Data-Mining of lifecycle information

Aaron Bentlage; Michael Zenker

ABSTRACT

The systematic analysis of life cycle data provides the potential to identify weaknesses of current products and starting points for product innovations. This paper presents a tool for an automated analysis of life-cycle data and describes the necessary steps of a data analysis. In particular, the merging of data from different systems, the pre-processing of raw data, the data analysis and interpretation are explained. In this context, the pre-processing of free text is described. Moreover, two models for an automated analysis of life cycle data are introduced. Finally, the implementation of this concept into a software tool is described.

1 INTRODUCTION

In 2012 2.8 zettabytes - or 2.8 trillion gigabytes – of data were created in total and the data volume is projected to going to reach 40 zettabytes by 2020 [1]. In the field of production technology a similar increase of data volume can be identified. During the lifecycle of a product a variety of data is gathered from different IT systems. The product development is supported by computer aided design and product lifecycle management systems. The manufacturing process is organized with manufacturing execution systems. The procurement and inventory management is done with enterprise resource planning systems. Moreover, service and maintenance activities are controlled and monitored by asset lifecycle management systems.

The analysis of this data has the potential to identify weaknesses of current products or production processes. The knowledge of these weaknesses can be used to facilitate product manufacturing, improve the product performance and to extend the product lifetime. By analyzing lifecycle data, companies can uncover valuable insights and gain a competitive advantage. Data is therefore becoming a new raw material of businesses [2].

Despite this great importance of a detailed data analysis recent studies concluded that only 0.5% of the collected data is evaluated specifically [1]. Therefore valuable knowledge that is hidden in the collected data stays undetected and cannot be used. The most important obstacles in order to implement a more detailed data analysis is a question of data privacy protection, the lack of a budget and missing know-how in the field of data analysis [3].

2 AUTOMATED DATA ANALYSIS

In order to overcome these obstacles an automated analysis of lifecycle data has been developed in the research project “LeWiPro”. The developed software tool analyzes data from the product life cycle and converts it into knowledge that can be used in product development. The distinguishing feature of this software is the possibility to analyze data from different life cycle phases in one very user-friendly interface. By using predefined analysis models on a preprocessed database, users without detailed knowledge about data-mining technologies or statistical analyses can use this tool. In this manner not only data scientists can analyze product lifecycle data but also design engineers or production planners.

The architecture of the developed software tool is based upon the knowledge discovery process in data bases (KDD) by Fayyad, which describes the process of discovering useful knowledge from data [4]. Fig. 1 shows this software architecture.

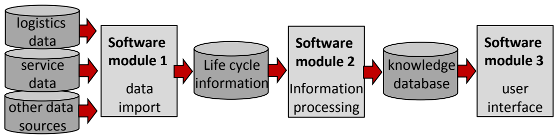


Fig. 1 software architecture

In the first software module the raw data is imported and preprocessed. Data from different source systems are merged into one life cycle information database. This database provides the input data for the following software module, the information processing. The information processing aims to transform data into knowledge. For this purpose two different types of data analysis methods are implemented. The first analysis method is the use of predefined analysis issues. The user can start an automated data analysis by choosing between several predefined analysis questions or analysis theses. For each analysis issue a data-mining model is preconfigured. The second analysis method is an interactive data analysis. The user can manually search for hidden knowledge in the data. For this purpose a simple and clear data dashboard is implemented in the third software module. The third software module visualizes the results of the data analysis clearly and comprehensible. In order to use the acquired knowledge in the product development process an interface for a product lifecycle management system has been developed. The details of these software modules are described in the following chapters.

2.1 DATA IMPORT

The required data to analyze the predefined theses is distributed in different data sources. Examples are manufacturing execution systems and asset lifecycle systems [5]. The developed software system allows accessing these disparate systems and bringing them together. The data of these disparate systems is replicated into a single all-encompassing MySQL-database. This predefined database has a consistent structure which is necessary for an automated analysis. The selection of data is based upon an analysis of the available data inventory of manufacturers of rail cars, printing presses, production measurement technology and catering supplies. The following figure 2 shows the structure of this database.

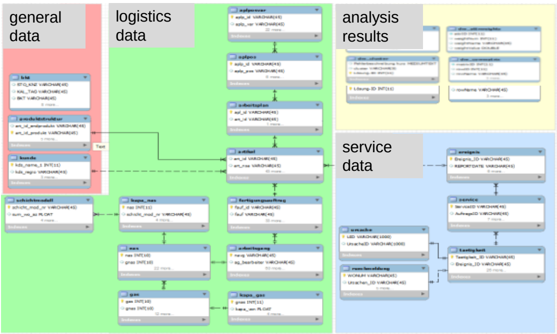
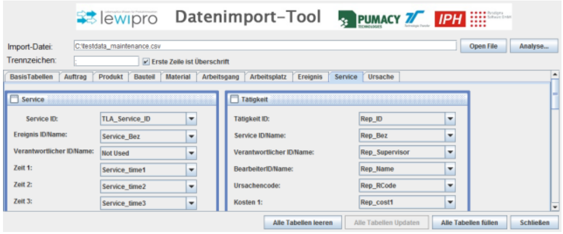


Fig. 2 structure of the life cycle database

The shown lifecycle database includes service and logistics data. Service data is all data that is collected through inspections, maintenance and repairs and brought by compression and classification in relation to each other, wherein the information content increases. The knowledge within this data is referred as service knowledge. It allows the designer to optimize the reliability and maintainability already while the product development (e. g. mean time to repair, mean time between failure, availability). Logistics data is all data that is collected from procurement to the material flow and order processing to delivery to the customer and are brought by compression and classification in relation to each other. The knowledge contained therein is referred as logistics knowledge. It allows the designer to optimize logistical key figures already in the product development (e. g. throughput time, utilization, stocks, punctuality). Moreover, general data like the structure of a product, customer information and date information is saved in this lifecycle information database. The results of the data analysis with predefined questions are also saved within this database. Data of further life cycle phases like the research and design or recycling can be added in this data base.

The analysis of available lifecycle data has shown that many companies collect similar data, but the labeling and the structuring of the data differs. For this reason, a generic import tool has been developed to import raw data into a predefined lifecycle database. The CSV ("Comma Separated Values") file format is used as a data exchange format, because most systems support this file format [6]. The following fig. 3 shows the user interface of this tool.



Import-Datei: C:\testdata_maintenance.csv
Trennzeichen: ;
☑ Erste Zeile ist Überschrift

□ Service

Service ID:	TLA_Service_ID
Ereignis ID/Name:	Service_Bez
Verantwortlicher ID/Name:	Not Used
Zeit 1:	Service_time1
Zeit 2:	Service_time2
Zeit 3:	Service_time3

□ Tätigkeit

Tätigkeit ID:	Rep_ID
Service ID/Name:	Rep_Bez
Verantwortlicher ID/Name:	Rep_Supervisor
BearbeiterID/Name:	Rep_Name
Ursachencode:	Rep_RCode
Kosten 1:	Rep_cost1

Fig. 3 user interface of the import tool

This import tool allows a mapping of company specific data to predefined data fields of the lifecycle database. The following data analysis in software module 2 does therefore not depend on a company specific data labelling. The data-mining processes always use the standardized data fields of the life cycle database as input data. In addition to the standardized labelling of the data fields the date types are also predefined. For example, item numbers are marked as polynomial values and costs as floats. If certain data fields are not maintained, they are characterized by the import tool as empty. Although certain predefined analysis models may not work without these data the whole system remains functional.

2.2 DATA PREPROCESSING

In addition to the data import, a data preprocessing is necessary for the automated data analysis. An important aspect of this preprocessing is the treatment of free text. This is particularly relevant to the service data as for example error descriptions are often documented as free text. Examples are the error descriptions: "oil loss in the gearbox area" or "leaking gearbox". Despite differences in formulation these texts describe the same error. To analyze the error descriptions using data-mining methods, these descriptions must be initially assigned to a common error cluster. Examples for useful clusters of failure descriptions are "oil loss transmission" or "defect on the speed controller". Due to the amount of data this step is automated by a text-mining algorithm. For this purpose the individual text fields are converted into word vectors. The word vector describes the frequency of relevant words within a text field. To create the word vector the following steps are performed: tokenization, filtering out stop words, replacing semantically similar terms, root-form reduction and the generation of word pairs.

In the tokenization the text of a document is divided into individual words. In the next step stop words are filtered out. Stop words are words that occur very frequently but have no relevance in the document. Examples are definite article in the German language (der, die, das) or prepositions. Then, terms with a semantically similar meaning are replaced on the basis of a predefined dictionary. As an example, this enables the replacement of the terms "oil loss" and "oil leak" through a common term. As a next step, a stem-form reduction is done. A stem-form reduction reduces various declensions or conjugation of a word to their word stem. In the last step of the word vector creation, word pairs are generated. The use of word pairs allows using the co-occurrence of words as additional information for the following analysis.

After the creation of the word vectors all text fields will be assigned to different clusters by the clustering algorithm Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [7]. This algorithm is capable to determine an appropriate number of required clusters. This is especially important as the number of required clusters is not known before the data analysis.

The algorithm DBSCAN determines areas with a particularly high density of word vectors. Setting-parameters for this algorithm are the neighborhood distance ε and the density threshold minPts. The neighborhood distance ε determines whether a point A is directly density-reachable from a point B. A point is directly density-reachable if the maximal distance is smaller than ε. The density threshold minPts determines, whether an area is classified as dense. An area is classified as dense if a point is surrounded by sufficiently many density-reachable points [8]. Every dense area is declared as a cluster. Word vectors which cannot be assigned to any of these clusters are marked as noise. This minimizes the risk that text fields are assigned to a wrong cluster and thus distort the subsequent data analysis.

The parameterization of this setting was carried out using a test data set with 330 entries. The first criterion for evaluating these automatically generated clusters is the number of useful associated text fields. Text fields are assigned as useful if they are not rated as noise and if not more than 50 percent of the text boxes are assigned to a common cluster. The second criterion for evaluating these automatically generated clusters is the cluster density performance. This performance indicator improves as the average euclidean distance between two objects within a common cluster decreases. Consequently, a high cluster density performance indicates a high similarity of objects within a common cluster. The best clustering results could be achieved with the setting parameters minPts=5 and ε=1,2. The following fig. 4 shows the influence of the setting parameters on the evaluation criteria.

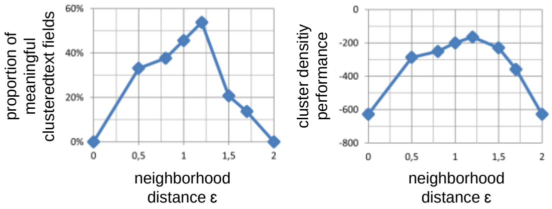


Fig. 4 DBSCAN parameterisation

Fig. 4 shows that both evaluation criteria could best be satisfied with ε=1,2. As a result, about 54% of the test data set could be divided into useful clusters. The remaining 46% of the data are identified as noise. A manual analysis of this noise cluster showed, that this cluster mainly includes unique failures which did not occur repeatedly. A more detailed clustering of these failure descriptions is therefore not useful. With these setting parameters similar results were obtained for other test data. This exemplary preprocessing step and further preprocessing operations like the transformation of data fields are implemented in the import tool. The user only has to start the preprocessing process. A detailed configuration is not necessary.

2.3 INFORMATION PROCESSING

After the preprocessing of the imported lifecycle data the user can start the information processing in software module 2. The information processing aims to transform data into knowledge. This knowledge acquisition can be done with a manual interactive data analysis or an automated analysis with predefined analysis issues. Every analysis issue includes a central question or thesis, a selection of potentially relevant data and a predefined data-mining process. The definition of the analysis issues was done together with companies from the field of production control, service planning and knowledge management. The use of the predefined analysis issues allows a simple analysis of a big amount of life cycle data.

2.3.1 Analysis of Logistics data

For the analysis of logistics data four theses were designed and implemented. Thesis 1 implies that products which have identical or highly similar workplans, also have approximately the same throughput times. If the throughput time fluctuates strongly, this may be an indication of unsafe processes. Product development should consider this in future products. Thesis 2 is especially suitable for the comparison of different product variants. By checking this thesis, it has been found out that the average time required for the production increases disproportionately with additional operations. In a following cause analysis the user can identify which operations or which operation combinations are responsible for these fluctuations. This can also be considered in the product development.

In the context of hypothesis 3 the causes of high throughput-time-fluctuations are examined under closer inspection of the operations. Two data mining methods are used in the analysis, the association, analysis and classification analysis.

Initially all production orders are marked with their associated work operations (pivoting). Afterwards the dataset is clustered to the deviation of the throughput time (see Thesis 1). Based upon this preprocessing an association analysis is performed in two consecutive steps. The first step is a search of recurrent patterns in the data with a FP-Growth algorithm (FP - Frequent Patterns). These recurring patterns are afterwards used to automatically define association rules. These if-then rules show cause-and-effect relationships. The effect is predefined by the subject matter. In this example only rules with the conclusion "rising cycle time variability" are interesting for the analysis. Thesis 4 is very similar to thesis 3 but instead of operations, work systems are considered.

The results of an analysis of thesis 3 and 4 are relatively difficult to interpret. If certain operations or work systems are often used together, e.g. a milling and deburring operation, it is difficult to identify which of these operations is responsible for a high throughput time fluctuation. Therefore, no direct conclusions should be drawn from the analysis results. The causality of the results can only be checked by comparing the analysis results with the production process.

2.3.2 Analysis of Service data

For the analysis of service data three analysis questions were designed and implemented. One of these questions will be explained with a short example.

In this example the user wants to analyze the predefined analysis question “Some maintenance measures show a high discrepancy between planned and actual processing time. What have these measurements in common?”. The first step of this analysis is the specification of the subject of investigation. The user has to decide whether he wants to analyze all data or just a subset of the data, e.g. only maintenance measures concerning the brake system. The next step is to define the threshold for a “high” discrepancy between planned and actual processing time. After this specification the user can start the automated data analysis. In this example a decision tree is generated automatically (see. fig. 5).

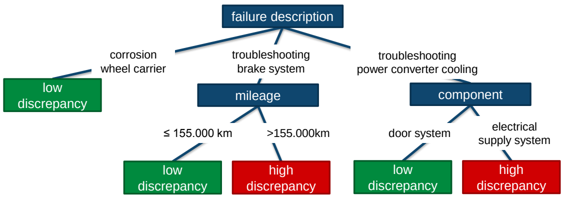


Fig. 5 decision tree

The generated decision tree shows under which conditions a high discrepancy between planned and actual processing time is likely going to occur. The setting parameters of this decision tree are optimization criteria, minimal split size, minimal leaf size, minmal gain, maximal depth, confidence, number of prepruning alternatives and pruning. All this setting parameters have been pre-configured for this analysis questions. Empirical tests using various datasets showed that the selected setting parameters achieve good results and can be commonly applied. If contrary to these tests the results of the automatically generated decision tree should not satisfy the expectation of the user, the user can choose between three different levels of detail. Through this preparation, users without detailed knowledge in the field of data mining can analyze life cycle data.

In this example a high discrepancy is likely when the failure description reads “troubleshooting brake system” and the mileage is higher than 155.000 km. Moreover, a high discrepancy is also likely when the failure description reads “troubleshooting power converter cooling” and the affected component is the electrical supply system. These identified patterns provide the user with potential starting points for a product improvement. The interpretation of the patterns has to be done by the user because data-mining algorithms can only show correlations and no causalities [9]. In this example the user has to decide whether the correlation between the mileage and discrepancy between planned and actual processing time is relevant or just a coincidence. But without the automated data analysis the user probably wouldn’t have found this correlation. Therefore, an automated analysis of lifecycle data can support designers to improve products.

2.4 SOFTWARE IMPLEMENTATION

The developed software tool is a demonstrator, not a marketable program, which serves to demonstrate the general functionality of the approach. The graphical user interface is implemented into the software jFast of the project partner GTT – Gesellschaft für Technologie Transfer mbH (see fig. 6).

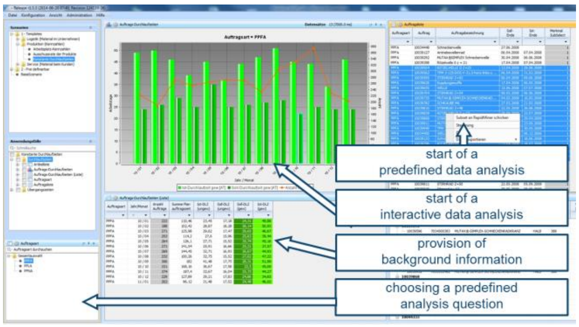


Fig. 6 User Interface

Within this user interface, the user can analyze the data manually or start the analysis of predefined questions. The results of the analysis are also shown in this graphical user interface. However, the actual data-mining is implemented with the software “Rapid Minder Studio 5”. The results of a data-mining process are automatically loaded into the user interface. So the analysis program works independently and remains hidden in the background. The advantage of this division of tasks is that the user can work in one very user friendly program. Besides that, the preparation of predefined analysis questions can be done in one of the most powerful analytics platforms. Similar to small applications for smartphones (“apps”), further analysis question can be added.

3 SUMMARY

If the potential of life cycle data is used better, weaknesses and experiences with current products can be easily identified and incorporated in the development of new innovative products. The developed software tool supports an effective analysis of life cycle data. By using data-mining techniques to identify potential relevant patterns within the data, the required time to analyze the data can be reduced. Moreover, even users without detailed knowledge about data-mining can use the developed software tool, as all data-mining algorithms and setting parameters are predefined. The developed tool can therefore support the users to transform data into knowledge.

4 REFERENCES

[1] Gantz, J.; Reinsel, D.: The Digital Universe in 2020; IDC study, online at http://idcdocserv.com/1414, 01.10.2014

[2] The Economist: “Data, data everywhere - A special report on managing information”, Volume Feb 27th, The Economist Newspaper Limited, London 2010.

[3] Fraunhofer IAIS: Big Data – Vorsprung durch Wissen Innovationspotenzialanalyse, online at http://www.iais.fraunhofer.de/fileadmin/user_upload/Abteilungen/KD/uploads_BDA/Innovationspotenzialanalyse_Big-Data_FraunhoferIAIS_2012.pdf, 01.10.2014.

[4] Fayyad, U. M. et al.: From Data Mining to Knowledge Discovery In: Fayyad, U. M. et al. (Hrsg.): Advances in Knowledge Discovery and Data Mining, AAAI Press, Cambridge 1996.

[5] Bentlage, A.; Zenker, M.; Kind, C.: Wissensschatz Lebenszyklusdaten. In: ERP-Management, GITO Verlag, Hannover 2013

[6] Rebstock, M.; Fengel, J.; Paulheim, H.; Naujok, K.-D.; Huemer, C.; Röder, P.; Tafreschi, O.: Ontologies-Based Business Intergration, Spinger Verlag, Heidelberg 2008.

[7] Ester, M. et. al.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: Simoudis, E. et al. (Hrsg.): Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, AAAI Press, Menlo Park 1996.

[8] Esmaelnejad, J. et al.: A novel Method to Find Appropriate ε for DBSCAN. In: Nguyen, N. T. et al. (Hrsg.): Intelligent Information and Database Systems: Second International Conference, Springer Verlag, Berlin 2010.

[9] Han, J.; Pei, J.: Mining frequent patterns by pattern-groth: methology and implications. ACM SIGKDD Explorations Newsletter, Volume 2, Issue 2, New York 2000.

Author: Bentlage, Aaron

University/ Company: IPH – Institut für Integrierte Produktion Hannover gGmbH

Department: production automation

E-Mail: bentlage@iph-hannover.de

Author: Zenker, Michael

University/ Company: IPH – Institut für Integrierte Produktion Hannover gGmbH

Department: logistics

E-Mail: zenker@iph-hannover.de