network. There is no possibility for single fault reserve.

(v) *Cord* is a chain of stations that can be switched off with one mutual switch.

The distribution part of the medium voltage network is mostly structured as rings, but sometimes also meshed and by exception there can be a radial part (for example in areas with a low density of costumers). It is practised radially, thus with one or more open net openings in the ring or mesh network. This way single fault reserve with energy interruption is possible.

One reason that ring and mesh networks are practised radially is that if there is a short circuit, it is not supplied with electricity from two or more directions. Also if it was not practised radially, the short circuit had to be switched off from the network in multiple places. This requires a complex and expensive security system to keep quality and availability consistent.

An example of a medium voltage ring is shown in Figure 2.5, where the black dots outside the substation are RMUs. Some of the assets in these stations have telemetry and are able to send signals to the Network Operation Centre. The topology can be seen as a concatenation of assets. Graph theory can be used to exploit the topology and divide the signals into the different medium voltage rings.

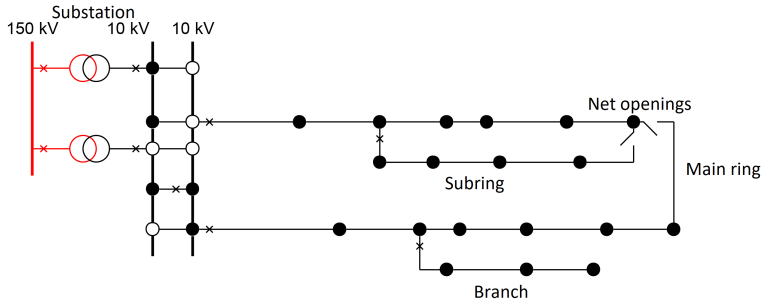

Fig. 2.5: Schematic example of a medium voltage ring[89].

2.2 Alarm management

Before diving into the literature of alarm management, a few definitions have to be provided. A signal is some kind of message sent by an asset to indicate a certain event or state. An alarm is basically a signal that is annunciated to the operator as an audible or visible indicating a possible problem in the system[34]. In the Network Operation Centre of Stedin, the alarms are displayed as a list and some important alarms have certain audible sounds. Note that not all signals are alarms, because not all signals are shown to the NOC operators on duty. Every signal consists of a date, a time, a location and a message. This message can be an event or a state.

There are approximately 25,000 stations in the medium voltage network of Stedin, of which approximately 3,000 have assets with telemetry. All these 3.000 assets are able to send signals to the Network Operation Centre. Note that the sensor values are not incorporated in the signals. The event is the relevant part of a signal and this indicates what happened.

A nuisance alarms is defined as a constant false signal for attention shift, which is a contributor to the problem of alarm management[124]. Note that this does not imply that a nuisance alarm is a false alarm. Examples of nuisance alarms are "door open", "door closed", "lost signal" and "re-established signal". These nuisance alarms may result in missing the important alarms that indicate an outage. The used

method could possibly filter out most of it, resulting in less attention shifts and more reliable monitoring.

2.2.1 Alarm systems

Alarm management is necessary for all systems that have important continuous processes. The most important examples are the power, the chemical, the mining, the pharmaceutical and the petroleum manufacturing industries[103]. If a system fails, it is important that this is noticed quickly so the problem can be resolved. The process of alarm management starts with sensors that sense a change in the system. These sensors then send signals to an alarm centre, where the alarms are displayed. The operator's role is to filter these alarms and detect abnormalities.

The way alarms are displayed can be divided into two categories, a parallel display and a serial display[125]. In a parallel display system the different categories of alarms are displayed on different screens, while in a serial display system the alarms are displayed in one list. Parallel has the advantage that the different categories of alarms are split, but the operators can be overwhelmed by the amount of screens they need to monitor. With serial display the operator only needs to monitor one screen, but the amount of alarms can overwhelm the operator in this case. At Stedin, the alarm management of the medium voltage electrical grid is done using serial display.

Most of the early work done on alarm management is about designing a rule based system that defines when a sensor sends a signal using certain threshold. This is be used in various systems, like nuclear power plants[125][102] and fluid infusion management systems[26]. There is also work on reducing unnecessary alarms in such systems[76][112][7]. At Stedin, there is also rule based alarm management where the assets with telemetry will only send signals when a change in the system is sensed. Adjusting this rule based system might result in more efficient alarm management.

2.2.2 Alarm fatigue

When the alarm frequency is high, there is a risk to become desensitized to the alarms and develop alarm fatigue[48][109][113][33]. The operator will be ignoring alarms, assuming they are false alarms. Also the operator will be busy processing

and checking false alarms, which takes the operator away from more important tasks[95].

The loads of false alarms is the result of manufacturers being biased towards alarming conservatively[84]. They act on the industry paradigm that too few alarms is worse than too many[101]. The manufacturer would rather produce extra false alarms than miss a true alarm. Poor alarm management can cost the company a lot of money and the affiliates can be affected longer than necessary[16]. Companies want to strive for a perfect alarm management system where the sensitivity would be 100% and the specificity would be 100%, which means the system will never miss a true alarm and the system will never give a false alarm[110].

A lot of work in alarm fatigue is done in the pharmaceutical sector, because there can be a lot of consequences regarding the patient's health, namely serious harm or even death[27]. Although there has been a lot of research on alarm fatigue in the pharmaceutical sector, there has been little work on reducing the amount of false alarms[66][110]. Some work has been done on individualizing alarm thresholds, but there does not exist any evidence to support the claim that this will work[59]. Before the alarms can be individualized by the staff, there needs to be a clear policy that defines what can and what cannot be done under certain circumstances[104]. Even if such a policy existed, the question whether the system improves or not stays.

2.2.3 Alarm analysis

A big part of alarm management is analyzing the alarms, from which the operator draws conclusions. This can be identifying problematic patterns in alarm data or locating the root cause of the problem. When there is an alarm flood (more than 10 alarms in 10 minutes)[6], it can be hard for the operator to analyze the alarms efficiently. Therefore a lot of work on alarm analysis has been done in the industrial sector.

Identifying patterns in alarm logs has been researched before with the use of event correlation analysis[88][121]. With such a method it is possible to find patterns, but nothing more. These patterns can help the company redesign its alarm management system according to the findings. Prediction or detection is not covered by such methods. Another problem with these methods is that it uses time to analyze consequential alarms. It is not possible to use time as a variable in this project, because it cannot be assumed that the signal time stamps are correct. It could be

possible that signals are received in the wrong chronological order, which might be a problem for such methods.

Other methods that are proposed to find patterns use similar alarm sequence alignment algorithms[6][25][60][67]. These methods can find patterns in the historical database and use these patterns used to detect outages early. Such methods could be useful, but need a chronological alarm log. As mentioned earlier, it is not possible to make this assumption. One specifically interesting paper on finding consequential alarm sequences uses association rule learning[120], which is the same method that will be used in this project. It is tested on a chemical plant system and it uses a fuzzy logic variant of association rule learning, but the principle remains the same.

There has also been work on a method for automated grouping of alarms[107] to reduce the amount of alarms. This method also uses the topology of the process and the alarm logs instead of sensor values, which makes this research very relevant to our project. Although it is very relevant, automated grouping does not imply that the system is able to draw useful conclusions from the data, which is what this project hopes to do.

Identifying the root cause of the problem is also relevant for this project. For instance, when there is a short circuit in the electrical grid, all the succeeding and proceeding assets will be affected and send signals to the Network Operation Centre. This does not mean that there is a problem in all these assets. The operator will approximately identify the root cause based on all the alarms. This consequential alarm analysis is done in many continuous processes that have sequences of sensors, like industrial processes and the electricity sector. One method uses a causal dependency graph and the sensor values to determine the root causes[28], but this is too complex for this project. The sensor values are not available and monitoring the medium voltage network is a completely different problem than monitoring a relatively small chemical process.

2.3 Graph theory

The Dutch electrical grid can be seen as one huge graph, where the medium voltage network is a subgraph of this graph. The graph of the medium voltage network is relevant for this project, because the topology can be used to divide the signals by medium voltage ring. The discussion in this section is based on several sources on graph theory [12][18][51][100]. The relevant information in this section is acquired from these sources.

A graph G is a mathematical structure consisting of two sets, namely the set of vertices $V(G) = \{v_1, v_2, \ldots, v_n\}$ and the set of edges $E(G) = \{e_1, e_2, \ldots, e_m\}$. Every edge e_i is a two-element subset of V, thus $e_i = \{v_j, v_k\}$. An edge is generally written as $v_j v_k$ instead of $\{v_j, v_k\}$. The electrical grid can be structured as a graph, where the assets are the vertices and the edges represent the connection between assets. This way it is possible to determine connectivity and other properties using graph theory.

A vertex v_j is called incident with an edge e_i when $v_j \in e_i$. Likewise, e_i is incident with v_j in this case. Two vertices are called adjacent when they are incident with a common edge. Similarly, two edges are called adjacent when they are incident with a common vertex.

A graph H with vertices $V(H)$ and edges $E(H)$ is a subgraph of G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. If H is a subgraph of G, G is called the supergraph of H. An example of a subgraph is shown in Figure 2.6. Note that if I is a subgraph of H and H is a subgraph of G, then I is a subgraph of G. As mentioned earlier, the medium voltage network can be seen as a subgraph of the electrical grid. Likewise a medium voltage ring can be seen as a subgraph of the medium voltage network and thus also a subgraph of the electrical grid.

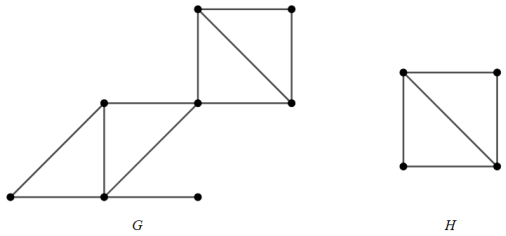

Fig. 2.6: An example where H is the subgraph of G.

If there is a path between every pair of vertices $v_i, v_j \in V$, then this graph is called a connected graph. Otherwise, the graph is called a disconnected graph. Connectivity is relevant to this project, because every medium voltage ring needs to be connected to be fully functional. When a medium voltage ring is disconnected, the part where there is no path to the source of power supply is affected.

Deletion of a vertex v or edge e from graph G results in a subgraph that is denoted as $G - v$ or $G - e$, respectively. An example of vertex and edge deletion is shown in Figure 2.7. Note that deletion of a vertex implies the deletion of all incident edges, but deletion of an edge does not imply the deletion of the incident vertices. A so-called cut vertex is a vertex in a connected graph G where its deletion results in a disconnected graph G with multiple components. In a similar way, a cut edge is an

edge in a connected graph G when its deletion results in a disconnected graph G with multiple components. An example of a cut vertex and a cut edge is also shown in Figure 2.7.

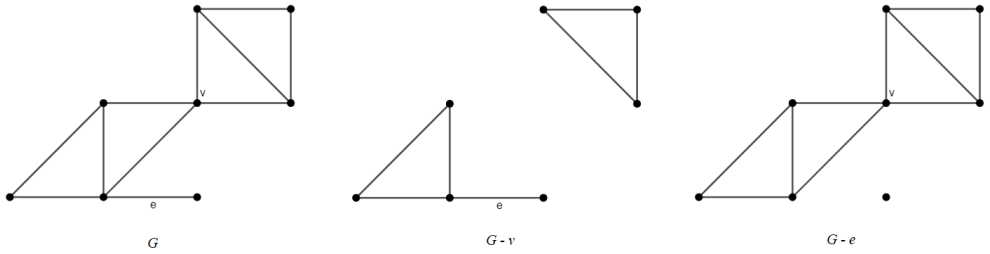

Fig. 2.7: Vertex deletion and edge deletion from a graph G. The deleted vertex is a cut vertex and the deleted edge is a cut edge.

Deletion is relevant in this project to detemine the medium voltage ring through connectivity. When deleting a cut vertex or cut edge, the graph of a medium voltage ring will be disconnected. As mentioned earlier, the component(s) of the disconnected graph where there is no path to the source of power supply are affected by the outage.

A $v_i - v_j$ walk is defined as an alternating sequence of vertices and edges from v_i to v_j. When no vertex is repeated in a walk, it is called a path. A path is closed if it begins and ends at the same vertex. A closed path is also called a k-cycle, where k represents the number of edges in the cycle. An example of a 3-cycle is a triangle. A tree is defined as a connected graph with no cycles. A property of a tree is that all edges are cut edges.

As mentioned in Section 2.1, there can be parallel connections in the transport part of the medium voltage network. These can be seen as k-cycles. The distribution part of the medium voltage network is often structured as a tree. Because all edges of a tree are cut edges, an outage in the distribution part of the medium voltage network will directly affect the disconnected components. In short, the medium voltage network can be structured as a graph of cycles and trees.

2.4 Association rule learning

Association rule learning is a machine learning method that finds patterns and correlations between different variables[20]. The field of machine learning focusses on creating algorithms that learn to how to perform a certain task without being

explicitly programmed for this task[83][32][98]. A machine learning algorithm will learn from examples, historical data or using real time data.

The process of finding the association rules consists of two steps[4][3]. First, the frequent item sets are determined. With these frequent item sets, the relevant association rules can be constructed. Historically association rule learning precedes frequent item set mining[2], but it should be seen as an extension of frequent item set mining[15].

This chapter will first discuss frequent item set mining, followed by a section on association rules. Two frequent item set mining algorithms will be discussed after this, namely Apriori and FP-Growth. Associative classification mining will follow after FP-Growth, which is a approach that builds a classifier using association rules. After this, a small section will discuss the possibilities of prediction using association rules. To conclude this chapter on association rule learning, some alternative methods will be discussed that could be useful, but are beyond the scope of this project.

2.4.1 Frequent item set mining

Frequent item set mining has its origin in the retail sector. It was used to analyze market baskets to find patterns in what costumers buy[2][20][53]. This way a company could increase their sales by adjusting their shop accordingly. An example of a frequent item set is that a costumer is likely to buy the set of items $\{bread, milk, butter\}$ together.

This method can be applied to many other problems, like predicting the risk of heart diseases[61]. As mentioned in Section 2.2, it has also been used for alarm analysis[120]. Frequent item set mining is very relevant for this project, because Stedin is dealing with a huge amount of signals that are received every day. These signals can be seen as items and patterns need to be found between these items.

First a formal definition of frequent item set mining is given. The set $B = \{i_1, i_2, \ldots, i_n\}$ is called the item base, where every transaction t_k is a set of items from B, thus $t_k \subseteq B$. Every transaction has its own transaction identifier (TID). The database is a list of transactions $T = [t_1, t_2, \ldots, t_m]$. Note that the database should not be represented as a set, because the same transaction can occur multiple times.

The cover of an item set I is defined as $K_T(I) = \{k \in \{1, 2, \ldots, m\} \mid I \subseteq t_k\}$ and represents the set of transaction identifiers that the item set I is contained in. Note

that the cover of $X \cup Y$ can be determined by intersecting the two covers, thus $K_T(X \cup Y) = K_T(X) \cap K_T(Y)$[43]. The support of an item set I is defined as $s_T(I) = |K_T(I)|$ and is basically the number of transactions that contain I. An item set I is called frequent if $s_T(I) \geq s_{min}$, where s_{min} is the predefined minimum support.

Frequent item set mining is a method that basically counts the number of occurrences of an item set I in the database T. Because this takes a lot of processing time, a lot of algorithms have been proposed to deal efficiently with large databases[118]. An example of a database and its frequent item sets is given in Figure 2.8. It is also possible to use the relative frequency, $\sigma_T(I) = s_T(I)/m$. In this case an item set I is called frequent if $\sigma_T(I) \geq \sigma_{min}$. The absolute support is used for now, because this can be computed from the relative support.

(a)

Transactions

0: {a, d, e}
1: {b, c, d}
2: {a, c, e}
3: {a, c, d, e}
4: {a, e}
5: {a, c, d}
6: {b, c}
7: {a, c, d, e}
8: {b, c, e}
9: {a, d, e}

(b)

Frequent item sets (with support)
(minimum support: $s_{\text{min}} = 3$)

0 items	1 item	2 items	3 items
∅: 10	{a}: 7	{a, c}: 4	{a, c, d}: 3
	{b}: 3	{a, d}: 5	{a, c, e}: 3
	{c}: 7	{a, e}: 6	{a, d, e}: 4
	{d}: 6	{b, c}: 3	
	{e}: 7	{c, d}: 4	
		{c, e}: 4	
		{d, e}: 4	

Fig. 2.8: An example of a database, together with the frequent item sets with minimum support 3 in the database.[15]

There are some nice properties that can be exploited to make the search for frequent item sets more efficient[15]. The support of an item set is so-called antimonotone, $\forall I \subseteq J \subseteq B : s_T(I) \geq s_T(J)$. This property together with the definition of the minimum support results in the so-called Apriori property[4][3], $\forall I \subseteq J \subseteq B : s_T(I) < s_{min} \implies s_T(J) < s_{min}$. The Apriori property says that the extension of an infrequent item set cannot be frequent. From this property follows that the set $\mathcal{F}_T(s_{min})$ of frequent item sets is downward closed, which is defined as $\forall I \in \mathcal{F}_T(s_{min}) : J \subseteq I \implies J \in \mathcal{F}_T(s_{min})$.

There have been many proposed frequent item set mining methods, like Apriori and FP-Growth. Apriori is mainly of historical value, because it is one of the first frequent item set mining algorithms and cannot compete with more recent ones. Because of its historical value, it will be discussed briefly in the following section. The algorithm that is used in this project will be discussed after that, namely FP-Growth[56]. There have been other proposed algorithms for mining frequent item sets with historical

value, like Partition[106] or Eclat[129], but these will not be discussed in this thesis because none of these algorithms is significantly better than the other[58][45].

Other frequent item set minings methods will be mentioned briefly. TreeProjection[1], which is a predecessor of FP-Growth and uses a similar tree structure. OpportuneProject[77] chooses between a similar tree representation or an array representation for the conditional databases. Medic[44], which is a variation of Eclat that does not need to maintain all covers and is memory efficient. H-mine[92][91] is an algorithm that is similar to FP-Growth but uses a different data structure called H-struct. Unlike FP-Growth, H-mine does not generate the conditional databases, which reduces the memory usage.

2.4.2 Association rules

An association rule $X \rightarrow Y$ means that a transaction that contains X, is likely to also contain Y, where X and Y are both item sets. Looking at the shopping example, $\{bread, milk\} \rightarrow \{butter\}$ might be an association rule that can be found. The support of an association rule is defined in the same way as for frequent item sets, thus $s_T(X \rightarrow Y) = s_T(X \cup Y)$. Let $X \cup Y$ be a frequent item set, the notion of confidence is defined as $c_T(X \rightarrow Y) = s_T(X \cup Y)/s_T(X)$. This represents the ratio where the item sets that contain X also contain Y. According to the predefined minimum confidence c_{min}, an association rule is accepted if $c_T(X \rightarrow Y) \geq c_{min}$.

When looking at alarm management, it would be very useful to find the association rules with an outage as consequent. This is possible by combining the signal log with the outage log and could give very interesting results. For instance, association rules like $I \rightarrow f$ could be found, where I is a set of signals and f is an outage in the electrical grid. Using this technique, it might be possible to automatically detect outages. This would help the operators in the Network Operation Centre to react faster and reduce the impact of an outage.

2.4.3 Apriori

The Apriori algorithm[4][3] is a breadth-first search algorithm. Let L_k be the set of frequent item sets with cardinality k and let C_k be the set of candidate frequent item sets with cardinality k. Apriori starts with determining L_1, so the frequent item sets with only one item. The algorithm does this by simply counting the number of occurrences for each item $i \in B$. Then Apriori iteratively generates C_k from L_{k-1}

```
1  L_1 = {{i} | i ∈ B ∧ s_T({i}) ≥ s_min};
2  for k = 2; L_{k-1} ≠ ∅ ; k++ do
3  |  C_k = {p ∪ q | p, q ∈ L_{k-1} ∧ p[1] == q[1] ∧ ... ∧ p[k-2] ==
   |    q[k-2] ∧ p[k-1] ≠ q[k-1]};
4  |  C_k = C_k − {c ∈ C_k | {c' | c' ⊂ c ∧ |c'| == k-1}L_{k-1}}
5  |  forall t ∈ T do
6  |  |  C_t = {c | c ∈ C_k ∧ c ⊆ t};
7  |  |  forall c ∈ C_t do  s_T(c)++ ;
8  |  end
9  |  L_k = {c ∈ C_k | s_T(c) ≥ s_min};
10 end
11 return ⋃_k L_k
```

Fig. 2.9: Pseudocode of the Apriori algorithm.

by a join step and generates L_k from C_k with a prune step. The join step joins L_{k-1} with L_{k-1} and the prune step then prunes the infrequent candidates, by counting the occurrences in the database. When Apriori is done iterating, it will return the union of all L_k, which is the set of all frequent item sets $\mathcal{F}_T(s_{min})$. Figure 2.9 shows pseudocode of the Apriori algorithm according to the implementation of Christian Borgelt[14].

As mentioned earlier, Apriori is mainly of historical value. The algorithm has to iterate through the database multiple times. For instance, when the cardinality of the biggest frequent item set is K, the Apriori algorithm will have to iterate through the database K times. When the database gets large or the cardinality of the frequent item sets gets large, this can be very time consuming. Algorithms like FP-Growth are more efficient and perform better in this case[86]. There have been several versions and extensions of the Apriori algorithm that are more efficient, but these will not be discussed in this thesis.

2.4.4 FP-Growth

The state-of-the-art FP-Growth usually outperforms other frequent item set mining algorithms[15] and that is why it will be used in this project. There have been extensions that could possibly increase the efficiency, but these will not be used. The historical signal log will be very long, so repeatedly scanning the database will be too expensive. FP-Growth is a depth-first search algorithm and only needs to scan the database twice, which is ideal for this project.

The idea behind FP-Growth is to compress the database into a compact structure, called a Frequent Pattern Tree (FP-Tree). Literature on alarm analysis using association rule learning[120] states that it is difficult to apply FP-Growth to alarm analysis, but no evidence is given in the paper. There are also no further references that mention this issue of FP-Growth. This is why this statement is ignored and FP-Growth will be used.

The first step in FP-Growth is scanning the database. When scanning the database for the first time, it counts the number of occurrences of each item and returns a list of frequent items, similar to Apriori. The list of frequent items is then arranged according to frequency in descending order. This ordering is used for constructing the FP-Tree, because this gives the best results related to the size of the FP-Tree[57].

The second step is to create a tree with an empty root. FP-Growth now starts scanning the database for a second time. For each transaction, the ordered list of its frequent items is constructed. An example of the ordered transaction database is shown in Figure 2.10. The algorithm then starts constructing the tree using the ordered transactions. The transactions with a common prefix, share a common part in the FP-Tree. The count of each item set is captured in the tree by a counter. The earlier shown sample database is worked out into an FP-Tree in Figure 2.10. It is fairly easy to determine the support of an item i by summing up the counters in the FP-Tree. The database is compressed into the FP-Tree and it is now possible to mine the tree for frequent item sets.

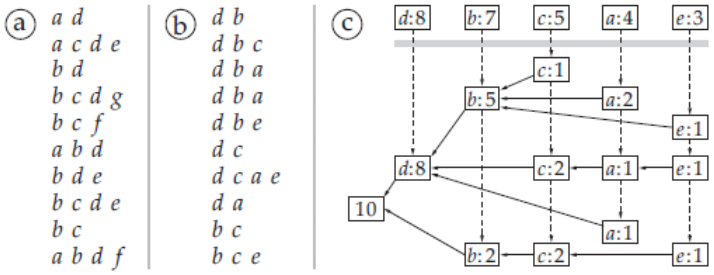

Fig. 2.10: An example of a database, together with the ordered transactions and the constructed FP-Tree.[15]

Mining the FP-Tree for frequent item sets is a recursive two-step process. First, the conditional database is determined and then the projected FP-Tree[13] is created. The reversed transaction ordering is used, so the algorithm starts with e in the case of Figure 2.10. For each frequent item i, FP-Growth starts a bottom-up approach and determines the conditional database. The conditional database T_I is defined as the list of transactions that contain i, but without I. In other words, $T_I = \{t - I \mid I \in t \wedge t \in T\}$, where $A - B$ is the relative complement of B in A and

```
1  FIS = [];
2  L_1 = {i ∈ B | s_T({i}) ≥ s_min};
3  FIS = L_1;
4  tree = createTree(T);
5  forall i ∈ L_1 do
6  |  T_i = determineConditionalDatabase(tree, i);
7  |  FIS_i = FP-Growth(T_i);
8  |  forall itemset ∈ FIS_i do FIS.append(itemset ∪ i) ;
9  end
10 return FIS;
```

Fig. 2.11: Pseudocode of the FP-Growth algorithm.

defined as $\{x \in A \mid x \notin B\}$[52]. For instance, the conditional database of $\{e\}$ is $\{\{d, b : 1\}, \{d, c, a : 1\}, \{b, c : 1\}\}$.

FP-Growth creates the conditional FP-Trees with these conditional databases. From the conditional FP-Trees, frequent item sets can be determined by summing up the counters of the items. Every item i in the conditional FP-Tree of item set I is frequent in the conditional database and it can be concluded that the set $I \cup \{i\}$ is frequent. This is repeated recursively until the conditional FP-Tree contains a single item. This recursive process determines all frequent item sets in an efficient way. To clarify how FP-Growth works, pseudocode is shown in Figure 2.11.

There have been various extensions on FP-Growth, which will be mentioned briefly. The method nonordfp[99] uses a more compact data structure and does not need to recursively build a projected tree for every conditional database. This method reduces the computation time and memory usage. FP-Growth*[49][50] is an improved method that uses an extra array to reduce one FP-Tree traversal. This improved version of FP-Growth performed really well at the FIMI Workshop competition[47][46]. Top-Down FP-Growth[122] is another proposed version that uses a top-down approach, as the name suggests. This way the algorithm does not have to generate the projected FP-Trees. Another alternative version of FP-Growth uses so-called FP-Bonsai[11][13], which is a tiny tree generated by pruning the FP-Tree. Because FP-Growth generates a lot of conditional databases and projected FP-Trees, this method reduces the memory usage of FP-Growth. AFOPT[75] is another alternative FP-Growth version that uses three different structures to represent the conditional databases, namely arrays, AFOPT-Trees[74] and buckets. Research shows that AFOPT reduces the memory usage a lot and outperforms other version of FP-Growth on both computation time and memory usage[105]. There is also a method that

uses Patricia tries, called PatriciaMine[96]. Patricia tries compress the data more efficiently than FP-Growth, which results in less memory usage.

There are also more recent FP-Growth variants from the last decade. One of those variants uses compressed FP-Trees and compressed arrays[108]. An even more recent version of FP-Growth only needs to scan the database once and it generates the frequent item sets without generating conditional FP-Trees[87].

To summarize, there are a lot of frequent item set mining algorithms and there are even more variants of algorithms. Different implementations can lead to significant differences in performance[45]. For example, research[131] showed that different algorithms round the minimum support differently, which results in an unequal amount of frequent item sets between the algorithms. All in all, it is impossible to conclude from the literature which version is the best. For now the original FP-Growth algorithm will suffice and will be used for this project. When problems regarding memory usage or computation time are encountered, an appropriate version can chosen to solve this.

2.4.5 Associative classification mining

Associative classification mining is the data mining technique that uses a small set of association rules to accurately classify unseen cases. This approach wants to find association rules with item sets as antecedent and classes as consequent. With these so-called class association rules (CARs), a classifier can be built. In this project, identifying patterns between signals is useful, but detecting outages using association rules will be the final goal. Therefore a classifier based on association rules is built and used to detect these outages.

The set of class association rules is generated from a similar transaction database, but every transaction is assigned to a class c. A ruleitem is defined as $condset \rightarrow c$, where $condset$ is an itemset and c is the related class[116]. The ruleitems that are frequent and satisfy the minimum confidence requirement form the set of class association rules. There are several different methods to generate these class association rules and build a classifier, like CBA[80], CMAR[71], CPAR[126] and MCAR[117]. For simplicity reasons, this project uses parts of the classification based on association rules (CBA) method. The classification problem is binary and it is therefore not necessary to look at more sophisticated approaches.

Generating the set of class assocation rules is done by taking a subset of the complete set of assocation rules found by FP-Growth. The ruleitems are those rules that have

an outage as consequent. This way it is still possible to generate rules between signals, but it is also possible to generate the ruleitems.

According to CBA, a classifier is build by pruning these and then sorting the ruleitems. Pruning is used to remove redundant rules or incorrect rules, which is optional[80]. Because binary classification is fairly simple and computing time is not an issue, there is no need for pruning. Sorting is used to choose the best rule for classifying an itemset. Sorting is irrelevant, because there are only rules for one class, namely outage. The other class is defined as the default class and there are no rules for this class.

When predicting an unseen case I, the classifier tries to find a class association rule $I' \rightarrow c$, where $I' \subseteq I$. Originally, the rule with the highest confidence is applied. In the case of this project, there are only rules for one class. This is the reason why the rules do not need to be sorted and it is not neccesary to find the rule with the highest confidence. In short, if an outage association rule exists that can be applied to the unseen case, it is classified as an outage. Otherwise, it is classified as the default class.

It is also possible to use negative association rules for more accurate classification[115][8]. This can generate more specific rules that can help classify unseen cases more accurately. For example, when there is a classifier with the ruleitems $A \rightarrow c_1$, $A \rightarrow c_2$ and $B \rightarrow c_2$ and case A needs to be classified, no decision can be made according to the current classifier. If negative rules are added, it might be possible to find the ruleitem $\neg B \rightarrow c_1$. With this ruleitem added to the classifier, it is now possible to classify case A as c_1.

While these negative rules might help choosing between different classes, sorting the rules as mentioned earlier can also help choosing between classes. Although the paper suggests it improves the classifier, this approach will not be used in this project. The concept of negative rules is still interesting, because this project has a binary classification problem where there needs to be chosen between $outage$ and $\neg outage$. This makes it interesting to look at the negative rules that have the class $\neg outage$ as consequent. This might reduce the amount of false positives.

2.4.6 Prediction using association rules

Prediction is something that is very interesting for Stedin. If an outage could be predicted some time before it occurs, Stedin could anticipate. The approach of prediction using association rules will be disccused briefly in this section, but it will

not be used in this project because of low data quality and too few outages to work with.

An interesting concept is finding so-called inter-transaction association rules. The previous discussed methods aim to find intra-transaction association rules, which are the patterns that occur frequently in a transaction. The example $\{bread, milk\} \rightarrow \{butter\}$ is an intra-transaction association rule. An inter-transaction association rule is a rule with items from different transactions. An example of an inter-transaction rule is $\{alarm_1, alarm_2\} \rightarrow \{an\ outage\ in\ 15\ minutes\}$. Several methods have been proposed for finding these inter-transaction rules, like E-Apriori and EH-Apriori[79], FITI[119], ITP-miner[68], or other algorithms[38][78][37][10][70].

Prediction using association rules has also been done in an approach called prediction mining[30]. It uses the antecedent to predict the consequent, which is exactly what is desired in the alarm management system of Stedin. While the method is very interesting, it uses some properties that can not be assumed in this project. For instance, it assumes that there is a time lag between the antecedent and the consequent. In Chapter 4 it will become clear that this assumption cannot be made in this project.

2.4.7 Alternative methods

There are a few methods that were considered, but seemed to complex for the scope of the project. These will be mentioned briefly in this section. In future research, it might be useful to look into these alternative methods.

One option is to use sequential pattern mining, which was introduced not long after frequent item set mining[5]. This method can for instance be used for web mining by analysing click streams[130][73]. Finding sequential patterns can be very useful in this project as well, considering that a certain sequence of signals can indicate a problem in the electricity network. The problem with sequential pattern mining is that it needs a chronological ordered database, which cannot be guaranteed in this project. This is why sequential pattern mining is not used.

When this assumption can be made in the future, this method will be very interesting to look into. Possibly useful proposed algorithms are GSP[111], SPIRIT[41], FreeSpan[55], SPADE[128], PrefixSpan[94][93] and MEMISP[72].

Another option is to use frequent item set mining algorithms for data streams, which is the problem of finding all frequent item sets in a time window $[t_i, t_j]$[69]. A

historical data set of one and a half year is used in this project, instead of streaming data. Broadening the scope of the project to streaming data will give new challenges. Time-related patterns, like seasonal patterns, will be very interesting. For now, this is beyond the scope of the project.

There have been several proposed methods that use frequent item set mining on data streams. Some of those are simple counting algorithms[82][65], but there are also methods that put more emphasis on the recent data[21]. As mentioned earlier, FP-Growth will be used for this project. The data stream version of FP-Growth, called FP-stream[42], is therefore particularly interesting. Another very interesting method is the mining of sequential pattern mining on data streams, like SSM[36] or MILE[23][22]. While none of those algorithms will be used in this project, it might be interesting to look into in the future.

It is also possible to use a weighted frequent item set mining method. Such a method makes it possible to put weights on different items. This can be useful for this project, because not all signals are of the same importance. Several algorithms have been proposed based on the Apriori algorithm[19][123][114]. There is also a proposed algorithm which is based on FP-Growth, called WFIM[127].

Lastly, when sensor values are available in the future, it could be possible to use other classification models[39][54][63] to detect outages. Classification based on sensor values is not applicable right now, but it can be considered by Stedin for future improvements.

3 Methodology

This chapter will explain how the problem is addressed. Useful patterns need to be found in the signals that arrive at the Network Operation Centre and these patterns will be used for detecting outages. One option is to look at all signals at the same time, but this seems a bit odd. Different medium voltage rings do often not influence each other, thus the patterns that will be found are not always relevant in this case. The solution is to divide the historical database into different data sets, where each group represents a medium voltage ring. Let's focus on the signals of one medium voltage ring for now, which makes the problem a lot more manageable.

The first section of this chapter will discuss how the topology is incorporated using graph theory. The methodology for finding patterns will be discussed in the second section. To conclude this chapter, the last section will discuss how a classifier is built using these patterns.

3.1 Medium voltage network as a graph

As mentioned earlier in Section 2.1, the medium voltage electrical grid can be structured as a graph. The vertices represent the different assets in the network, like cables, switches, breakers, fuses, transformers, busbars or other assets. The edges represent the connections between different assets of the network, which defines the topology of the network. Some of the assets on the vertices have telemetry and are able to send signals to the Network Operation Centre. In short, the medium voltage electrical grid is a concatenation of edges and vertices, where some of the vertices have the ability to send signals.

The medium voltage rings can be determined from the complete graph of the medium voltage network. This is done by using the connectivity concept of graph theory. A medium voltage ring always starts with a feeder that is coming from the busbar of a substation or switching station. The ring consists of the feeder coming from the busbar and everything that is connected with the feeder from there on. The results is a subgraph that represent the medium voltage ring. The signal and outage

data can be divided with the used of these subgraphs, which makes it possible to distinguish between the different medium voltage rings.

The topology of a medium voltage ring might also be interesting to look into. In that case, the relative position in the graph becomes important. The reason behind this will be explained in the following section, but first some definitions have to be given. When looking from the perspective of a vertex, the preceding vertices are those that are closer to the source of power supply (thus closer to the feeder). On the other hand, vertices are considered succeeding when they are further away from the source of power supply. The succeeding vertices with telemetry will be denoted as $successor$. Note that distance does not make a difference. If v_1 has a successor v_2 and v_2 has a successor v_3, then v_3 is also a successor of n_1. Similarly, the predecessor of a vertex is denoted as $predecessor$. The vertex itself is denoted as $identity$. An example of the relative position is shown in Figure 3.1.

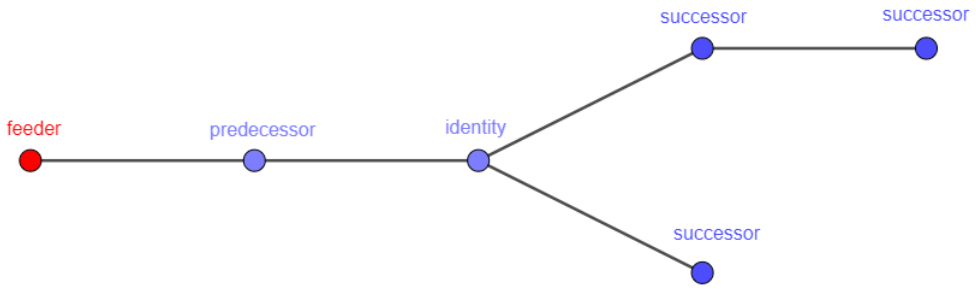

Fig. 3.1: An example of the relative positions in a medium voltage ring.

When an outage is determined according to the signals, the affected part of the graph can be determined using connectivity. As mentioned earlier, the vertex with the outage can be deleted using vertex deletion. The disconnected component(s) that are no longer connected to the source of power supply is affected. This is not in the scope of this project, but might be interesting for the operators.

3.2 Finding patterns

The first step in this project is to find patterns. The signal log and outage log are merged into one database. This database will be transformed to a transaction database using mean-shift clustering. How the transaction database is generated will discussed here. This section will also explain how the frequent item sets and the association rules are found.

3.2.1 Generating transaction databases

The used database consists of one long historical log, which is combination of signals and outages. As mentioned in the previous section, this database is then divided into different data sets using the topology. Every data set represents the historical log of a medium voltage ring.

From these different data sets, transactions can be generated that can be used for frequent item set mining. This can be done in several ways. One option is to use a sliding window, which is used in most of the literature. There are several interesting options, like a landmark window or a sliding window[64]. A landmark window is a window that has a starting time point i and an ending time point t, where i is static and t is the current time. A sliding window has a time span $w \in \mathbb{R}^+$ and all items inside the window $[t - w + 1, t]$ are considered, where t is the current time. Care has to be taken when specifying the time span w. Some signal information is possibly lost when using a small sliding window, where a large sliding window will result in the likelihood that external factors will affect the signal information[120].

The problem with a sliding window is that the duration of a signal sequence can vary a lot. Sometimes several signals arrive in one minute, while other times related signals arrive for the duration of 15 minutes. This is very inconsistent and therefore there is no correct way to specify the time span w.

Instead of using a sliding window, the events will be clustered according to their time. The idea of clustering time-series data into transactions is not new, it has been used before to discover rules from time-series[29]. Which method should be used is not clear, because there is no best algorithm[85]. The choice should be made on basis of the data set. A one-dimensional clustering algorithm is needed, because there needs to be clustered according to the time. In this project the operators can validate if the clusters are correct, which is an important element when choosing the most appropriate clustering approach[9]. A method is chosen according to the data and the clustering results and these results are then validated by operators.

A method like k-means[81] could be used, but the problem is that the number of clusters (k) needs to be specified. k is not known beforehand thus k-means cannot be used for this project.

Another option is to use DBSCAN[35], where there is no need to specify the number of clusters beforehand. The problem with DBSCAN is that it ignores outliers. While this might be preferable in some cases, in this project the outliers must be treated as

separate clusters. A single signal might contain enough information to imply there is an outage. Therefore DBSCAN is not used.

Some other clustering methods were tested, but one method performed the best according to the validation of the clusters, which was mean-shift clustering[40][24][31]. It was verified by a NOC operator that these clusters were correct. Each cluster forms a transaction, which results in a transaction database for each medium voltage ring. Merging all the different transaction databases results in the used transaction database for the first experiment. Note that no distinction is made between the transactions of different medium voltage rings. The reason behind this is the quantity of the data. There are too few outages to distinguish between patterns of different medium voltage rings.

3.2.2 Adding relative graph positions

The relative graph positions could possibly help locate the source of the problem and improve the accuracy of the model. Outages always occur on breakers or fuses, but there can be multiple breakers and fuses in a medium voltage ring. This means that determining the affected medium voltage ring is not the final step in locating the outage. Generating databases from the perspective of breakers and fuses might be the solution. For instance, if an outage occurs, the succeeding and preceding assets will be affected and might sent signals. Finding patterns in the signals relative to the outage might therefore be interesting. An example of a useful association rule is $\{s(A_1), p(A_2)\} \rightarrow outage$, where $s(A_1)$ is a signal A_1 coming from a successor asset and $p(A_2)$ is a signal A_2 coming from a predecessor asset.

The way the transaction database is generated is slightly different. Instead of having a data set per medium voltage ring, a data set is created per breaker or fuse. The event of a signal in a medium voltage ring is sent to all relevant breakers and fuses with a relative graph position label. These events with labels form a new data set per breaker or fuse. Note that only signals are labelled and sent to breakers and fuses. Outage events only occur in the database of the respective breaker or fuse.

The events are transformed as follows. Suppose an event e arrives from an asset in a medium voltage ring, then all succeeding breakers and fuses will receive it as $p(e)$. This indicates that an event e occurred at a preceding asset. Similarly, all preceding breakers and fuses will receive it as $s(e)$, indicating an event e occurred at a succeeding asset. When an event occurs exactly at a breaker or fuse, it will be send to the respective database as $i(e)$. This way the relative topology gets incorporated into the signals.

From the created asset databases, it is possible to generate transactions using clustering. Again, merging all the transaction data sets results in a transaction database. This transaction database will be used for the second experiment, which desires to automatically pinpoint the outage and improve the accuracy of the classifier.

3.2.3 Association rules

Finding the frequent item sets is done using the FP-Growth algorithm, which has been explained extensively in Section 2.4. Given a user specified minimum support, the algorithm finds the frequent item sets in the transaction database. These frequent item sets will then be used to construct the association rules as described in Section 2.4. The user specified minimum confidence determines the accepted association rules.

Rules between different signals are interesting and will be presented in the results. These rules are not relevant for this project, but are very interesting for the NOC operators. Analyzing the signals this way, it is possible to find redundant signals or even noise signals. This can be insightful and might help improve alarm management at Stedin.

The most important rules are those that have an outage as consequent. These rules are also insightful for the operators. They can verify that the algorithm works properly and the patterns are indeed related to outages, but it could also identify patterns that are unknown. The association rules related to outages will be used to build a classifier, which will be discussed in the next section.

3.3 Building a classifier

With the outage related association rules, a classifier can be build to detect the outages. This will be done according to Section 2.4. All rules will be used, because there is no reason for pruning. When an unseen case arrives, the classifier tries to find a rule that can be applied to the unseen case. If an outage association rule can be applied, the unseen case is classified as an outage. Otherwise if no rule can be applied, the transaction will be classified as the default class which means there is no outage. When a transaction is classified as an outage, the respective medium voltage ring is automatically known.

The experiment where the relative graph positions are incorporated are done in a similar way. When there exists a rule that is applicable to the unseen case, it is classified as an outage. Otherwise, it is classified as the default class and there is no outage. The breaker or fuse of the outage is automatically known when a transaction is classified as an outage.

3.3.1 Negative rules

There will also be an experiment with negative class association rules, like $I \rightarrow \neg outage$. Instead of looking at detecting outages, it might also be possible from a different perspective and try to detect the non-outages. Every transaction that is not an outage, is a non-outage. The negative class association rules are used to build a classifier. For simplicity reasons, no experiment with negative rules will be done for the database with relative graph positions.

It is not yet clear how the negative rules are incorporated. The optimal solution would be to combine the positive and negative rules, but there is no literature on how to sort these rules correctly. If the rules are sorted according to algorithms like CBA[80], the positive rules are overwhelmed by the negative rules and the probability that a positive rule will be applied is very low. This results in the fact that almost no outages will be predicted. Therefore sorting the rules was not tried, but a classifier is built using only negative class association rules. When no rule can be applied to the unseen case, it is classified as the default class, which means there is an outage.

4 Experimental setup and data

This chapter will discuss the data and the experimental setup. Data gathering and processing will be discussed in the first section. This section will also discuss the data quality, which had an impact on this project. The second section will explain the different experiments that will be conducted for this project.

4.1 Data gathering and processing

Three different sources of data were used. First of all, an export of the network topology was used in the form of XML files. Secondly, the signal log between April 2017 and October 2018 was used. Lastly, the outage log between January 2017 and October 2018 was used. Some other sources were used to link the signal and outage log to the topology, but these will not be discussed.

4.1.1 Topology

The topology of the medium voltage electrical grid is captured in XML files. There are XML objects that represent different assets of the network, like a cable, switch, breaker, fuse, transformer, busbar or other assets. All these assets together represent the topology of the electrical grid. There are also other XML objects that indicate that an asset has telemetry. This is used to identify what parts of the network have telemetry. These object are also used to link between the signals and the topology. Then there are also so-called vaults, which represent the RMUs in the network. This XML object captures which assets lie inside the Ring Main Unit.

The topology of the medium voltage network can be seen as one big graph. With the use of the XML objects in the XML files, a graph is constructed. All assets are intentionally placed on the vertices (even cables). The edges only have a connecting purpose. There can be multiple assets on the same vertex, which are added to the vertex as attributes. Each vertex also has a Boolean variable which represents if the asset has telemetry or not.

To start, the complete medium voltage part of the electrical grid is constructed from these XML files. This project focuses on the medium voltage rings and these are determined from the complete graph using the connectivity concept from graph theory. The medium voltage ring consists of the feeder coming from the busbar and everything that is connected with the feeder from there on. There were approximately 2,600 medium voltage rings at the moment of this project, where around 2,200 have telemetry. The medium voltage rings are grouped by substation area, which is determined by the busbar where the feeder is connected.

In Section 2.1 it became became clear that there is a transport part and a distribution part in the medium voltage network. The medium voltage rings represent the distribution part, which are easy to determine using connectivity. Regrettably, it is not possible to precisely determine the different transport parts from the data. There are multiple parallel connections, which complicates the task of dividing the data into relevant groups.

The topology of the network changes daily, which might influence the performance of the model. Medium voltage rings can change over time and an asset might currently lie in a different medium voltage ring than a year ago. This means that the signals that came from this asset a year ago, are placed in the wrong medium voltage ring database. In the optimal situation the graph is dynamic and should be adjusted accordingly every day, but this was not feasible within the time frame of this project. The changes in the system are not always logged. A static graph is therefore used, but it should be noted that this can influence the results.

The amount of telemetry can vary for each medium voltage ring. Some medium voltage rings have no telemetry at all. This inconsistency will influence the model. For instance, when there is a fault detector in a medium voltage ring while another medium voltage ring does not have a fault detector, it could be harder to predict outages with the amount of signals coming from the ring that lacks a fault detector.

Another data quality problem is that the data of the network might not be correct. Some cables were placed in the ground so many years ago, that there is no record of how they are connected. The data of these cables is based on their geographical position.

4.1.2 Signal log

The signal log is a table where each row is one signal that is logged. It consist of almost 10 million signals from April 2017 until October 2018. The signal log has

6 columns. The datetime column, category and text columns were used for this project. The datetime is in GMT to the exact second and nothing had to be adjusted. The category column was used for filtering, because some categories of signals were irrelevant for detecting outages. Therefore operator actions, system status updates and other irrelevant signals were filtered out. There was also a location column that could be used to link an alarm to a substation area, but this was not used in the project because it does not give a correct representation of the topology.

The text column is the most relevant column in the signal log. As the name suggest, it is a string and consists of relevant information. The device type, the device ID and the event are incorporated in this string. With the device type and the device ID, the signal can be located in the network. Sometimes the string was built up in a different way and it had to be located differently.

This project only uses the signals that can be located, the other signals are filtered out. From the 10 million signals, around 1.1 million alarms were located and used. The signal log is then divided by medium voltage ring. The resulting data set is a signal log for each medium voltage ring.

The standards for events has changed throughout the years, which results in inconsistency within the data. There are several signals that basically describe the same event. It is clear that this will influence the performance of frequent item set mining by a lot. Firstly, there are a lot more items and thus the computing time will increase. Secondly, there are less frequent item sets generated, because the events are distributed over several different signals. The solution for this might be decreasing the minimum support, but this might result in overfitting. At the start of this project, operators from Stedin were asked to group the different signals together by event. Regrettably, this could not be completed in the time frame of this project. Therefore the original signals were used for frequent item set mining instead of the events.

4.1.3 Outage log

The outage log is a table which is logged manually. The data quality was therefore not optimal. There were approximately 1,600 outages from December 2016 until October 2018. This means there are around 2-3 outages a day on average. In a similar way as with the signal log, some data processing needs to be done. Firstly, the datetime in the outage log was CET and this had to be converted to GMT. The datetime for the outage log was precise on the minute, but because it was logged manually.

Secondly, the outages had to be located in the network. This is where most of the data is lost. The location of the outage was only logged properly when a breaker or fuse switched off. In the case of a breaker not switching off during an outage, the location was logged in a different way. Regrettably, this data was too inconsistent and messy to work with. In short, only the outages were used where a breaker or fuse was switched off and this breaker or fuse could be located in the network. The outages were removed where there was no telemetry in the medium voltage ring. This because there will be no signal preceding or succeeding the outage and thus no patterns can be found. For similar reasons, outages were removed where no signals preceded or succeeded the outage. In this case it could be related to data quality. It is impossible to work with outages that are not detectable by signals and therefore these outages are omitted.

4.1.4 Final database

The processed signal log and the processed outage log are merged together to form the final database. The signals in the signal log are from between April 30 2017 until October 24 2018 and the outages in the outage log are from December 31 2016 until October 10 2018. Because the minimum and maximum datetime of the two different logs do not agree, the minimum datetime was set to the minimum datetime of the signal log and the maximum datetime was set to the maximum of the outage log. From this final database, the transactions will be generated as described in Chapter 3.

4.2 Experimental setup

There are three different experiments that are conducted for this project. This section will first discuss the experiment evaluation, which stays the same throughout the different experiments. The three experiments only differ in the rules that are used to build a classifier. The experiments will be discussed after the experiment evaluation.

4.2.1 Experiment evaluation

Evaluating the used method is a subject on its own. For this project, cross-validation will be used to evaluate the model, specifically k-fold cross-validation[39][63]. The

transaction database is the training data T and this is divided into k folds $T_1, \ldots, T_k$. Then for each fold T_i, the model is trained on the other $k-1$ folds and then tested on T_i. When testing, every transaction from the test set T_i is classified. For this project $k = 10$ will be used.

This project has a classification problem, namely whether there is an outage or not. The model needs to be optimized so that it detects most of the true positives, but gives as few false positives as possible. One possible evaluation method would be to use the error rate, which is defined by the formula $CV = \frac{1}{n}\sum_{i=1}^{n} Err_i$[63]. Here $Err_i = I(y_i \neq \hat{y}_i)$ represents whether the predicted outcome $\hat{y}_i$ is equal to the actual outcome y_i.

The problem with error rate is there a huge class imbalance in this project. There are only 357 transactions classified as outages of the total 152,431 transactions. Therefore, precision, recall and f-measure[97] will be used as measures for evaluating the method. The following definitions are given:

- A true positive (tp) is a correctly classified outage
- A false negative (fn) is an incorrectly classified outage
- A false positive (fp) is an incorrectly classified non-outage
- A true negative (tn) is a correctly classified non-outage.

The amount of all instances is summed up over the 10 different folds. The summed up totals are used to calculate the precision, recall and f-measure. The definitions for these measure are as follows:

- precision = $\frac{tp}{tp+fp}$
- recall = $\frac{tp}{tp+fn}$
- f-measure = $2 * \frac{precision*recall}{precision+recall}$

For each measure, the optimal value is 1 and it is desired that the model would approach 1 for these measures. A high value of precision means there are less false positives in comparison to the true positives. Similarly a high value of recall implies there are less false negatives in comparison to the true positives.

In this project, the recall measure would be more important because it is not desired to miss an outage. The precision measure is also very relevant though, because this project tries to reduce the amount of false alarms. The f-measure combines the two measures in one, which can assist in finding the optimal model.

4.2.2 Experiment using outage related association rules

For this experiment, the optimal classifier needs to be determined and evaluated. An absolute minimum support of 15 was used, which is a relative support of 0.0001. This was not changed throughout the experiment. The minimum confidence will be varied throughout the experiment from 0 to 1 with intervals of 0.01. The precision, recall and f-measure will be computed for each minimum confidence. A graph can be plotted against the minimum confidence and the optimal confidence threshold can be determined from these results.

4.2.3 Experiment using negative association rules

This experiment is done in a similar way as the previous one. An absolute minimum support of 15 will again be used to have a fair comparison between the two experiments. Also the confidence threshold will be varied throughout the experiment from 0 to 1 with intervals of 0.01.

4.2.4 Experiment using relative graph positions

As described in the methodology chapter, the data needs to be transformed differently for the second experiment. Every signal that could be located is send to the relevant breakers with a relative position label. Every breaker has its own database and similarly those databases are clustered according to time. From these clusters, a transaction database is generated.

With this transaction database, the experiment is very similar to the last two experiments. The classifier tries to correctly classify the transactions as outages or non-outages. The same absolute minimum support of 15 will be used, together with the varying confidence from 0 to 1.

5 Results

This chapter will discuss the results of the experiments in this project. First, some rules between signals will be presented. Those rules have high support and high confidence and are therefore relevant for improving alarm management at Stedin. Note that these rules are examples and it is not representative for the management in the NOC. After this, the results from detecting outages are presented. The results of the experiment with negative rules follows. Lastly, this chapter will present and discuss the results of the experiment with the relative graph position labels.

5.1 Rule analysis

This section will highlight some patterns that were found between signals. Also some interesting findings related to outages will be highlighted. This section might not be relevant for the project, but it is insightful for Stedin to see some of the patterns. Redundant signals can be found by analyzing the signal log this way.

The following rules are rules between signals. Only simple rules are shown to keep the patterns clear, but a lot more patterns were found during this analysis. Some of these patterns are already known by Stedin, but there are also some patterns that are new and insightful. The minimum relative support was set to 0.001 (absolute support of 152) and the minimum confidence was set to 0.99. The following rules were found.

{'IED BED_BEV COMM GESTOORD'} $\rightarrow$ 'IED BED_BEV COMM PARAAT'

{'MAXIMAAL BEV I> TRIP UITKOMM'} $\rightarrow$ 'MAXIMAAL BEV I> TRIP NORMAAL'

{'MAXIMAAL BEV I» TRIP UITKOMM'} $\rightarrow$ 'MAXIMAAL BEV I» TRIP NORMAAL'

{'MAX BEV I» AANGESPR'} $\rightarrow$ 'MAX BEV I» HERSTELD'

{'BEVEILIGING RELAIS AANGESPR'} $\rightarrow$ 'BEVEILIGING RELAIS HERSTELD'

These rules show that certain events are triggered and then a lot of the times return to normal shortly after. This has probably something to do with the rule based

system, maybe the thresholds need to be adjusted. This is something for Stedin to further look into.

It is also useful to look at the association rules related to outages. The minimum support was set to 0.0001 (absolute support of 15), which will also be used in the experiments. The confidence threshold was set a bit lower, because most of the association rules did not have such a high confidence. Only two rules have a higher confidence than 0.9, which are the following.

{'STORINGSVERKLIKKER AANGESPR', 'VERMOGENSCHAKELAAR UIT', 'AARDFOUT AANGESPR'} $\rightarrow$ 'OUTAGE'

{'VERMOGENSCHAKELAAR UIT', 'STORINGSVERKLIKKER AANGESPR'} $\rightarrow$ 'OUTAGE'

The operators in the NOC verified that these rules are correct. Other interesting rules can be shown, but that will be beyond the scope of this thesis. It is clear that Stedin can learn a lot from association rule learning. For example, a very important signal 'SCHAKELSTAND UIT' has a very low confidence rule that implies an outage, namely 0.057 confidence. This means that most of the times, this signal does not imply an outage. In combination with other alarms, for example 'AARDFOUT AANGESPR', the confidence increases to 0.586.

The NOC operators of Stedin were surprised by the low confidence of 'SCHAKELSTAND UIT'. The signal 'AARDFOUT AANGESPR' has a higher confidence of 0.084. Stedin has prioritized 'SCHAKELSTAND UIT' above 'AARDFOUT AANGESPR' and from this analysis, it can be concluded that this might not be the correct prioritization.

5.2 Detecting outages

This section will discuss the most important experiment of the project, namely detecting outages using the signal log. Different classifiers are built and evaluated using the association rules related to outages found in the training set. Each classifier has a different minimum confidence threshold, as explained in Chapter 4. For every transaction in the test set, the model tries to predict the related class. The results of precision, recall and f-measure are shown in Figure 5.1. Figure 5.2 shows the number of true positives, false positives and false negatives. Because most of the cases are true negatives and are therefore not relevant, it is not shown in the figure.

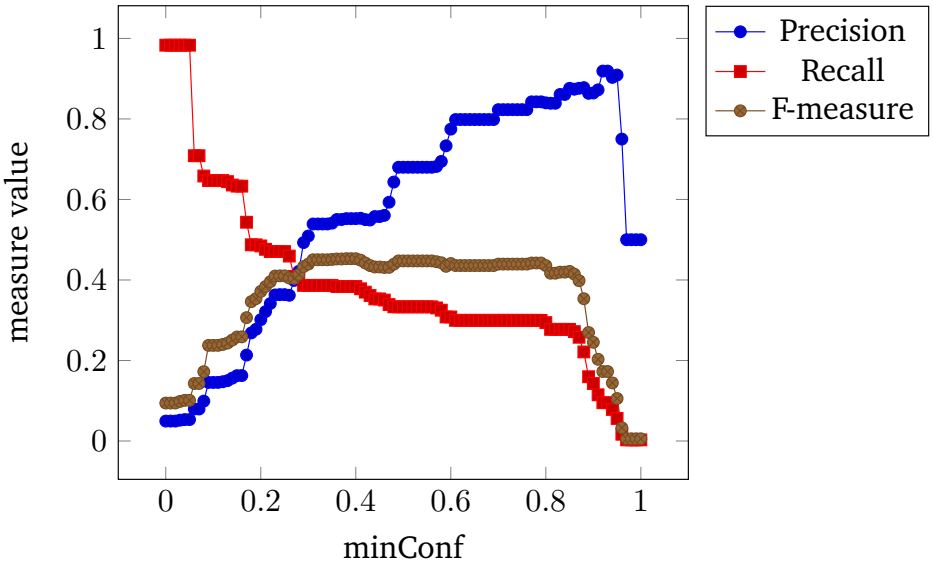

Fig. 5.1: The precision, recall and f-measure results of the first experiment.

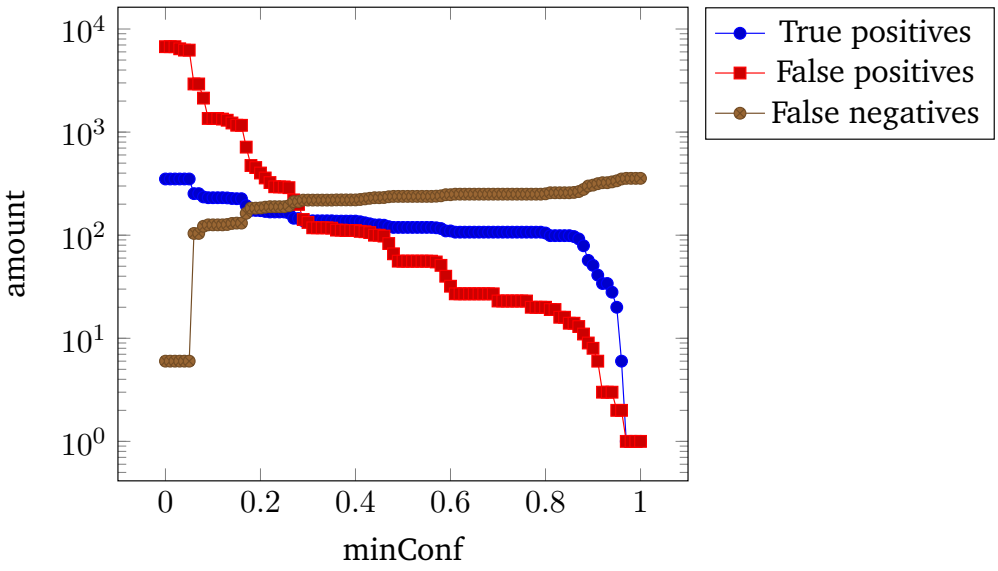

Fig. 5.2: The amount of true positives, false positives and false negatives of the first experiment.

Looking at the precision, recall and f-measure, It is clear that the model did not perform well. The maximum f-measure was recorded when the model was trained with a minimum confidence of 0.38, 0.39 or 0.40, which was 0.453. This is very low, an optimal model would approximate the f-measure of 1. This model has 111 true positives and 220 false negatives, which means it misses two thirds of the outages. This means that this model does not meet the expectations.

On the other hand, when look at an optimal recall, it can be seen that the model performs best with a minimum confidence of 0.04 or 0.05. The recall here is 0.983, which means that the model predicted almost all outages. From the 357 outages, only 6 are missed. Note that these false negative might also have to do with data quality, as described in Chapter 4. However, the precision is 0.050, which means 6,240 false positives. This can be seen as a bad performance in terms of these measures, but in terms of Stedin these results are great, which is a statement that is supported by the client. From all outages, 98.3% is detected and the amount of signals is reduced from 1.1 million signals or 150,000 transactions to only 6,600 possible outages. This is certainly an improvement for the Network Operation Centre. Instead of having multiple operators monitor 1.1 million signals, they would only have to check around 6,600 possible outages.

The great part about this model is that it works per medium-voltage ring. It can therefore instantly identify in which medium-voltage ring the outage occurred. A lot of time can be saved this way, because it can sometimes take up to 5 minutes to find the medium-voltage ring where the outage occurred.

5.3 Detecting outages using negative rules

For this experiment, a different perspective was used and it was tried to detect outages using negative class association rules. The negative rules were used to build different classifiers. The confidence threshold is varied to find the optimal model. The results of precision, recall and f-measure are shown in Figure 5.3. Figure 5.4 shows the number of true positives, false positives and false negatives.

Evidently this model performed worse than the classifier from the previous experiment. The optimal f-measure of 0.180 was recorded with a minimum confidence 0.95. Note that the confidence threshold is very high in this case, because a higher minimum confidence results in less negative rules and thus more cases will be classified as outage. Only 260 of the 357 outages were predicted, which is not desirable. In the optimal situation, all outages are predicted.

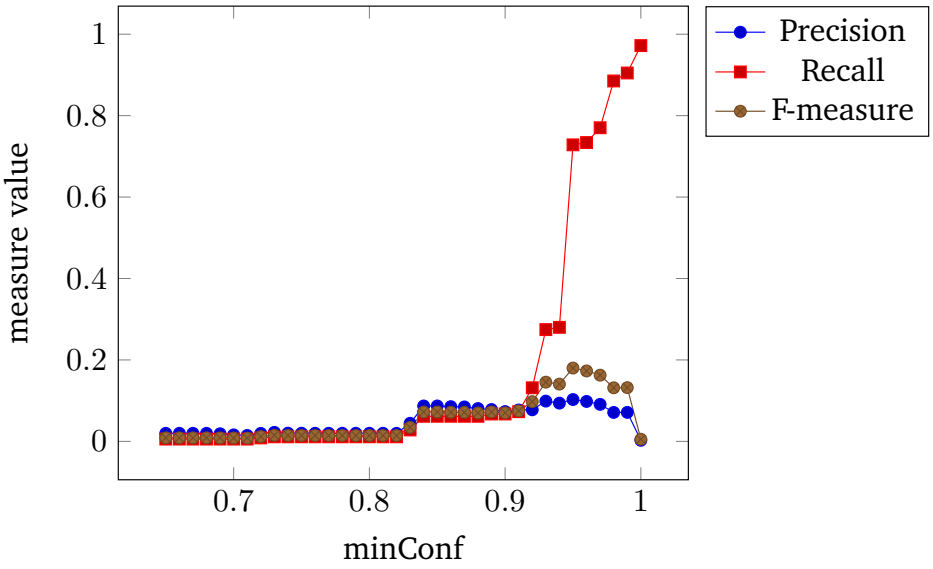

Fig. 5.3: The precision, recall and f-measure results of the second experiment.

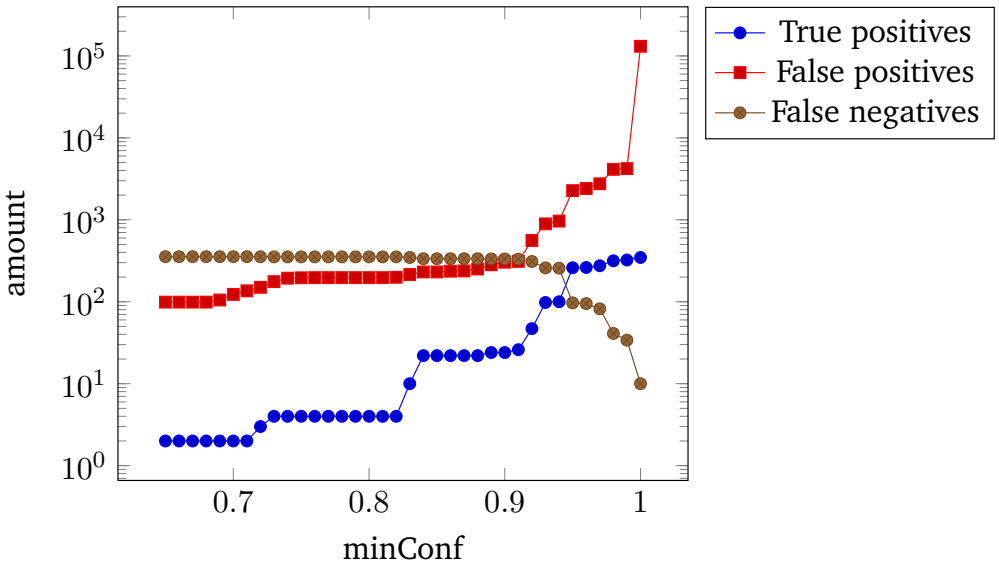

Fig. 5.4: The amount of true positives, false positives and false negatives of the second experiment.

Looking at an optimal recall, the model performs best with a minimum confidence of 1 where the recall measure is 0.972. From the 357 outages, 347 are predicted, but there are 131,119 false positives. It can be concluded that this model did not perform well and the model from the previous experiment significantly outperforms the current model. It is not recommended to use negative rules in for detecting outages in the manner it was used in this project. It might be interesting in the future when more research has been done on this subject.

5.4 Detecting outages with relative graph positions

The final experiment that was conducted incorporated the relative graph positions. The outage related association rules were used to build a classifier. Again the confidence threshold was varied throughout the experiment. The results of precision, recall and f-measure are shown in Figure 5.5. Figure 5.6 shows the number of true positives, false positives and false negatives.

From the results it can be concluded that this model did not meet its expectations. With a minimum confidence between 0.54 and 0.73, an optimal f-measure of 0.408 was acquired. This f-measure is not significantly worse than optimal f-measure from the first experiment, but again two thirds of the outages are missed.

In the case of an optimal recall, the model performed best with a minimum confidence of 0. With a recall of 0.969, the model detected 346 of the 357 outages. The model predicted most of the outages, even though it did not perform as well as the model from the first experiment. However, the model also predicted 31,718 false positives. The objective of this project is to reduce the amount of false alarms, which means this amount of false positives is not desired. Therefore it is not advised to use this method.

It is still a very interesting concept and therefore some outage related association rules were analyzed to give a better insight in the method. The following rules were discovered with a minimum support of 15 and a minimum confidence of 0.9.

{'i(SCHAKELSTAND UIT)', 's(STORINGSVERKLIKKER AANGESPR)'} $\rightarrow$ 'OUTAGE'

{'i(VERMOGENSCHAKELAAR UIT)', 's(STORINGSVERKLIKKER AANGESPR)'} $\rightarrow$ 'OUTAGE'

{'i(VERMOGENSCHAKELAAR UIT)', 'p(AARDFOUT AANGESPR)'} $\rightarrow$ 'OUTAGE'

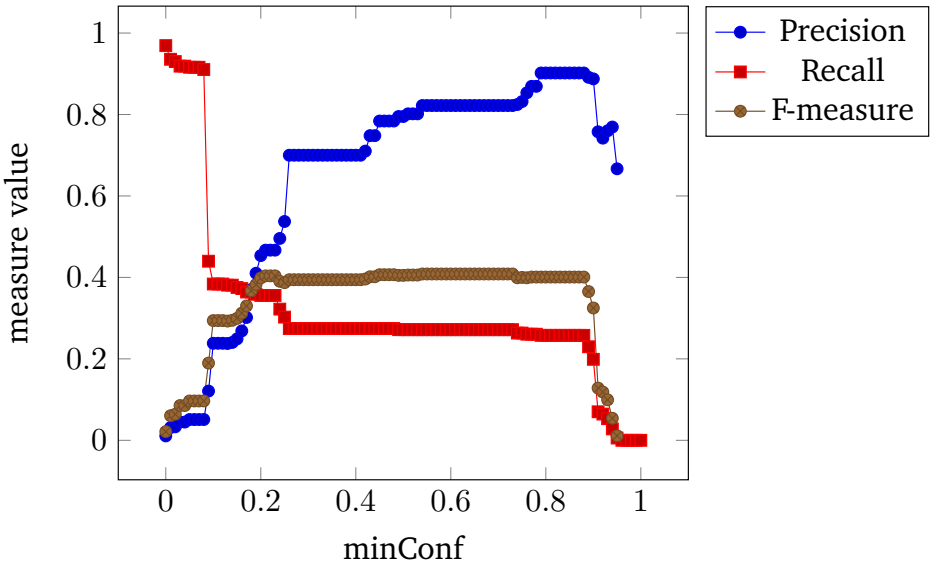

Fig. 5.5: The precision, recall and f-measure results of the third experiment

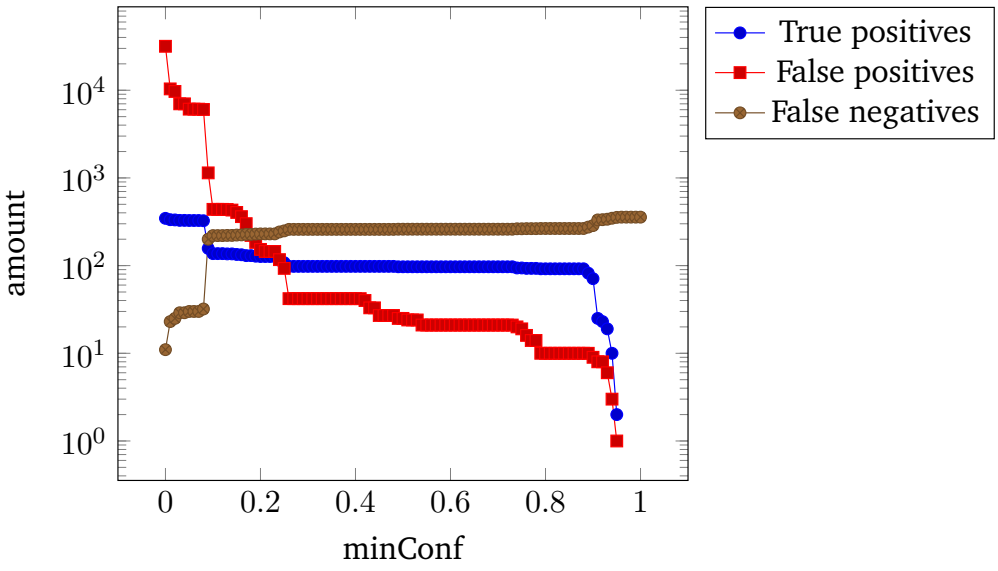

Fig. 5.6: The amount of true positives, false positives and false negatives of the third experiment

{'i(SCHAKELSTAND UIT)', 's(AARDFOUT AANGESPR)', 's(STORINGSVERKLIKKER AANGESPR)'} $\rightarrow$ 'OUTAGE'

The first rule is very interesting, because it seems that a triggered fault detector is always located further in the medium voltage ring. However, the last two rules show that the relative position of triggered assets is not always consistent. The earth fault is detected by preceding as well as succeeding assets.

From the results of the experiments and the rule analysis, it cannot be concluded that this approach works. When data quality improves and the amount of telemetry in the medium voltage rings is more consistent, it might be interesting to revisit this approach and reproduce the conducted experiment.

Conclusion

6

This project has shown that there is great potential for detecting outages using the signals. Extensive literature study was done to help understand the challenge. The chosen method for solving the assigned problem was association rule learning. This is an insightful method, where it is possible to verify the patterns that are discovered. Using association rule learning, patterns have been identified. These patterns were used to build a classifier, which can detect and locate the outages.

At the start of the thesis, two research questions arose. The first research question is: '*Is it possible to filter out the vast majority of signals and detect the outages?*'. To answer this, the following two sub-questions had to be answered: '*What percentage of true positives is the machine learning able to detect?*' and '*What percentage of signals is the machine learning technique able to filter out without missing additional true positives?*'. The optimal classifier was able to detect 98.3% of the outages. Approximately 6,600 outages were predicted, which means that the NOC operators only needs to look into these 6,600 outage signals instead of monitoring 1.1 million signals. This means the amount of signals is reduced by 99.4%. It can be concluded that it is indeed possible to filter out the vast majority of signals and detect the true positives.

The second research question is: '*How can you make a machine learning system that can find the source of the problem?*'. To answer this, the following two sub-questions had to be answered: '*To what extend is the machine learning system able to find the correct source of the problem?*' and '*How much time is saved by using machine learning?*'. The used classifier will detect outages per medium-voltage ring and can therefore instantly identify the medium-voltage ring where the outage occurred. Operators of the NOC sometimes need 5 minutes to identify the source of the problem. This can be reduced because the classifier will instantly identify the affected medium-voltage ring. How much time is saved exactly cannot be given, because it is not tested in the NOC. The results do suggest that it will be a substantial improvement. Operators need to react and locate the problem, which can take some time where the classifier does this instantly.

Both research questions have been answered and it can be concluded that this project has successfully shown the potential of detecting and locating outages using machine learning.

6.1 Recommendations

A third question arose at the start of the project, which was: *In what way can the applied method help Stedin ensure more efficient alarm management?*. This section will answer this question and give recommendations for Stedin.

6.1.1 Data

The recommendation to Stedin is to first improve their data quality. The signal log should have a separate column that indicates where the signal is coming from. This way the text column does not have to be parsed for a location. In a similar way, the event should be in a separate column. A lot of data is lost by processing the data, which could have influenced the used approach.

In case of the outage log, the location of the outage should be logged properly. From the approximately 1,600 outages, only 357 were used in this project. Only outages were used that could be assigned to a location in the network. The outages where no breaker was switched off, should be logged properly.

The topology data was of high quality. The only improvement might be that the graph representation is not dynamic. If Stedin could log the changes in the network properly, then it could be possible to create a dynamic graph. This will ensure that the signals are always correctly grouped together by medium voltage ring.

6.1.2 Model

Almost all outages can be predicted and the amount of false positive is reduced considerably. If all outages can be assigned to a correct location in the network, the model will be ready to be trained again and implemented at the NOC. The expectation is that the precision, recall and f-measure will increase.

When properly trained, the model is ready to be implemented on streaming data. The signals can be assigned to the correct medium voltage ring and from these databases a prediction can be made. It is not possible to cluster the streaming data in a similar way as was used in this project. Therefore, some other method should be applied to group these alarms together. For example, when an alarm arrives within the next 5 minutes, add this item to the transaction. Otherwise when no alarm arrives within the next 5 minutes, close the transaction and make a final prediction.

Every time an item is added to the item set, a prediction should be made. When it's likely to be a outage, the operators should be informed that there is a possible outage. They can then instantly send people to the related medium voltage ring to check for problems.

6.2 Future work

The results of this project indicate that it is possible to use a classifier to detect outages. Future work might be able to improve and extend this. Firstly, as mentioned earlier it is possible to use data streams for frequent item set mining. This can be used on real-time data to build a constantly improving model which can be applied directly at the NOC.

When it is possible to make the assumption that the chronological order of the data is correct, it might be useful to look into sequential pattern mining. The order of signals could be relevant for detecting outages. Future research could possibly find interesting sequential patterns related to outages, which might result in an improved model.

Furthermore it is possible to add other scenarios to the model, like maintenance. The model will learn patterns that imply there is maintenance going on in the network. This will add a new class to the problem and might reduce the number of false positives. Unfortunately this maintenance log was not yet available at the moment of this project.

In the far future streaming sensor values might be available, which will open a whole new window of opportunities. Using sensor values, a different approach can be used to build a more specific and sophisticated classifier that will be able to detect outages more accurately. The noise signals from maintenance and other factors will for example not be incorporated in such a model, which could result in a better classifier.

Bibliography

[1] Ramesh C Agarwal, Charu C Aggarwal, and VVV Prasad. “A tree projection algorithm for generation of frequent item sets”. In: *Journal of parallel and Distributed Computing* 61.3 (2001), pp. 350–371 (cit. on p. 20).

[2] Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. “Mining association rules between sets of items in large databases”. In: *Acm sigmod record*. Vol. 22. 2. ACM. 1993, pp. 207–216 (cit. on p. 18).

[3] Rakesh Agrawal, Heikki Mannila, Ramakrishnan Srikant, Hannu Toivonen, A Inkeri Verkamo, et al. “Fast discovery of association rules.” In: *Advances in knowledge discovery and data mining* 12.1 (1996), pp. 307–328 (cit. on pp. 18–20).

[4] Rakesh Agrawal, Ramakrishnan Srikant, et al. “Fast algorithms for mining association rules”. In: *Proc. 20th int. conf. very large data bases, VLDB*. Vol. 1215. 1994, pp. 487–499 (cit. on pp. 18–20).

[5] Rakesh Agrawal and Ramakrishnan Srikant. “Mining sequential patterns”. In: *Data Engineering, 1995. Proceedings of the Eleventh International Conference on*. IEEE. 1995, pp. 3–14 (cit. on p. 26).

[6] Kabir Ahmed, Iman Izadi, Tongwen Chen, David Joe, and Tim Burton. “Similarity analysis of industrial alarm flood data”. In: *IEEE Transactions on Automation Science and Engineering* 10.2 (2013), pp. 452–457 (cit. on pp. 14, 15).

[7] Nobuo Akamatsu, Taisuke Ishida, Nobuhiro Niina, and Yasunori Kobayashi. *Alarm management system*. US Patent 7,345,580. Mar. 2008 (cit. on p. 13).

[8] Maria-Luiza Antonie and Osmar R Zaıane. “An associative classifier based on positive and negative rules”. In: *Proceedings of the 9th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*. ACM. 2004, pp. 64–69 (cit. on p. 25).

[9] Sami Äyrämö and Tommi Kärkkäinen. “Introduction to partitioning-based clustering methods with a robust example”. In: *Reports of the Department of Mathematical Information Technology. Series C, Software engineering and computational intelligence* 1/2006 (2006) (cit. on p. 31).

[10] Christos Berberidis, Lefteris Angelis, and Ioannis Vlahavas. “Inter-transaction association rules mining for rare events prediction”. In: *Proc. 3rd Hellenic Conference on Artificial Intellligence*. 2004 (cit. on p. 26).

[11] Francesco Bonchi and Bart Goethals. “FP-Bonsai: the art of growing and pruning small fp-trees”. In: *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer. 2004, pp. 155–160 (cit. on p. 23).

[12]John Adrian Bondy, Uppaluri Siva Ramachandra Murty, et al. *Graph theory with applications*. Vol. 290. Citeseer, 1976 (cit. on p. 15).

[13]Christian Borgelt. "An Implementation of the FP-growth Algorithm". In: *Proceedings of the 1st international workshop on open source data mining: frequent pattern mining implementations*. ACM. 2005, pp. 1–5 (cit. on pp. 22, 23).

[14]Christian Borgelt. "Efficient implementations of apriori and eclat". In: *FIMI'03: Proceedings of the IEEE ICDM workshop on frequent itemset mining implementations*. 2003 (cit. on p. 21).

[15]Christian Borgelt. "Frequent item set mining". In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 2.6 (2012), pp. 437–456 (cit. on pp. 18, 19, 21, 22).

[16]ML Bransby and J Jenkinson. *The management of alarm systems*. Citeseer, 1998 (cit. on p. 14).

[17]*Brochure Strategie Stedin Groep 2018*. `https://www.stedingroep.nl/~/media/files/stedin/stedin-groep/strategie_stedingroep_brochure_online.pdf?la=nl-nl`. [Accessed 31-July-2018]. 2018 (cit. on p. 2).

[18]Fred Buckley and Marty Lewinter. *A friendly introduction to graph theory*. Prentice Hall, 2003 (cit. on p. 15).

[19]Chun Hing Cai, Ada Wai-Chee Fu, CH Cheng, and WW Kwong. "Mining association rules with weighted items". In: *Database Engineering and Applications Symposium, 1998. Proceedings. IDEAS'98. International*. IEEE. 1998, pp. 68–77 (cit. on p. 27).

[20]Aaron Ceglar and John F Roddick. "Association mining". In: *ACM Computing Surveys (CSUR)* 38.2 (2006), p. 5 (cit. on pp. 17, 18).

[21]Joong Hyuk Chang and Won Suk Lee. "Finding recent frequent itemsets adaptively over online data streams". In: *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2003, pp. 487–492 (cit. on p. 27).

[22]Gong Chen, Xindong Wu, and Xingquan Zhu. "Mining sequential patterns across data streams". PhD thesis. University of Vermont, 2005 (cit. on p. 27).

[23]Gong Chen, Xindong Wu, and Xingquan Zhu. "Sequential pattern mining in multiple streams". In: *null*. IEEE. 2005, pp. 585–588 (cit. on p. 27).

[24]Yizong Cheng. "Mean shift, mode seeking, and clustering". In: *IEEE transactions on pattern analysis and machine intelligence* 17.8 (1995), pp. 790–799 (cit. on p. 32).

[25]Yue Cheng, Iman Izadi, and Tongwen Chen. "Pattern matching of alarm flood sequences by a modified Smith–Waterman algorithm". In: *chemical engineering research and design* 91.6 (2013), pp. 1085–1094 (cit. on p. 15).

[26]James E Coutre, Wayne P Griffin, and Charles M Crisler. *Infusion fluid management system*. US Patent 5,317,506. May 1994 (cit. on p. 13).

[27]Maria Cvach. "Monitor alarm fatigue: an integrative review". In: *Biomedical instrumentation & technology* 46.4 (2012), pp. 268–277 (cit. on p. 14).

[28]Fredrik Dahlstrand. “Consequence analysis theory for alarm analysis”. In: *Knowledge-Based Systems* 15.1-2 (2002), pp. 27–36 (cit. on p. 15).

[29]Gautam Das, King-Ip Lin, Heikki Mannila, Gopal Renganathan, and Padhraic Smyth. “Rule Discovery from Time Series.” In: *KDD*. Vol. 98. 1. 1998, pp. 16–22 (cit. on p. 31).

[30]Jitender Deogun and Liying Jiang. “Prediction mining–an approach to mining association rules for prediction”. In: *International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing*. Springer. 2005, pp. 98–108 (cit. on p. 26).

[31]Konstantinos G Derpanis. “Mean shift clustering”. In: *Lecture Notes* (2005) (cit. on p. 32).

[32]Pedro Domingos. “A few useful things to know about machine learning”. In: *Communications of the ACM* 55.10 (2012), pp. 78–87 (cit. on p. 18).

[33]Barbara J Drew, Patricia Harris, Jessica K Zègre-Hemsey, et al. “Insights into the problem of alarm fatigue with physiologic monitor devices: a comprehensive observational study of consecutive intensive care unit patients”. In: *PloS one* 9.10 (2014), e110274 (cit. on p. 13).

[34]Engineering Equipment and Materials Users’ Association. *Alarm systems: A guide to design, management and procurement*. Engineering Equipment and Materials Users Association London, 1999 (cit. on p. 12).

[35]Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei 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 (cit. on p. 31).

[36]CI Ezeife and Mostafa Monwar. “SSM: A frequent sequential data stream patterns miner”. In: *Computational Intelligence and Data Mining, 2007. CIDM 2007. IEEE Symposium on*. IEEE. 2007, pp. 120–126 (cit. on p. 27).

[37]Ling Feng, Tharam Dillon, and James Liu. “Inter-transactional association rules for multi-dimensional contexts for prediction and their application to studying meteorological data”. In: *Data & Knowledge Engineering* 37.1 (2001), pp. 85–115 (cit. on p. 26).

[38]Ling Feng, Hongjun Lu, Jeffrey Xu Yu, and Jiawei Han. “Mining inter-transaction associations with templates”. In: *Proceedings of the eighth international conference on Information and knowledge management*. ACM. 1999, pp. 225–233 (cit. on p. 26).

[39]Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*. Vol. 1. 10. Springer series in statistics New York, NY, USA: 2001 (cit. on pp. 27, 38).

[40]Keinosuke Fukunaga and Larry Hostetler. “The estimation of the gradient of a density function, with applications in pattern recognition”. In: *IEEE Transactions on information theory* 21.1 (1975), pp. 32–40 (cit. on p. 32).

[41]Minos N Garofalakis, Rajeev Rastogi, and Kyuseok Shim. “SPIRIT: Sequential pattern mining with regular expression constraints”. In: *VLDB*. Vol. 99. 1999, pp. 7–10 (cit. on p. 26).

[42]Chris Giannella, Jiawei Han, Jian Pei, Xifeng Yan, and Philip S Yu. “Mining frequent patterns in data streams at multiple time granularities”. In: *Next generation data mining* 212 (2003), pp. 191–212 (cit. on p. 27).

[43]Bart Goethals. “Frequent set mining”. In: *Data mining and knowledge discovery handbook*. Springer, 2005, pp. 377–397 (cit. on p. 19).

[44]Bart Goethals. “Memory issues in frequent itemset mining”. In: *Proceedings of the 2004 ACM symposium on Applied computing*. ACM. 2004, pp. 530–534 (cit. on p. 20).

[45]Bart Goethals. “Survey on frequent pattern mining”. In: *Univ. of Helsinki* 19 (2003), pp. 840–852 (cit. on pp. 20, 24).

[46]Bart Goethals and Mohammed J Zaki. “Advances in frequent itemset mining implementations: report on FIMI’03”. In: *Acm Sigkdd Explorations Newsletter* 6.1 (2004), pp. 109–117 (cit. on p. 23).

[47]Bart Goethals and Mohammed J Zaki. “FIMI’03: Workshop on frequent itemset mining implementations”. In: *Third IEEE International Conference on Data Mining Workshop on Frequent Itemset Mining Implementations*. 2003, pp. 1–13 (cit. on p. 23).

[48]Kelly Creighton Graham and Maria Cvach. “Monitor alarm fatigue: standardizing use of physiological monitors and decreasing nuisance alarms”. In: *American Journal of Critical Care* 19.1 (2010), pp. 28–34 (cit. on p. 13).

[49]Gösta Grahne and Jianfei Zhu. “Efficiently using prefix-trees in mining frequent itemsets.” In: *FIMI*. Vol. 90. 2003 (cit. on p. 23).

[50]Gösta Grahne and Jianfei Zhu. “Fast algorithms for frequent itemset mining using fp-trees”. In: *IEEE transactions on knowledge and data engineering* 17.10 (2005), pp. 1347–1362 (cit. on p. 23).

[51]Jonathan L Gross, Jay Yellen, and Ping Zhang. *Handbook of graph theory*. Chapman and Hall/CRC, 2013 (cit. on p. 15).

[52]Paul R Halmos. *Naive set theory. The university series in undergraduate mathematics*. 1960 (cit. on p. 23).

[53]Jiawei Han, Hong Cheng, Dong Xin, and Xifeng Yan. “Frequent pattern mining: current status and future directions”. In: *Data Mining and Knowledge Discovery* 15.1 (2007), pp. 55–86 (cit. on p. 18).

[54]Jiawei Han, Jian Pei, and Micheline Kamber. *Data mining: concepts and techniques*. Elsevier, 2011 (cit. on p. 27).

[55]Jiawei Han, Jian Pei, Behzad Mortazavi-Asl, et al. “FreeSpan: frequent pattern-projected sequential pattern mining”. In: *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2000, pp. 355–359 (cit. on p. 26).

[56]Jiawei Han, Jian Pei, and Yiwen Yin. “Mining frequent patterns without candidate generation”. In: *ACM sigmod record*. Vol. 29. 2. ACM. 2000, pp. 1–12 (cit. on p. 19).

[57]Jiawei Han, Jian Pei, Yiwen Yin, and Runying Mao. “Mining frequent patterns without candidate generation: A frequent-pattern tree approach”. In: *Data mining and knowledge discovery* 8.1 (2004), pp. 53–87 (cit. on p. 22).

[58]Jochen Hipp, Ulrich Güntzer, and Gholamreza Nakhaeizadeh. “Algorithms for association rule mining—a general survey and comparison”. In: *ACM sigkdd explorations newsletter* 2.1 (2000), pp. 58–64 (cit. on p. 20).

[59]Alicia M Horkan. “Alarm fatigue and patient safety”. In: *Nephrology Nursing Journal* 41.1 (2014), p. 83 (cit. on p. 14).

[60]Wenkai Hu, Jiandong Wang, and Tongwen Chen. “A local alignment approach to similarity analysis of industrial alarm flood sequences”. In: *Control Engineering Practice* 55 (2016), pp. 13–25 (cit. on p. 15).

[61]M Ilayaraja and T Meyyappan. “Efficient data mining method to predict the risk of heart diseases through frequent itemsets”. In: *Procedia Computer Science* 70 (2015), pp. 586–592 (cit. on p. 18).

[62]*Jaarverslag Stedin Groep 2017*. `https://www.stedingroep.nl/~/media/files/stedin/stedin-groep/investor-relations/jaarverslag-stedin-groep-2017.pdf?la=nl-nl`. [Accessed 31-July-2018]. 2017 (cit. on p. 3).

[63]Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An introduction to statistical learning*. Vol. 112. Springer, 2013 (cit. on pp. 27, 38, 39).

[64]Ruoming Jin and Gagan Agrawal. “Frequent pattern mining in data streams”. In: *Data Streams*. Springer, 2007, pp. 61–84 (cit. on p. 31).

[65]Richard M Karp, Scott Shenker, and Christos H Papadimitriou. “A simple algorithm for finding frequent elements in streams and bags”. In: *ACM Transactions on Database Systems (TODS)* 28.1 (2003), pp. 51–55 (cit. on p. 27).

[66]Avinash Konkani, Barbara Oakley, and Thomas J Bauld. “Reducing hospital noise: a review of medical device alarm management”. In: *Biomedical Instrumentation & Technology* 46.6 (2012), pp. 478–487 (cit. on p. 14).

[67]Shiqi Lai and Tongwen Chen. “A method for pattern mining in multiple alarm flood sequences”. In: *Chemical Engineering Research and Design* 117 (2017), pp. 831–839 (cit. on p. 15).

[68]Anthony JT Lee and Chun-Sheng Wang. “An efficient algorithm for mining frequent inter-transaction patterns”. In: *Information Sciences* 177.17 (2007), pp. 3453–3476 (cit. on p. 26).

[69]Victor E Lee, Ruoming Jin, and Gagan Agrawal. “Frequent pattern mining in data streams”. In: *Frequent Pattern Mining*. Springer, 2014, pp. 199–224 (cit. on p. 26).

[70]Qing Li, Ling Feng, and Allan Wong. “From intra-transaction to generalized inter-transaction: landscaping multidimensional contexts in association rule mining”. In: *Information Sciences* 172.3-4 (2005), pp. 361–395 (cit. on p. 26).

[71]Wenmin Li, Jiawei Han, and Jian Pei. “CMAR: Accurate and efficient classification based on multiple class-association rules”. In: *icdm*. IEEE. 2001, p. 369 (cit. on p. 24).

[72]Ming-Yen Lin, Suh-Yin Lee, et al. “Fast discovery of sequential patterns through memory indexing and database partitioning”. In: *Journal of Information Science and Engineering* 21.1 (2005), pp. 109–128 (cit. on p. 26).

[73]Bing Liu. “Association rules and sequential patterns”. In: *Web Data Mining*. Springer, 2011, pp. 17–62 (cit. on p. 26).

[74]Guimei Liu, Hongjun Lu, Yabo Xu, and Jeffrey Xu Yu. “Ascending frequency ordered prefix-tree: Efficient mining of frequent patterns”. In: *Database Systems for Advanced Applications, 2003.(DASFAA 2003). Proceedings. Eighth International Conference On*. IEEE. 2003, pp. 65–72 (cit. on p. 23).

[75]Guimei Liu, Hongjun Lu, Jeffrey Xu Yu, Wei Wang, and Xiangye Xiao. “AFOPT: An Efficient Implementation of Pattern Growth Approach.” In: *FIMI*. 2003 (cit. on p. 23).

[76]Jun Liu, Khiang Wee Lim, Weng Khuen Ho, et al. “The intelligent alarm management system”. In: *IEEE software* 20.2 (2003), pp. 66–71 (cit. on p. 13).

[77]Junqiang Liu, Yunhe Pan, Ke Wang, and Jiawei Han. “Mining frequent item sets by opportunistic projection”. In: *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2002, pp. 229–238 (cit. on p. 20).

[78]Hongjun Lu, Ling Feng, and Jiawei Han. “Beyond intratransaction association analysis: mining multidimensional intertransaction association rules”. In: *ACM Transactions on Information Systems (TOIS)* 18.4 (2000), pp. 423–454 (cit. on p. 26).

[79]Hongjun Lu, Jiawei Han, and Ling Feng. “Stock movement prediction and n-dimensional inter-transaction association rules”. In: *1998 ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery, Seattle, WA, USA, ACM, New York, USA*. 1998 (cit. on p. 26).

[80]Bing Liu Wynne Hsu Yiming Ma and Bing Liu. “Integrating classification and association rule mining”. In: *Proceedings of the fourth international conference on knowledge discovery and data mining*. 1998 (cit. on pp. 24, 25, 34).

[81]James MacQueen et al. “Some methods for classification and analysis of multivariate observations”. In: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Vol. 1. 14. Oakland, CA, USA. 1967, pp. 281–297 (cit. on p. 31).

[82]Gurmeet Singh Manku and Rajeev Motwani. “Approximate frequency counts over data streams”. In: *VLDB’02: Proceedings of the 28th International Conference on Very Large Databases*. Elsevier. 2002, pp. 346–357 (cit. on p. 27).

[83]Tom Michael Mitchell. *The discipline of machine learning*. Vol. 9. Carnegie Mellon University, School of Computer Science, Machine Learning Department, 2006 (cit. on p. 18).

[84]Mike Mitka. “Joint commission warns of alarm fatigue: multitude of alarms from monitoring devices problematic”. In: *Jama* 309.22 (2013), pp. 2315–2316 (cit. on p. 14).

[85]Laurence Morissette and Sylvain Chartier. “The k-means clustering technique: General considerations and implementation in Mathematica”. In: *Tutorials in Quantitative Methods for Psychology* 9.1 (2013), pp. 15–24 (cit. on p. 31).

[86]MS Mythili and AR Mohamed Shanavas. “Performance evaluation of apriori and FP-growth algorithms”. In: *International Journal of Computer Applications* 79.10 (2013) (cit. on p. 21).

[87]Meera Narvekar and Shafaque Fatma Syed. “An optimized algorithm for association rule mining using FP tree”. In: *Procedia Computer Science* 45 (2015), pp. 101–110 (cit. on p. 24).

[88]Junya Nishiguchi and Tsutomu Takai. “IPL2 and 3 performance improvement method for process safety using event correlation analysis”. In: *Computers & Chemical Engineering* 34.12 (2010), pp. 2007–2013 (cit. on p. 14).

[89]Peter van Oirsouw and JFG Cobben. *Netten voor distributie van elektriciteit*. Phase to Phase, 2011 (cit. on pp. 7–9, 11, 12).

[90]*Paris Agreement*. United Nations Treaty Collection XXVII 7.d (cit. on p. 1).

[91]Jian Pei, Jiawei Han, Hongjun Lu, et al. “H-Mine: Fast and space-preserving frequent pattern mining in large databases”. In: *Iie Transactions* 39.6 (2007), pp. 593–605 (cit. on p. 20).

[92]Jian Pei, Jiawei Han, Hongjun Lu, et al. “H-mine: Hyper-structure mining of frequent patterns in large databases”. In: *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*. IEEE. 2001, pp. 441–448 (cit. on p. 20).

[93]Jian Pei, Jiawei Han, Behzad Mortazavi-Asl, et al. “Mining sequential patterns by pattern-growth: The prefixspan approach”. In: *IEEE Transactions on Knowledge & Data Engineering* 11 (2004), pp. 1424–1440 (cit. on p. 26).

[94]Jian Pei, Jiawei Han, Behzad Mortazavi-Asl, et al. “Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth”. In: *icccn*. IEEE. 2001, p. 0215 (cit. on p. 26).

[95]JoAnne Phillips and Jane H Barnsteiner. “Clinical alarms: improving efficiency and effectiveness”. In: *Critical care nursing quarterly* 28.4 (2005), pp. 317–323 (cit. on p. 14).

[96]Andrea Pietracaprina. “Mining frequent itemsets using patricia tries”. In: (2003) (cit. on p. 24).

[97]David Martin Powers. “Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation”. In: (2011) (cit. on p. 39).

[98]Dorian Pyle and Christina San Jose. “An executive’s guide to machine learning”. In: *McKinsey Quarterly* 3 (2015), pp. 44–53 (cit. on p. 18).

[99]Balázs Rácz. “nonordfp: An FP-growth variation without rebuilding the FP-tree.” In: *FIMI*. 2004 (cit. on p. 23).

[100]Santanu Saha Ray. “Subgraphs, Paths, and Connected Graphs”. In: *Graph Theory with Algorithms and its Applications*. Springer, 2013, pp. 11–24 (cit. on p. 15).

[101]Michelle Raymond, Liana Maria Kiff, Sophie Burkart, et al. *High volume alarm management system*. US Patent 8,554,714. Oct. 2013 (cit. on p. 14).

[102]Charles F Ridolfo. *Alarm management system*. US Patent 6,492,901. Dec. 2002 (cit. on p. 13).

[103]Douglas H Rothenberg. *Alarm management for process control: a best-practice guide for design, implementation, and use of industrial alarm systems*. Momentum Press, 2009 (cit. on p. 13).

[104]Keith J Ruskin and Dirk Hueske-Kraus. “Alarm fatigue: impacts on patient safety”. In: *Current Opinion in Anesthesiology* 28.6 (2015), pp. 685–690 (cit. on p. 14).

[105]Aiman Moyaid Said, PDD Dominic, and Azween B Abdullah. “A comparative study of fp-growth variations”. In: *International Journal of Computer Science and Network Security* 9.5 (2009), pp. 266–272 (cit. on p. 23).

[106]Ashok Savasere, Edward Robert Omiecinski, and Shamkant B Navathe. *An efficient algorithm for mining association rules in large databases*. Tech. rep. Georgia Institute of Technology, 1995 (cit. on p. 20).

[107]Markus Schleburg, Lars Christiansen, Nina F Thornhill, and Alexander Fay. “A combined analysis of plant connectivity and alarm logs to reduce the number of alerts in an automation system”. In: *Journal of process control* 23.6 (2013), pp. 839–851 (cit. on p. 15).

[108]Benjamin Schlegel, Rainer Gemulla, and Wolfgang Lehner. “Memory-efficient frequent-itemset mining”. In: *Proceedings of the 14th International Conference on Extending Database Technology*. ACM. 2011, pp. 461–472 (cit. on p. 24).

[109]Sue Sendelbach. “Alarm fatigue”. In: *Nursing Clinics* 47.3 (2012), pp. 375–382 (cit. on p. 13).

[110]Sue Sendelbach and Marjorie Funk. “Alarm Fatigue A Patient Safety Concern”. In: *AACN advanced critical care* 24.4 (2013), pp. 378–386 (cit. on p. 14).

[111]Ramakrishnan Srikant and Rakesh Agrawal. “Mining sequential patterns: Generalizations and performance improvements”. In: *International Conference on Extending Database Technology*. Springer. 1996, pp. 1–17 (cit. on p. 26).

[112]Asuman Suenbuel, Thomas Odenwald, and Brian S Mo. *False alarm mitigation using a sensor network*. US Patent 7,250,855. July 2007 (cit. on p. 13).

[113]Tanya Tanner. “The problem of alarm fatigue”. In: *Nursing for women’s health* 17.2 (2013), pp. 153–157 (cit. on p. 13).

[114]Feng Tao, Fionn Murtagh, and Mohsen Farid. “Weighted association rule mining using weighted support and significance framework”. In: *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2003, pp. 661–666 (cit. on p. 27).

[115]Wei-Guang Teng, Ming-Jyh Hsieh, and Ming-Syan Chen. “On the mining of substitution rules for statistically dependent items”. In: *null*. IEEE. 2002, p. 442 (cit. on p. 25).

[116]Fadi Thabtah. “A review of associative classification mining”. In: *The Knowledge Engineering Review* 22.1 (2007), pp. 37–65 (cit. on p. 24).

[117]Fadi Thabtah, Peter Cowling, and Yonghong Peng. “MCAR: multi-class classification based on association rule”. In: *Computer Systems and Applications, 2005. The 3rd ACS/IEEE International Conference on*. IEEE. 2005, p. 33 (cit. on p. 24).

[118]Akhilesh Tiwari, Rajendra K Gupta, and Dharma P Agrawal. “A survey on frequent pattern mining: Current status and challenging issues”. In: *Information Technology Journal* 9.7 (2010), pp. 1278–1293 (cit. on p. 19).

[119]Anthony KH Tung, Hongjun Lu, Jiawei Han, and Ling Feng. “Breaking the barrier of transactions: Mining inter-transaction association rules”. In: *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 1999, pp. 297–301 (cit. on p. 26).

[120]Jia Wang, Hongguang Li, Jingwen Huang, and Chong Su. “Association rules mining based analysis of consequential alarm sequences in chemical processes”. In: *Journal of Loss Prevention in the Process Industries* 41 (2016), pp. 178–185 (cit. on pp. 15, 18, 22, 31).

[121]Jia Wang, Hongguang Li, Jinwen Huang, and Chong Su. “A data similarity based analysis to consequential alarms of industrial processes”. In: *Journal of Loss Prevention in the Process Industries* 35 (2015), pp. 29–34 (cit. on p. 14).

[122]Ke Wang, Liu Tang, Jiawei Han, and Junqiang Liu. “Top down fp-growth for association rule mining”. In: *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer. 2002, pp. 334–340 (cit. on p. 23).

[123]Wei Wang, Jiong Yang, and Philip S Yu. “Efficient mining of weighted association rules (WAR)”. In: *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2000, pp. 270–274 (cit. on p. 27).

[124]David D Woods. “The alarm problem and directed attention in dynamic fault management”. In: *Ergonomics* 38.11 (1995), pp. 2371–2393 (cit. on p. 12).

[125]David D Woods, William C Elm, Melvin H Lipner, George E Butterworth III, and James R Easter. *Alarm management system*. US Patent 4,816,208. Mar. 1989 (cit. on p. 13).

[126]Xiaoxin Yin and Jiawei Han. “CPAR: Classification based on predictive association rules”. In: *Proceedings of the 2003 SIAM International Conference on Data Mining*. SIAM. 2003, pp. 331–335 (cit. on p. 24).

[127]Unil Yun and John J Leggett. “WFIM: weighted frequent itemset mining with a weight range and a minimum weight”. In: *Proceedings of the 2005 SIAM international conference on data mining*. SIAM. 2005, pp. 636–640 (cit. on p. 27).

[128]Mohammed J Zaki. “SPADE: An efficient algorithm for mining frequent sequences”. In: *Machine learning* 42.1-2 (2001), pp. 31–60 (cit. on p. 26).

[129]Mohammed Javeed Zaki, Srinivasan Parthasarathy, Mitsunori Ogihara, Wei Li, et al. "New Algorithms for Fast Discovery of Association Rules." In: *KDD*. Vol. 97. 1997, pp. 283–286 (cit. on p. 20).

[130]Qiankun Zhao and Sourav S Bhowmick. "Sequential pattern mining: A survey". In: *ITechnical Report CAIS Nayang Technological University Singapore* 1 (2003), p. 26 (cit. on p. 26).

[131]Zijian Zheng, Ron Kohavi, and Llew Mason. "Real world performance of association rule algorithms". In: *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2001, pp. 401–406 (cit. on p. 24).