Mining Industrial Logs for System Level Insights

Sebastian Czora[1], Marcel Dix[2], Hansjörg Fromm[1], Benjamin Klöpper[2], Björn Schmitz[1]

Abstract: Industrial systems are becoming more and more complex and expensive to operate. Companies are making considerable efforts to increase operational efficiency and eliminate unplanned downtime of their equipment. Condition monitoring has been applied to improve equipment availability and reliability. Most of the condition monitoring applications, however, focus on single components, not on entire systems. The objective of this research was to demonstrate that a combination of visual analytics and association rule mining can be successfully used in a condition monitoring context on system level.

Keywords: condition monitoring; predictive maintenance; log file analysis; data mining

1 Introduction

Most of the condition monitoring applications known today are focusing on single units of equipment (e.g. pumps, motors, transformers, turbines, etc.), not on entire systems (e.g. factories, sub-stations, power plants). This leaves possible interactions and cause-and-effect relationships among different units of equipment unconsidered. The objective of this research was to demonstrate that a combination of visual analytics and association rule mining can be successfully used in a condition monitoring context on system level. More precisely, the goal was to identify patterns of events (recorded in log files) that occur closely together in time or are correlated with certain types of alarms or failures.

The study was conducted for the downstream section of a petrochemical plant with data that was captured by the installed SCADA (supervisory control and data acquisition) system. The SCADA system generates a log file that reports events (alarms and warnings) originating from condition monitoring systems attached to various types of equipment (pumps, motors, handling equipment). In their monitoring function, SCADA systems tend to be event-driven rather than process-driven [GH13]. The step from continuous process data to discrete event data (called logs) reduces the amount of data communicated significantly. On the other hand, some detailed information is lost. Still, a SCADA of even a small installation can generate thousands of potentially alarming events per day [HBH12] and the data volume generated by a factory-wide SCADA system can reach an order of magnitude that easily can be considered as big data

[1] Karlsruhe Institute of Technology (KIT), Karlsruhe Service Research Institute (KSRI), Kaiserstr. 89, 76133 Karlsruhe, Germany
[2] ABB AG, Corporate Research Center, Wallstadter Str. 59, 68526 Ladenburg, Germany

[FB14,ZE11]. Monitoring these vast amounts of data exceeds human capabilities by far and calls for technologies that are nowadays called big data analytics [ZE11].

Two important requirements were formulated for this analytics solution: Firstly, there should be no extensive software development. Algorithms should be used that are available in commercial data mining tools or open source solutions with a strong development community. Secondly, the algorithms should be comprehensible ('white box' instead of 'black box') and generate results that are explainable and understandable by domain and plant experts. It will be shown in the course of this paper that these requirements could be fulfilled.

2 Related Work

Most existing condition monitoring solutions concentrate on errors of one particular equipment [Ei15]. However, there can be complex interdependencies between equipment-level faults and system-level faults [Lu09]. To improve the availability and reliability of the entire system, a system-level condition monitoring is required [Ei15]. The processing of multiple sensor data, however, quickly reaches a complexity that exceeds computational tractability. To still maintain a system-wide view, the data from multiple sources must be reduced in volume by appropriate preprocessing or filtering techniques. This is exactly done by SCADA systems which transform the raw process data into events and report these events in a logging system.

Log message contain information about the conditions under which certain events occurred. Typical attributes are: the time of occurrence, a message sequence number, the event origin (e.g. a unit of equipment), a severity code (in case of alarms), a message type and a standardized message text. A collection of log messages is called a log file. Equipment event logs have been investigated by Devaney et al. [HGH15] using natural language processing and case-based reasoning. Sipos et al. [Si14] have applied multi-instance learning to mine event logs.

Log files are not only used in industrial systems, but also in computer systems and telecommunications networks. Similar are *web logs*, which record user activities when users browse a web site. Different data mining algorithms have been used for log file analysis in these areas. Among them are clustering, association/frequent pattern mining, sequential pattern mining, multi-instance learning, naïve Bayes classifiers, hidden Markov and semi-Markov models, support vector machines, text mining, and visual analytics. The extensive literature on these methods cannot be listed in this short paper.

3 Industrial Case Study

The study was conducted for the aforementioned downstream section of a chemical plant with data that was captured by the installed SCADA system. The data consisted of log

messages (alarms and events) originating from the condition monitoring systems of the individual units of equipment. The main phases of the project that was conducted according to the CRISP-DM[3] process were *data understanding and preparation, followed by the actual data analysis*.

3.1 Data Understanding and Preparation

As already mentioned, data for this study originates from a SCADA system installed at the downstream section of a petrochemical plant. A SCADA system is a distributed hierarchical IT system connecting local control units attached to physical devices with a SCADA control center. Log entries are generated by additional servers in the network communicating over standard OPC4 protocols, the OPC Alarm and Event Server (OPC AE), the OPC Data Access Server (OPC DA), and a Data Concentrator.

Log entries contain the time of occurrence (Timestamp), a message sequence number (MessageId), the event origin (DeviceId), a severity code (Severity), a message type (StatementId) and a corresponding message text (Statement). Every time an equipment condition changes, the OPC server records an event and sends it to the central controller of the SCADA system. Warning and failure alarms are events of a higher severity code (i.e. Severity = 500, Severity = 1000) than simple condition changes.

We used an iterative approach in order to develop an understanding of this data. Several workshops were conducted with industrial domain experts, who ultimately work with the analysis results. Besides talking to the industrial domain experts, a visual, interactive data exploration tool that was developed based on R's Shiny web application framework5 was extremely helpful. Its primary objective was to visualize event patterns and to develop a deeper understanding of relationships that might exist in the data set. Data quality was assessed by checking missing values, plausibility of data, and inconsistencies between the fields in the log file. Identified quality problems resulted in adjustments in the real system to improve data collection in the future. Since the analysis was based on data that had been collected over a long timeframe in the past, the existing data had to be corrected and improved as best as possible for further processing.

3.2 Data Analysis

As already mentioned, the objective of this research project was to find patterns of events that occur closely together. This co-occurrence of events is a problem that was originally investigated in *market basket analysis* and is known as *pattern mining*, *frequent itemset mining* or *association analysis*. A number of algorithms have been developed over time to address the problem of association rule mining. Among them are

[3] CRISP-DM = Cross Industry Standard Process for Data Mining
[4] OLE (Object Linking and Embedding) for Process Control (OPC), https://opcfoundation.org/about/what-is-opc/
[5] https://cran.r-project.org/package=shiny

Aprior, *Eclat* and *FP-growth* [HKP11].

In our context, transactions correspond to buckets of 100 events that precede a warning alarm, and groups of items correspond to groups of events. Let A and B be nonempty, disjoint subsets of an overall set of possible events I. Then, an association rule is an implication of the form

$$A \Longrightarrow B$$

where A is called the *antecedent*, and B is called the *consequent* of the rule. This means: if a group of events A is occurring in a certain time window (e.g. a number of alarms), then a group of events B (e.g. a failure) is likely to occur in the same time window. *Support* and *confidence* are measures to describe the significance of the association rule:

$$A \Rightarrow B\ [support = 2\%, confidence = 60\%]$$

A support of 2% means that 2% of all transactions (observed time intervals) show that the events A and events B are occurring together. A confidence of 60% means that 60% of all intervals which contain events A also contain events B. Formally, let $P(A)$ be the probability that a transaction contains the set of events A, and, accordingly, $P(A \cup B)$ be the probability that a transaction contains the *union* of the sets A and B, or *both* A and B. Then $P(A \cup B)$ is called the *support* of the rule $(A \Rightarrow B)$

$$support(A \Rightarrow B) = P(A \cup B)$$

and $P(B \mid A)$ is called the *confidence* of the rule $(A \Rightarrow B)$

$$confidence(A \Rightarrow B) = P(B \mid A)$$

If *support (A)* is greater than a predefined support threshold, then A is called a *frequent itemset*. In market basket analysis, the focus is on identifying co-occurring *frequent* itemsets. In our context, the focus lies on co-occurring *rare* patterns - patterns that occur infrequently (i.e. with low support) but are of critical importance. Such events are warnings, failures or other deviations from normal operating conditions which are expected to occur infrequently. *Rare pattern mining* has also become important in other areas such as fraud and intrusion detection. A review of the field is given by Koh and Ravana [KR16].

Classical association mining algorithms generate rules that exceed a minimum support threshold (*minsup*), i.e. they are designed for discovering frequent patterns. For rare patterns, the minsup threshold has to be set very low (in our case, minsup < 1%). As a consequence, the algorithms generate a very high number of rules, and not all of these rules are interesting [KR16]. To identify relevant rules and filter out irrelevant ones, additional measures of interestingness are required that reveal more intrinsic pattern characteristics than only support and confidence. Such measures are *lift*, the *Kulczynski measure*, and the *imbalance ratio*. Wu et al. [WCH10] compare seven measures of interestingness and come to the conclusion that the *Kulczynski* measure [Ku27] in conjunction with the *imbalance ratio* is a very promising combination. Both measures

are null-invariant. *Kulczynski* proposed a measure based on the average of the two conditional probabilities

$$Kulc(A,B) = \frac{1}{2}\left(P(A \mid B) + P(B \mid A)\right)$$
$$= \frac{1}{2}\left(confidence(B \Rightarrow A) + confidence(A \Rightarrow B)\right)$$

A *Kulczynski* value near 0 or 1 indicates that the patterns A and B are either negatively or positively correlated and we have an interesting rule. If the value is 0.5, there is no such indication. For this case, the *imbalance ratio* [WCH10]

$$IR(A,B) = \frac{\left| support(A) - support(B) \right|}{support(A) + support(B) + support(A \cup B)}$$

provides more insight. It assesses the imbalance of two patterns A and B where 0 is perfectly balanced and 1 is very skewed. If *Kulc(A,B) = 0.5* and *IR(A,B) = 0*, the rule is completely uninteresting. If *Kulc(A,B) = 0.5* and *IR(A,B) = 1*, the rule might be worth looking at.

By applying measures like the Kulczynsi measure or the imbalance ratio, additional concepts such as *multilevel mining* and *constraint-based mining* can be used [HKP11]. Since we were interested in using easy-to-understand algorithms, we focused on the well-known, standard pattern mining algorithms available in commercial or well-supported open source packages and selected an algorithm that has proven to perform well with low support thresholds. Based on an evaluation by Saabith et al. [SSB15], FP-growth was chosen.

3.3 Big Data Impementation

FP-growth typically performs well on large datasets if the minsup parameter is set high enough to prune the combinatorial search space. The smaller minsup is chosen - and this is necessary for discovering rare patterns such as failures in industrial plants -, the larger the search space and the longer the algorithm's runtime and memory requirements. Experiments demonstrated that the large search space did not fit entirely into the memory of a single computer. This calls for new technical solutions that cope with the complexity of the problem. These solutions are based on massively parallel architectures and are known as *big data technologies* [FB14] [Ma11].

We chose IBM's InfoSphere BigInsights [ZE11], which is based on the distributed big data processing environment Apache Hadoop. IBM's InfoSphere BigInsights allows the integration of open source analytics packages and we decided to use Mahout's parallel FP-growth algorithm[6] to take full advantage of the distributed hardware architecture. As an extension of Mahout's FP-growth implementation, we implemented more

[6] https://github.com/apache/mahout/tree/mahout-0.8/core/src/main/java/org/apache/mahout/fpm/pfpgrowth

sophisticated pattern evaluation measures. The implementation and the experiments were conducted at the Smart Data Innovation Lab (SDIL).[7] The hardware architecture used in this project consists of 6 IBM Power S822 full rack nodes, each with 20 cores, 512 GB RAM, 260 hard disk drives with over 300 TB storage capacity and a 40 GB/s network connection. For the implementation and the experiments, Hadoop version 2.7, version 3 of IBM's InfoSphere BigInsights, and version 0.8 of the Mahout library were available.

The automated log data transformation and the parallel FP-growth algorithm as well as the pattern analysis were embedded into an overall workflow with MapReduce jobs for data transformation, the parallel FP growth, and the calculation of interestingness measures and the validation and interpretation of results. We implemented configurable MapReduce jobs that automatically create transactions (baskets) for the FP-growth algorithm. These MapReduce jobs implement the logic how the baskets should be constructed from the logs that should be mined. Here, we decided to split the log file every time a warning alarm occurs. Furthermore, each transaction consists of 100 events that occurred before the warning alarm was registered in the log.

3.4 Results

The result of the analysis phase is a collection of association rules together with their corresponding measures of rule interestingness. A short excerpt of this rule collection is shown in Table 1. Every row in this table represents an association rule. Thus, a row can be interpreted as a set of events that occur together in the analyzed log file. The first column contains the *RuleId*, the second column the consequent of the association rule and the further columns the association rule's antecedent set. Additionally, the last columns represent the measures of rule interestingness *support*, *confidence*, *Kulczynski* and *imbalance ratio*.

The association rules shown in Table 1 describe co-occurrence of failures across different units of equipment like pumps, fans and lifts. The *Kulczynski* value for the respective association rules varies in a range between 0.75 and 1. Therefore, the *imbalance ratio* does not need to be considered for further interpretation and the association rules are interesting. The rules with *RuleIds* 69, 85, and 195 describe situations where the data concentrator lost its connection to the central controller. Consequently, each of the equipment units connected to the respective data concentrator also lost their connection to the central controller. This is a situation that was already observed in the preprocessing phase with the visual data exploration tool.

Additionally, further interesting association rules were found. For example, a rule representing a situation in which the communication between a motor starter power module and the control unit of the motor starter is disrupted. Further analysis of the alarms in the antecedent set disclosed that the co-occurrence of the respective alarms reflect the process of intentionally disconnecting equipment from the condition

[7] Smart Data Innovation Lab: http://www.sdil.de/en

monitoring system. Another interesting association rule describes a situation in which equipment exceeded the predefined failure level of one of its operational parameters due to a specific handling of the equipment.

The discussion with industrial domain experts confirmed that the above mentioned association rules are understandable and explainable and denote either undesired communication problems, problems in setting up a manufacturing facility, or actual operational issues in the manufacturing systems. Thus, they are valid and interesting from an engineering point of view and can be used to improve plant operations. Limitations of association rule mining have been demonstrated by comparing patterns that are identified by the visual data exploration tool with the patterns identified by the parallel FP-growth algorithm. To sum up, the study has shown that visual analytics in combination with association rule mining are suitable methods for condition monitoring on the system level.

Rule ID	Rule	S	C	K	IR
69	Fan ID A67 is damaged or missing & Material Life AB 69 and Oil pump is damaged or missing => Concentrator ID 1 is missing	0.31	1	1	0
85	Immersion Pump P1015 is damaged or missing & Fan ID A67 & Hydrauliuc Oil pump is missing => Concentrator ID 2 is missing	0.17	0.75	0.75	0
195	Fan ID A89 is damaged & Fan ID A89 is damaged & Lift Workshop is damaged or missing => Concentrator ID 21 is missing	0.31	1	1	0

Tab. 1: Excerpt of the association rules together with their measures of interestingness S=support, C=confidence, K=Kulczyinski measure, IR=imbalance ratio

4 Conclusion

In the real-world industrial environment described in the beginning, we were able to demonstrate that a system-level analysis of the log files generated by a SCADA system is achievable with appropriate algorithms and technology. For the detection of rare events with low support and confidence, association rule mining requires new technical approaches. A pattern mining algorithm could successfully be implemented on a scalable big data architecture. Not surprisingly, good data quality is one of the key requirements in achieving meaningful results.

References

[Ei15] Eickmeyer, J.; Li, P.; Givehchi, O.; Pethig, F.; Niggemann, O.: Data driven modeling for system-level condition monitoring on wind power plants. In: 26th International

Workshop on Principles of Diagnosis (DX 2015), 2015.

[FB14] Fromm, H.; Bloehdorn, S.: Big Data – Technologies and Potential, In: Enterprise-Integration, ed. by Schuh, G.; Stich, V., Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, chap. 9, pp. 107–124.

[GH13] Galloway, B.; Hancke, G. P.: Introduction to industrial control networks, Communications Surveys & Tutorials, IEEE 15/2, pp. 860–880, 2013.

[HBH12] Hadziosmanović, D.; Bolzoni, D.; Hartel, P. H.: A log mining approach for process monitoring in SCADA, International Journal of Information Security 11/4, pp. 231–251, 2012.

[HKP11] Han, J.; Kamber, M.; Pei, J.: Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2011.

[HN15] Herzig, K.; Nagappan, N.: Empirically detecting false test alarms using association rules. In: Proceedings of the 37th International Conference on Software Engineering-Volume 2, IEEE Press, pp. 39–48, 2015.

[KR16] Koh, Y. S.; Ravana, S. D.: Unsupervised Rare Pattern Mining: A Survey, ACM Transactions on Knowledge Discovery from Data (TKDD) 10/4, 45:1–45:29, 2016.

[Ku27] Kulczynski, S.: Die Pflanzenassoziationen der Pieninen, Bulletin de l'Academie polonaise des sciences. Serie des sciences biologiques, pp. 57–203, 1927.

[Lu09] Lu, B.; Li, Y.; Wu, X.; Yang, Z.: A review of recent advances in wind turbine condition monitoring and fault diagnosis. In: 2009 IEEE Power Electronics and Machines in Wind Applications, IEEE, Lincoln, NE, pp. 1–7, 2009.

[Ma11] Manyika, J.; Chui, M.; Brown, B.; Bughin, J.; Dobbs, R.; Roxburgh, C.; Byers, A. H.: Big data: The next frontier for innovation, competition, and productivity, 2011.

[Si14] Sipos, R.; Fradkin, D.; Moerchen, F.; Wang, Z.: Log-based predictive maintenance. In: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining – KDD '14, ACM Press, New York, NY, USA, pp. 1867–1876, 2014.

[SSB15] Saabith, A. L. S.; Sundararajan, E.; Bakar, A. A.: Comparative Analysis of Different Versions of Association Rule Mining Algorithm on AWS-EC2. In: International Visual Informatics Conference. 2015, Springer International Publishing, 2015.

[WCH10] Wu, T.; Chen, Y.; Han, J.: Re-examination of interestingness measures in pattern mining: a unified framework, Data Mining and Knowledge Discovery 21/3, pp. 371–397, 2010.

[ZE11] Zikopoulos, P.; Eaton, C.: Understanding big data: Analytics for enterprise class hadoop and streaming data. McGraw-Hill Osborne Media, 2011.