Author(s): Wartiainen, Pekka; Heimbürger, Anneli; Kärkkäinen, Tommi

Title: Context-sensitive framework for visual analytics in energy production from biomass

Year: 2014

Version:

Please cite the original version:

Wartiainen, P., Heimbürger, A., & Kärkkäinen, T. (2014). Context-sensitive framework for visual analytics in energy production from biomass. In B. Thalheim, H. Jaakkola, & Y. Kiyoki (Eds.), Proceedings of the 24th International Conference on Information Modelling and Knowledge Bases - EJC 2014 (pp. 508-515). Kiel University. Kiel Computer Science Series, 4/2014. https://www.numerik.uni-kiel.de/~discopt/kcss/kcss_2014_04_v1.0_print.pdf

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.

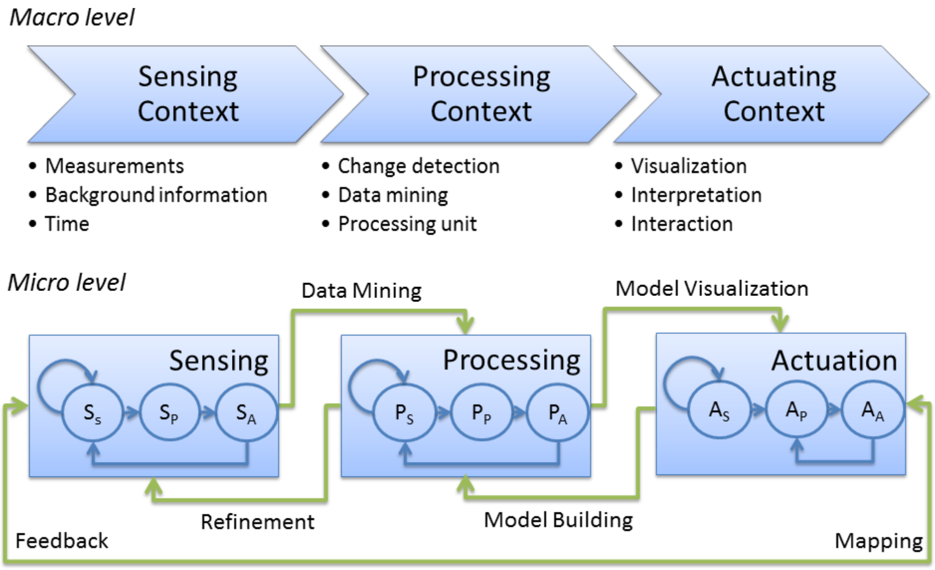

Figure 1. The framework of iterative SPA with elements of visual analytics. On macro level, the analysis process is straight forward from measurements to analysis. On micro level, there is a more detailed structure in the analysis chain, where each process phase contains also iterative sub-phases. The 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

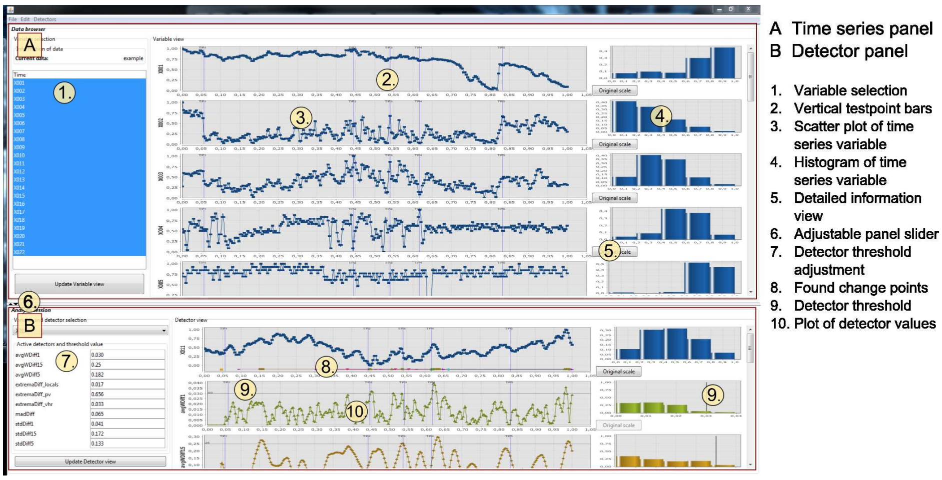

Figure 2. Java GUI with example data. In the upper panel, time-series data is visualized. Change points are analyzed on the lower panel.

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.