JYVÄSKYLÄ STUDIES IN COMPUTING

215

Pekka Wartiainen

Detector-Based Visual Analysis of Time-Series Data

JYVÄSKYLÄN YLIOPISTO

JYVÄSKYLÄ STUDIES IN COMPUTING 215

Pekka Wartiainen

Detector-Based Visual Analysis of Time-Series Data

Esitetään Jyväskylän yliopiston informaatioteknologian tiedekunnan suostumuksella
julkisesti tarkastettavaksi yliopiston Agora-rakennuksen auditoriossa 2
kesäkuun 29. päivänä 2015 kello 12.

Academic dissertation to be publicly discussed, by permission of
the Faculty of Information Technology of the University of Jyväskylä,
in building Agora, auditorium 2, on June 29, 2015 at 12 o'clock noon.

UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2015

Detector-Based Visual Analysis of Time-Series Data

JYVÄSKYLÄ STUDIES IN COMPUTING 215

Pekka Wartiainen

Detector-Based Visual Analysis of Time-Series Data

UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2015

Editors
Timo Männikkö
Department of Mathematical Information Technology, University of Jyväskylä
Pekka Olsbo, Ville Korkiakangas
Publishing Unit, University Library of Jyväskylä

URN:ISBN:978-951-39-6244-9
ISBN 978-951-39-6244-9 (PDF)

ISBN 978-951-39-6243-2 (nid.)
ISSN 1456-5390

Jyväskylä University Printing House, Jyväskylä 2015

ABSTRACT

Wartiainen, Pekka
Detector-Based Visual Analysis of Time-Series Data
Jyväskylä: University of Jyväskylä, 2015, 62 p.(+included articles)
(Jyväskylä Studies in Computing
ISSN 1456-5390; 215)
ISBN 978-951-39-6243-2 (nid.)
ISBN 978-951-39-6244-9 (PDF)
Finnish summary
Diss.

This work focuses on the analysis of industrial time series (TS) using techniques derived from visual analytics and data mining. An industrial measurement process, for example in the context of a power plant, produces a great amount of multivariate time-series data from various sources such as fine particles, temperature, pressure, gas content, and automated powerplant controls. In addition, the production of energy from biomass involves significant background knowledge and contextual information (meta data) that describes the process and that may change over time. To understand and find deeper meaning in this data mass is a demanding task for both humans and computers. Computers are good at computations while humans excel at reasoning. Visual analytics combines the fields of automated data analysis and visual data exploration with visualization methods, data analysis, and user interaction. In this research, a context-sensitive framework based on visual analytics was developed to utilize robust knowledge extraction using data mining techniques and change-point detection. These methods and algorithms are tested and applied in dynamic real-world scenarios. First, the context-sensitive approach is introduced. Second, the tools, such as statistical and predictive change-point detection and time series profiling via clustering, are presented. Third, the whole analysis chain is applied to a case of analyzing fine particulates produced in the use of biomass as a fuel and their relationship to the power plant controls. The related studies show the value of preprocessing and the possibilities of the visual analysis. As a result, a graphical user interface (GUI) was produced, along with automated data analysis methods, that follows the principles of the described context-sensitive framework. As the final result, a detector-based visual analysis framework is demonstrated and evaluated.

Keywords: visual analytics, change-point detection, context, visualization, user interaction, visual data exploration, graphical user interface, knowledge discovery, data mining, clustering, neural network

Author

Pekka Wartiainen
Department of Mathematical Information Technology
University of Jyväskylä
Finland

Supervisors

Professor Tommi Kärkkäinen
Department of Mathematical Information Technology
University of Jyväskylä
Finland

Senior Lecturer Anneli Heimbürger
Faculty of Information Technology
University of Jyväskylä
Finland

Reviewers

Research Director Yrjö Hiltunen
Department of Environmental Science
University of Eastern Finland
Finland

Professor Martti Juhola
School of Information Sciences
University of Tampere
Finland

Opponent

Professor Risto Ritala
Department of Automation Science and Engineering
Tampere University of Technology
Finland

ACKNOWLEDGEMENTS

I would like to express my gratitude to my supervisors, Prof. Tommi Kärkkäinen and Dr. Anneli Heimbürger for their guidance and support, and to Dr. Sami Äyrämö for supervising at the beginning of my work.

I would also like to thank my colleagues and collaborators, Mr. Paavo Nieminen, Mr. Alexandr Maslov, Mrs. Mirka Saarela, Mr. Joonas Hämäläinen, Mr. Alexey Kornilov, and all the other colleagues who have helped me during this work.

I am grateful to the Department of Mathematical Information Technology of the University of Jyväskylä for giving me this opportunity to work on my dissertation. All this would have not been possible without the funding from the OSER-project, managed by Dr. Pasi Vainikka, as the part of Regional Development Fund Programme, and the funding from Regional Council of Central Finland, cordinated by Mrs. Hilkka Laine. I thank the Fortum Foundation for partly funding my research.

Finally, I would like to thank my family for the never-ending support and encouragement. Also, the support from my friends at the university coffee rooms, from personal hobbies, and elsewhere have been very important to me. Thank you all!

Jyväskylä
June 12, 2015

Pekka Wartiainen

GLOSSARY

BFB	Bubling fluidized bed
CUSUM	Cumulative sum control chart
ELPI	Electronic low-pressure impactor
EM	Expectation maximization
GPU	Graphical processing unit
GUI	Graphical user interface
KD	Knowledge discovery
KDD	Knowledge discovery in databases
MLP	Multilayered perceptron
OSER	Osaava energia Keski-Suomessa
PCA	Principal component analysis
SPA	Sensing, processing, and actuation
SOR	Sequential over-relaxation
TS	Time series

LIST OF FIGURES

LIST OF TABLES

CONTENTS

LIST OF INCLUDED ARTICLES

PI Pekka Wartiainen, Tommi Kärkkäinen, Anneli Heimbürger, and Sami Äyrämö. Methods of Visual Analytics in Knowledge Mining. *21st European-Japanese Conference on Information Modelling and Knowledge Bases, pp. 117–121*, 2011.

PII Pekka Wartiainen, Tommi Kärkkäinen, Anneli Heimbürger, and Sami Äyrämö. Context-Sensitive Approach to Dynamic Visual Analytics of Energy Production Processes. *Information Modelling and Knowledge Bases XXIV, IOS Press, pp. 15–22*, 2013.

PIII Tommi Kärkkäinen, Alexandr Maslov, and Pekka Wartiainen. Region of interest detection using MLP. *In 22nd European Symposium on Artificial Neural Networks (ESANN)*, 2014.

PIV Pekka Wartiainen, Anneli Heimbürger and Tommi Kärkkäinen. Context-Sensitive Framework for Visual Analytics in Energy Production from Biomass. *Information Modelling and Knowledge Bases XXVI, IOS Press, pp. 449–456*, 2015.

PV Pekka Wartiainen and Tommi Kärkkäinen. Hierarchical, prototype-based clustering of multiple time series with missing values. *In 23nd European Symposium on Artificial Neural Networks (ESANN)*, 2015.

PVI Pekka Wartiainen, Tommi Kärkkäinen, Anneli Heimbürger, and Merja Hedman. Data Analysis Process for Particulate Measurements in a Bubbling Fluidized-Bed Boiler. *Expert Systems, Wiley, submitted*, 2015.

Pekka Wartiainen has contributed as the main author to each of these articles except [PIII]. The co-authors made substantial contributions to the conception, design, and empirical analysis, actively drafting and revising all the joint articles. For [PIII], Tommi Kärkkäinen is the main author, Wartiainen authored the section introducing the industrial time series application point of view and was responsible for the technical editing of the paper. For [PV], Tommi Kärkkäinen played a significant role in writing the theoretical section.

1 INTRODUCTION

People have invented various ways to collect and apply data, ranging from industrial contexts to everyday life. Collecting data involves measuring observations with various sensors and producing new data based on a phenomena. Storing data is relatively easy, but processing massive data sets still poses great challenges to humankind. The enormous quantities of data dealt with might contain so many variables that it would be almost impossible to analyze them in one piece. On the other hand, the data might contain a lot of unnecessary information that would not shed any light on the problem under investigation. Analyzing these massive data sets requires many resources, especially in real-time scenarios. In addition, data is often inconsistent, containing outliers and noise in real life.

Furthermore, real-life scenarios and processes have all kinds of dynamic variables and meta data that can indirectly affect the knowledge extraction. This background information defines the context of the environment [58]. Applied to real life, the methods of any other software development application should take into consideration the context and use contextual information as a support in computing [57]. A context-sensitive application includes context-based functions and uses context to provide relevant information and services to the user, where relevancy depends on the user's situation [58].

Industrial processes are providing a challenging basis for software development due to their dynamic nature. To describe the whole process, a versatile architecture framework that contains all parts of the process is needed. Here, the focus is on a case of energy production using biomass as a fuel. To analyze the case, a software architecture approach is introduced with a few novel tools and techniques to support the analysis process. Due to the real-world measurements, preprocessing and transformation of the target data is emphasized. For example, noise is often induced in the data by the measurement process. Both the equipment and the measured phenomena have noise factors that increase the total noise in the data. The amount of noise can be reduced with noise removal methods, but there is always a compromise between smooth data and lost information. Too smooth a result may mean the loss of important spikes from the data, which would have significant meaning in the interpretation of the results. Robust methods for achieving reliable results are discussed and applied in this work.

Data mining methods have been actively developed for the last 25 years [54]. The difference between statistical and data mining methods is that traditional statistical methods try to confirm hypotheses using the data, while data mining methods try to use the data to find new hypotheses and to resolve unknown facts (exploratory data analysis or EDA) [113]. Dimension reduction methods offer one approach to the challenge of handling data masses [54, 72]. In a dimension reduction process as part of data transformation, the number of variables in the data is reduced. This can be done simply by leaving out any unnecessary parameters or combining parameters together. Dimension reduction methods result in more responsive systems because their data can be managed in a more compact form [54]. Still, the analysis of huge scenarios is time consuming and requires a high level of expertise in information technology on the part of the end user.

Visual analytics refers to interactive methods and technologies that could be applied in presenting the results of the data mining process to users [112]. Visual analytics has been defined as ”a science of analytical reasoning, facilitated by interactive visual interfaces” [112]. However, the definition and meaning of visual analytics has since evolved, as follows. ”Visual analytics combines automated analysis techniques with interactive visualizations for an effective understanding, reasoning and decision making on the basis of very large and complex datasets” [72].

Visual analytics can be applied in many different fields. One such field is information security, where visual analytics has been applied in anomaly detection [17, 73, 84, 22]. In information security, the challenge is to find a suitable representation format when advising the administrator about an ongoing attack on the network. However, unlike the network administrator, managers of the company want to see a concise overview and not the small details of specific TCP/UDP packets. In energy production, this scenario applies to the company manager who is trying to maximize the profits of the power plant and to the analyst or operator who is interested in details and adjustments in the power plant mechanics. Each of the previous examples relate to two different stakeholders; end users are interested in different details in terms of the results. The idea behind visual analytics supports distinctive user roles (stakeholders) with versatile visualizations and data views.

Still, the general process of visual analytics is limited to using contextual information, the domain, or the end users in the analysis. Hence, a context-sensitive approach, that contains three main phases was developed. These three sections correspond to each step in the analysis of an industrial process: sensing, processing, and actuation. The analysis process begins with the sensing phase, where data collection methods and the domain-related research environment are defined. The second phase (processing) includes automated data handling where the data gathered in the first phase is analyzed and numerical results are produced. In the last phase (actuation), information is delivered to the researcher by visual interactive interfaces. With this approach, the entire process of visual analytics is thus observed.

1.1 Research objective

The objective of the research is to improve the analysis of industrial time-series (TS) data in real-world situations. To achieve this, the problem was approached from two different perspectives. One approach is a general view of the data analysis from which a suitable framework was defined. The other approach is on the application level method development utilizing clustering and change-point detection methods to analyze industrial time series in depth.

The main research questions for this study are:

1. How can knowledge extraction from multivariate time series in real-world scenarios be improved?
2. How can the analysis of an energy production process using biomass be improved?
3. Is it possible to make the interpretation of the results more understandable?
4. Is robust clustering a reliable dimension reduction method for industrial time-series data?
5. How to find cause effect relations from industrial time-series data?

This research is conducted using the constructive research method [28]. The goal is to construct artifacts that provide a theoretical framework and practical methods for data analysis as part of OSaava EneRgia Keski-Suomessa (OSER) project. Several industrial parties have participated in the project. These artifacts have been evaluated in the papers and in Chapter 6.

1.2 Structure of the work

The format of this dissertation is the collection of articles. The study was begun at the same time as the OSER project in January 2011. The early stages of the research consisted of an overall review of the research area and the review of a conference series on visual analytics. It was noted that there are not many processes or tools to visually analyze industrial data. Based on these findings, a context-sensitive analysis approach was introduced. Next, a graphical user interface (GUI) for the analysis of time series was implemented to provide a framework for a context-sensitive approach. Meanwhile in terms of the development of the GUI, change-point detection methods were implemented as part of the framework. The context-sensitive approach was then improved with an iterative design. The last phases of the study consisted of the evolution of the time series profiling algorithm using clustering methods and the test scenario for the whole

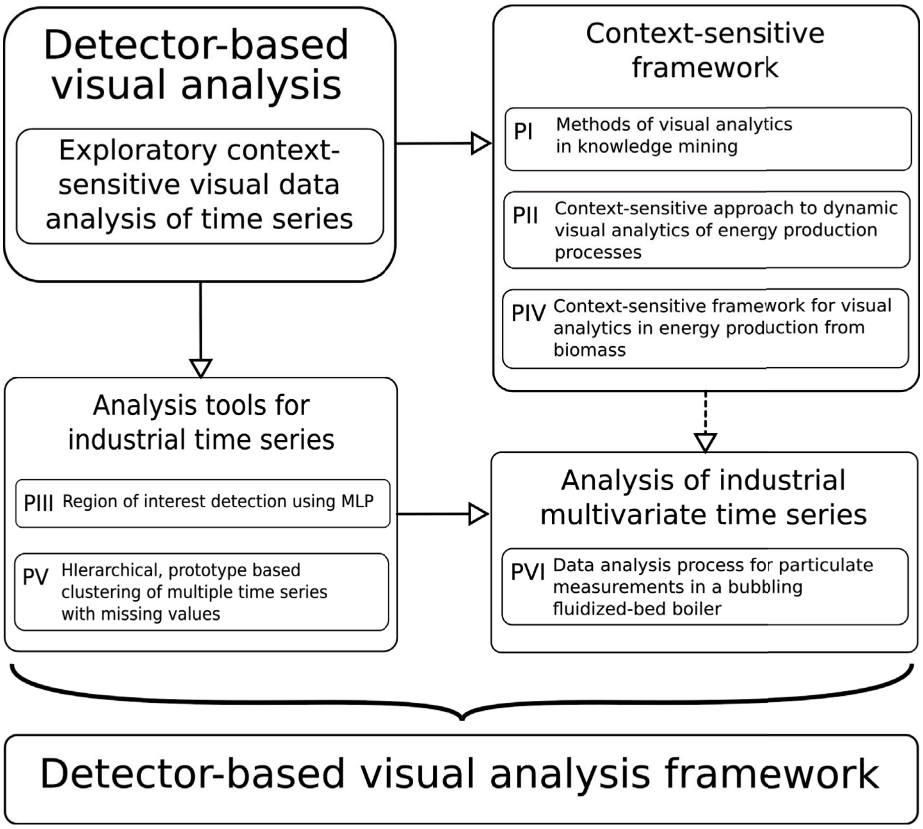

Detector-based visual analysis
Exploratory context-sensitive visual data analysis of time series
Context-sensitive framework
PI Methods of visual analytics in knowledge mining
PII Context-sensitive approach to dynamic visual analytics of energy production processes
PIV Context-sensitive framework for visual analytics in energy production from biomass
Analysis tools for industrial time series
PIII Region of interest detection using MLP
PV HIerarchical, prototype based clustering of multiple time series with missing values
Analysis of industrial multivariate time series
PVI Data analysis process for particulate measurements in a bubbling fluidized-bed boiler
Detector-based visual analysis framework

FIGURE 1 The main phases of the Ph.D. work related to the published papers.

analysis chain. The structure of the work and related papers therein is presented in Figure 1.

Chapter 1 introduces the motivation, goals, and methods for this study, as well as the structure of the work. Chapter 2 gives a short introduction to the energy production using biomass as an application area. The measurement data sets definite challenges that are introduced from the analysis point of view. Chapter 3 introduces the general framework; knowledge discovery (KD)and visual analytics processes are presented as a relevant part of the developed context-sensitive approach. Chapter 4 explains the tools used in this study to analyze industrial time series and explore them visually. Chapter 5 presents the published articles relating to this study. The key elements in the objectives, findings, and scientific contribution are given, as well as a summary of the main results. The findings are also related to the original research questions. Chapter 6 introduces the suggestion of a detector-based visual analysis framework. This new contribution demonstrates and evaluates the results even further. Chapter 7 discusses the key challenges and discoveries of the work. In addition, the tasks for future research are examined. Chapter 8 concludes this study and summarizes the phases in the work.

2 ON ENERGY PRODUCTION USING BIOMASS

The thesis contributes to the analysis of multivariate industrial time-series data. The main domain for this work has been in the field of energy production using biomass as a fuel [98, 11]. Our test data was collected from a full-scale power plant using bubbling fluidized bed (BFB) boiler technology (Figure 2). The process control of such an industrial process is carried out by an automation system, but the analysis of the collected data and the research and development work based on it is often done manually. Process control itself in the power plant is a result of advanced engineering, and the work presented in this thesis is not meant to improve this automation.

However, it is known that utilizing biomass is challenging because of the organic compounds [107]. Alkali chlorides are classified as fine particles, and they are released from the biomass due to the burning process. Alkali chlorides form chloride acids in high temperatures, which produces corrosion and fouling in the boiler [107, 101]. The quality of the burning process also affects the emissions and the environment; therefore, it should be taken into account when utilizing biomass as a renewable fuel [18, 7]. Understanding the behavior of the fine particles is crucial to improve the use of biomass as a fuel.

The basic operating procedure for a BFB boiler is that the fuel is fed to a furnace from a fuel feeding chute [98]. The flames of the burning fuel heat water that is circulating in the walls of the boiler. Gas from the burning process heats the vaporized water flowing in the superheaters. The difference between the BFB boiler and a traditional grate boiler is that in the BFB boiler the bed material consists of sand that is bubbling, and some of the exhaust gas is fed back to the bed. Electricity is produced by massive generators that are rotated by the superheated steam.

In a real-world situation, a huge number of process variables are measured in a power plant [8, 62]. In this study, the focus is on measurement data relating to a BFB boiler (particle concentration of gas and particulates) and measurement data relating to process (temperatures, pressures, airflows, etc.). Other contextual information (time, location, measuring equipment, etc.) is also required. All these background details affect the calculations. For instance, there is a difference in

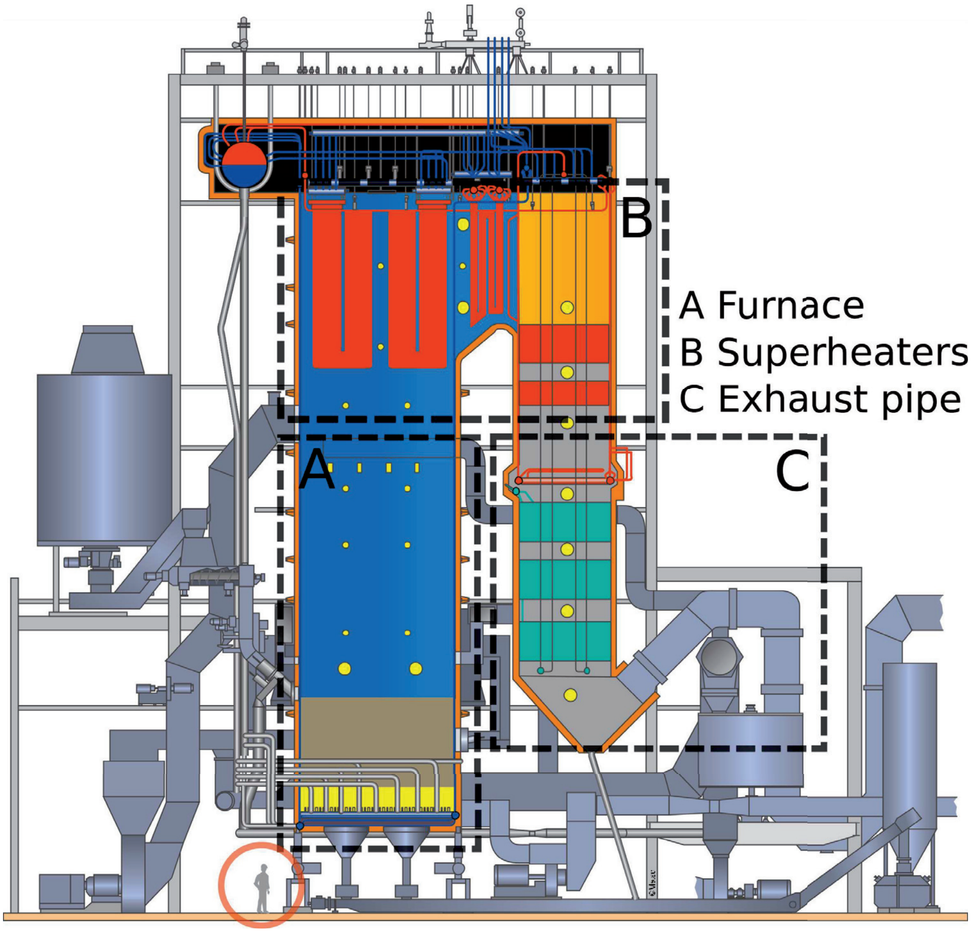


FIGURE 2 Example of a full-scale BFB boiler power plant. Notice the figure of a man below the furnace (A) to give an idea of size. The exhaust gas flows from the furnace to the exhaust pipe (C) through the superheater area (B). The picture is owned by Valmet [115] and is used with permission.

gas concentration environments depending on whether the measurement is taken inside the furnace or just before the exhaust pipe. In the furnace, the conditions may be skewed; for example, 60% of the gas might be flowing up against the right wall while 40% of the gas might be flowing against the left wall. Here, if the first measurement includes only 60% of the total gas and the last one includes 100%, there is a measurable difference in gas concentration.

The main data type in this study is time-series data, a well-known data type [15, 51] for which there are applications in on different domains [75, 71, 121, 104, 50]. A time series is a sequence of data points typically consisting of successive measurements made over a time interval [15]. Any periodically measured or recorded phenomenon produces a time series. For example, the temperature measured in superheaters every minute produces a time series by a length of 60 for every hour. One piece of data typically contains several measurements. Multiple time series in one data set is called multivariate time series.

Regarding the data used for this study, different data sets were measured with different kinds of equipment. Each piece of equipment recorded and stored the data in its own format and with different settings. The largest data was recorded every 60 seconds, but one data set used a 61-second recording interval, and other data sets were using faster frequencies (5 and 20 seconds, respectively). Some of the data was recorded for long periods, totaling one month of recording using different mixtures of the fuel. The amount and the distribution of fine particles were carefully considered during the analysis.

The physical measurement locations were located around the power plant and therefore the distance between two measurement locations may be tens of meters. However, after a discussion with experts in the field, it was agreed to assume zero delays in the overall analysis between measurement locations. The main reason was that the data sets were downscaled to match the slowest update interval (one minute) so the exhaust gas will have enough time to the travel from the furnace to open air through all the measurement locations. Still, while building the framework for the analysis, the option for possible delays for the future analysis cases was left.

Another challenge, which is related to almost all recordings from the real-world scenarios, was the noise in the data. The amount and quality of the noise varies depending on the measurement equipment, and some of the data sets are already averaged within their sampling period. Two extreme examples are the process data set with a 60-second recording interval, which is relatively noiseless data, and the particle distribution data recorded every five seconds, which contains a lot of noise. This noise was partly due to the features of the measurement device, an electronic low-pressure impactor (ELPI) [29], and partly due to the fast recording interval. The challenges are to decide on the noise reduction filter, when it is used, and how much the data needs to be filtered. There is no single correct answer, which also makes the validation and evaluation procedure difficult.

Next, there is the challenge of both high dimensional and large amounts of data. One power plant produces a lot of data with its numerous sensors. Even if the data is recorded only every minute, the long running times propagates the size of the data. For example, in one process measurement data set, a nine-day-long continuous test run produces 12960 observations for 64 variables. All the data sets in use were not recorded all the time, but the total number of variables is over a hundred. Still, the quality of the data is a key factor for useful analysis. Some variables or data may contain missing values. The solution depends on the situation. If the variable is constant or otherwise does not provide relevant information to the analysis, it can be deleted, but otherwise data imputation is recommended (hot or cold deck imputation) [6]. All in all, robust analysis methods are required to achieve reliable results in industrial data analysis.

3 TOWARDS A CONTEXT-SENSITIVE ANALYSIS FRAMEWORK

In this chapter, the knowledge discovery process is briefly examined. Next, an overview of visual analytics is given, and the context-sensitive approach is introduced.

3.1 Knowledge discovery process

The knowledge discovery (KD) process and knowledge discovery in databases (KDD) describes the entire analysis process, where knowledge is mined from data [37, 36, 38]. Generally, the principles of KDD are the same as in EDA. The purpose is to come up with new hypotheses based on the findings from the data [55]. Hand et al. focus more on the data mining part of the KDD process but maintain the high-level process in the loop[55]. They also focus on the data mining part but integrate the database section in the analysis [54].

The KDD process onsists of the five phases (shown in Figure 3) briefly introduced below [36].

Data selection is the first phase in the KDD process. This phase is highly dependent on the domain and the goal of the study. The data for the analysis is selected from the data mass to narrow down the sources of information, that is, to analyze only the relevant data. The target data is defined.

In the *preprocessing*, the data is prepared for analysis. First, the defined target

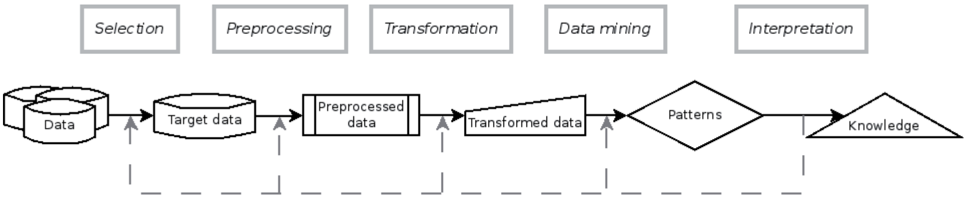


FIGURE 3 General overview of the KDD process according to Fayyad et al. [36].

data is cleaned and converted to the correct format. The target data may contain noise and can be inconsistent, especially if the data is measured from the real world. Many of the transformation and data mining methods require consistent and good quality data. Therefore, normalization, noise removal, and data imputation methods are applied in this phase. This phase is crucial for the success of the remaining phases and is often underestimated in studies. In addition, in case of multiple data sets, the sources need to be synchronized.

Transformation of the preprocessed data means to alter the data to make it suitable for data mining tasks. This includes, for example, feature extraction, selection, and dimension reduction. The transformation methods depend on the data mining methods that will be applied. In some cases, the data mining methods can be applied as part of the transformation methods.

Data mining methods are the key in the KDD process. Here, the results and patterns are extracted from the transformed data. There is a wide variety of data mining methods; for example, clustering, classification, regression, and association rule learning are used for different applications. The method used depends on the goal and on the data type. Data mining methods are classified as unsupervised or supervised. Supervised methods require training data, which again has to be prepared in the previous steps. Furthermore, data mining methods can be divided into off-line and on-line methods to analyze dynamic data.

Interpretation is the last phase of the KDD process. The obtained results and patterns of data mining are evaluated, as there might have been problems or mistakes in each step of the process. Finally, the objective is to address the goal of the study based on the findings. In most cases, if the goal is achieved, the produced knowledge has business value and may contribute to scientific knowledge.

The method development in this research concentrates on data mining methods. They are applied in the transformation phase and as part of statistical analysis (change-point detection). Data mining is a well-researched area. A brief general introduction was given by Fayyad et al. [37], Hand et al. [55], and Glymour et al. [48]. In data mining (and machine learning), problems, tasks, and methods are typically classified as *supervised* or *unsupervised* [2, 54], although, so-called semisupervised scenarios (partially labeled data) have also emerged [43].

All in all, there are still challenges related to the application of the KD process. One has to be an experienced data miner to be able to take full advantage of the great potential of data mining. In addition, the KD process is not too flexible in a multidisciplinary research group, where new information and data sources are added into the study on the fly.

3.2 Visual analytics

As concluded during the KD process, the raw data has no value in itself; only the extracted information has value. Also, in real-world scenarios, time and money are wasted and opportunities are lost if the results are not acquired in a reason-

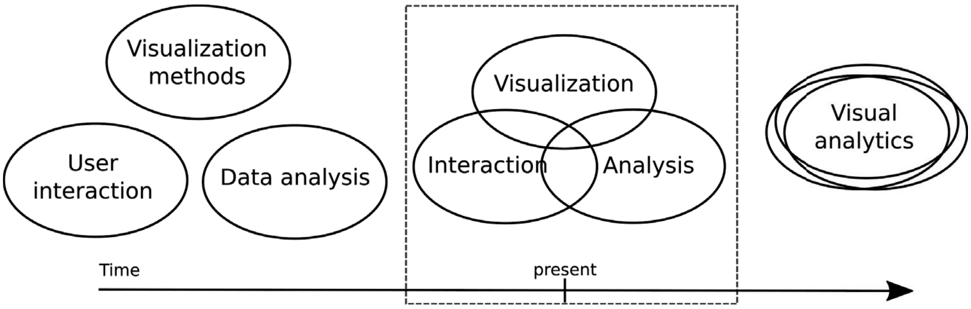


FIGURE 4 The present situation of visual analytics according to the VTT report [63].

able time frame. In these research tasks, success depends on the availability of the right information. Visual analytics aims at making data and information processing transparent and combines the strengths of humans and computers [72]. By definition, ”visual analytics combines automated analysis techniques with interactive visualizations for an effective understanding, reasoning and decision making on the basis of very large and complex datasets” [72].

Visual analytics was first developed for security monitoring and analysis after the events of 11 September 2001 [112]. The main focus was to create a scalable framework to analyze large amounts of information from different sources and to be able to develop a solution as quickly as possible [112]. The analytical reasoning process is supported by data analysis, visualizations, and user interaction [112, 72, 86, 96, 65]. Earlier, these aspects were separate research areas, but visual analytics tries to combine them (Figure 4). The visual analytics process utilizes the approach of the KD process and integrates user interaction and visualization into it via highly interactive GUIs (Figure 5).

Visual analytics is often related to very high-dimensional data. EDA provides many methods to work with many dimensions. Traditional data mining methods, for example classification and dimension reduction, are studied to process data in the application of visual analytics [87, 61, 24, 45, 91]. Clustering has also been used widely as a data mining method well suited for many tasks [25, 40, 19, 103, 66]. In visual analytics applications, data mining methods have usually developed from the basic forms to answer the challenges caused by the dynamic visualization interface [122, 21, 24, 27]. Visual analytics applications are often complex, and solving the problem requires the use of multiple data types. For the problems that require visual analytics, one of the most common variables is time (temporal data and time-series data) [119, 122, 81]. For certain types of tasks, statistical methods have been proven useful [4, 109, 119].

Visual analytics emphasizes the meaning of visualization in the analytical reasoning process [112, 72]. The visualization is traditionally done with histograms, graphs, scatter plots, and other 2D and 3D representations in which the numerical results are presented [39]. Some of the visualization methods are developed for a specific purpose, such as a variable relationship visualization [117, 118]. In the advanced visualization of multidimensional data, a common solution is to use parallel coordinates and different modified scatter plots in the

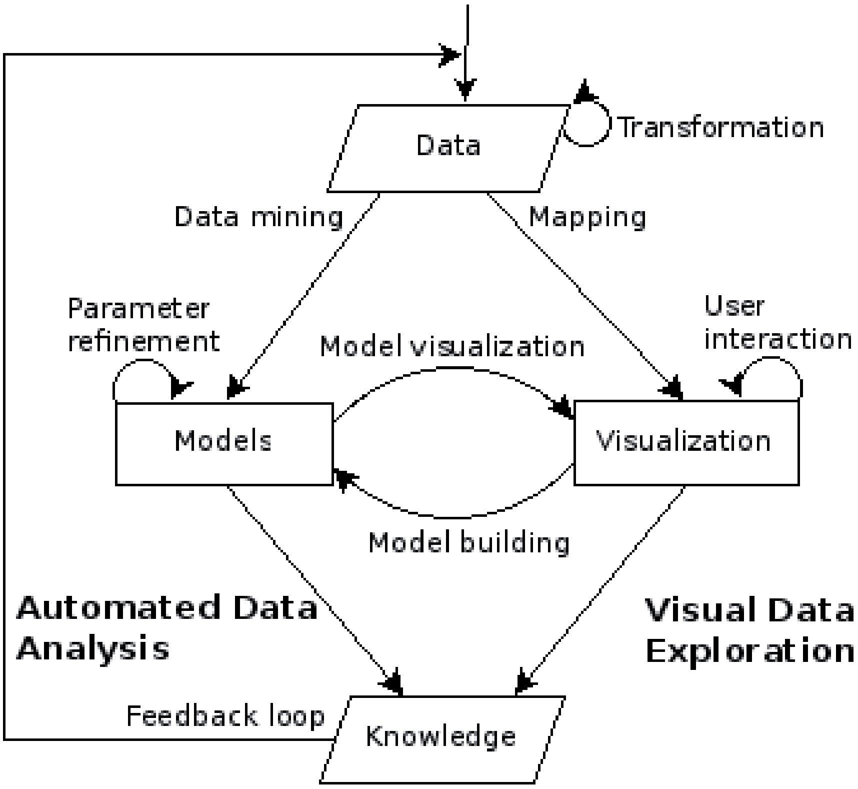


FIGURE 5 The visual analytics process according to Keim et al. [72]. Automated data analysis is shown on the left side, and visual data exploration is presented on the right side.

basic visualization of the data [21, 111]. Still, in visualizing high-dimensional data, data mining methods are often used to process the data suitable for 2D or 3D visualizations [1, 9]. The traditional way is to apply data mining to the data and then produce an interactive visualization to explore the data [83, 114, 96, 89]. However, in visual analytics, the results of data mining are not necessarily presented in a single figure but rather in more than one figure to provide different view points [94]. With interactive interfaces, the challenge is to integrate analysis with automated analysis and visualizations [111].

Typically, the interaction possibilities in visual analytics applications are numerous. Interaction in GUIs and other visualizations consists of common interactions, for example, select, explore, reconfigure, encode, and filter [72]. These resourceful interactions make it possible to apply the researcher's own knowledge to the analysis and to obtain a deeper understanding of the high-dimensional data [47, 120]. In many cases, a new GUI is developed to analyze a specific data [26, 67, 24]. To be able to analyze complicated data, the GUI can be made more complex. The influence of personality on interaction in the interface has been studied [52].

Visual analytics software and frameworks have been actively developed in the last few years [95, 19]. Recently, visual analytics has been applied in the area of energy analysis. Goodwin et al. modeled and visualized energy consumption at home [49]. However, examples of industrial applications of visual analytics are few. One possible reason for this might be that visual analytics requires the tight integration of software and algorithms; user interaction, data analysis, and visualization methods should make up a seamless analysis process [72]. Even if there are applications where users may input existing knowledge into the analysis via labeling [87, 106] and automatically generate visualizations based on context [60], in industrial applications, the meaning of meta data and contextual background should not be underestimated. Therefore, a novel context-sensitive framework tailored for industrial needs was developed in this study.

3.3 Context-sensitive approach

The concept of context sensitivity fits well with the idea of visual analytics. In visual analytics, the researcher is seen as a part of the knowledge mining process by contributing his/her background knowledge [72]. That background knowledge is important for a deeper understanding of the problem and is helpful in finding more reliable solutions, for example, understanding the content in document analysis [108]. Still, it is necessary to recognize and define the context before using it [12]. The sensing of contextual facts provides more information, which in this case is necessary for a proper and valid analysis [3].

Context sensitivity in the field of computing can be defined, as in [58], ”A computational method, a computer system, or an application is context-sensitive if it includes context-based functions and if it uses context to provide relevant information and services to the user, where relevancy depends on the user's situation." Regarding industrial applications, one could add that context sensitivity depends not only on the user's situation but also on the stage of the process and the state of the environment.

Context awareness is often used in relation to location awareness, mobile devices, and web-page document analysis where location and other sensor measurement information is part of the data [102, 20, 30, 57]. In this study, the term "context-sensitive" is used to refer to industrial process circumstances, as it gives more choices and sets fewer limitations.

In analysis cases, contextual information contains all meta data related to the test or measurement environment, software architecture, data processing, visualization, and end users, as well as to preliminary knowledge from the related application area. In a complete analysis chain, contextual information should be taken into account in every analysis phase and should be included as a part of automated processing. An industrial application, which for example uses burning biomass to produce heat and electricity, has a very specific context basis. Therefore, a general context-sensitive approach has been proposed, where the contextual background and other meta data are constantly considered during the analysis [PII, PIV]. This approach consists of three separate phases: sensing, processing, and actuation (SPA) (Figure 6(a)). The steps are explained as follows.

Sensing starts with obtaining all possible data related to the measurement environment and equipment, as well as background knowledge about the application area. The sensing phase is directly related to the application area and can be updated based on the changes in the environment. Data sensing in industrial processes comes down to measuring a process. Process measurements provide valuable information about the state of the process used to monitor and control industrial processes and for research and development purposes.

Processing describes the content of the preprocessing, transformation, and refinement aspects of the analysis. The context in this step is not directly associated with the application area. Therefore, the same set of computations is modifiable to the analysis of the same kind of data in different environments. Still, it is necessary to observe and use contextual background knowledge for processing, especially if the context changes in any previous phase. Data processing becomes immediately more challenging when time is a variable [88]. The challenge is that time adds dynamicity to the environment, meaning that the environment changes over time. While analyzing a dynamic system, it must be remembered that different contextual sources, for example the temporal context that includes the time stamp and date, can affect processing methods [57].

Actuation defines the actions based on the results in the processing phase. Different contexts, for example different user roles, will be selected according to the current situation. In the ultimate scenario, these contexts could be selected automatically based on the knowledge gathered from the user. In a dynamic setting, dynamic visualization methods are also needed. A key feature in visualizing dynamic data is visualization of the change in the process. Rapid changes in values are typically caused by events that are anomalous. Another key feature in

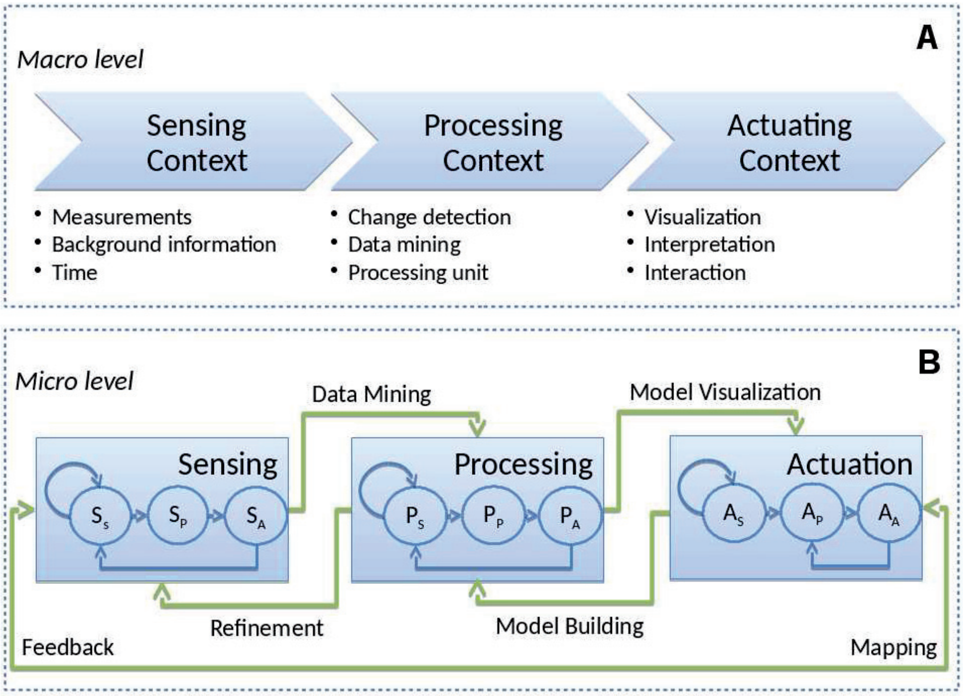


FIGURE 6 The framework of iterative SPA with elements of visual analytics. The framework was introduced in [PIV]. The top section (A) presents the general level context-sensitive approach on a macro level, whereas the bottom section (B) introduces the iterative SPA framework on a micro level.

visual analytics is the help it provides for the researcher or operator in making the big picture of the process understandable. In dynamic visualization, the importance of context sensing becomes emphasized. On one hand, it is important to get the different users to understand the process. On the other hand, the users with different user profiles should be able to use the system efficiently.

However, the SPA process framework may be too straightforward for real-world applications. Therefore, an iterative SPA approach was developed (Figure 6(b)) [PIV]. The iterative SPA framework is divided into two hierarchical levels. On the general macro level, the energy production process is analyzed as a straight-forward analysis process, from variable measurements to visualization and the interpretation of results. Different contexts are related to each SPA step and are considered in the overall analysis process. For example, the sensing context includes information about how and where measurements are taken, the processing context describes software analysis methods, and the actuating context defines the kind of visualization used.

By its nature, the SPA framework is scalable for different application areas. In this thesis, industrial processes are considered. In theory, all measurements should remain constant while using the same fuel. However, the environment keeps changing, which causes variability of the measurements. The possible reasons for a change can be, for instance, fouling and differently sized particles in

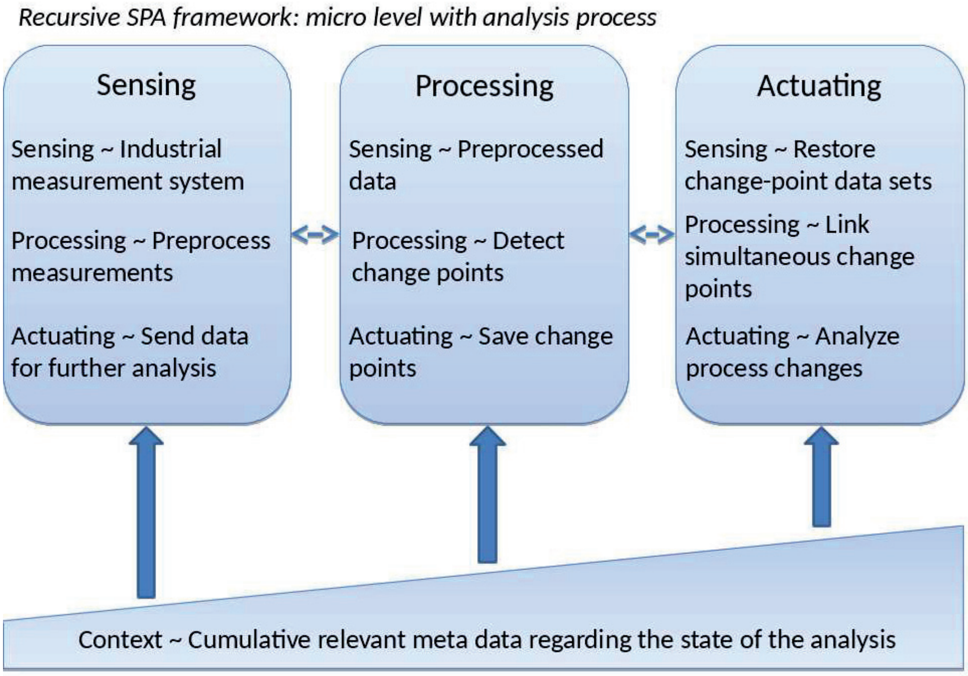

Recursive SPA framework: micro level with analysis process
Sensing
Sensing ~ Industrial measurement system
Processing ~ Preprocess measurements
Actuating ~ Send data for further analysis
<->
Processing
Sensing ~ Preprocessed data
Processing ~ Detect change points
Actuating ~ Save change points
<->
Actuating
Sensing ~ Restore change-point data sets
Processing ~ Link simultaneous change points
Actuating ~ Analyze process changes
Context ~ Cumulative relevant meta data regarding the state of the analysis

FIGURE 7 Cumulative addition of the meta data to the analysis process in different phases.

the bed material. For data processing, the challenge is to detect the significant changes among the general noise and variability. Contextual information plays an important role in reasoning and decision making. One example is a case where gas content measurements suddenly start varying more than usual between different measurement locations. This indicates that either something is wrong with the sensors or that something unusual is occurring in the process. With contextual background knowledge, the change could be spotted and explained with more accuracy. Also, in actuation the role of the end user may change during the analysis, for example if an operation fault has been detected. This should be reported to superiors without forgetting the earlier findings or the environment. Thus, the context in the actuation process changes, and the system should adapt to the new situation.

All in all, new meta data and contextual information is cumulatively added in each phase of the SPA process, which is illustrated in Figure 7. The increasing amount of meta data sets requirements for the software to integrate more information into the analysis. For example, setting labels and marking the data are ways for the end user to input additional information into the study [24].

4 VISUAL INSPECTION OF INDUSTRIAL TIME SERIES

In this chapter, the background of the tools and techniques used in this study are presented. The first section familiarizes the reader with the clustering method. Next, change-point detection is briefly explained from statistical and predictive points of view. The last section is about visual analysis and the GUI.

4.1 Data profiling using clustering

The goal in clustering is to find groups of objects such that the objects in one group are similar to one another and dissimilar to the objects in other groups, that is, the inter-cluster distances between objects in the same group are minimized [54, 6]. Han and Kamber classified clustering algorithms into five major categories, but in this study only two are relevant, the partitional and the hierarchical algorithms [54]. All partitional algorithms consist of the same expectation maximization (EM) kind of iterative relocation of prototypes, where, first, the closest prototype for each observation is detected and then the statistical estimate to determine the cluster representative is computed [16]. In a clean setting, this provides a disjoint division of the given data into subgroups, which are represented by the corresponding prototypes.

One of the oldest but still most popular partitional clustering algorithms is the *k-means* [41]. Its popularity is due to tis efficiency, simplicity, and scalability when clustering large data sets. However, there are many challenges related to *k*-means: i) one needs to detect the number of clusters *k* separately, ii) an appropriate initialization is needed because the iterative relocations necessitate a local search, and iii) it is prone to outliers and non-Gaussian errors in data due to the use of the mean as the statistical estimate for the prototypes. Related to the latter, the sparsity of any given data in the form of a missing value can be thought of as an ultimate outlier because any value (in the variable's value range) could be the one unavailable.

Therefore, second-order statistics that rely on the normally distributed error are not the best choice to define a prototype, especially with sparse data with missing values. Instead, one can and should use the so-called non-parametric, robust statistical techniques [59]. Two simple and robust location estimates are the median and the spatial median. The median, however, is univariate and utilizes only the available values of an individual variable. The spatial median, on the other hand, is truly a multidimensional location estimate and can take advantage of the pattern of available data as a whole. The spatial median has many attractive statistical properties. In particular, its breakdown point is 0.5, meaning that it can handle up to 50% of data contamination.

In [6], a robust approach utilizing the spatial median to cluster sparse and noisy data was introduced, namely the *k-spatialmedians* clustering algorithm. It minimizes the score (cluster error) function of the form locally:

$$\mathcal{J} = \sum_{k=1}^{K} \sum_{i \in I_k} \|\boldsymbol{P}_i(\boldsymbol{x}_i - \boldsymbol{c}_k)\|_2. \tag{1}$$

Here, I_k refers to the observations that are closest to the kth prototype c_k and the projections $\boldsymbol{P}_i, i = 1, \ldots, N$ specify the available values of the ith observation:

$$(\boldsymbol{P}_i)_j = \begin{cases} 1, \text{if } (\boldsymbol{x}_i)_j \text{ exists}, \\ 0, \text{otherwise}, \end{cases}$$

where j is the j^{th} value in p-dimensional object $x_i \in \mathbb{R}^p$.

In the actual realization of k-spatialmedians, the projected distance in (1) is used first, and the recomputation of the prototypes is based on the sequential over-relaxation (SOR) algorithm [6] with the overrelaxation parameter $\omega = 1.5$.

As explained above, a prototype-based clustering algorithm requires that the number of clusters corresponding to the number of prototypes be given. The so-called cluster indices measure the quality of the final result of a relocation algorithm in detecting k [53]. Generally, these methods take into account the clustering error (1) by combining it with the distance between the prototypes. For example, the Ray-Turi [99], the Davies-Bouldin, and the Davies-Bouldin* [77] indices are well known clustering evaluation indices.

The approach in hierarchical clustering differs from prototype-based partitional clustering. Hierarchical methods gradually merge observations (agglomerative) or divide superclusters (divisive) in order to optimize a score function [55]. For example, in one agglomerative clustering method, the observations are merged such that the minimum distance between each cluster is maximized. Hierarchical and partitional clustering methods often produce different results due the score function [6]. For instance, the purpose of the k-means method is to minimize the distance between the observations and the centroids, whereas Ward's method minimizes local variance within the clusters. The clustering error (1) measures the distance between the observation and the cluster center, and therefore the partitional methods often produce results with smaller error in the environmental setting used in this study.

The advantage of using clustering methods over traditional dimension reduction methods, such as principal component analysis (PCA) [64] in the transformation phase, is that clustering does not lose the original variable space. Therefore, which original time-series measurements are combined in the clustered prototype can be easily indicated. With time-series data in particular, these prototype variables have the same characteristic properties as the original measurements. For example, the changes in mean and variance are preserved within an acceptable temporal margin. Moreover, the principal components are oriented along the direction of the greatest variability of data [70] so that the direction of level changes of certain variables can be completely lost.

Clustering has been used in various domains and applications [33, 92, 35, 100, 78]. Recently, clustering methods have proved to be solid tools for visual analytics and data exploration [89, 14]. With dynamic data, for example multivariate time series, it is common to modify the original clustering algorithm to meet the challenges relating to the nature of temporal data [104, 80]. However, clustering sequences of time series have been noted to be pointless [74]. Clustering as part of the KD process is still important and is a good way of processing data streams [44].

4.2 Change-point detection

Change-point detection is the identification of abrupt changes in the generative parameters of sequential data [10]. Change-point detection is defined as the determination of those time stamps where statistical properties change significantly accordingly to some predefined criterion [10]. These change points are often considered points of interest in a data stream, especially in industrial data [44]. In practice, a change in the behavior of a measured value can indicate a change in the process, such as the temperature of the bubbling bed starting to rise suddenly. In order to assess the whole process, the change points of variables should be detected and analyzed. There are a few implementations of change-point detection algorithms used in the industry [71, 110]. Still, there are challenges in this domain [42].

The basic types of statistical change are based on first- and second-order statistics, that is, changes in the level or variability of the signal [10, 79]. The change might be instant (i.e., an abrupt change) or gradual (i.e., a trend). For example, a rapidly increasing variance may indicate a high risk of malfunction of a mechanical device, whereas slowly decreasing temperature in the superheaters is correlated with fouling in the furnace in the power plant. For change-point detection, typical solutions rely on a statistical model of data distribution [23, 82]. The challenge for real-world applications is that the probability function has to be estimated [76, 44]. The advantage is the speed and the simplicity of the detection after the parameters have been adapted for the application at hand. Cumulative sum control chart (CUSUM) is one of the commonly used methods for detecting level-based changes [23]. The challenge in using CUSUM in an industrial environment is its inability to detect multiple change points.

For time-series analysis and prediction, the most commonly used classical methods (autoregressive, AR; integrated, I; moving average, MA; both separately and together, or possibly with seasonal ARIMA, SARIMA) are composed of a class of linear models based on the Box-Jenkins methodology [15]. However, the behavior of modern industrial processes is non-linear (see, e.g., [116] and the articles therein). Using feedforward neural networks is one possible technique to capture and represent non-linear behavior [13, 56]. In particular, there are many recent examples of the use of the multilayered perceptron (MLP, aka back-propagation neural network, BPNN, or single hidden layer feedforward network, SLFN) neural network for time-series prediction (e.g., [121, 97, 34, 116, 31, 69]). Zhang and Kline [121] underline the importance of preprocessing, data preparation and transformation, and variable (feature) selection when using neural networks for seasonal time-series forecasting. In this direction, differencing was reported to be especially useful for the MLP in [97]. Still, applications with noisy real-world data require robust methods [68].

To apply change-point detection in visual analytics, the detection algorithm is wrapped as an automated routine, a detector. The detector detects change points from the time series in a moving window. In multivariate time series, one detector is run for each univariate time series. The principle behind a detector is as follows.

First, (univariate) time-series data is a sequence of indexed values

$$\mathbf{y} = \langle y(t) \rangle = \langle y(t_1), \ldots, y(t_l) \rangle,$$

where t_i, $i = 1, \ldots, l$, is the sampling time; l determines the number of points, that is, the length of the time series; and $\langle \cdot \rangle$ denotes an ordered row vector. A multiple (or multivariate) time-series of the n univariate time series with the same sampling rate can then be simply stored and represented by a matrix $\mathbf{Y} \in \mathbb{R}^{n \times l}$ with the individual time series in the rows. Typically, with industrial measurements in the same process, the different recordings can have different sampling frequencies and/or can represent different time intervals. Thus, to produce one time-series matrix $\mathbf{Y}$ for further analysis, temporal alignment and downsampling or oversampling are necessary.

Now for the detector, it is necessary to determine manually the window length w, where $w < l$. Then, let $I_j = \langle t_{j-w}, \ldots, t_j \rangle$ be a subset of time indices such that $t_j \in]w+1, l]$ and $\mathbf{y}' = \langle \mathbf{y}(t_{j-w}), \ldots, \mathbf{y}(t_j) \rangle$. Now, let $f(\mathbf{y}')$ be a statistical estimation method, for example, moving average or variance, applied to the window values $\mathbf{y}'$. Next, statistical change-point estimates are computed for each subinterval I_j as $s_j = s(t_j) = f(\mathbf{y}'(I_j)) - f(\mathbf{y}'(I_{j-1}))$. In this way, a vector containing the detector values $\mathbf{s} = \langle 0 \ldots s_j \rangle$ for $j = w+2, \ldots, l$ is achieved. The final binary change-point detection result is then computed with the threshold h as

$$\hat{\mathbf{s}} = \langle \hat{s}_j \rangle = \begin{cases} 1, & \text{if } s_j > h, \\ 0, & \text{otherwise,} \end{cases}$$

where the binary change-point matrix for matrix $\mathbf{Y}'$ is denoted as $\hat{\mathbf{S}} \in \mathbb{R}^{k \times l}$. For each different change-point detection method, that is, a detector, one matrix $\hat{\mathbf{S}}$ is produced, so at the end, there are m change-point matrices for m detectors. The threshold h is set for each $\hat{\mathbf{S}}$ separately. By default, the threshold is computed for 95% of the data mass.

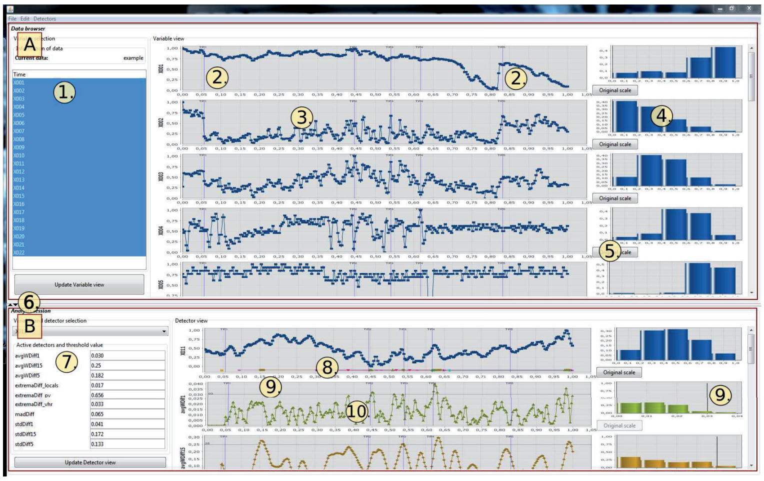
4.3 Visual data analysis

Motivated by the challenges and versatility of visual analytics (see Section 3.2), a visual framework, DebVA, was built for this study to visualize time-series variables focusing on the change points. The goal of variable visualization is to represent the absolute level of the variable and changes in relation to time. The user interaction and interactive visualizations have been examined during the design process [105, 72]. From a data analysis point of view, time is the only unifying factor for these data sets recorded with different equipment. Therefore, special requirements are set for both data preprocessing and data visualization.

MATLAB [85] was chosen as the computational engine of the framework because of its large and versatile base of methods and toolboxes and the weakly-typed core language. MATLAB has a collection of implemented algorithms, and development using it is often faster than with traditional languages. However, building a GUI in MATLAB is a challenge because of the visual analytics requirements for user interactions. It is also known that after a version upgrade in MATLAB, all the features may not function properly in old interfaces. Therefore, Java was chosen as the language for GUI development. A Java GUI was built using the Java Swing library [32], and visualizations use the JFreeChart library [46]. The main advantage of using a Java GUI on top of MATLAB is that all Java visualization tools can be combined into powerful data processing techniques in MATLAB.

The framework is mainly targeted to explore data measured in energy production processes related to power plants, but due to the modular interface, the same approach can be used for other application domains as well. One can specify and modify settings for the framework and implement new detectors to achieve the goals of different types of time series analysis tasks. More precisely, the modular approach provides tools for quick adaptation, for example, to include a new change-point detection method as a detector, thus allowing the comparison of different detectors. A screen shot of the GUI with its major features summarized is shown in Figure 8.

Generally, the GUI (Figure 8) is divided into two main panels that are aligned horizontally. The upper panel, *Data browser*, offers tools for analyzing time-series visually, and the lower panel, *Analysis session*, provides an interface to apply change-point detection and to adjust detector values. The sizes of the two main panels can be adjusted vertically with a slider (Number 6 in Figure 8), so that the user may concentrate only on relevant part of the analysis. All variable- and



Active detectors and threshold value

avgWDiff1	0.030
avgWDiff15	0.25
avgWDiff5	0.182
extremaDiff_locals	0.017
extremaDiff_pv	0.656
extremaDiff_vhr	0.033
madDiff	0.065
stdDiff1	0.041
stdDiff15	0.172
stdDiff5	0.133

A Time-series panel
1. Variable selection
2. Illustration of test point
3. Plot of variable
4. Histogram of variable
5. Launch individual variable explorer

6. Adjustable panel slider

B Detector panel
7. Detector threshold adjustment
8. Indication of found change points
9. Illustration of threshold and detector histogram
10. Time-series plot of detector

FIGURE 8 DebVA user interface with example data. In the upper panel (A), time-series data is visualized. Change points are analyzed in the lower panel (B).

detector-related selections are provided on the left-hand side, which then affects the visualizations in the center of the GUI. It should be noted that in both *Data browser* and *Analysis session*, the visualized time-series are scaled into $[0, 1]$. A user can launch a separate window with the original scale (Number 5 in Figure 8), where other relevant information like the current set of change points are also visualized similarly to the main GUI. Some options for visualizations and analysis, for example, the amount of bins in histograms and the window size of detectors, are global are can be changed from the menu bar of the GUI.

The main contribution of the graphical part of the framework is the ability to visualize time-series variables with additional (contextual) information. Therefore, the main focus of the GUI is reserved for graph visualizations. The main graphical elements are 2D charts where time series are visualized in time and a 1D histogram illustrating the discrete density distribution.

In all time-series plots, change points are visualized as colored markers below the corresponding item in time. Colored markers for change points follow the graphical outline of the detectors. Because all change points are aligned, one can use the interaction possibilities of the interactive interface to browse and compare different change-point detectors within the data set. In the detector visualization panel, each detector is plotted with a different color and marker combination.

Color design in a GUI plays an important role in successfully understanding and analyzing data. Colors can be used advantageously to highlight correct information, but the whole interface may become unusable if colors are used carelessly. In this framework, general guidelines GUI design have been followed. The color palette toolkit provided by NASA's Ames Research Center [93] is used for the color design of plots.

5 SUMMARY OF INCLUDED ARTICLES

5.1 Context-sensitive approach

5.1.1 Paper [PI]: Methods of Visual Analytics in Knowledge Mining

This article was published in 2011: J. Henno & Y. Kiyoki & T. Tokuda & N. Yoshida (Eds.), 21st European- Japanese Conference on Information Modelling and Knowledge Bases. Tallinn, Estonia, June 06–10, 2011, vol.1, pp.117–121.

Objective of this study

Visual analytics is a multidisciplinary field that combines many technological aspects with human technology. One objective is to show the difference between visual analytics and traditional data mining. Otherwise, a general review of the research and applicable methods are discussed.

Findings and contribution

In this article, an introduction to visual analytics in knowledge mining is given. When the data masses grow, special methods are required in the analysis. Dimension reduction methods are well studied in the area of data mining, but most can be applied to visual analytics with small modifications. Visualization and interaction are very important elements in visual analytics. The challenges will increase after the dimension of the data exceeds three. Processing is then required to be able to visualize the data. However, temporal data and online settings create another set of challenges.

5.1.2 Paper [PII]: Context-Sensitive Approach to Dynamic Visual Analytics of Energy Production Processes

This article was published in 2013: Peter Vojtáš & Y. Kiyoki & T. Tokuda & N. Yoshida (Eds.), Information Modelling and Knowledge Bases XXIV. IOS Press, Amsterdam, pp.15–22.

Objective of this study

Data masses require significant data processing. Traditional methods are limited in terms of usability features and easily understandable results. Based on the open research questions and challenges in the paper [PI] relating to industrial data analysis, the objective of this paper was to define and propose a context-sensitive approach. The analysis of the approach begins with the measurement procedures (sensing), followed by an analysis of the data (processing) and an analysis of the results (actuation). Another aspect of this approach is to take all meta data into account throughout the whole analysis process. In this article, the focus is on the field of energy production where biomass is utilized as a fuel.

Findings and contribution

In this paper, a three-phase SPA (Sensing, Processing, Actuation) analysis framework for analyzing real-world industrial cases is designed and proposed. It takes into account the whole process from the measurements to end user decision making. The first phase, sensing, describes the measurements (equipment, location, etc.) and the preprocessing of the data. Then, the computations, including dimension reduction, are performed for the preprocessed data in the processing phase. Finally, the numerical results are visualized and the obtained knowledge is utilized in the decision-making process. Background knowledge is utilized in all phases as contextual knowledge. A highly interactive user interface plays a significant role in this framework. At the end of this article, one practical example of the usage of the framework is given.

5.1.3 Paper [PIV]: Context-Sensitive Framework for Visual Analytics in Energy Production from Biomass

This article was published in 2015: B. Thalheim & H. Jaakkola & Y. Kiyoki & N. Yoshida (Eds.), Information Modelling and Knowledge Bases XXIV. IOS Press, Amsterdam, pp.449–456.

Objective of this study

The SPA framework proposed in paper [PII] was improved in this paper. It was discovered that the preliminary model was too limited and the process too straightforward. Here, an iterative SPA model to integrate more elements of visual analytics into the framework was introduced. In addition to visual analytics, the authors emphasize the importance of sensing contextual information and background knowledge. To be able to utilize this framework, a GUI to analyze industrial time series was developed. The framework was tested with real-world data collected from a BFB power plant.

Findings and contribution

After the research in [PII], the context-sensitive approach was improved with an iterative element. In this iterative context-sensitive proposal, the improvement is to be able to move back and forth between SPA phases. In addition, each of the phases are divided into internal SPA substeps. For example, if it is noted in the middle of the processing that there is too much noise, it can be backtracked to the previous phase, to apply noise filtering, and then the analysis can be continued. The idea is to utilize the context information, the meta data, in the analysis to obtain more knowledge from the data. To fulfill the requirements of the iterative framework, a GUI was developed to automate data processing and visualization. It was recognized that it is challenging to validate the results with real-world data, as there is not just one correct answer. The value and reliability of the computational methods are emphasized.

5.2 Tools for the analysis

5.2.1 Paper [PIII]: Region of Interest Detection Using MLP

This article was published in 2014: 22nd European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, Bruges, Belgium, April 23–25, 2014.

Objective of this study

The objective of this paper was to test the predictive change-point detection method. A novel technique to detect regions of interest in a time series as deviations from the characteristic behavior was proposed. The technique uses a reliably trained MLP neural network with detailed complexity management and cross-validation-based generalization assurance. This technique is tested with a simulated and a real data set (the latter found in the UCI Machine Learning repository).

Findings and contribution

In the field of industrial time series, neural networks have traditionally been utilized mainly for classification purposes. In this paper, an algorithm was proposed to detect deviation from the deterministic behavior of the time series data. The MLP neural network was adapted for this task with a reliable training method. Ten-fold cross-validation was used to assure proper generalization of the obtained MLP network. The simulated data showed reliable results and the change points were detected with high accuracy. The results for detecting multiple changes in the real data set were promising, regardless of the extra noise. The validation of the real data was challenging, as there is no absolutely correct answer. All in all, using neural networks for predictive change-point detection for time series is a promising novel approach.

5.2.2 Paper [PV]: Hierarchical, Prototype-based Clustering of Multiple Time Series with Missing Values

This article was published in 2014: 23rd European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, Bruges, Belgium, April 22–24 2015.

Objective of this study

In this paper, a technique is presented to divide a given set of multiple time series containing missing values into disjoint subsets. The subsets are stored in a dynamic tree-like structure generated from the results of the recursive usage of a robust clustering algorithm. The unsupervised clustering is applied here in a supervised setting, that is, with a set of multiple time series with given labeling. The experiment is run with the same data set used in paper [PIII] found in the UCI Machine Learning repository.

Findings and contribution

Clustering algorithms are divided into partitional and hierarchical methods. In the proposed algorithm, the hybrid clustering method is applied recursively to construct a tree-like structure. In the algorithm, a robust partitional clustering method, *k*-spatialmedians, was utilized. The number of clusters and the division of the data were detected using cluster evaluation indices in a mixture-of-experts fashion. The algorithm is applied recursively to the nodes until the stopping criterion is satisfied. The original data is stored in the leaf nodes of the generated tree. Combined with labeling information, each node represents a characteristic prototype of the data. Moreover, this algorithm was a first step towards a decision tree using recursive clustering for time series.

5.3 Analysis of industrial multivariate time series

5.3.1 Paper [PVI]: Improving Analysis of Particulates in Energy Production using Visual Analytics

This article was submitted to *Expert Systems*.

Objective of this study

While burning biomass as a fuel in a power plant, the behavior of the small particulates is not well known. In this paper, the computational methods and the analysis process itself are described with a real-world use case analyzing behavior of the fine particles. The paper introduces the phases of the approach to analyze industrial time series. The results from earlier papers are utilized in the analysis process. The fine particles are profiled with the clustering approach (proposed in [PV]), and the NeuroDetector introduced in [PIII] is applied for change-point detection. Also, the iterative framework and the graphical tool illustrated in the paper [PIV] form the basis of the visual analysis. With these tools, a sample of full-scale time-series data measured in a power plant is analyzed. The objective was to find cause-effect relationships between variables and particularly to find an explanation for the behavior of fine particles in the flue gas.

Findings and contribution

This study proposed an approach to analyze industrial multivariate time series data. As an application area, energy production provided many challenges. First, the measured real-world data is noisy, and different data sources are not directly comparable. The dimension of the measured data was high; therefore, there were challenges regarding visualization and understandability of the data. The presented solution emphasizes the preprocessing of the multivariate time-series data. The preprocessing consisted of the traditional preprocessing tasks and a novel way of clustering the time series as a whole. With vector quantization using clustering of the multivariate time-series data, the dimension of the data was vastly reduced, and certain profiles for the deterministic behavior of the data were generated. After the dimension reduction, the time series were visualized separately in the graphical tool where change-point detection was applied. The results of the change-point detection were loaded back into the graphical tool and the possible cause-effect relationships were considered. It was noted that the evaluation of the results was challenging because there are no absolutely correct answers. The results were validated by the experts in the field of energy production. The proposal gives a new point of view in terms of analyzing measured industrial time-series data.

5.4 Summary of the results

The results of the papers have contributed to two main aspects: general-level context-sensitive framework and application-level method development. The main results are summarized in Table 1. The obtained results are compared to the original research questions in Table 2. The context-sensitive framework is oriented as a general-level approach to solve research problems involving multiple data sources and a multidisciplinary research group. In this study, the target was industrial-oriented applications with time-series data. This application environment is defined as a new context in the early stages of the iterative context-sensitive framework. Therefore, the same model can be utilized in several differ-

TABLE 1 Summary of the results of included articles.

Obtained results	*General*	*Application*	*Paper No.*
(a) Theory base	State-of-the-art analysis		[PI]
(b) Context-sensitive approach	Design of analysis approach to energy production from biomass		[PII]
(c) Iterative SPA framework	Design of analysis framework in energy production from biomass		[PIV]
(d) NeuroDetector	Method application	Change-point detection	[PIII]
(e) Time-series profiling	Method development	Transformation of time series	[PV]
(f) Visual analysis process	Design of analysis process	Analysis of industrial time series	[PVI]

TABLE 2 Positioning of obtained results to each research question.

Research questions	*Obtained results*
RQ1	OR (a,c,e,f)
RQ2	OR (b,c)
RQ3	OR (a,b,c)
RQ4	OR (e,f)
RQ5	OR (d,f)

ent cases by modifying only the sensing part of the framework.

The specific domain in this study was the analysis of measured data from a BFB power plant using biomass as a fuel. To be able to reliably meet the objectives of this study using real-world time-series data, new tools were required. One was the application of a robust and reliable way of detecting abrupt changes from univariate time series. For this, an MLP neural network with robust training was applied. The second tool was developed for transformation of the multiple time series to form characteristic profiles of the deterministic behavior of the data. This time series vector quantization method was implemented using unsupervised *k*-means clustering in a supervised manner. The resulting profiles were combined with meta data that were used to construct a dynamic decision tree. In addition, a way of exploring the data and presenting the results in an understandable form was needed. For this, a graphical visualization framework was developed. Altogether, the individual results and the GUI, utilized in the manner described in the context-sensitive framework, answer the challenge of analyzing industrial cases with time series data.

6 DETECTOR-BASED VISUAL ANALYSIS FRAMEWORK

In order to further demonstrate and evaluate the proposed detector-based visual analysis framework (Figure 9), an example of the analysis chain is given. The familiar Dodgers data from the UCI Machine Learning repository [5] was used in this case (see the description of the data in [PIII, PV]). Based on the traffic sensor readings, the goal was to detect a match that occurred in the evening. The analysis chain presented in [PVI] was applied to Dodgers data. The preprocessing and transformation phases followed the same principles introduced in [PV] (Figure 9, Sensing). Thus, after the clustering, a dynamic tree structure containing the profiles of different traffic patterns was formed. For this study, only the traffic from weekdays was selected for consideration. In practice, in the first round, all weekday data was clustered in one cluster (122 days) and weekend data in the other cluster by the accuracy of 96%. The weekday traffic was then divided into six different clusters: one contained all matches scheduled in the evening (46 days), another contained all normal traffic with a few matches in the daytime (71 days including 4 daytime match days), and the rest contained inconsistent measurements or days that the sensor was offline. The prototypes were computed for these two main clusters as match day traffic and normal day traffic. The daytime match days were ignored as part of statistical variation.

Next, the match day traffic prototype was declared a test time series, the goal of which is to detect the match time. The usage of a known prototype is rationalized due to the validation of the results. The knowledge of the match time was not used in the training or in the detection of change points. In addition to these two time series, a different time series was computed in order to highlight the statistical deviations. The difference time series ($\mathbf{y}_2$) was computed simply by normalizing and reducing the prototype of normal traffic pattern ($\mathbf{y}_0$) from the test time series ($\mathbf{y}_1$), as follows:

$$\mathbf{y}_2 = \mathbf{y}_1 / max(\mathbf{y}_1) - \mathbf{y}_0 / max(\mathbf{y}_0).$$

These, time series ($\mathbf{y}_{\{0,1,2\}}$) were then combined with timestamps as a suit-

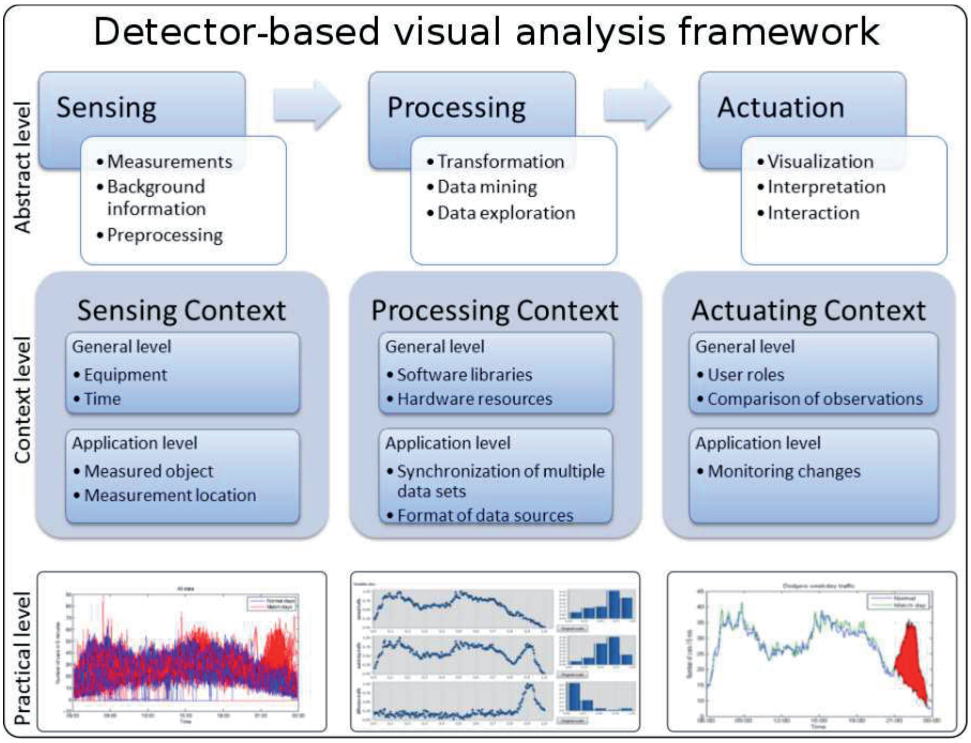


FIGURE 9 The detector-based visual analysis framework. The Dodgers data was used in the demonstration and evaluation of the framework. On the practical level, the raw data is first formatted to target data. The data was then transformed ready for analysis. In the last phase, the findings were interpreted and evaluated. The red area in the third figure on the practical level marks the pattern for detected match ending. This was detected from the traffic sensor readings illustrated in the first figure on the practical level.

able data set matrix (**Y**) for further analysis, which was done in the GUI (Figure 10). The produced three time-series prototypes are part of the processing phase of the framework (Figure 9, Processing). The NeuroDetector was trained for predictive change-point detection. The selection of training data was done based on the known meta data from the study environment. The goal was to predict if there is a match in the evening. Also, it is known from the history that the match typically starts at 19:00 and ends at 22:00 on weekdays. Therefore, the training data for the NeuroDetector was determined as a section of the normal traffic pattern prototype in the evening (i.e., traffic between 17:00 and 24:00).

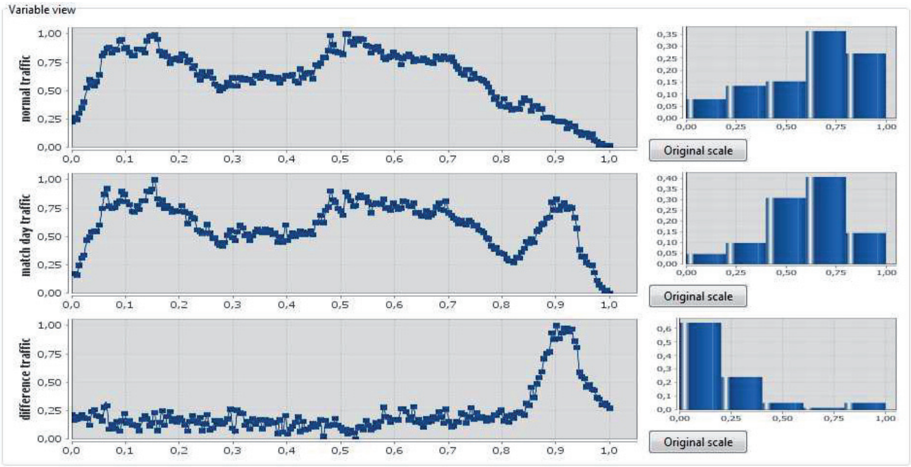
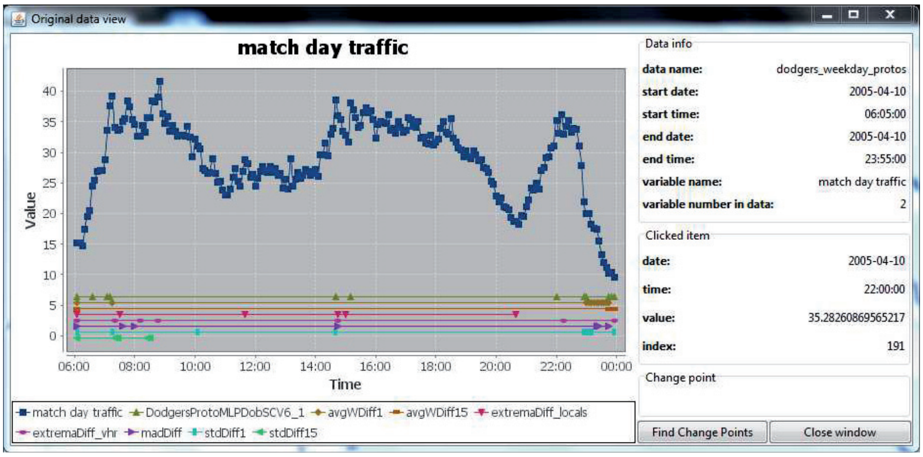


FIGURE 10 Visualization of Dodgers data: prototypes from weekday traffic. The normal traffic pattern is on the top, the test pattern with the evening match is in the middle, and the computed difference pattern is on the bottom.

The change-point detection was applied to the test time series. Unfortunately, the beginning of the game was not detected by any of the detectors. As can be seen from the difference pattern in Figure 10, there is no sign of deviation from the normal weekday traffic pattern when the match starts. The match traffic is mixed into the daily rhythm and routines. However, the end of the match was detected with the NeuroDetector at 22:00 (Figure 11). Please note that the knowledge of the average match time was not used in the detection, only in the validation of the detected change points. The NeuroDetector also detected the beginning of the normal daily routines (the increasing traffic around 07:00 and 15:00). The statistical detectors also detected changes, but the behavior was not as coherent.

The change points were also detected in the difference time series. In this case, it was pointless to apply the predictive NeuroDetector because the deterministic pattern of the difference time series did not follow the real traffic pattern. The statistical detectors were applied to the difference time series, and the results were relatively favorable (Figure 12). The first indication of a change in the traffic pattern was determined at 21:20. A clear change point was then detected with almost all detectors about 20 minutes later. The interpretation of the



Data info

data name:	dodgers_weekday_protos
start date:	2005-04-10
start time:	06:05:00
end date:	2005-04-10
end time:	23:55:00
variable name:	match day traffic
variable number in data:	2

Clicked item

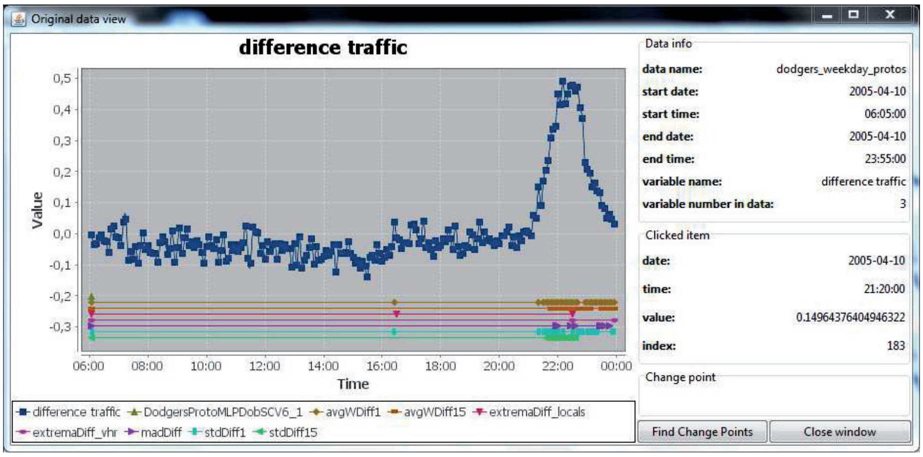
date:	2005-04-10
time:	22:00:00
value:	35.28260869565217
index:	191

Change point

FIGURE 11 Detailed view of the NeuroDetector used to test time series. In this test, it was known that the evening match typically starts at 19:00 and ends at 22:00.

detected region of change points is illustrated in the last phase of the framework (Figure 9, Actuation). It is noted that one false positive detection was made by a few detectors after 16:00.

As shown in these results, the suggested framework unifies the whole analysis process. The main findings on the practical level are that different detectors are suitable for different tasks and that preprocessing of the data is important. The statistical detectors find the statistical changes easily, whereas the predictive detector succeeds in detecting deviation in deterministic behavior. The preprocessing and transformation phases are again emphasized in the pursuit of reliable results. Using clustering in the dimension reduction of time series forms solid profiles for different scenarios. In addition, with fine tuning of the parameters, the suggested framework can be applied in online settings. For example, in the case of a new observation, the observation can be compared to each of the profiles to determine the closest prototype, or the trained detectors can be applied to the observation to find interesting regions or change points.



Data info

data name: dodgers_weekday_protos

start date: 2005-04-10

start time: 06:05:00

end date: 2005-04-10

end time: 23:55:00

variable name: difference traffic

variable number in data: 3

Clicked item

date: 2005-04-10

time: 21:20:00

value: 0.14964376404946322

index: 183

Change point

FIGURE 12 Detailed view of detected change points with different statistical detectors. Typically, the evening match starts at 19:00 and ends at 22:00.

7 DISCUSSION

In this section, the most essential findings related to the challenges of the research are discussed. The findings are:

- Temporal synchronization of separate data sets is a compromise.
- Dealing with similarly behaving variables (i.e., variables with high correlation).
- The difficulty of validating the results.
- A few ideas, how to improve the GUI.
- About the quality and reliability of the detectors.
- Using multiple detectors.
- Novelty of the work.

While analyzing real-time data, it should be noted that preprocessing is one of the most important steps in any successful case. Synchronization of multiple data sets with different, varying sampling frequencies is mostly manual work using MATLAB (or any other language). The question is always which sampling frequency should be chosen. If the lowest is selected, then there is a possibility that interesting details are missed. But if the fastest frequency is selected, the most suitable way of interpolating the slower data sets and whether they are valid for computations must be determined. Often in real-world scenarios, selecting the slowest frequency is a solid choice to produce reliable results that are not necessarily the fastest.

Ultimately, an overall framework for analyzing time-series data was obtained based on the results presented in the papers discussed. In the analysis of power plant data, there are often similarly behaving variables. From a computational point of view, the smaller number of variables, the better. Therefore, computing the profiles for certain types of phenomena in the analysis case is recommended and produces a valid result by itself. The presented clustering algorithm tracks the results of manually combining the variables quite well based on their location and type. In this way, this preprocessing step could be automated in certain scenarios.

On the other hand, in the analysis, the relationships between variables are difficult to validate. Similarly behaving variables may behave similarly because they react to the same input, or a change in one variable may influence another variable. Simply put, there are often no correct answers. The experts in the field have the best knowledge, and validation of the results is performed in discussion meetings. The addition of meta data to analyze the behavior of the variables supports the final interpretation and validation.

In order to utilize the iterative SPA framework, a GUI for MATLAB was developed that integrates an interactive user interface and visualizations with powerful automated computations. The development work is still in progress. One clear improvement would be to make context information more transparent to the researcher. However, preliminary work with real-world data provided by Valmet has given positive results.

Still, a very important aspect in the use of the GUI framework presented in this thesis is to validate the quality of the detectors. If detectors are not detecting change points correctly, the results are unreliable. It should be noted that one detector does not fit all types of variables. One detector works for noisy data, while another does not. Therefore, validating detectors with simulated data sets is not sufficient for reliable decision making. The results of each detector should be verified with data gathered from the real world using experts' opinions. One way to increase reliability is to add more detectors and vote for the change points. This is reliable due to statistical phenomena, but minor changes might be lost.

It was found during the analysis that one type of detector does not provide optimal results for all signals. We have a few differently tuned variants of the basic average and variance detectors, and it was clearly seen that some variants worked and others did not. With a different signal, the settings and some conclusions were different. This implies that a single statistical detector is not sufficient for diverse analysis.

All together, the results presented above were proven useful for both the industrial parties and to researcher. Even if there was no completely new method developed, the applications of the methods provided novel approaches. Probably the most beneficial result was the decision tree-like structure produced by hybrid clustering of time series. Its diverse application makes it possible to utilize the algorithm in, for example, clustering, classification, profiling of data, prediction, dimension reduction, and anomaly detection. Also, the produced GUI for MATLAB, together with the context-sensitive framework, is a key point for the future and provides some business value.

8 CONCLUSION AND FURTHER RESEARCH

The work in this thesis has been greatly influenced by the ideas of visual analytics. However, there are still challenges with visual analytics, especially in industrial applications. In this thesis, the context-sensitive approach, the analysis tools for visual inspection, and the detector-based visual analysis framework were introduced and applied in real-world scenarios. In the analysis of measured data from the real world, the meaning of preprocessing is emphasized and the use of robust analysis methods is necessary. The suggested framework can be adapted to different scenarios by fine tuning the parameters of the applied methods. The sensing phase depends on the scenario, whereas the processing and actuation phases are related to the research question and objectives.

The dynamic tree of the data generalization decreases the complexity of the data set but also gives profiles to certain phenomena traditionally measured with several variables. The tools and the framework have been introduced in the papers, but were also demonstrated and evaluated in this thesis. The results are difficult to validate because of the real-world scenario, but the results look promising to continue the research. After discussing with the experts the usability of this kind of approach to analyze energy production processes, it was agreed the idea was novel and would provide useful information in the analysis.

In the future, it is necessary to improve the context-sensitive SPA framework and the visual analysis approach. In visual analytics, the real-time results are one of the key factors. By using different detectors, particularly predictive detectors, the computation time might become a new challenge. One solution would be to parallelize the change-point detection either by detector or by variable. The most beneficial solution would be to parallelize the training of the MLP neural network for the NeuroDetector. This idea could be developed even further to divide all tasks into small subtasks. These could then be computed even faster in graphical processing units (GPUs). Computation in GPUs is an interesting research field that improves computing time significantly, as shown in [90].

One other improvement would be a proper detector evaluation and the application of several more detectors to obtain more reliable voting results. With numerous detectors, a challenge might be the delay in getting results due to different delays in the change-point detection. Computations are fast enough with a small amount of data, but the detection speed of the detectors varies. Figure 12 shows there are differences in detection speed between detectors. Another improvement would be to integrate meta data into the framework, making even more solid collaboration between data mining experts and application analysts possible.

Furthermore, the GUI is merely a prototype for such an analysis case. The visualizations, user interactions, and possibility of adding more meta data should be enhanced. At this stage, the GUI does not fully support views for different stakeholders. However, detailed information can be obtained from a variable in its original value space. Java offers versatile possibilities for GUI development, and the creation of context-dependent stakeholder profiles is achievable. However, this must be left for another study.

YHTEENVETO (FINNISH SUMMARY)

Tämä väitöskirja, *aikasarja-aineistojen detektoripohjainen visuaalinen analyysi*, keskittyy teollisten aikasarjojen analysointiin visuaalisesta analytiikasta ja tiedonlouhinnasta johdettujen tekniikoiden avulla. Tietokoneet murskaavat numeroita nopasti ja tehokkaasti, mutta ihmiset ovat loistavia päättelemään asiayhteyksiä. Visuaalisen analytiikan tutkimusala yhdistää visualisoinnin, data analyysin sekä ihmisen ja koneen välisen vuorovaikutuksen. Teollinen mittausprosessi, esimerkiksi voimalaitoskontekstissa, tuottaa suuren määrän monimuuttuja-aikasarja-aineistoa useista eri lähteistä. Mitattuja suureita ovat esimerkiksi pienhiukkaset, lämpötila, paine, kaasupitoisuus ja automatisoidut voimalan säädöt. Työn keskeisin sovellusalue on energian tuotanto biomassasta. On tunnettua, että biomassan käyttö on haastavaa sen sisältämien orgaanisten yhdisteiden vuoksi. Lisäksi energiantuotantoprosessi kokonaisuudessaan sisältää runsaasti analysoinnin kannalta merkittävää taustatietoa sekä kontekstitietoa (metadataa), joka tulee ottaa huomioon analyysin edetessä. Oman haasteensa tuo se, että kontekstitieto voi muuttua ajan suhteen ja sen määrä yleensä kasvaa prosessin edetessä. Tämän suuren datamassan syvemmän merkityksen ymmärtäminen on vaativa tehtävä sekä ihmisille että tietokoneille.

Tässä väitöskirjassa kehitettiin visuaaliseen analytiikkaan perustuva kontekstiherkkä viitekehys, jossa huomioidaan koko analyysiprosessi mittauksista loppuanalyysiin. Viitekehys koostuu teoreettisesta kontekstiherkästä lähestymistavasta, jossa määritellään prosessin eri vaiheet ja niiden toimenpiteet, sekä käytännönläheisestä menetelmäkehityksestä, jossa sovelletaan robusteja tiedonlouhinta- sekä muutospisteiden havaitsemismenetelmiä aikasarja-aineistoon. Näihin sovellettuihin menetelmiin kuuluvat tilastolliset ja ennustavat muutospisteiden havaitsemismenetelmät sekä klusterointimenetelmät aikasarjojen profilointiin. Työssä esitellyt menetelmät ja algoritmit muodostavat työkaluja, joita on testattu ja sovellettu reaalimaailman skenaarioissa. Esimerkiksi, koko analyysiketjua sovellettiin tapaukseen, jossa analysoitiin biomassan käytöstä johtuvaa pienhiukkasten muodostumisen suhdetta voimalan automatisoituihin säätöihin. Oheisissa tutkimuksissa nousi esille esikäsittelyn tärkeys sekä visuaalisen analyysin mahdollisuudet. Työn osatuloksena kehitettiin graafinen käyttöliittymä, jonka avulla voidaan käyttää automatisoituja aikasarjojen analyysimenetelmiä. Käyttöliittymä noudattaa määritellyn kontekstiherkän viitekehyksen periaatteita. Lopullisena yhteenkokoavana tuloksena demonstroitiin ja arvioitiin detektoripohjaista visuaalisen analyysin viitekehystä liikennevirrasta mitatun aikasarjan avulla.

REFERENCES

[1] G. Albuquerque, M. Eisemann, D. J. Lehmann, H. Theisel, and M. Magnor. Improving the visual analysis of high-dimensional datasets using quality measures. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST)*, pages 19–26, 2010.

[2] E. Alpaydin. *Introduction to Machine Learning*. The MIT Press, Cambridge, MA, USA, 2nd edition, 2010.

[3] Z. Alshaikh and C. Boughton. Notes on synthesis of context between engineering and social science. In *Modeling and Using Context*, pages 157–170. Springer, 2013.

[4] G. Andrienko, N. Andrienko, M. Mladenov, M. Mock, and C. Pölitz. Discovering bits of place histories from people's activity traces. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST)*, pages 59–66, 2010.

[5] A. Asuncion and D. Newman. UCI machine learning repository, 2007.

[6] S. Äyrämö. *Knowledge Mining Using Robust Clustering*, volume 63 of *Jyväskylä Studies in Computing*. University of Jyväskylä, 2006.

[7] L. S. Bäfver, M. Rönnbäck, B. Leckner, F. Claesson, and C. Tullin. Particle emission from combustion of oat grain and its potential reduction by addition of limestone or kaolin. *Fuel Processing Technology*, 90(3):353 – 359, 2009.

[8] J. Bakker, M. Pechenizkiy, I. Žliobaitė, A. Ivannikov, and T. Kärkkäinen. Handling outliers and concept drift in online mass flow prediction in cfb boilers. In *Proceedings of the Third International Workshop on Knowledge Discovery from Sensor Data*, SensorKDD '09, pages 13–22, New York, NY, USA, 2009. ACM.

[9] S. Barlowe, T. Zhang, Y. Liu, J. Yang, and D. Jacobs. Multivariate visual explanation for high dimensional datasets. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST) '08*, pages 147–154, 2008.

[10] M. Basseville and I. V. Nikiforov. *Detection of Abrupt Changes: Theory and Application*. Prentice Hall, 1993. change point detection.

[11] P. Basu. *Combustion and Gasification in Fluidized Beds*. CRC Press, 2006.

[12] M. Bazire and P. Brézillon. Understanding context before using it. In *Modeling and using context*, pages 29–40. Springer, 2005.

[13] C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, Inc., New York, NY, USA, 1995.

[14] A. Bouguettaya, Q. Yu, X. Liu, X. Zhou, and A. Song. Efficient agglomerative hierarchical clustering. *Expert Systems with Applications*, 42(5):2785 – 2797, 2015.

[15] G. E. Box, G. M. Jenkins, and G. C. Reinsel. *Time Series Analysis: Forecasting and Control*, volume 4th edition. John Wiley & Sons, 2008.

[16] P. Bradley, U. Fayyad, and C. Reina. Clustering very large databases using em mixture models. In *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, volume 2, pages 76–80 vol.2, 2000.

[17] Y. Cai, R. d. M. Franco, and M. García-Herranz. Visual latency-based interactive visualization for digital forensics. *Journal of Computational Science*, 1(2):115 – 120, 2010.

[18] A. Calvo, L. Tarelho, E. Teixeira, C. Alves, T. Nunes, M. Duarte, E. Coz, D. Custodio, A. Castro, B. Artiñano, and R. Fraile. Particulate emissions from the co-combustion of forest biomass and sewage sludge in a bubbling fluidised bed reactor. *Fuel Processing Technology*, 114(0):58 – 68, 2013.

[19] J. A. Castellanos-Garzón, C. A. García, P. Novais, and F. Díaz. A visual analytics framework for cluster analysis of dna microarray data. *Expert Systems with Applications*, 40(2):758–774, 2013.

[20] S. Ceri, F. Daniel, M. Matera, and F. M. Facca. Model-driven development of context-aware web applications. *ACM Transactions on Internet Technology*, 7(1), Feb. 2007.

[21] S. Chan, L. Xiao, J. Gerth, and P. Hanrahan. Maintaining interactivity while exploring massive time series. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST) '08*, pages 59–66, 2008.

[22] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Computing Surveys*, 41(3):15:1–15:58, July 2009.

[23] N. Cheifetz, A. Samé, P. Aknin, and E. de Verdalle. A cusum approach for online change-point detection on curve sequences. *Computational Intelligence and Machine Learning, Bruges (Belgium)*, pages 25–27, 2012.

[24] Y. Chen, S. Barlowe, and J. Yang. Click2annotate: Automated insight externalization with rich semantics. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST)*, pages 155–162, 2010.

[25] J. Choo, S. Bohn, and H. Park. Two-stage framework for visualization of clustered high dimensional data. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST) 2009*, pages 67–74, 2009.

[26] J. Choo, H. Lee, J. Kihm, and H. Park. iVisClassifier: An interactive visual analytics system for classification based on supervised dimension reduction. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST)*, pages 27–34, 2010.

[27] C. D. Correa, Y. Chan, and K. Ma. A framework for uncertainty-aware visual analytics. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST) 2009*, pages 51–58, 2009.

[28] G. D. Crnkovic. Constructive research and info-computational knowledge generation. In L. Magnani, W. Carnielli, and C. Pizzi, editors, *Model-Based Reasoning in Science and Technology*, volume 314 of *Studies in Computational Intelligence*, pages 359–380. Springer Berlin Heidelberg, 2010.

[29] Dekati Ltd. Home page. http://www.dekati.com, 2012.

[30] A. K. Dey. Understanding and using context. *Personal Ubiquitous Comput.*, 5(1):4–7, Jan. 2001.

[31] W. Du, S. Y. S. Leung, and C. K. Kwong. Time series forecasting by neural networks: A knee point-based multiobjective evolutionary algorithm approach. *Expert Systems with Applications*, 41(18):8049 – 8061, 2014.

[32] R. Eckstein, M. Loy, and D. Wood. *Java Swing*. O'Reilly, Beijing, 2. edition, 2002.

[33] B. S. Everitt, S. Landau, and M. Leese. *Cluster Analysis*. Wiley Publishing, 4th edition, 2009.

[34] X. Fan, S. Li, and L. Tian. Chaotic characteristic identification for carbon price and an multi-layer perceptron network prediction model. *Expert Systems with Applications*, 42(8):3945 – 3952, 2015.

[35] U. Fayyad, D. Haussler, and P. Stolorz. Mining scientific data. *Communications of the ACM*, 39(11):51–57, Nov. 1996.

[36] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. From data mining to knowledge discovery in databases. *AI Magazine*, 17(3):37, 1996.

[37] U. M. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. Advances in knowledge discovery and data mining. chapter From Data Mining to Knowledge Discovery: An Overview, pages 1–34. American Association for Artificial Intelligence, Menlo Park, CA, USA, 1996.

[38] U. M. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. Knowledge discovery and data mining: Towards a unifying framework. In *KDD*, volume 96, pages 82–88, 1996.

[39] U. M. Fayyad, A. Wierse, and G. G. Grinstein. *Information visualization in data mining and knowledge discovery*. Morgan Kaufmann, 2002.

[40] B. J. Ferdosi, H. Buddelmeijer, S. Trager, M. H. F. Wilkinson, and J. B. T. M. Roerdink. Finding and visualizing relevant subspaces for clustering high-dimensional astronomical data using connected morphological operators. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST)*, pages 35–42, 2010.

[41] E. W. Forgy. Cluster analysis of multivariate data: efficiency versus interpretability of classifications. *Biometrics*, 21:768–769, 1965.

[42] L. Fortuna, S. Graziani, A. Rizzo, and M. G. Xibilia. *Soft Sensors for Monitoring and Control of Industrial Processes*. Spronger, 2007.

[43] B. Frénay and M. Verleysen. Classification in the presence of label noise: a survey. *IEEE Transactions on Neural Networks and Learning Systems*, 25(5):845–869, 2014.

[44] J. Gama. *Knowledge Discovery from Data Streams*. Chapman and Hall / CRC Data Mining and Knowledge Discovery Series. CRC Press, 2010.

[45] S. Garg, I. V. Ramakrishnan, and K. Mueller. A visual analytics approach to model learning. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST)*, pages 67–74, 2010.

[46] D. Gilbert. Jfreechart. http://www.jfree.org/, 2000.

[47] M. Gleicher. Explainers: Expert explorations with crafted projections. *Visualization and Computer Graphics, IEEE Transactions on*, 19(12):2042–2051, Dec 2013.

[48] C. Glymour, D. Madigan, D. Pregibon, and P. Smyth. Statistical themes and lessons for data mining. *Data Mining and Knowledge Discovery*, 1(1):11–28, 1997.

[49] S. Goodwin, J. Dykes, S. Jones, I. Dillingham, G. Dove, A. Duffy, A. Kachkaev, A. Slingsby, and J. Wood. Creative user-centered visualization design for energy analysts and modelers. *Visualization and Computer Graphics, IEEE Transactions on*, 19(12):2516–2525, 2013.

[50] T. Górecki and M. Łuczak. Multivariate time series classification with parametric derivative dynamic time warping. *Expert Systems with Applications*, 42(5):2305 – 2312, 2015.

[51] C. W. Granger. Some properties of time series data and their use in econometric model specification. *Journal of Econometrics*, 16(1):121–130, 1981.

[52] T. M. Green and B. Fisher. Towards the personal equation of interaction: The impact of personality factors on visual analytics interface interaction. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST)*, pages 203–210, 2010.

[53] M. Halkidi, Y. Batistakis, and M. Vazirgiannis. On clustering validation techniques. *Journal of Intelligent Information Systems*, 17(2-3):107–145, 2001.

[54] J. Han, M. Kamber, and J. Pei. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 3rd edition, 2011.

[55] D. J. Hand, H. Mannila, and P. Smyth. *Principles of data mining*. MIT press, 2001.

[56] S. Haykin. *Neural Networks: A Comprehensive Foundation (3rd Edition)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2007.

[57] A. Heimbürger, Y. Kiyoki, T. Kärkkäinen, E. Gilman, K.-S. Kim, and N. Yoshida. On context modelling in systems and applications development. In *Proceedings of the 2011 Conference on Information Modelling and Knowledge Bases XXII*, pages 396–412, Amsterdam, The Netherlands, The Netherlands, 2011. IOS Press.

[58] A. Heimbürger, M. Nurminen, T. Venäläinen, and S. Kinnunen. Modelling contexts in cross-cultural communication environments. In *Proceedings of the 2011 Conference on Information Modelling and Knowledge Bases XXII*, pages 301–311, Amsterdam, The Netherlands, The Netherlands, 2011. IOS Press.

[59] T. P. Hettmansperger and J. W. McKean. *Robust nonparametric statistical methods*. Edward Arnold, London, 1998.

[60] J. Hullman, N. Diakopoulos, and E. Adar. Contextifier: Automatic generation of annotated stock visualizations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 2707–2716, New York, NY, USA, 2013. ACM.

[61] S. Ingram, T. Munzner, V. Irvine, M. Tory, S. Bergner, and T. Möller. Dimstiller: Workflows for dimensional analysis and reduction. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST)*, pages 3–10, 2010.

[62] A. Ivannikov, M. Pechenizkiy, J. Bakker, T. Leino, M. Jegoroff, T. Kärkkäinen, and S. Äyrämö. Online mass flow prediction in cfb boilers. In *Proceedings of the 9th Industrial Conference on Advances in Data Mining. Applications and Theoretical Aspects*, ICDM '09, pages 206–219, Berlin, Heidelberg, 2009. Springer-Verlag.

[63] P. Järvinen, K. Puolamäki, P. Siltanen, and M. Ylikerälä. Visual analytics. Final report. Technical report, VTT Technical Research Centre of Finland, 2009.

[64] I. Jolliffe. *Principal component analysis*. Springer-Verlag New York, 2002.

[65] N. Kadivar, V. Chen, D. Dunsmuir, E. Lee, C. Qian, J. Dill, C. Shaw, and R. Woodbury. Capturing and supporting the analysis process. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST) 2009*, pages 131–138, 2009.

[66] E. Kandogan. Just-in-time annotation of clusters, outliers, and trends in point-based data visualizations. In *Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on*, pages 73–82, Oct 2012.

[67] Y. Kang, C. Gorg, and J. Stasko. Evaluating visual analytics systems for investigative analysis: Deriving design principles from a case study. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST) 2009*, pages 139–146, 2009.

[68] T. Kärkkäinen and E. Heikkola. Robust formulations for training multilayer perceptrons. *Neural Computation*, 16:837–862, 2004.

[69] T. Kärkkäinen, A. Maslov, and P. Wartiainen. Region of interest detection using mlp. In *22nd European Symposium on Artificial Neural Networks*, 2014.

[70] T. Kärkkäinen and M. Saarela. Robust principal component analysis of data with missing values. *To appear in the Proceedings of the 11th International Conference on Machine Learning and Data Mining MLDM*, 2015.

[71] Y. Kawahara and M. Sugiyama. Change-point detection in time-series data by direct density-ratio estimation. In *Proceedings of the 2009 SIAM International Conference on Data Mining*, volume 4, pages 389–400. SIAM, 2009.

[72] D. A. Keim, J. Kohlhammer, G. Ellis, and F. Mansmann. *Mastering The Information Age - Solving Problems with Visual Analytics*. Eurographics, November 2010.

[73] D. A. Keim, F. Mansmann, and J. Thomas. Visual analytics: How much visualization and how much analytics? *SIGKDD Explor. Newsl.*, 11(2):5–8, May 2010.

[74] E. Keogh and J. Lin. Clustering of time-series subsequences is meaningless: implications for previous and future research. *Knowledge and Information Systems*, 8:154–177, 2004.

[75] E. Keogh, S. Lonardi, and B. Y. Chiu. Finding surprising patterns in a time series database in linear time and space. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pages 550–556, New York, NY, USA, 2002. ACM.

[76] D. Kifer, S. Ben-David, and J. Gehrke. Detecting change in data streams. In *Proceedings of the Thirtieth International Conference on Very Large Data Bases - Volume 30*, VLDB '04, pages 180–191. VLDB Endowment, 2004.

[77] M. Kim and R. Ramakrishna. New indices for cluster validity assessment. *Pattern Recognition Letters*, 26(15):2353–2363, 2005.

[78] A. Kitamoto. Spatio-temporal data mining for typhoon image collection. *Journal of Intelligent Information Systems*, 19(1):25–41, July 2002.

[79] M. Lavielle. Using penalized contrasts for the change-point problem. *Signal Processing*, 85(8):1501 – 1510, 2005.

[80] T. W. Liao. Clustering of time series data - a survey. *Pattern Recognition*, 38:1857–1874, 2005.

[81] Z. Liao, Y. Yu, and B. Chen. Anomaly detection in gps data based on visual analytics. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST)*, pages 51–58, 2010.

[82] S. Liehr, K. Pawelzik, J. Kohlmorgen, S. Lemm, and K.-R. Müller. Hidden markov gating for prediction of change points in switching dynamical systems. In *Proceedings of the European Symposium on Artificial Neural Networks*, pages 405–410. Citeseer, 1999.

[83] D. M. Maniyar and I. T. Nabney. Visual data mining using principled projection algorithms and information visualization techniques. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '06, pages 643–648, New York, NY, USA, 2006. ACM.

[84] F. Mansmann, F. Fischer, D. A. Keim, and S. C. North. Visual support for analyzing network traffic and intrusion detection events using treemap and graph representations. In *Proceedings of the Symposium on Computer Human Interaction for the Management of Information Technology*, CHiMiT '09, pages 3:19–3:28, New York, NY, USA, 2009. ACM.

[85] MATLAB. *version 7.11.0.584 (R2010b)*. The MathWorks Inc., 2010.

[86] S. Mehta, S. Parthasarathy, and R. Machiraju. Visual exploration of spatio-temporal relationships for scientific data. In *Proceedings of the IEEE Symp Visual Analytics Science And Technology On*, pages 11–18, 2006.

[87] M. Migut and M. Worring. Visual exploration of classification models for risk assessment. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST)*, pages 11–18, 2010.

[88] T. Mitsa. *Temporal Data Mining*. Chapman & Hall/CRC, 1st edition, 2010.

[89] E. Müller, I. Assent, R. Krieger, T. Jansen, and T. Seidl. Morpheus: interactive exploration of subspace clustering. In *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '08, pages 1089–1092, New York, NY, USA, 2008. ACM.

[90] M. Myllykoski, T. Rossi, and J. Toivanen. Fast Poisson solvers for graphics processing units. In P. Manninen and P. Öster, editors, *Applied Parallel and Scientific Computing*, volume 7782 of *Lecture Notes in Computer Science*, pages 265–279. Springer Berlin Heidelberg, 2013.

[91] S. A. Najim and I. S. Lim. Trustworthy dimension reduction for visualization different data sets. *Information Sciences*, 278(0):206 – 220, 2014.

[92] E. J. Nam, Y. Han, K. Mueller, A. Zelenyuk, and D. Imre. Clustersculptor: A visual analytics tool for high-dimensional data. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST) 2007*, pages 75–82, 2007.

[93] NASA Ames Research Center. Using color in information display graphics. http://colorusage.arc.nasa.gov/, 2013.

[94] B. Nouanesengsy, S.-C. Seok, H.-W. Shen, and V. J. Vieland. Using projection and 2d plots to visually reveal genetic mechanisms of complex human disorders. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST) 2009*, pages 171–178, 2009.

[95] K. Ooms, G. L. Andrienko, N. V. Andrienko, P. D. Maeyer, and V. Fack. Analysing the spatial dimension of eye movement data using a visual analytic approach. *Expert Systems with Applications*, 39(1):1324–1332, 2012.

[96] N. Pelekis, G. Andrienko, N. Andrienko, I. Kopanakis, G. Marketos, and Y. Theodoridis. Visually exploring movement data via similarity-based analysis. *Journal of Intelligent Information Systems*, 38(2):343–391, 2012.

[97] M. Qi and G. P. Zhang. Trend time–series modeling and forecasting with neural networks. *Neural Networks, IEEE Transactions on*, 19(5):808–816, 2008.

[98] R. Raiko, I. Kurki-Suonio, J. Saastamoinen, and M. Hupa. *Poltto ja palaminen*. International Flame Reasearch Foundation - Suomen kansallinen osasto, 1995.

[99] S. Ray and R. H. Turi. Determination of number of clusters in k-means clustering and application in colour image segmentation. In *Proceedings of the 4th international conference on advances in pattern recognition and digital techniques*, pages 137–143, 1999.

[100] D. M. Rocke and J. Dai. Sampling and subsampling for cluster analysis in data mining: With applications to sky survey data. *Data Mining and Knowledge Discovery*, 7:7–215, 2003.

[101] K. Salmenoja. *Field and Laboratory Studies on Chlorine-induced Superheater Corrosion in Boilers Fired with Biofuels*. Report - Åbo Akademi. Process Chemistry Group. Kirjapaino Hermes, 2000.

[102] B. Schilit, N. Adams, and R. Want. Context-aware computing applications. In *Mobile Computing Systems and Applications, 1994. WMCSA 1994. First Workshop on*, pages 85–90, 1994.

[103] T. Schreck, J. Bernard, T. Tekusova, and J. Kohlhammer. Visual cluster analysis of trajectory data with interactive kohonen maps. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST) '08*, pages 3–, 2008.

[104] A. P. Serra and L. E. Zárate. Characterization of time series for analyzing of the evolution of time series clusters. *Expert Systems with Applications*, 42(1):596 – 611, 2015.

[105] S. B. Shneiderman and C. Plaisant. *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Pearson Addison Wesley, 5 edition, 2005.

[106] Y. B. Shrinivasan, D. Gotzy, and J. Lu. Connecting the dots in visual analysis. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST) 2009*, pages 123–130, 2009.

[107] J. Silvennoinen and M. Hedman. Co-firing of agricultural fuels in a full-scale fluidized bed boiler. *Fuel Processing Technology*, 2011.

[108] J. Stasko, C. Görg, and Z. Liu. Jigsaw: supporting investigative analysis through interactive visualization. *Information Visualization*, 7(2):118–132, 2008.

[109] C. A. Steed, J. E. Swan, T. J. Jankun-Kelly, and P. J. Fitzpatrick. Guided analysis of hurricane trends using statistical processes integrated with interactive parallel coordinates. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST) 2009*, pages 19–26, 2009.

[110] M. Sugiyama, T. Suzuki, S. Nakajima, H. Kashima, P. Bünau, and M. Kawanabe. Direct importance estimation for covariate shift adaptation. *Annals of the Institute of Statistical Mathematics*, 60(4):699–746, December 2008.

[111] A. Tatu, G. Albuquerque, M. Eisemann, J. Schneidewind, H. Theisel, M. Magnor, and D. Keim. Combining automated analysis and visualization techniques for effective exploration of high-dimensional data. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST) 2009*, pages 59–66, Oct 2009.

[112] J. J. Thomas and K. A. Cook. *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. National Visualization and Analytics Ctr, 2005.

[113] J. Tukey. *Exploratory data analysis*. Addison-Wesley, 1977.

[114] J. J. Valdés, E. Romero, and A. J. Barton. Data and knowledge visualization with virtual reality spaces, neural networks and rough sets: Application to cancer and geophysical prospecting data. *Expert Systems with Applications*, 39(18):13193 – 13201, 2012.

[115] Valmet Oy. Home page. http://www.valmet.com, 2015.

[116] L. Wang, Y. Zeng, and T. Chen. Back propagation neural network with adaptive differential evolution algorithm for time series forecasting. *Expert Systems with Applications*, 42(2):855 – 863, 2015.

[117] C. Weaver. Multidimensional visual analysis using cross-filtered views. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST) '08*, pages 163–170, 2008.

[118] C. Weaver. Multidimensional data dissection using attribute relationship graphs. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST)*, pages 75–82, 2010.

[119] K. Wongsuphasawat and B. Shneiderman. Finding comparable temporal categorical records: A similarity measure with an interactive visualization. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST) 2009*, pages 27–34, 2009.

[120] J. Yi, Y. Kang, J. Stasko, and J. Jacko. Toward a deeper understanding of the role of interaction in information visualization. *Visualization and Computer Graphics, IEEE Transactions on*, 13(6):1224–1231, Nov 2007.

[121] G. P. Zhang and D. M. Kline. Quarterly time-series forecasting with neural networks. *Neural Networks, IEEE Transactions on*, 18(6):1800–1814, Nov 2007.

[122] H. Ziegler, M. Jenny, T. Gruse, and D. A. Keim. Visual market sector analysis for financial time series data. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST)*, pages 83–90, 2010.

ORIGINAL PAPERS

PI

METHODS OF VISUAL ANALYTICS IN KNOWLEDGE MINING

by

Pekka Wartiainen, Tommi Kärkkäinen, Anneli Heimbürger, and Sami Äyrämö

2011

21st European-Japanese Conference on Information Modelling and Knowledge Bases, pp. 117–121

PII

CONTEXT-SENSITIVE APPROACH TO DYNAMIC VISUAL ANALYTICS OF ENERGY PRODUCTION PROCESSES

by

Pekka Wartiainen, Tommi Kärkkäinen, Anneli Heimbürger, and Sami Äyrämö

2013

Information Modelling and Knowledge Bases XXIV, IOS Press, pp. 15–22

PIII

REGION OF INTEREST DETECTION USING MLP

by

Tommi Kärkkäinen, Alexandr Maslov, and Pekka Wartiainen 2014

In 22nd European Symposium on Artificial Neural Networks (ESANN)

ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7.
Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

Region of interest detection using MLP

Tommi Kärkkäinen[1] and Alexandr Maslov[1] and Pekka Wartiainen[1] *

Department of Mathematical Information Technology
P.O. Box 35, 40014 University of Jyväskylä - Finland

Abstract. A novel technique to detect regions of interest in a time series as deviation from the characteristic behavior is proposed. The deterministic form of a signal is obtained using a reliably trained MLP neural network with detailed complexity management and cross-validation based generalization assurance. The proposed technique is demonstrated with simulated and real data.

1 Introduction

Change point detection from a time series is defined as determination of those time stamps where statistical properties change significantly accordingly to some predefined criterion. Typical algorithms for this purpose rely on detecting changes in first or second order statistics, like mean, median, or standard deviation, or parameters of a statistical model of data distribution [1, 2, 3]. The challenge for real application, e.g., with industrial measurements [4], is that one needs to define and estimate the probability distribution which establishes the basis for change detection [5] (see also [6] for an overview). In this context, a region of interest (ROI) is considered as a subsequence containing one or more change points.

Traditionally, neural networks have been utilized with industrial time series data mainly for classification tasks [7]. MultiLayered Perceptron (MLP) neural networks are known to be universal nonlinear regression approximators [8]. However, for real applications this is just the beginning, as summarized by Hornik, Stinchcombe, and White [9]: "We have thus established that such 'mapping' networks are universal approximators. This implies that any lack of success in applications must arise from inadequate learning, insufficient numbers of hidden units or the lack of a deterministic relationship between input and target." Therefore, a reliable network training needs to address two other principal characteristics of a data based model in addition to its accuracy: *complexity* and *generalization to unseen data*.

Here we first describe an MLP training algorithm which takes into account these targets by a detailed management of network's *structural* (size of hidden layer) and *functional* (size of weights) complexity, targeting at highly reliable generalization using the well-known cross-validation technique [10]. This training framework is then applied to the given time series to train MLP capturing its deterministic behavior. To this end, those subintervals in time where there are significant deviations greater than predetermined threshold from the model's

*The authors gratefully acknowledge the support from Jenny and Antti Wihuri Foundation (TK) and from the OSER project (AM and PW).

ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence
and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7.
Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

predictions, are suggested as *ROIs.* We emphasize that the proposed technique is *unsupervised*, i.e., it does not utilize any labeling of ROIs, even if they are known in advance.

The contents of the rest of the paper are as follows: we describe the proposed method in Section 2, and report and conclude the computational experiments in Section 3.

2 The method

2.1 MLP Training

Action of MLP in a layer wise form can be given by (e.g., [11])

$$\mathbf{o}^0 = \mathbf{x}, \quad \mathbf{o}^l = \mathcal{F}^l(\mathbf{W}^l \tilde{\mathbf{o}}^{(l-1)}) \text{ for } l = 1, \dots, L. \tag{1}$$

By $\tilde{}$ we indicate the vector enlargement for the bias and $\mathcal{F}^l(\cdot)$ denotes the activation function. This places biases in a layer as first column of the layer's weight matrix which then have the factorization $\mathbf{W}^l = \begin{bmatrix} \mathbf{W}_0^l & \mathbf{W}_1^l \end{bmatrix}$.

Using the given learning data $\{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^N$, $\mathbf{x}_i \in \mathbb{R}^{n_0}$ and $\mathbf{y}_i \in \mathbb{R}^{n_L}$, the unknown weight matrices $\{\mathbf{W}^l\}_{l=1}^L$ in (1) are determined as a solution of an optimization problem

$$\min_{\{\mathbf{W}^l\}_{l=1}^L} \mathcal{J}(\{\mathbf{W}^l\}). \tag{2}$$

Here we restrict ourselves to MLP with one hidden layer and our cost functional reads as follows:

$$\mathcal{J}(\{\mathbf{W}^1, \mathbf{W}^2\}) = \frac{1}{2N} \sum_{i=1}^N \left\| \mathbf{W}^2 \tilde{\mathcal{F}}^1(\mathbf{W}^1 \tilde{\mathbf{x}}_i) - \mathbf{y}_i \right\|^2 + \frac{\beta}{2n_1} \sum_{(i,j)} \left(|\mathbf{W}_{i,j}^1|^2 + |(\mathbf{W}_1^2)_{i,j}|^2 \right) \tag{3}$$

for $\beta \geq 0$. The special form of regularization omitting the bias-column $\mathbf{W}_0^2$ is due to Corollary 1 in [12]: *Every locally optimal solution to* (2) *provides an unbiased regression estimate having zero mean error.*

The universal approximation property guarantees accuracy of an MLP network, but in practical applications we also need to address *simplicity* and *generalization.* Simplicity is further divided into *structural simplicity*, which means favoring small size of the hidden layer, and *functional simplicity*, which refers to favoring small weights improving the network's fault tolerance [13]. Hence, in our actual training method we use a grid search for both size of the hidden layer n_1 and size of the regularization coefficient β. Moreover, 10-fold cross-validation is used as a technique to assure proper generalization of the obtained MLP network. To this end, the usual gradient based optimization methods for minimizing (3) act locally and, therefore, the solution depends on the initialization. In order to explore the search landscape better towards global optimization, we repeat the random started optimization solver three times and select, as the solution, the one with minimal training error. In the final training of MLP with fixed n_1 and β, we test five iterations for slightly more thorough globalization.

Algorithm 1 Reliable determination of MLP neural network.

```
Input: Training data $\{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^{N}$ .
Output: MLP neural network.
Define $\vec{\beta}$, $n1max$, and $nfolds$
Create $nfolds$ using random sampling
for $n_1 \leftarrow 1$ to $n1max$ do
    for $regs \leftarrow 1$ to $|\beta|$ do
        for $k \leftarrow 1$ to $nfolds$ do
            for $i \leftarrow 1$ to 3 do
                Initialize $(\mathbf{W}^1, \mathbf{W}^2)$ from $\mathcal{U}([-1, 1])$
                Minimize (3) with current $n_1$ and $\vec{\beta}(regs)$, and the $CV_{Tr}$
                Store Network for smallest $P_{err,Tr}$
            Compute $P_{err,Te}$ for the stored Network
        Store $n_1^* = n_1$ and $\beta^* = \beta$ for the smallest mean $P_{err,Te}$
for $i \leftarrow 1$ to 5 do
    Initialize $(\mathbf{W}^1, \mathbf{W}^2)$ from $\mathcal{U}([-1, 1])$
    Minimize (3) using $n_1^*, \beta^*$ and the whole training data
    Select the network with smallest $P_{err}$
```

Minimization of (3) is based on MATLAB's unconstrained minimization routine *fminunc*[1] with self realized MLP cost function and gradient calculations along the lines of [12] (these are done in full matrix form and then reshaped to and from one long weight vector for the optimizer). The vector of regularization parameters is defined as $\vec{\beta} = 10^{-i}, i = 1, \ldots, 6$. The prediction error ($P_{err,[Tr|Te]}$) is computed as the mean Euclidian error. In MLP, the sigmoidal activation function $s(x) = \frac{1}{1+\exp(-x)}$ is used. Moreover, all input and output variables are preprocessed into the range $[0, 1]$ of $s(x)$ to balance their scaling with each other and with the range of the overall transformation [12].

2.2 Application of MLP for ROI detection

Assume that a time series $\{s(t_i)\}_{i=1}^{T}$ is given. The learning data for MLP is created in the usual way: first a window length $L \in \mathbb{N},\ L > 1$, is fixed and then we associate $y_i = s(t_i), i = L, \ldots, T$, and $\mathbf{x}_i = \{s(t_{i-j}),\ j = L - 1, \ldots, 1\}$.

Then Algorithm 1 is applied to train a reliably generalizing network capturing the deterministic behavior of the time series. With this model, the absolute prediction error time series

$$e_i = |\mathcal{N}(\{\mathbf{W}^l\})(x_i) - y_i|, \quad i = L, \ldots T,$$

is created. To this end, a threshold $\tau \in \mathbb{R}$ is fixed and those indices, for which $e_i > \tau$, are proposed as members of ROIs.

[1] *http://www.mathworks.se/help/optim/ug/fminunc.html*

3 Experimental results and conclusions

We illustrate the proposed algorithm using two examples, a simulated and a real data set one. The threshold τ is set to 0.05 and $L = 8$ is used as the data window size. We use separate learning data and validation data to assess the method's performance to detect ROIs.

Example 1 *We created a simulated case with sinusoidal wave form and added normally distributed degradations of different strength to four subregions. Similar form with three noncharacteristics subregions is used as validation data. (see Figure 1)*

Example 2 *We use the Dodgers data set from UCI repository*[2]*. The data describes a five minute sampled traffic sensor reading storing the amount of cars passing a ramp on a freeway in Los Angeles. The learning problem is to determine the times of football games which are provided in another file. In the whole data there is almost six months of measurements (10-Apr-2005 00:00 – 01-Oct-2005 23:55), but occasionally the sensor is off. From the measurements, we used the indices {380–2466} (11-Apr-2005 07:35 – 18-Apr-2005 13:25, first five matches) as training data filtering out periods where the sensor is off. As validation data, the indices {3284-12529} (21-Apr-2005 09:35 – 23-May-2005 12:00, next 18 matches) are used.*

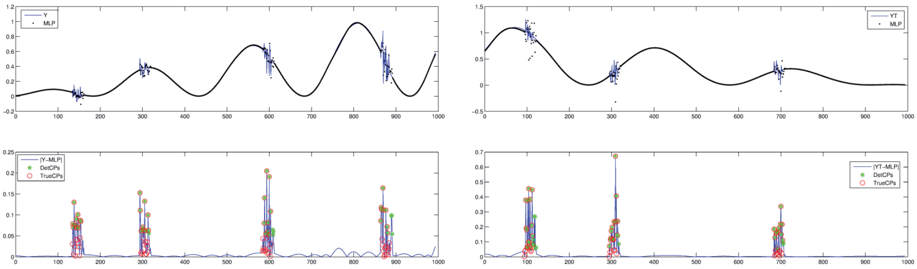


Fig. 1: Detected change regions for sinusoidal training data (left) and test data (right).

In the result figures (Figures 1,2,3), real change points in ROIs are depicted with red circles and, correspondingly, ROI indices proposed by the thresholded MLP error are given by green stars. Because Example 2 is related to traffic near the football stadium, a reasonable assumption is to assume uncharacteristic traffic patterns also before and after the actual match times. Therefore, we accept as correct indication of ROI one hour before and after the game.

Because deviation from a normal behavior of a signal can correspond to noise or actual change, we assume that the time series is not dominated by noise.

[2] *http://archive.ics.uci.edu/ml/datasets/Dodgers+Loop+Sensor*: "These loop sensor measurements were obtained from the Freeway Performance Measurement System (PeMS)"

Therefore, in the Dodgers example we first apply mean filtering with window size 11 for denoising. The filter size of 11 is selected empirically in order to achieve smallest window size for removing noise but sustaining real behaviour of the data.

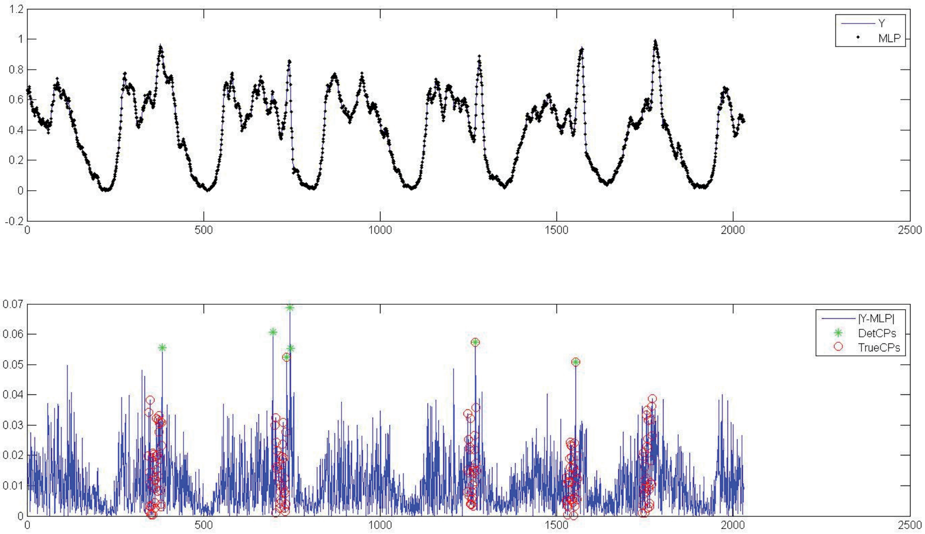


Fig. 2: MLP training compared to detected ROIs for Dodgers training data.

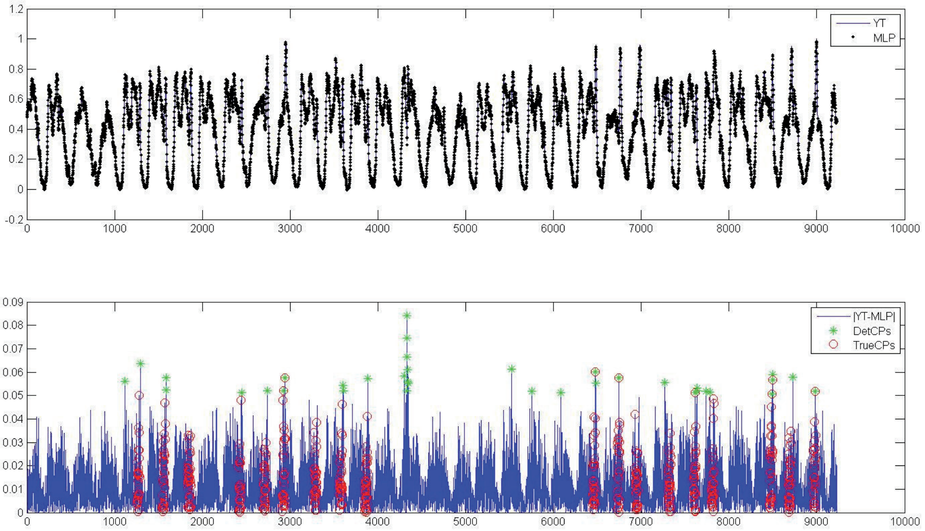


Fig. 3: MLP testing compared to detected ROIs for Dodgers test data.

From the figures we conclude that changes from normal deterministic behaviour are detected very well with the simulated case, both with training and validation data (Figure 1). Also, preliminary results are promising with real data. In the training data, 4 of 5 games were detected (Figure 2). With this trained network, 14 out of 18 ROIs were successfully located from the validation data (Figure 3) with 6 false-positive alerts. However, this was the best result obtained after several runs of the overall algorithm, which is not fully deterministic due to random creation of folds in cross-validation.

Our numerical experiments confirm the potential of the proposed approach. When the characteristic behavior of a time series is smooth and deviation clearly visible, as in Example 1, the results are as expected. Even if no such separation exists, we were able to identify potential and in many cases correct ROIs in Example 2. As can be seen, however, in such cases it might be difficult to say whether a noisy behavior (even after denoising) or actual change regions are captured. Therefore, reliable denoising is a prerequisite for good performance of the approach. The method could be improved, e.g., by feature extraction to replace the raw time series values as MLP input.

References

[1] Igor V. Nikiforov and Michèle Basseville. Detection of Abrupt Changes - Theory and Application, 1998.

[2] Nicolas Cheifetz, Allou Samé, Patrice Aknin, and Emmanuel de Verdalle. A cusum approach for online change-point detection on curve sequences. *Computational Intelligence and Machine Learning, Bruges (Belgium)*, pages 25–27, 2012.

[3] Marc Lavielle. Using penalized contrasts for the change-point problem. *Signal Processing*, 85(8):1501 – 1510, 2005.

[4] Luigi Fortuna, Salvatore Graziani, Alessandro Rizzo, and Maria G. Xibilia. *Soft Sensors for Monitoring and Control of Industrial Processes.* Spronger, 2007.

[5] Daniel Kifer, Shai Ben-David, and Johannes Gehrke. Detecting change in data streams. In *Proceedings of the Thirtieth International Conference on Very Large Data Bases - Volume 30*, VLDB '04, pages 180–191. VLDB Endowment, 2004.

[6] João Gama. *Knowledge Discovery from Data Streams.* Chapman and Hall / CRC Data Mining and Knowledge Discovery Series. CRC Press, 2010.

[7] Silvio Simani and Ronald J. Patton. Neural networks for fault diagnosis of industrial plants at different working points. In Michel Verleysen, editor, *ESANN*, pages 495–500, 2002.

[8] Allan Pinkus. Approximation theory of the MLP model in neural networks. *Acta Numerica*, pages 143–195, 1999.

[9] Kurt Hornik, Maxwell Stinchcombe, and Halber White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2:359–366, 1989.

[10] Larson S. The shrinkage of the coefficient of multiple correlation. *Journal of Educational Psychology*, 22:45–55, 1931.

[11] M. T. Hagan and M. B. Menhaj. Training feedforward networks with the Marquardt algorithm. *IEEE Trans. Neural Networks*, 5:989–993, 1994.

[12] Tommi Kärkkäinen. MLP in layer-wise form with applications in weight decay. *Neural Computation*, 14:1451–1480, 2002.

[13] Salvatore Cavalieri and Orazio Mirabella. A novel learning algorithm which improves the partial fault tolerance of multilayer neural networks. *Neural Networks*, 12:91–106, 1999.

PIV

CONTEXT-SENSITIVE FRAMEWORK FOR VISUAL ANALYTICS IN ENERGY PRODUCTION FROM BIOMASS

by

Pekka Wartiainen, Anneli Heimbürger and Tommi Kärkkäinen 2015

Information Modelling and Knowledge Bases XXVI, IOS Press, pp. 449–456

Context-Sensitive Framework for Visual Analytics in Energy Production from Biomass

Pekka WARTIAINEN [a,1], Anneli HEIMBÜRGER [a] and Tommi KÄRKKÄINEN [a]

[a] *Department of Mathematical Information Technology, University of Jyväskylä, Finland*

AbstractData masses require a lot of data processing. Data mining is the traditional way to convert data into knowledge. In visual analytics, humans are integrated into the process as there is continuous interaction between the analyst and the analysis software. Data mining methods can be utilized also in visual analytics where the priority is given to the visualization of the information and to dimension reduction. However, the provided data is not always enough. There is a large amount of background contextual information, which should be included into the automated process. This paper describes a context-sensitive approach, in which we utilize visual analytics by studying all phases in the process according to our ”sensing, processing and actuation” framework. Experimental studies show that our framework can be very useful in the process of analyzing causes for and relations between variable changes with laboratory-scale power plant data.

Keywords. Visual Analytics, Context, Context Sensitive, Energy Production, Visualization

Introduction

While analyzing industrial processes, especially those of energy production, the context information plays a significant role in acquiring reliable results. High level computational methods are mandatory for processing data, but, if the context is not defined, results are not fully understood. Energy production from biomass has been a challenging area due the organic compounds in the biomass [19]. Among other things, gas from burning biomass forms chloride acids in high temperatures. Another important factor to be taken into account when utilizing biomass is the amount of small particulates produced.

Visual Analytics is a relatively new research field – the first book in the field was published in 2005 [21]. It refers to interactive methods and technologies that could be applied for presenting the results of data mining process to users [21]. Visual analytics combines automated analysis techniques with interactive visualizations for an effective understanding, reasoning and decision making on the basis of very large and complex datasets [12].

[1] Corresponding Author: Pekka Wartiainen, Department of Mathematical Information Technology, University of Jyväskylä, P.O. Box 35 (Agora) FI-40014 University of Jyväskylä, Finland; E-mail: pekka.wartiainen@jyu.fi.

Industrial process monitoring is a well-suited genre for applications of visual analytics due its elements of data analysis and visualization. However, the set of available examples is scarce. One possible reason for this is that visual analytics is requires a lot from software methods and algorithms: user interaction, data analysis and visualization methods are too far apart from each other [12]. In visual analytics, these methods should be used simultaneously without restrictions imposed on them in some of these areas. Highly interactive interfaces combined with automated data analysis and visualization will reduce the gap between the user and the computer. Interaction possibilities are not limited only to parameter tuning and data exploration: also contextual information can be made available.

The concept of contextual sensing fits well also to the idea of visual analytics. In visual analytics, the researcher is seen as a part of knowledge mining process with his/her background knowledge [12]. That background knowledge is important for deeper understanding of the problem and helpful in finding more reliable solutions. In the same way, sensing of contextual facts provides more information, which in this case is necessary for a proper and valid analysis.

Based on this, if something changes in the context, it will have direct effect to the process and measurements. Change-point detection addresses the problem of discovering time points at which properties of time-series data change [11]. There are numerous ways of implementing change-point detection algorithms, but traditionally they are all based on statistical methods and properties [20]. In this paper, we introduce a context-sensitive framework and an implementation of it, which utilizes context-sensitive approach and change-point detection methods for visual analytics.

Our paper is organized as follows. Related work is discussed in Section 1. The principles of our iterative context-sensitive framework are introduced in Section 2. In Section 3, we present the preliminary implementation of the framework. Conclusions will be given in Section 4.

1. Background

Although visual analytics and data mining have been extensively researched, there are still many challenges [22]. Huge data sets and data bases require enormous amounts of storing capacity and almost incomprehensibly fast data transfer connections. In many application areas, data is complex or inconsistent. Also, it may happen that even if there is a huge amount of data with many variables, there is still very little data that is suitable for training [22]. Therefore, handling big data is challenging and finding the optimal solution for feature selection and evaluation of results is difficult. Existing visual analytics software and frameworks are still most likely related to certain application fields [1, 17].

Context-sensitivity in the field of computing can be defined, e.g. as in [8], thus: ”A computational method, a computer system, or an application is context-sensitive if it includes context-based functions and if it uses context to provide relevant information and services to the user, where relevancy depends on the user’s situation”. Regarding industrial applications, one could add that context-sensitivity depends not only on the user’s situation but also on the stage of the process and the state of the environment.

Context awareness is often encountered in telecommunications and web-page document analysis where location and other sensor measurement information is part of the data [2, 18]. System is context-aware if it ”uses context to provide relevant information and/or services to the user, where relevancy depends on the user’s task” [3]. Architecture of context-aware applications is presented in [7]. In our work, we prefer using the term context-sensitive approach, since it gives more choices and sets fewer limitations.

Traditionally, change-point detection uses statistical methods for finding fluctuation in the data. Recently, there was an effort to use anomaly detection in data mining to detect change-points. By defining an anomalous pattern as one ”whose frequency of occurrences differs substantially from that expected, given previously seen data”, Keogh et. al. presented a way where anomalies are not explicitly formulated [14]. Otherwise, fault detection and pattern finding related to industrial purposes have been researched a lot [5, 9–11, 13].

2. Iterative Context-sensitive Approach

In industrial processes, contextual information contains all meta data related to measurement environment, software architecture, data processing and visualization and also to preliminary knowledge from the related application area. In a complete analysis chain, contextual information should be taken into account in every analysis phase and should be included as a part of automated processing. An industrial application, which for example uses burning biomass to produce heat and electricity, has a very specific context basis. Therefore, in order to compute reliable results, different context environments are required in analyses for different industrial applications.

To meet these challenges, we are extending the EJC2012 context-sensitive SPA framework [23] with the iterative SPA approach (Figure 1). The iterative SPA framework is divided into two hierarchical levels. On general macro level, the energy production process is analyzed as a straight-forward analysis process, from variable measurements to visualization and interpretation of results. Different contexts are related to each SPA step which are considered in the overall analysis process. For example, the sensing context includes information about how and where measurements are taken, processing context describes software analysis methods, and actuating context defines the kind of visualization used.

From the visual analytics point of view, this straight-forward macro level is not enough to provide holistic and accurate information view on user. The model of visual analytics allows the process move between automated processing and visualizations while also mapping the raw data for visualizations [12]. Hence, we introduce an iterative micro level with the SPA structure. The main additional benefit of this is the possibility to move back and forth and jump over the SPA steps. Micro level is defined below macro level in the hierarchy and contains all the same contextual elements. In addition, each SPA step is divided into iterative sub-phases $S_{\{S,P,A\}}$, $P_{\{S,P,A\}}$, and $A_{\{S,P,A\}}$ that follow the SPA structure. Here, a sub-process contains one set of iterative SPA sub-phases. In general, sensing sub-phase $\{S, P, A\}_s$ includes an update loop that initializes a new set of parameters and restarts the sub-process. A more detailed description of the SPA phases and their sub-phases is given in the following chapters.

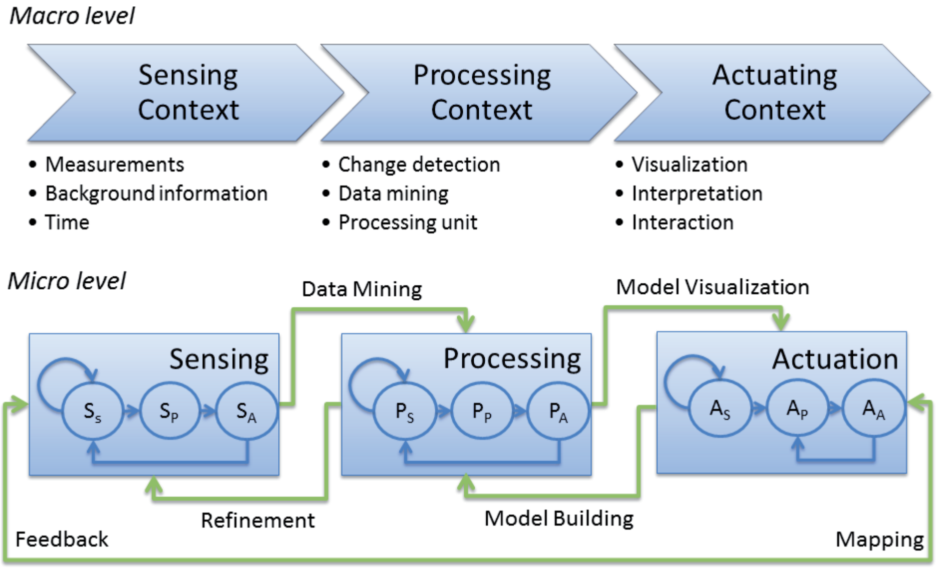


Figure 1. The framework of iterative SPA with elements of visual analytics. On macro level, the analysis process is straight forward from measurements to analysis. On micro level, there is a more detailed structure in the analysis chain, where each process phase contains also iterative sub-phases. The next phase can be chosen by the type of the action required. For example, starting from the **Sensing** phase, the user can do *data mining* by advancing to the **Processing** phase or *mapping* and visualize raw data directly in the **Actuation** phase.

2.1. Sensing contextual information

Sensing starts from inquiring all possible data related to the measurement environment and equipment as well as background knowledge from application. Also, when conducting an experiment, everything should be logged as well as possible, especially if adjustments are made. This is the main contribution in sub-phase S_S. The more information is gathered the better. Irrelevant information can be automatically reduced afterwards.

Once the experiment has been completed, all measured data and context information is processed into variables (subphase S_P). While creating new variables from context information, one has to decide whether to use them in computation. In general, quantitative or continuous variables can be used in computation, but qualitative or categorical variables provide meta-data for researcher for the interpretation and evaluation of results and later for decision making. Variables are also classified as input- or output-type variables according to their function. For example, temperature and pressure measurements are output-type (monitored) variables while fuel feeding and bottom coarse conveyor belt are input-type (controlling) variables.

In the last sub-phase S_A, the next actions are decided. In dynamic industry environment, e.g. in a power plant, context environment may change during time. For example, it is possible to change the type or quality of fuel. A radical change, like that of fuel in the context environment, affects the rest of the analysis chain. While analyzing the efficiency of a certain fuel type, a model of parameter settings is built. Due the some elemental facts, different fuels behave differently, and the original model will fail to produce reliable results. If we find a change in the context environment, the process model should be updated accordingly.

2.2. *Processing Context*

Changes in the context environment have to be adapted in the processing phase. Sensing these changes can be done manually or automatically, depending on the case in phase P_S. Possible changes could be triggered by a different fuel type or an abnormal behavior of the signals. As a response to these changes, the processing model and parameters are updated, but also going back to sensing phase is possible.

Processing phase is computationally heavy since all software computations are done in this phase (more specifically, step P_P). However, with modern computers, computations are relatively fast with small and large data sets. Here, a large data set contains more than 100 variables, with several thousands of measurements. In our framework, computations will include the steps of preprocessing (e.g. synchronization), transformation (e.g. dimension reduction) and change-point detection.

In the last phase P_A, the results of processing are briefly verified. For example, the values for each change-point detector are checked and detection thresholds are fine-tuned. A more explicit explanation of detectors and threshold is given in Section 3. Based on human decision, the analysis process may be continued to Actuation phase or iterated back to phase P_S if some tuning of parameters for computation is needed.

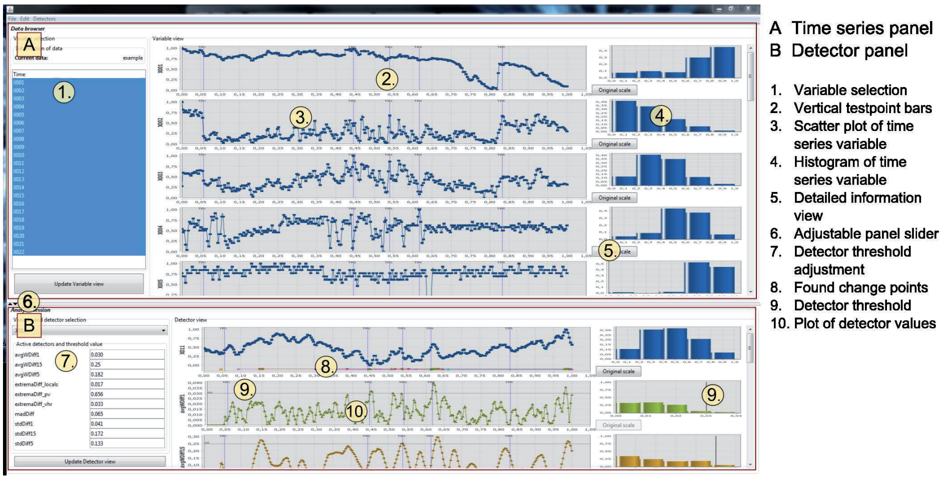
2.3. *Actuation context in visual framework*

Actuation phase starts with (sub-phase A_S) sensing the context at the user's end. Different contexts, e.g. different user roles, will be selected according to current situation. In the ultimate scenario, these contexts could be selected automatically based on the knowledge gathered from the user, For example, different factors could be formed of the role of the user, expertise of the working group, etc. In practice, an overall view is presented automatically, but the user may find more detailed information of the data when needed.

After the initialization of the user interface, the results of computations are visualized and examined (sub-phase A_P). Often, the cases with industrial data require discussions and interpretation. Validation of the results, with data from real world, is always a huge challenge. One of the best validation methods (perhaps the only) is to trust experts' opinion. In sub-phase A_A, decisions are made based on acquired knowledge. If the results are insufficient, the user can go back to the previous SPA steps and for example adjust the parameters or use different methods. On the other hand, if the results are proven reliable, measured information has been transformed to some knowledge which hopefully solves the research problem.

3. Analysis framework for effective computation and visualization

Motivated by the lack of contextual support in existing applications, we started the development of a visual analytics software for analyzing time-series data. Matlab [15] was chosen as a base of the framework because of its computational features. Matlab has a collection of implemented algorithms, and development with it is often faster than with traditional languages. However, building a GUI in Matlab is a challenge because of visual analytics requirements for user interactions. Also, it is known that after a version upgrade in Matlab all features may not function properly in old interfaces. Therefore Java was



Active detectors and threshold value

avgWDiff1	0.030
avgWDiff15	0.25
avgWDiff5	0.182
extremaDiff_locals	0.017
extremaDiff_pv	0.656
extremaDiff_vhr	0.033
madDiff	0.065
stdDiff1	0.041
stdDiff15	0.172
stdDiff5	0.133

Figure 2. Java GUI with example data. In the upper panel, time-series data is visualized. Change points are analyzed on the lower panel.

chosen as the language for GUI development. A Java GUI was built using Java's Swing library [4], and visualizations utilize the JFreeChart library [6]. The main advantage of using Java GUI on top of Matlab is that all Java visualization tools can be combined into powerful data processing techniques in Matlab.

The GUI (Figure 2) is divided into two main panels which are aligned horizontally. The upper panel, *Data browser*, offers tools for analyzing time-series visually, and the lower panel, *Analysis session*, provides an interface to apply change-point detection and to adjust detector values. In a basic workflow, the user selects variables for visual inspection in the *Data browser* and then, in the *Analysis session*, a variable for further analysis, e.g. change-point detection. Variables in scatter plots are normalized between 0 and 1, and the plots are synchronized in time for easier comparison.

After change-point detection, the findings can be exported and loaded back into the system as a new data set and the results can be inspected again on the *Data browser*. In this phase, the crucial feature for finding relations between variables is the ability to mark interesting points in time as test points, which are plotted on each variable (Number 2 in Figure 2). Now the user may find similarly behaving variables before and after any test point. Of course, the automatically computed similarities are given to the user, but an expert's opinion is required to achieve reliable results with real world data.

In the experiments, we found out that the key element in mining reliable results is the change-point detection. Change-point detection is operated in the lower panel *Analysis session* of the GUI. On the left, one variable can be selected and then plotted on the top graph of the lower panel. The rest of the space is reserved for detector plots, where the results of different change-point methods are visualized. Each detector has a horizontal threshold bar (Number 9 in Figure 2). All detector values exceeding the threshold level are considered as change points. The user can fine-tune a threshold value for each detector separately, based on his/her expertise on the field. Detected change points can be drawn into the top plot with the corresponding variable.

Color design in graphical user interfaces plays an important role in a successful data analysis and understanding process. Colors can be used for advantage by highlighting correct information, but with careless usage of colors the whole interface may become unusable. In this framework, general guidelines of GUI design have been followed. The color design of scatter plots is chosen with a color palette toolkit provided by NASA's Ames Research Center [16].

The GUI is also scalable in the contextual sense. The big picture from data can be seen on the main window, but more detailed information is available. For example, in the original space window (opened with the *Original scale* button) the requested variable is plotted in a different context and with its original values and time scale containing more specific information about the measurements and the change-points. Thus the researcher can find more relevant information of the interesting points in time.

Considering the iterative SPA framework, the GUI concentrates mainly to processing and actuation phases. The data visualized in the upper panel of the GUI supports actuation and decision making in every SPA sub-phase. The lower panel implements the processing phase with visualizations computed by automated methods in Matlab. However, the sensing phase contains still a few steps, which are conducted manually in Matlab, for example, adding context information into the system.

4. Conclusions

In case of a power plant, the idea of visual analytics is to give tools for research and development tasks rather than to build a fast controlling system. We have done to this by extending the context-sensitive SPA framework with an iterative structure. By offering ways to loop back and provide feedback to previous steps, iterative SPA framework facilitates getting deeper understanding of measured information.

In order to utilize the iterative SPA framework, we started developing a GUI for Matlab, which integrates an interactive user interface and visualizations to powerful automated computations. The development work is still in progress with analysis methods and to make context information more transparent. However, preliminary work with real world data provided by Valmet (previously called Metso Power) has given positive results. Our novel context-sensitive approach and framework, including the analysis-chain, have given new insights and ways to data analysis. Our tests show that the change-point detection methods have the key role in achieving reliable results.

In the future, we would like to concentrate our efforts on creating a massive library of change-point methods. This versatile library would provide tools for different tasks, as no particular method is the best in every situation. The second improvement to our framework is not to deal only with the context of a single point in time but with the context of a region of interest (ROI). In industrial applications, changes might be slow and take for example half hour to complete. During that time it is impossible to set a single point in time for the change, but it is happening in a region of time points.

Acknowledgements

The authors thank the OSER-project funded by European Social Fund. Also, our thanks to Valmet, which has been providing material for the research.

References

[1] José A. Castellanos-Garzón, Carlos Armando García, Paulo Novais, and Fernando Díaz. A visual analytics framework for cluster analysis of dna microarray data. *Expert Syst. Appl.*, 40(2):758–774, 2013.

[2] Stefano Ceri, Florian Daniel, Maristella Matera, and Federico M. Facca. Model-driven development of context-aware web applications. *ACM Trans. Internet Technol.*, 7(1), February 2007.

[3] Anind K. Dey. Understanding and using context. *Personal Ubiquitous Comput.*, 5(1):4–7, January 2001.

[4] Robert Eckstein, Marc Loy, and Dave Wood. *Java Swing*. O'Reilly, Beijing, 2. edition, 2002.

[5] Ada Wai-chee Fu, Eamonn Keogh, Leo Yung Hang Lau, and Chotirat Ann Ratanamahatana. Scaling and time warping in time series querying. In *Proceedings of the 31st international conference on Very large data bases*, VLDB '05, pages 649–660. VLDB Endowment, 2005.

[6] David Gilbert. Jfreechart. `http://www.jfree.org/`, 2000.

[7] Anneli Heimbürger, Yasushi Kiyoki, Tommi Kärkkäinen, Ekaterina Gilman, Kyoung-Sook Kim, and Naofumi Yoshida. On context modelling in systems and applications development. In *Proceedings of the 2011 Conference on Information Modelling and Knowledge Bases XXII*, pages 396–412, Amsterdam, The Netherlands, The Netherlands, 2011. IOS Press.

[8] Anneli Heimbürger, Miika Nurminen, Teijo Venäläinen, and Suna Kinnunen. Modelling contexts in cross-cultural communication environments. In *Proceedings of the 2011 Conference on Information Modelling and Knowledge Bases XXII*, pages 301–311, Amsterdam, The Netherlands, The Netherlands, 2011. IOS Press.

[9] Chun-Chin Hsu, Mu-Chen Chen, and Long-Sheng Chen. Intelligent ica-svm fault detector for non-gaussian multivariate process monitoring. *Expert Syst. Appl.*, 37(4):3264–3273, April 2010.

[10] Chun-Chin Hsu, Mu-Chen Chen, and Long-Sheng Chen. A novel process monitoring approach with dynamic independent component analysis. *Control Engineering Practice*, 18(3):242 – 253, 2010.

[11] Yoshinobu Kawahara. Change-point detection in time-series data by direct density-ratio estimation. *Direct*, 4(2):389–400, 2009.

[12] Daniel A. Keim, Joern Kohlhammer, Geoffrey Ellis, and Florian Mansmann, editors. *Mastering The Information Age - Solving Problems with Visual Analytics*. Eurographics, November 2010.

[13] Eamonn Keogh and Shruti Kasetty. On the need for time series data mining benchmarks: A survey and empirical demonstration. *Data Mining Knowledge Discovery*, 7(4):349–371, October 2003.

[14] Eamonn Keogh, Stefano Lonardi, and Bill 'Yuan-chi' Chiu. Finding surprising patterns in a time series database in linear time and space. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pages 550–556, New York, NY, USA, 2002. ACM.

[15] MATLAB. *version 7.11.0.584 (R2010b)*. The MathWorks Inc., 2010.

[16] Ames Research Center NASA. Using color in information display graphics. `http://colorusage.arc.nasa.gov/`.

[17] Kristien Ooms, Gennady L. Andrienko, Natalia V. Andrienko, Philippe De Maeyer, and Veerle Fack. Analysing the spatial dimension of eye movement data using a visual analytic approach. *Expert Syst. Appl.*, 39(1):1324–1332, 2012.

[18] B. Schilit, N. Adams, and R. Want. Context-aware computing applications. In *Mobile Computing Systems and Applications, 1994. WMCSA 1994. First Workshop on*, pages 85–90, 1994.

[19] Jaani Silvennoinen and Merja Hedman. Co-firing of agricultural fuels in a full-scale fluidized bed boiler. *Fuel Processing Technology*, 2011.

[20] Silvio Simani and Ronald J. Patton. Neural networks for fault diagnosis of industrial plants at different working points. In Michel Verleysen, editor, *ESANN*, pages 495–500, 2002.

[21] James J. Thomas and Kristin A. Cook. *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. National Visualization and Analytics Ctr, 2005.

[22] Tatiana von Landesberger, Sebastian Bremm, Matthias Kirschner, Stefan Wesarg, and Arjan Kuijper. Visual analytics for model-based medical image segmentation: Opportunities and challenges. *Expert Systems with Applications*, 40(12):4934 – 4943, 2013.

[23] Pekka Wartiainen, Tommi Kärkkäinen, Anneli Heimbürger, and Sami Äyrämö. Context-sensitive approach to dynamic visual analytics of energy production processes. In Yasushi Kiyoki and Takehiro Tokuda, editors, *22th European-Japanese Conference on Information Modelling and Knowledge Bases*. MATFYZPRESS - Univerzity Karlovy, 2012.

PV

HIERARCHICAL, PROTOTYPE-BASED CLUSTERING OF MULTIPLE TIME SERIES WITH MISSING VALUES

by

Pekka Wartiainen and Tommi Kärkkäinen 2015

In 23nd European Symposium on Artificial Neural Networks (ESANN)

Reproduced with kind permission of ESANN.

PVI

DATA ANALYSIS PROCESS FOR PARTICULATE MEASUREMENTS IN A BUBBLING FLUIDIZED-BED BOILER

by

Pekka Wartiainen, Tommi Kärkkäinen, Anneli Heimbürger, and Merja Hedman
2015

Expert Systems, Wiley, submitted