Cluster Rule Based Algorithm for Detecting Incorrect Data Records

Nadia El Bekri
Fraunhofer IOSB
Karlsruhe, Germany
nadia.elbekri@iosb.fraunhofer.de

Elisabeth Peinsipp-Byma
Fraunhofer IOSB
Karlsruhe, Germany
elisabeth.peinsipp-byma@iosb.fraunhofer.de

Andre Syndikus
Karlsruhe Institute of Technology
Karlsruhe, Germany
andre.syndikus@kit.edu

***Abstract* - Software applications have become an indispensable integral part of this world. In all areas of everyday life they are used to store information. Users of software applications rely on the data correctness. Incorrect data within the data set can cause a reduced user acceptance. To avoid incorrect data sets the process of knowledge discovery in databases (KDD) is a powerful instrument. The application of this process comprises five different steps. The steps are applied successively. One of the core steps is the use of data mining algorithms. This paper outlines the possibilities of combining various data mining algorithms to improve the correctness of the data.**

Keywords - Data Mining; KDD; Cluster Analysis; Association Rules

I. INTRODUCTION

One of the challenges of today's software applications is in the field of data quality caused by the huge amount of data sets. The example that is considered in this work is manually added data through different of users. The data set has a direct influence on the usability of the software. The goal is to design a quality assurance process for databases. In contrast of various other data cleaning techniques, the designed quality assurances process depicts a continuous process. The continuous assurance process ensures that the data quality of the data set do not decrease at any time. The continuity of the quality assurance process will be achieved by the direct correction of the actual incorrect record. The user corrects the error autonomous. Incorrect records can occur by users while creating new records. This can extend from simple misspellings errors to an incorrect handling of the software. For example by manually adding records to a library database a wrong location of the books can occur easily. The insertion of duplicates would be an example of an incorrect use of the software. In order to locate such errors within the database it requires the application of automatic procedures that discover these types of errors. Discovering the incorrect data manually is not feasible within large data set. In this case every data record needs to be validated which would be a very time consuming and error-prone process. The automatic identification is possible by extracting knowledge from the already stored data sets. Therefore an adjusted version of the Knowledge Discovery in Databases process with a focus on the combination of data mining algorithms is applied. The KDD process extracts knowledge from the existing dataset and serves therefore as abstract framework. The KDD process consists of five steps: the selection, the pre-processing, the transformation, the data mining and the interpretation. A more detailed description of the KDD process is described in the work of Fayyad et al. [1]. The goal is to use the extracted knowledge and give users while adding a new data set hints on possible errors they entered. Moreover the system offers proposals for the correction possibilities based upon the previous extracted knowledge.

II. RELATED WORK

There are several procedures and systems that maintain the data quality for a data set. First, various works in the field of the data cleaning procedures are illustrated to achieve a high quality data set. Data cleaning procedures are used previous the data mining step in the step of the preprocessing. A detailed description of the data cleaning process is explained in the work of Müller et al. [2] and Raman et al. [3]. The work of Batini et al. [4] illustrates a detailed overview of the data cleaning types. The first step of the data cleaning procedure is often named different. For example, it is named Data Auditing, Data Assessment or Data Analysis. In the following, the first step of the procedure is designated as Data Auditing. This step searches for rules and patterns that lead to the conclusion of possible errors. The second step is the transformation of the data with the found rules and patterns. The transformation causes the correction or the elimination of the error. In the last step the results are checked by the users and if necessary conflicts can be solved then.

One application in the field of data cleaning is named *Potter's Wheel.* This application is introduced in the work of Raman et al. [3]. The goal is to delete errors from the data set. By means of a graphical user interface the user implements operations that modify the data set. Thereby the user gradual creates a sequence of operations that are collected in one transformation. An elimination of the errors within the data set is thereby done by the transformation as well.

III. Definition of Data Quality

The International Organization for Standardization defines the term quality as: “Quality is the degree to which a set of inherent characteristics of an object fulfils requirements“. The quality of a data set is than measured by how many of the system requirements are complied. The requirements differ from user to user. Every user has its own idea of an application and thereby defines his own quality goals. Quality is therefore a measure that cannot be defined as a general statement because it depends on the subjective impression of every user. So called data quality assessment processes circumvent the problem by defining for every process an own quality goal. This definition makes it possible to define dimensions that describes the quality of any data set. The four basic dimensions in Batini et al. [4] are accuracy, completeness, consistency and timeliness. Although most data quality assessment methods integrate all these dimensions, the exact definitions of the dimensions differ and are based on an intuitive understanding. El Bekri et al. [5] offers a more detailed description of the dimensions.

Different types of errors lead to a reduced quality of the data set. In order to find different type of errors, it is important to classify different error types. One distinction is between the involved amounts of data sources. Afterwards it needs to be distinguished between the scheme and the instance level. Table I illustrates the classification for data errors. The scheme level describes the object type level, the instance level object level.

Table I. Instance and Scheme level

Scheme Level	Instance Level
Uniqueness, attribute dependencies…	Typing error, missing values…

IV. Methodology

This chapter describes the rule-based identification of the incorrect data set. Through the application of the rule-based error identification two aspects should be classified automatically. The first aspect that needs to be identified are wrong values, the second are missing values. Both of these quality aspects can be traced back to incorrect user behaviour. With the help of the quality assurance process and the user interaction the data set than needs to be corrected. In order to recognize errors through an association rule-based approach the FP-Growth algorithm is used.

The data analysis using association rules is one of the most common used data-mining procedures. Association rules illustrate frequent occurring dependencies within the data set. The FP-Growth algorithm uses as index structure the frequent pattern tree. With the help of this index structure it is not necessary to generate frequent item sets. This reduces the runtime of the association analysis. In the first step the Frequent Pattern (FP) Growth algorithm detects association rules by counting the relative frequency. Afterwards all items that did not reach the *minimum support (minsup)* are discarded. After preparing the transaction, the construction of the FP-tree follows. For the construction of the FP-Tree step by step nodes are added regarding the items. This procedure repeats until every transaction of the database is represented by a tree. A detailed introduction of the FP-Growth algorithm is illustrated in the work of Han et al. [6]. This work presents an algorithm that uses a cluster analysis before the association analysis with the FP-Growth algorithm and named in the following *Clustered Rule Based Algorithm* (*CRB)* algorithm.

First, the DQM algorithm of Hipp et al. [7] is discussed because the CRB algorithm is based on this. Subsequently, the extension of the algorithm is then described.

The DQM algorithm identifies errors by measuring a likelihood of the error for the data record. Based upon the association rules a key is calculated in which the case of the “if, then” rules matches. Based on a list *R* of association rules and their confidence values, the key number *s (d)* is calculated for every data record. A record *d* violates against an association rule *r = X* → *Y*, if the premise *X* of the association rule is satisfied but the consequence *Y* not. For the data set this means it contains the items of the premise *X* but at least one item of the consequence *Y* is absent or incorrect. For example, the data record *d = {A, B, F}* violates the association rule *A* → *CY*, but not the association rule *B* → *AF*. The function *violatesData (d, r)* is one if the record *d* violates against the association rules *r* and otherwise zero.

$$s(d) = \sum_{k=0} \text{Confidence } (r)^{r} * violates(d,r) \qquad (1)$$

The index *s (d)* is calculated from the summed confidence values of the association rules, to which the record is not conform. The parameter $r \in R_0$ serves for weighting the confidence values. The bigger r is selected, the higher the confidence values are weighted.

Hipp et al. [7] state out that for this application association rules with a higher confidence value of 0.75 are of a particular interest. The confidence value of an association rule provides information about the likelihood of the occurrence. The calculation of the key value *s (d)* illustrated that the confidence value is related to occurrence of errors within the data set. A data set that violates against an association rule with a high confidence value obviously contains an error. The importance of an association rule is very strong related to the confidence value. At this point the CRB algorithm takes part. Although, the confidence value is the crucial factor while the error identification the DQM algorithm can sort out association rules with a high confidence value. The filtering of the association rules is done by the FP-Growth algorithm. The problem by using the FP-Growth algorithm for the association analysis is that in some cases, rules with a high confidence value are sorted out because rules with a low support value are not generated and thereby considered. For example the association rule *{Tesla} = {electric motor}* is maximal because Tesla only produces vehicles with electric motors. The basic issue is that Tesla only produces a few models, Tesla does not appear often in the data set. As consequence the support of the item set Tesla is very low. The FP-Growth algorithm does not produce this kind of rule then because the item sets with Tesla have a low support. There are two different approaches to solve this problem. First, the threshold value of the *minsup* can be decreased. Second, the size of the data set can be changed to influence the relative frequency of the item. The approach of decreasing the value *minsup* increases the amount of the found association rules enormous. Hence, this approach is not further pursued. In addition for calculating the key value all association rules must be reviewed. This would cause an increased runtime. The second approach is illustrated in the work of Goller et al. [8] and Plasse et al. [9] by combining various data mining algorithms. There the cluster analysis is performed before the association analysis with the Apriori algorithm. By combining those two algorithms, the problem can be minimized. The results of the association analysis are improved by the previous executed cluster analysis.

Figure 1 and Figure 2 illustrate the two different approaches of the DQM and the CRB algorithm. The DQM algorithm only generates association rules for the biggest clusters, data sets from smaller clusters are not considered because of the low support value (*minsup*). In comparison with the DQM algorithm the CRB algorithm generates clusters. After generating the clusters the association analysis is performed on every cluster. In addition a classificator is trained on the clusters that is used to add new data directly a cluster. The allocation is necessary in order to apply the correct list of association rules for the error identification. In order to check the belonging for a data entry to a specific cluster the previous trained model is used. The model assigns the new data entry a cluster and thereby specifies the association rules. The next step calculates the key number. The detailed explanation for the calculation of the key number is described in the work of Hipp et al. [7]. An error is present if the calculated key number is above the threshold value.

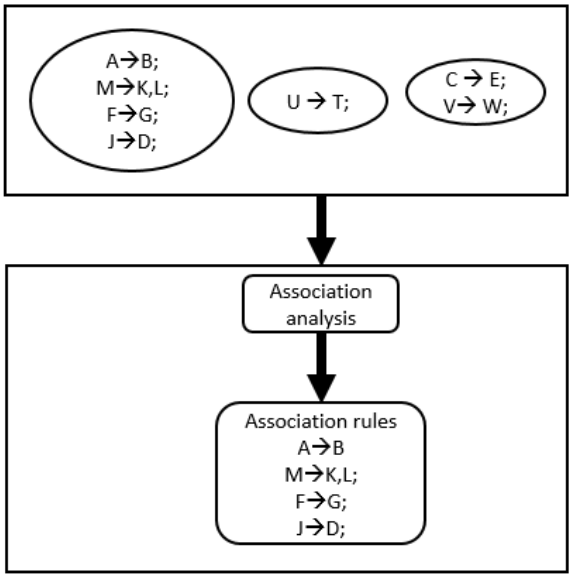


Figure 1. DQM algorithm.

An open issue in the work of Grimmer et al. [10] is the correction of the encountered error. This work solves this problem through the involvement of the user. Thus, after an error has been detected, an information is displayed for the user. The information contains the association rules that the present data entry violates against. The following example illustrates the process of this interaction with the user.

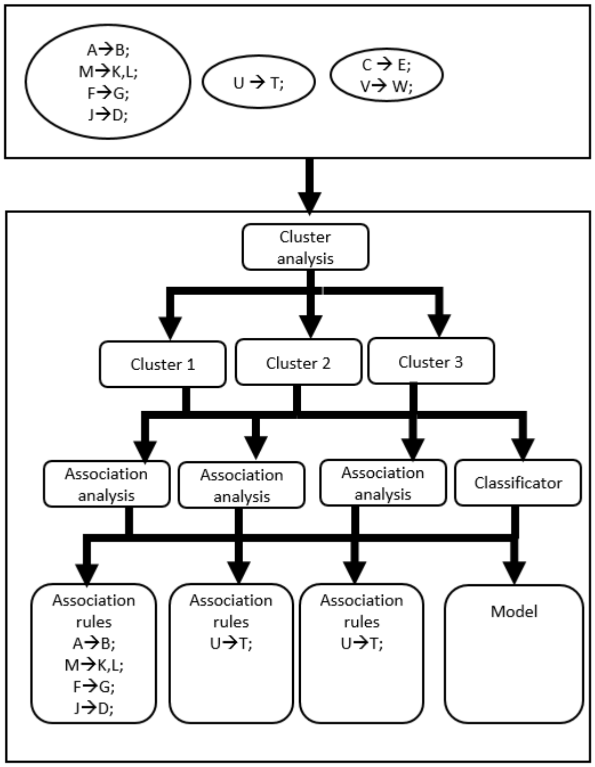


Figure 2. CRB algorithm.

Figure 3 outlines the conceptual flow of interaction between the user. First, the user creates a new data object. In this example, the user creates a data entry with the manufacturer *Tesla* and as type of drive *petrol engine*. Before the association rule *{Tesla}* → *{electric motor}* with a maximum confidence value from the existing data was generated. With the application of the CBR algorithm the new data record is verified, whether the new entry, is violating against an association rule. In this case, the quality assurance process recognizes that the record violates against the association rule *{Tesla}* → *{electric motor}*. The new record contains the premise *Tesla*, but not the consequence electric *motor*. The user will be notified that there is a possible error. Afterwards the user can decide whether the error information was correct or incorrect.

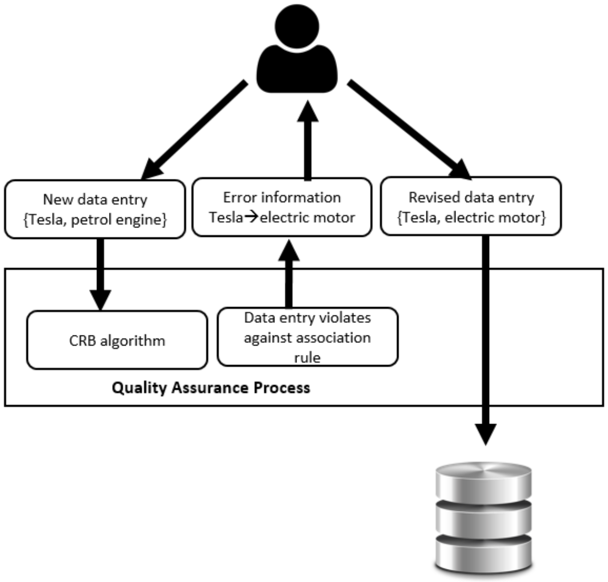

New data entry {Tesla, petrol engine}
Error information Tesla→electric motor
Revised data entry {Tesla, electric motor}
CRB algorithm
Data entry violates against association rule
Quality Assurance Process

Figure 3. Process of rule based error identification.

V. Evaluation

In this section, the CRB and the DQM algorithm are evaluated by different key numbers.

A. *The data set*

The evaluation dataset contains various data of vehicles, which are available for purchase in the United States. This database is intended to allow U.S. citizens to find the most economical and most environmentally friendly vehicle. For this purpose, the database contains a variety of key numbers for consumption and emissions of individual vehicles. For the consumers the key numbers are available online [12].

Both algorithms are trained on the data set of vehicles between 2013 and 2015. The evaluation is done by using the test data set of vehicles from the year 2016.

B. *Manipulation of the test data set*

This part describes the generation of artificial errors. The goal is to produce errors that are really close to real errors. As a starting point to generate the incorrect data, the data of vehicles from the year 2016 is used. This record contains 1189 vehicles. The correct data sets of the vehicles from 2016 will undergo some transformations to produce errors.

First, 112 random records are copied. The data then contains 112 duplicates. Since the transformations to generate artificial errors are also executed on the duplicates, duplicates after the error generation will not be necessarily identical. This approach complicates the identification of duplicates and checks the fuzzy matching of the duplicate detection. The generation of incorrect data is done in three ways. First, the swapping of attribute values of two random vehicles, the removing of random attribute values and the insertion of random values. Thereby the test data set contains 728 incorrect and 572 correct records.

C. *Procedure for evaluation*

First, the different cases of error identification are considered. For this purpose, Dietterich et al. [11] describe the creation of a truth matrix. Table II illustrates the four possible outcomes of the prediction. If the data record is incorrect and is recognized as an error, it is the case true positive. The false positive case describes the result that a correct data record was incorrectly marked as an error. If the process predicts no errors within the data set there are two cases: false negative and true negative. False negatives refers to the case that an incorrect data record is not recognized as such by the process. The case true negative predicts correct data record is also predicted to be correct.

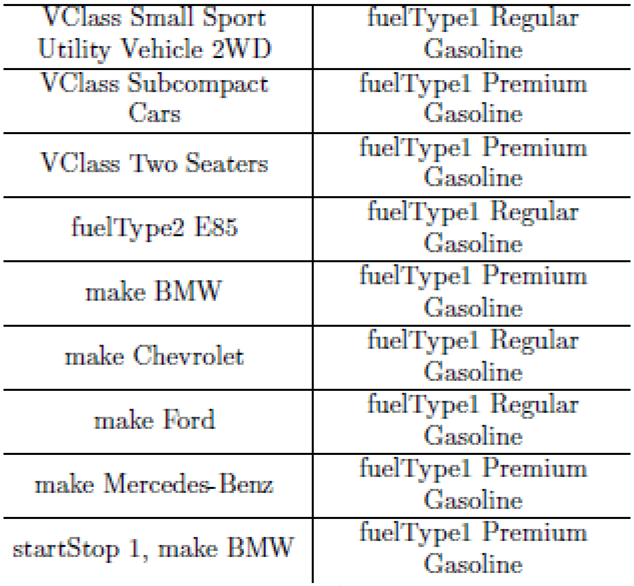
TABLE II. CONFUSION MATRIX.

	Incorrect data set	**Correct data set**
Error predicted	Right positive	False positive
No error predicted	False Negative	Right positive

From the indicators of Table III, the conditional probabilities *hit rate, accuracy* the *default rate* can be estimated. The hit rate describes the probability that an error is predicted correctly when an incorrect data record is present. The accuracy is the probability of predicting a failure properly. The default rate considers the probability, with which an error by the model is indeed predicted, but a correct data record set is present. With these indicators the evaluation is done.

D. *The evaluation*

Already at the extraction of the association rules, a clear difference between both algorithms is evident. The DQM algorithm uses the FP-Growth algorithm without the clustering analysis. This provides very few association rules with a confidence value above the required threshold value of 0.75. Figure 4 illustrates the found association rules.



VClass Small Sport Utility Vehicle 2WD	fuelType1 Regular Gasoline
VClass Subcompact Cars	fuelType1 Premium Gasoline
VClass Two Seaters	fuelType1 Premium Gasoline
fuelType2 E85	fuelType1 Regular Gasoline
make BMW	fuelType1 Premium Gasoline
make Chevrolet	fuelType1 Regular Gasoline
make Ford	fuelType1 Regular Gasoline
make Mercedes-Benz	fuelType1 Premium Gasoline
startStop 1, make BMW	fuelType1 Premium Gasoline

Figure 4. Association rules from DQM algorithm

The found association rules are applied to the test dataset in order to identify erroneous records. For example, the rule *{make: BMW}* → *{fuelType1: Premium Gasoline}* states that vehicles from the manufacturer BMW have with a probability of 90 % as primary fuel type Premium Gasoline. All records of the manufacturer BMW that are entered with a different primary fuel type will be marked as errors.

All found association rules are in less than 10% of the data set. These nine association rules cover the dataset barely. They are unsuitable for a sufficient error identification. This also confirms the evaluation by the test dataset.

By applying the association rules from Figure 4 on the test dataset of vehicles from the year 2016 only 38 of 273 incorrect records are detected. More detailed results are illustrated in Table IV.

TABLE III. KEY INDICATORS FOR RULE BASED ERROR IDENTIFICATION.

Parameters	**Hit rate**	**Accuracy**	**Default rate**
DQM minConf 0,75t = 0,5	0,11	0,5	0,03
DQM minConf 0,75t = 0,75	0,09	0,62	0,01
CRB minConf 0,6t = 0,5	0,16	0,43	0,05
CRB minConf 0,6t = 0,75	0,15	0,47	0,04

With the lowest threshold of 0.25, the DQM process reached a hit rate 0.15. This corresponds to the hit rate of the CRB error identification with the highest threshold value of 0.75. Furthermore, it is observed that with the increase of the threshold value the accuracy of both processes increases and the failure rate decreases. The increase of the threshold value sorts out data sets with a low probability of error occurrence.

VI. CONCLUSION

The confusion matrix demonstrated that the CRB process in contrast to the DQM method achieved an improvement for the rule-based error identification. The results have determined that the cluster analysis can be a promising approach to use before the association analysis. Especially with large volumes of data it is necessary to perform a cluster analysis. An important aspect is that the results depend on the quality of the cluster analysis. This means that results will show a potential improvement of the CRB process for the error identification through an improved cluster analysis.

ACKNOWLEDGMENT

The underlying project to this article is funded by the Federal Office of Bundeswehr Equipment, Information Technology and In-Service Support under promotional references. The authors are responsible for the content of this article.

REFERENCES

[1] U. Fayyad, P. Gregory, S. Padhraic, “From data mining to knowledge discovery in databases,” *AI magazine*, vol. 3, pp. 37 -39, 1996.

[2] H. Müller, J.C. Freytag, “Problems, methods, and challenges in comprehensive data cleansing,” 2005.

[3] V. Raman; J. M. Hellerstein, “Potter’s wheel: An interactive data cleaning system,” *VLDB*, vol 1, pp. 381–390, 2001.

[4] C. Batini, C. Cappiello, C. Francalanci, A. Maurino, “Methodologies for data quality assessment and improvement,” *ACM Computing Surveys (CSUR)*, vol 41 , p. 16, 2009.

[5] N. El Bekri, E. Peinsipp-Byma, ”An Approach for Min(d) the Quality of Data,” *The 2015 International Conference on Data Mining (DMIN)*, pp. 62-64, 2015.

[6] J. Han, J. Pei , Y. Yin, “Mining frequent patterns without candidate” generation” *ACM SIGMOD*, vol. 29, pp. 1–12, 2000.

[7] J. Hipp, U. Güntzer, U. Grimmer, “Data Quality Mining-Making a Virute of Necessity,” *DMKD*, 2001.

[8] M. Goller, M. Humer, M. Schrefl, “Beneficial Sequential Combination of Data Mining Algorithms,” *ICEIS*, vol 2, pp. 135–143, 2006.

[9] M. Plasse, N. Niang, G. Saporta, A. Villeminot, L. Leblond, “Combined use of association rules mining and clustering methods to find relevant links between binary rare attributes in a large data set,” *Computational Statistics & Data Analysis 52*, vol. 1, pp. 596–613, 2007.

[10] U. Grimmer, H. Hinrichs, “A Methodological Approach to Data Quality Management Supported by Data Mining,” *IQ*, pp. 217–232, 2001.

[11] T.G. Dietterich, “Approximate statistical tests for comparing supervised classification learning algorithms,” *Neural computation*, vol 10 , pp. 1895–1923, 1998.

[12] https://www3.epa.gov