Metadata–based data auditing

H. Hinrichs, T. Wilkens

Oldenburg Research and Development Institute for Computer Science Tools and Systems (OFFIS), Germany.

Abstract

Data quality is one of the most challenging problems today’s organisations have to cope with. Knowledge discovery in databases and data mining seem to be promising concepts with regard to error detection and correction in databases. In this paper, we propose a process model for mining–based data cleansing, so-called data auditing, which we derived from the ”classical” KDD model. Furthermore, we present a software system for data auditing that implements this process model. The auditing system makes extensive use of metadata, providing a flexible adaptability to various application domains. Our metadata model is based on an established standard to ensure system interoperability. The prototype of our data auditing system has been evaluated by means of cancer registry data.

1 Introduction

The increasing popularity of data warehouse systems (Inmon [1]) reflects the rising requirement to make strategic use of data integrated from heterogeneous sources. Data integration often reveals deficiencies of data quality, e. g. inconsistent and incomplete data. To avoid the so–called ”garbage in, garbage out” effect, it is inevitable to cleanse warehouse data before evaluating it. As a rule, cleansing is made difficult by large data quantities and hidden data semantics. Since not all inconsistencies can be detected by applying explicit integrity rules, it is necessary to work out a supplementary solution concerning the data quality problem.

The concepts of knowledge discovery in databases (KDD) and data mining (Fayyad et al. [2]) seem to be predestined for such an approach because they were construed to discover previously unknown regularities in large databases. Data values that do not obey these regularities are exceedingly suspicious of being in-
consistent. By correcting such inconsistencies and inserting previously missing values according to the regularities, the quality of a data set can be improved.

Starting at this point, our aim was to develop a generic software system for data mining–based data cleansing, so–called data auditing. The attribute "generic" corresponds to the intention that the auditing system should be easily adaptable to various application domains. This flexibility is achieved by keeping domain–specific information within a standardised metadata repository.

The remainder of this paper is organised as follows: In Sect. 2, a process model for data auditing is proposed which was derived from the KDD model. In Sect. 3, we present a metadata–based software system that implements this process model. Related work is reviewed in Sect. 4. Section 5 describes an evaluation of our concepts and implementation by means of a real world application. Finally, an overview of future work is given in Sect. 6.

2 A process model for data auditing

The objective of data auditing is to detect and correct inconsistencies and to predict missing values. This differs from the usual objective of KDD, which is to detect significant, previously unknown patterns. To meet this difference, a process model for data auditing will inevitably deviate from the "classical" KDD model. Apart from differences within the particular subphases, which will be pointed out in the following sections, there are some aspects worth mentioning in advance:

We believe that data auditing has to be embedded into a holistic model of data quality management (DQM) (Hinrichs [3]). DQM consists of four sequential phases, namely data quality planning, measurement, analysis, and improvement (see Fig. 1).

Quality planning includes the specification of requirements on quality aspects of data (consistency, completeness, redundancy, etc.). Measurement of the actual quality and subsequent analysis of measurement results determine whether these requirements are fulfilled. If so, the data may be made available for further use. If not, improvement methods should be applied. Quality improvement comprises both data scrubbing (Chaudhuri and Dayal [4]) and data auditing. After execution
of an improvement process the measurement step has to be repeated. Suitable abort criteria should ensure the termination of this iterative process. If no sufficient improvements can be achieved, the data have to be rejected.

Since data auditing is supposed to detect data deficiencies that cannot be found by conventional scrubbing methods, we assume that data scrubbing has already been done when data auditing comes into play. For this reason, our process model for data auditing does not have an explicit data cleansing phase. Moreover, a data reduction and transformation phase is not necessary because data auditing has to operate on the original values of a data set. Since there are some data auditing-specific preparation steps that are dependent on the data mining task and method to be applied, we changed the order of the corresponding phases (see Sect. 2.1.3 and 2.1.4).

Our process model comprises two superior phases, namely a data analysis phase (see Sect. 2.1) and a subsequent data correction phase (see Sect. 2.2), each consisting of several subphases (see Fig. 2).

In contrast to many proprietary solutions, we assume that the data whose quality is to be assured are stored within a relational database system (RDBS) and not in flat files. A dedicated metadata repository manages schema information which is retrieved at runtime. This information is used to generate SQL queries dynamically, providing maximum flexibility. Apart from schema information, domain-specific information is being used during the data auditing process in order to be
able to benefit from explicitly defined relationships between attributes. These relationships can be specified as if–then rules which are evaluated at runtime, ensuring domain–independence and adaptability.

2.1 Data analysis phase

The data analysis phase comprises five sequential steps which are described in the following subsections.

2.1.1 Selection of target and base attributes

In this step, the user selects a subset of data that is going to be analysed by the subsequent data mining process.

First of all, one or more so-called target attributes are specified. These are the attributes whose data values are to be checked for quality deficiencies. For each target attribute, one or more so-called base attributes have to be stated. By means of the values of these base attributes, the quality of the target attribute values will be assessed. If, for example, 99 of 100 records with coinciding base attribute values show the same value in their target attribute, and only one record shows a different value, this one record is exceedingly suspicious of being inconsistent.

Target and base attributes do not need to originate from the same database table: If the target attribute belongs to a table A and there is a 1:mc, 1:m, 1:1, or 1:c relationship between table A and another table B, the base attributes may also be taken from table B.

The specification of base attributes is simplified by the metadata support of the process model. If domain–specific relationships between attributes are given, an automated preselection of base attributes in respect of a certain target attribute is performed. This preselection can be modified interactively by adding or removing base attributes. The selection of target and base attributes improves the quality of results, extremely reduces the search space and thus increases time efficiency.

2.1.2 Specification of the auditing objective

Data auditing is only suited for detecting and correcting quality deficiencies of some specific type. For example, it cannot be used to trace obsolete data. The main deficiency types that may be influenced by data auditing are data consistency and completeness. In this step, the user specifies which of these two types the data auditing process will concentrate on.

2.1.3 Specification of a data mining task and method

In this step, the user selects a data mining task and an associated method which is suited for performing this task. Our process model currently supports classification, concept description, prediction, dependency analysis, and deviation detection as mining tasks. The set of data mining methods currently comprises decision trees, clustering, k–nearest–neighbours, and rule induction. Other methods like neural networks, regression, and discriminant analysis will be added later. However, not all task–method–combinations are qualified for data auditing in the same
Figure 3: Suitable task–method–combinations for data auditing.

way. Figure 3 shows the most promising combinations which we identified in the scope of a study described in Wilkens [5].

2.1.4 Data auditing specific preparations

Before a data mining process for data auditing can be initiated, some prearrangements have to be made, dependent on the task–method–combination specified in the previous step.

In order to reduce the number of regularities that will be found, similar attribute values can be grouped into categories (sets of intervals in case of numeric attributes) each of which is then treated as one superior value.

Furthermore, the user has to decide how null values (indicating missing values) concerning base attributes are to be handled. To understand the following comments, it is necessary to know that many data mining methods distinguish between training records and test records. Training records are database records which help the respective mining method to build up basic “knowledge” of characteristics of the data set under consideration. This knowledge is used afterwards to check if the test records also show these characteristics. Based on this distinction, our process model offers two different strategies to handle null values: The first is to always take training records with null values in base attributes into account, entailing a representative view on the data set. The second is to consider such records only if at least one test record also contains null values in its base attributes, thus reducing analysis complexity, e.g. by omitting null value edges in decision trees.

Training records containing a null value in their target attribute are not considered since they do not supply useful information for data auditing.

2.1.5 Data mining

Based on the specifications made in the previous steps, an automated data mining process is performed. Information about potential data defects detected during this
process is written to a database. In the data correction phase of data auditing (see Sect. 2.2), these results are evaluated interactively. Although a fully automated data correction would also be imaginable, we preferred the interactive solution in order to be able to verify and tune system behaviour.

2.2 Data correction phase

The data correction phase consists of three subphases. The subphase “presentation of results” plays a central role since it represents the starting point from which the execution of the other phases can be triggered.

2.2.1 Presentation of results

In this phase, the analysis results currently stored in the database are presented to the user. For each result record, the user may request the correction of the corresponding database record (see Sect. 2.2.2) and/or the deletion or archiving of the result record, respectively (see Sect. 2.2.3). With regard to the interaction of the analysis and correction phases, it is particularly possible to analyse the same subset of a database multiple times using different data mining methods, without having to execute a correction process between two analysis processes.

2.2.2 Correction of target data

The objective of this step is to change the values of a database record which was determined to show a qualitative defect. To reach this goal, a so-called correction instruction is created whose execution triggers an update of the database record under consideration. This instruction is formulated as an SQL UPDATE statement that may be modified by the user with regard to which attributes should be set to which values. Every single database update is logged in the corresponding result record to enable backtracking in case of subsequent inconsistencies caused by the update.

2.2.3 Deletion or archiving of results

When the user has finished the processing of a result record, he/she may initiate its deletion. Alternatively, the record may be archived. A deletion or archiving should be performed only if the concerned record is not needed anymore, i.e. either if it describes an exception to a rule instead of a data defect or if the corresponding database record has been corrected without entailing subsequent inconsistencies.

3 Architecture of the data auditing system

Based on the process model described in Sect. 2, we developed a software system called MEDAS (Metadata–based Data Auditing System) and implemented it prototypically.
3.1 System components

The system architecture of MEDAS comprises four layers (see Fig. 4). At the top layer, a graphical user interface handles all user interactions. It encapsulates the functionality of the data auditing kernel below. This kernel controls both the analysis and the correction processes, making use of a data mining library (exchangeable), a repository engine, and a rule engine. While the repository engine manages metadata, especially schema information, the rule engine is responsible for inferring knowledge from domain-specific business rules. The auditing kernel accesses business data and analysis results – stored within a relational database system – via a standard database interface. Since many data mining libraries are file-based, an interface to the file system is provided as well.

3.2 Implementation

A prototype of MEDAS has been implemented on the Windows platform using Visual C++ 6.0. As a data mining library, we chose MLC++ (Kohavi et al. [6]) which provides decision trees, clustering, k-nearest-neighbour, and rule induction algorithms. Up to now, merely the decision tree algorithm ID3 has been incorporated into MEDAS, but we are eagerly working on the integration of additional algorithms. Moreover, we integrated the rule engine of ILOG Rules [7] and the Microsoft Repository engine (Bernstein et al. [8]). Our metadata model is based on the Open Information Model (OIM) defined by the Meta Data Coalition [9]. Sup-
porting this established metadata standard guarantees interoperability. Metadata are accessed by the Microsoft Repository via Component Object Model (COM) calls. Business data and analysis results are managed by Oracle8i.

4 Related work: WizRule

The commercial data auditing tool WizRule by WizSoft [10], Israel, is the most prominent representative of its kind. Before a data set is analysed, the user of WizRule may select relevant attributes, specify the data format of numbers, currencies, and dates, and finally pinpoint the lower limit of rule accuracy and the minimum number of cases required to establish a rule.

WizRule generates if-then rules like “if product is 'Notebook xy’ then branch is 'London’”, formula rules like "sum_price = unit_price * number_of_units”, and spelling rules like "value 'London’ appears 25 times in the branch field, 1 case with a similar value ('Lundon’)”.

The merits of WizRule lie in its support of multiple data formats (MS Access, ODBC, ASCII, etc.), a very fast algorithm, and a convenient graphical user interface. On the other hand, WizRule provides merely a rule induction algorithm. Other data mining methods like, for example, nearest neighbour or decision tree algorithms are not supported. Since it does not offer an open system architecture, WizRule is hardly extensible and difficult to integrate into a superior system for data quality management. The proprietary nature of WizRule (no metadata standard supported) restrains interoperability and sharing of resources.

5 Evaluation

To evaluate the MEDAS prototype and compare it to WizRule, we applied it to cancer registry data. Epidemiological cancer registries constitute an important foundation of research in the field of cancer documentation and prevention (Jensen et al. [11]). Of course, the quality of epidemiological studies can be only as good as the quality of the underlying data. Therefore, a quality assurance of cancer registry data is absolutely necessary.

5.1 Evaluation method

As test data, we chose a set of tumour records which had been sent to the cancer registry of Lower-Saxony, a federal state of Germany, by one of its reporting organisations. The test data had already been checked to suffice the given representational format and basic integrity rules like domain value ranges. Altogether, the test data comprised 706 tumour records. Although this data set is quite small, it should be sufficient to test the functionality of MEDAS. (Furthermore, the demo version of WizRule 3.2A, to which MEDAS should be compared, is restricted to 1,000 records anyway.) We also plan to evaluate MEDAS using larger data sets from different domains, especially with regard to time efficiency.
Table 1: Configuration (excerpt) and results of a selected test run.

<table>
<thead>
<tr>
<th></th>
<th>WizRule</th>
<th>MEDAS prototype</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mining method</td>
<td>Rule induction</td>
<td>ID3 Decision tree algorithm</td>
</tr>
<tr>
<td>Attributes considered</td>
<td>Gender, site code</td>
<td>Target attribute: Gender</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Base attribute: Site code</td>
</tr>
<tr>
<td>Objective</td>
<td>(not selectable)</td>
<td>Data consistency</td>
</tr>
<tr>
<td>Null value strategy</td>
<td>Always considered</td>
<td>Always considered</td>
</tr>
<tr>
<td>Miscellaneous</td>
<td>Minimum accuracy: 95%</td>
<td>Percentage of test records per decision tree: 50%</td>
</tr>
<tr>
<td></td>
<td>Minimum number of cases: 20</td>
<td></td>
</tr>
<tr>
<td>Potential inconsistencies</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Confirmation (%)</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>Elapsed time (sec)</td>
<td>3</td>
<td>8</td>
</tr>
</tbody>
</table>

The following tumour attributes were considered: year and month of diagnosis, gender, morphology code, site code, behaviour, side, extent, and grading (codes according to the International Classification of Diseases for Oncology ICD-O, 2nd edition). Additionally, a numeric identifier was assigned to each tumour record. We then applied both the MEDAS prototype and WizRule to the data set.

5.2 Evaluation results

Table 1 summarises the chosen configurations and the obtained results of one selected test run that concentrated on the gender and site code attributes. In this test run, domain knowledge was used by preselecting attributes between which a correlation was expected. For example, prostate cancer is restricted to males, breast cancer to females. Whereas WizRule would not allow for a use of more complex domain knowledge, MEDAS can integrate arbitrary business rules that are expressible by means of the pattern-matching language OPS5 (Brownston et al. [12]).

After the test run, the potential inconsistencies that had been found by the software tools were checked by the medical staff of the registry. As Tab. 1 shows, the quality of the results was very good, both with WizRule and MEDAS. As expected, WizRule beat MEDAS with regard to time efficiency. This is due to the fact that MEDAS has not been optimised in view of time efficiency yet. To eliminate this shortcoming, we plan to integrate a cache that minimises the number of database accesses.

For us, the most important result of the evaluation was that MEDAS can keep up with a commercial data auditing tool without giving up the benefits of openness, interoperability, and a methodically founded process model.
6 Conclusion

In contrast to other data auditing tools like WizRule (cf. Sect. 4), MEDAS provides an open, highly modular architecture which is universally applicable and easily extensible. Moreover, it supports an established metadata standard, enabling sharing and reuse of information. The evaluation of MEDAS by means of cancer registry data has shown that significant amounts of inconsistencies could be detected which had not been found by the conventional data scrubbing algorithms that had been applied in advance.

Data mining methods seem to be very convenient to detect and possibly correct inconsistent and incomplete records in large databases. Extensive use of metadata facilitates the tailoring to domain-specific needs. The process model proposed in this paper offers guidelines for an effective data auditing.

Future work includes the implementation of multiuser support and the integration of MEDAS into a data quality management system providing, amongst other things, extensive quality planning and measuring facilities (cf. Sect. 2).

References