Data Pre-processing for knowledge discovery

Mortadha M. Hamad , Banaz A. Qader
College of Computer , University of Anbar

Abstract

Data pre-processing stage is also known as (data preparation) stage and it is a fundamental stage for data analysis and knowledge discovery. If there is much irrelevant and redundant information or noisy and unreliable data, then knowledge discovery during analysis and mining phase will be more difficult. Therefore we consider the pre-processing stage as an important step for knowledge discovery process and has a significant impact on predictive accuracy. Essentially, while each customer attribute may require special treatment for each algorithm, so the choices of data pre-processing (DPP) depend on the individual dataset or database used. In this paper we have chosen and explained two different pre-processing techniques which are (consistency, reduction) depending on our data warehouse of marketing which contains inconsistent attributes and also contains duplicated records. We have also proposed two new algorithms for reduction named (Removing Duplication Algorithm) and for consistency named (Resolving Inconsistency Algorithm) so that achieving the best performance for their data set. In this paper we applied and implemented our two new algorithms on our data warehouse using (C# programming language) and (Microsoft Access file), and gained cleaning data warehouse with consistent attributes and empty of duplicated records that is ready for preparing quality data as input to the algorithms of data mining process or any other analysis method which also influences of knowledge quality that is discovered during data mining process.

Keywords: data pre-processing, data mining, knowledge discovery, data cleaning.

1- Introduction

Recently, progress in computational and storage capacity has enabled the accumulation of ordinal, nominal, binary and unary demographic and psychographic customer centric data, inducing large, rich datasets of heterogeneous scales. Essentially, each customer attribute may require special treatment for each algorithm, such as discretization of numerical features, rescaling of ordinal features and encoding of categorical ones. The phase of data preprocessing (DPP) represents a complex prerequisite for data mining in the process of knowledge discovery in databases aiming to maximize the predictive accuracy of data mining [1]. In many computer science fields, such as pattern recognition, information retrieval, machine learning, data mining, and Web intelligence, we need to prepare quality data by pre-processing the raw data. Data preprocessing (preparation) is therefore a crucial research topic. However, much work in the field of data mining was built on the existence of quality data. The emergence of knowledge discovery in databases (KDD) as a new technology has been brought about with the fast development and broad application of information and database technologies. The process of KDD is defined as an iterative sequence of four steps which are (defining the problem, data pre-processing (data preparation) , data mining, and post data mining) [3]. So in this paper we study the pre-processing stage especially its some important techniques, where it is necessary to prepare the data for knowledge discovery and data mining. Because preprocessing stage consists of many analysis methods which transforms the raw data in to the cleaning and high quality data, this lead to get the accurate and high quality knowledge that is discovered during knowledge discovery phase by applying data mining.

Corporate data mining faces the challenge of systematic knowledge discovery in large data streams to support managerial decision making, because the application of each data mining algorithm requires the presence of data in a mathematically feasible format which achieved through DPP. Therefore, data pre-processing (DPP) represents a prerequisite phase for data mining in the process of knowledge discovery in databases (KDD). While research in operations research, direct marketing and machine learning focuses on the analysis and design of data mining algorithms, so there is need to interaction of data mining with the preceding phase of data pre-processing. Data pre-processing (DPP) tasks are distinguished in data reduction, aiming at decreasing the size of the dataset by means of instance selection and/or feature selection, and data projection, altering the representation of the data, e.g. mapping continuous variables to categories or encoding nominal attributes. While some of these are imperative for the valid application of a method, such as scaling for neural network, others appear to be more general to facilitate method performance in general [1].

2- Data Pre-Processing Definition

Data pre-processing is also known as (data preparation). It comprises those techniques concerned with analyzing raw data so as to yield quality data. It mainly includes data collection and integration, data transformation, data cleaning, data reduction, and data discretization [3].

Data pre-processing is an important step in the data mining process, because if there is much irrelevant and redundant information or noisy and unreliable data, then knowledge discovery during the analysis and training phase will be more difficult [7]. Data pre-processing is considered as a data mining technique that involves transforming raw data into an understandable format. Real-world data is often incomplete, inconsistent, and/or lacking in certain behaviors or trends, and is likely to contain many errors. Data preprocessing is a proven method of resolving such issues. Data preprocessing prepares raw data for further processing like (data mining). Data preprocessing is used in database-driven applications such as customer relationship management and rule-based applications (like neural networks) [10].

3- Importance of Data Pre-processing (Preparation)

Over the years, there has been significant advancement in data-mining techniques. This advancement has not been matched with similar progress in data pre-processing (preparation). Therefore, there is now a strong need for new techniques and automated tools to be designed that can significantly assist us in preparing quality data. Data pre-processing can be more time consuming than data mining, and can present equal, if not more, challenges than data mining [3]. High-performance mining systems require quality data, because quality data yields high-quality patterns. In order to investigate quality data, it is important to apply data preprocessing. Data preprocessing importance can be included in the following three tasks [3][6]:

1- Data pre-processing (preparation) generates a dataset smaller than the original one, which can significantly improve the efficiency of data mining. This task includes:

- Selecting relevant data: attribute selection (filtering and wrapper methods), removing anomalies, or eliminating duplicate records.
- Reducing data: sampling or instance selection.

2- Data pre-processing (preparation) generates quality data, which leads to quality patterns (knowledge). For example, we can:

- Recover incomplete data: filling the values missed, or reducing ambiguity.
- Purify data: correcting errors, or removing outliers (unusual or exceptional values).
- Resolve data conflicts: using domain knowledge or expert decision to settle discrepancy.

3- Data preprocessing is used because real-world data may be incomplete, noisy, duplicated and inconsistent, which can disguise useful patterns. This is due to the following reasons [3][9]:

- Incomplete data because of lacking attribute values, lacking certain attributes of interest, or containing only aggregate data.
- Noisy data because of containing errors or outliers.
- Inconsistent data because of containing discrepancies in codes or names.
- duplicated data because of containing duplicated records (transactions) or attributes with similar names or contents.

4- Categorizing of Data Pre-processing Techniques

The techniques that clean and prepare data for processing and mining to gain high quality and accurate knowledge can be categorized in to the following [4][5][9]:

- **Data Cleaning:** Data is cleansed through processes such as filling in missing values, smoothing the noisy data, or resolving the inconsistencies in the data which we explain and apply in this paper, as shown in figure (1).
- **Data Integration:** it includes the integrating of several databases and files in to one unified file or database as we have applied in this paper. Data with different representations are put together and conflicts within the data are resolved, as shown in figure (1) .
- **Data Transformation:** Data is normalized, aggregated and generalized.
- **Data Reduction:** This step aims to present a reduced representation of data in a data warehouse which we study in this paper. In many cases the amount of data is too huge, therefore it is necessary to reduce the utilized data without losing any of its predicting accuracy and describing ability, as shown in figure (1).
- **Data Discretization:** Involves the reduction of a number of values of a continuous attribute by dividing the range of attribute intervals.

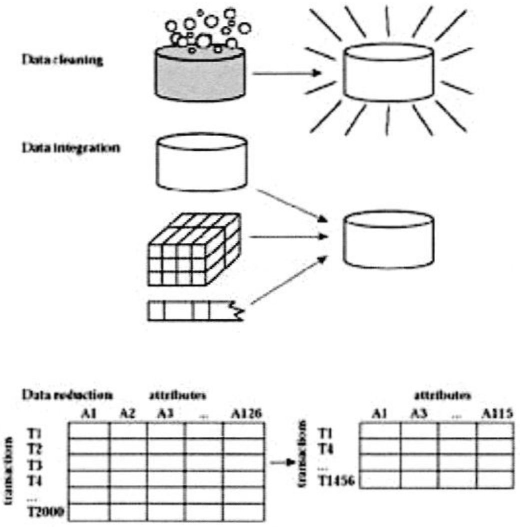

Figure (1) Types of Data Pre-processing Techniques [4]

5- Methodology of Applying The Proposed Pre-processing Algorithms on The Data Warehouse (DW)

In this paper, we proposed two new algorithms for preparing data in the data warehouse (DW) to be ready and prepared for further processing like Knowledge Discovery and Data Mining. in this section, we apply a methodology for applying and executing these algorithms on a marketing DW. this methodology consists of the following four steps :

5.1 Data Gathering and Integrating

In this stage which is the first stage of our work, we have collected data about sales and selling items of a market from several sources and files such as (text file, excel, access, ...etc) that have been existed in multiple sales departments of the market. Where collecting data from different sources usually presents many challenges, because different departments will use different styles of record keeping, different conventions, different time periods, different degrees of data aggregation, different primary keys, and will have different kinds of error. So the data must be assembled, integrated, and cleaned up during (preprocessing stage). In this stage after collecting data from different sources of market departments, we also integrate these different data files into one unified file which is (Microsoft Access file) to be ready for importing in to the C# environment for other data pre-processing techniques. Integration step of pre-processing may lead to appearing one or more of challenges which are: incomplete, irrelevant, redundant, noisy, inconsistent, duplicated and unreliable data that need pre-processing, and in this paper integration step led to occurring duplicated records (transactions) and inconsistent attributes.

5.2 Proposing Algorithms of Data Pre-processing

After data integration which led to appearance of duplicated records and inconsistent attributes in the data warehouse, In this stage we apply two techniques of data preprocessing which are consistency as one of the data cleaning techniques and data reduction technique. In this paper we have proposed and built new algorithms for the two techniques mentioned above and apply these algorithms on the data warehouse (DW) which special of marketing. The DW contains inconsistent attributes which is (currency) attribute and (sales_date) attribute, so we will apply the proposed algorithm to resolve the inconsistency and get consistent attributes. Our marketing DW also contains duplicated records which are removed and processed by applying the proposed algorithm for the reduction. We will explain these two data pre-processing techniques below as well as illustrating their algorithms:

a) **Data Reduction:**

There are two different concepts of data reduction, namely the attribute\ record reduction and the discretization of real value attribute. Both concepts are useful as data pre-processing methods for knowledge discovery process. Attribute\record reduction eliminates duplicated attributes\records and favors those attributes\records which are most relevant to the mining process. Discretization eliminates insignificant differences between real values by partition of real axis into intervals [2].

In this paper because of appearance duplicated records in the Data Warehouse, we trend toward the attribute\record reduction concept and removing all duplicated records by applying the following algorithm which we have proposed and built for records reduction.

1- *Algorithm of Reduction (Removing Duplication)*

- ***Input:*** *Table with duplicated records*
- ***Output:*** *Clean table from duplicated records*

```
1- start
2- set PROTABLE is new empty table
3- set DUPTABLE is the table that contains the duplicated records
4- set SPF is any specified field (attribute) which according to it the processing is performed, where $SPF \in DUPTABLE$
5- sort the DUPTABLE according to SPF ascending or descending.
6- begin
a.set PROROW is the first row of DUPTABLE
b. insert PROROW into PROTABLE
c.for each $ROW \in DUPTABLE$
d. If SPF of CURRENTROW not equal to SPF of PROROW then
e.insert CURRENTROW into PROTABLE
f. end if
g. PROROW= CURRENTROW
h. end for
7- end begin
8- end start
```

b) **Data Consistency**

It is one of the data cleaning techniques. Data that is collected from many different sources contains different formats, Where these different formats contain the same concept but the name of the data format is different [6][8]. In this paper, after data integration step, it is noticed that the creating Data Warehouse contains inconsistent attributes which are (currency and sales_date) as well as containing duplicated records. For resolving all inconsistent data formats in the inconsistent attributes, we apply the following algorithm which we have proposed and built for resolving inconsistency and make consistent attributes.

2- *Algorithm of Resolving Inconsistency*

- ***Input:*** *Table with inconsistent attributes*
- ***Output:*** *Clean table with consistent attributes*

```
1- start
2- set INCONTABLE is the table that contains inconsistent attributes
3- create new empty attribute named by PROSSFIELD in INCONTABLE
4- choose the inconsistent attribute named by INCONFIELD based on its type from INCONTABLE
5- Identify the type of INCONFIELD
a) If type of INCONFIELD is date then
       Call procedure Date-algorithm
b) If type of INCONFIELD is Currency or Temperature then
Call procedure Cur-Temp-algorithm
6- end start
```

2.a Procedure Cur-Temp-algorithm

```
1- begin
2- for each ROW in the INCONFIELD
```

```
3- convert the currency or temperature to string
4- split the currency or temperature in to two parts : numeric (num) and symbol (sym)
5- process the (num) part and convert it to the required and specified currency or temperature format
6- compare the (sym) part with the specified currency or temperature symbol named by (SPSYM)
   If (sym) equal to SPSYM then
      OK
   Else convert the (sym) to true SPSYM
7- re-merge the (num) part with the (sym) part
8- save the new format of the currency or the temperature in the PROSSFIELD
9- end for
10- end begin
```

2.b <u>Procedure Date-algorithm</u>

```
1- begin
2- for each ROW in the INCONFIELD
3- split the date format into parts D1, D2, D3 depending on specified format such as D1 represents the year , D2 represents the month and D3 represents the day.
   • If D3 greater than 31 then
      D3= D3 mod 31
   • If D3 equal to 31 and ( D2 equal to 4 or D2 equal to 6 or D2 equal to 9 or D2 equal to 11) then D3 =30
   • If D1 less than or equal to 0 then
      D1 = current year
   • If D2 less than or equal to 0 then
      D2 = current month
   • If D3 less than or equal to 0 then
      D3 = current day
4- re-merge the D1, D2, D3 into the required and specified format
5- Save the new format of the date in the PROSSFIELD
6- end for
7- end begin
```

5.3 Implementation

In this sub section and in this stage of applying pre-processing methodology, we implement and apply the two proposed algorithms which are (resolving inconsistency algorithm) and (removing duplication algorithm) on our marketing data warehouse (DW) using C# programming language and Microsoft Access file in order to get cleaning data warehouse (DW) with consistent attributes and empty of duplicated records and to be prepared as input to the algorithms of data mining technique for knowledge discovery or any other analysis methods. For performing implementation, we have designed an interface which through it we can identify the size of data warehouse (DW) to be processed and identify the pre-processing algorithms which are (resolving inconsistency algorithm) and (removing duplication algorithm) to be applied on the data warehouse (DW) to clean and prepare it for further processing techniques like data mining to discover knowledge as well as showing the results which will be discussed in the next sub section. The implementation interface also shows the data warehouse (DW) before and after applying pre-processing algorithms. This stage shows the execution of (removing duplication algorithm) and (resolving inconsistency algorithm) with showing the DW after and before removing duplication and emerging its results according to particular parameters like processing time, number of reduced records , number of records that have been processed for inconsistent attributes and rate of correct.

5.4 Results

During the implementation stage of applying pre-processing methodology and through the execution of the proposed algorithms on the marketing data warehouse (DW), we gained the results have been demonstrated below in tables (1) and (2) which mainly depend on time parameter.

As shown in table (1), the resolving inconsistency algorithm has been applied on different sizes of DW that are (100000 , 200000, 300000, 400000, 500000) records. It was noticed that when the DW size (number of records) increases, the consuming time of processing also increases, but still too little. By applying the resolving inconsistency algorithm on 100000 records of DW size , the processing time that it takes is(7 seconds) for scanning 100000 records and converting all inconsistent records which are (41687) to consistent records according to a specified attribute type which is (currency or date) in our DW, observing that by increasing the number of records to 200000 and applying the algorithm , the processing time also increases to (16 seconds) because this algorithm scans 200000 records, finding (83617) inconsistent records, and they are converted to consistent records during (16 seconds) which is little time, and thus for other different sizes of DW until reaching to applying this algorithm on the maximum size of the DW 500000 where in spite of increasing the processing time to (51 seconds), we observe that this time is little because through less than one minute this algorithm scans 500000 records once for a specified attribute (currency or date) and resolving (209564) inconsistent records. So these results lead to inferring that there is ejective proportion between processing (execution) time and DW size and inferring that increasing DW size (number of records) has a great influence on increasing the processing time. It was also proved that this algorithm can resolve inconsistent attributes at a very little time and give consistent quality data because of scanning large sizes of DW once.

Table (1) Results of Executing Proposed Resolving Inconsistency Algorithm

Database size for processing	Processing time	Rate of inconsistent records that processed	Rate of processing (consistency)
100000	00:00:07.72	41687/100000	41.68%
200000	00:00:16.96	83617/200000	41.80%
300000	00:00:26.81	125714/300000	41.90%
400000	00:00:38.49	167542/400000	41.88%
500000	00:00:51.36	209564/500000	41.91%

Through the execution of the removing duplication algorithm that has also been applied on different sizes of DW which are (100000, 200000, 300000, 400000, 500000) records and getting results that are shown below in table (2), we inferred that the parameters influenced on the processing time are DW size (number of records) that are scanned and number of duplicated records that are reduced by executing this algorithm. The increasing of DW size that will be processed by applying the algorithm of removing duplication (reduction) leads to increasing of processing time because of reducing larger number of duplicated records by this algorithm, but the consuming time for processing still little. By applying the removing duplication algorithm on 100000 records of DW size , the processing time that it takes is (1.45 second) for scanning 100000 records and removing all duplicated records which are (13257) , but by increasing the number of records to 200000 and applying this algorithm , the processing time is also increased to (2 seconds) because of scanning larger number of records and reducing larger number of duplicated records which is (26297) , and thus for other different sizes of DW. It was also proved that this algorithm can scan our DW of 500000 records and reduce (65308) duplicated records at a very little time that is just (6 seconds) and give quality data and cleaning DW empty of duplication.

Table (2) Results of Executing proposed Removing Duplication (Reduction) Algorithm

Data Warehouse size before reduction	Data Warehouse size after reduction	Processing time	Number of reducing records	Rate of reduction
100000	86743	00:00:01.45	13257	13.25%
200000	173703	00:00:02.60	26297	13.14%
300000	260670	00:00:04.21	39330	13.11%
400000	347611	00:00:05.82	52389	13.09%
500000	434692	00:00:06.99	65308	13.06%

As it has been shown in both above tables (1) and (2), the rate of processing (consistency) and (reduction) are still stable in spite of applying the (resolving inconsistency algorithm) and (removing duplication algorithm) on different sizes of our DW because by increasing the DW size that is scanned, the number of inconsistent records according to specified attribute that are resolved and the number of duplicated records that reduced are increased too, making the rate of processing approximately stable.

6- Conclusion

After the implementation of the proposed algorithms of Data Pre-Processing (DPP) which are (resolving inconsistency algorithm) and (removing duplication algorithm) and from obtained results , we concluded the following:

1- Data Pre-processing (DPP) is an important and necessary stage for knowledge discovery because Data pre-processing lead to create cleaning DW which will be ready and prepared for knowledge discovery process by using Data Mining technique or any analysis method and affect the quality and accuracy of discovering knowledge.

2- Efficiency of proposed resolving inconsistency algorithm to process large DW size with 500000 records at a very little time is (51.36 seconds), and efficiency of proposed removing duplication (reduction) algorithm to scan DW with 500000 records and removing (65308) duplicated records at a very little time is (6.99 seconds).

3- Performance and processing rate of both data reduction and data consistency techniques by using the proposed algorithms is high.

4- Applying resolving inconsistency algorithm and removing duplication algorithm on DW leads to create cleaning DW contains consistent attributes and empty of duplicated records and attributes during very little time and provide quality data for data mining and knowledge discovery.

References

1- Sven F. Crone, Stefan Lessmann, Robert Stahlbock ,"**The impact of preprocessing on data mining: An evaluation of classifier sensitivity in direct marketing**" , 2005, European Journal of Operational Research 173 (2006) 781–800.

2- Hung Son Nguyen, "**Approximate Boolean Reasoning: Foundations and Applications in Data Mining",** 2006, Warsaw University Banacha 2, 02-097 Warsaw, Poland.

3- Shichaozhang, Chengqi Zhang, Qiangyang, " **Data Preparation For Data Mining**", Copyright # 2003 Taylor & Francis, Applied Artificial Intelligence, 17:375–381, 2003.

4- Han, J. and Kamber, M. (2001), **"Data Mining: Concepts and Techniques"**. Morgan Kaufmann Publishers, USA.

5- Zhang, D. Ha, Q.L. and Lu, M. (2001**), "Mining California Vital Statistics Data"**. IEEE International Conference on Data Mining.

6- Baydaa Sulaiman Bahnam , (2006) , " **The Use of the Modified Clustering Algorithms in Data Mining"** , M. Sc. Thesis\ University of Mosul\chapter two.

7- S. B. Kotsiantis, D. Kanellopoulos and P. E. Pintelas, **"Data Preprocessing for Supervised Leaning"**, (2006), INTERNATIONAL JOURNAL OF COMPUTER SCIENCE VOLUME 1 NUMBER 2 2006 ISSN 1306-4428.

8- Alaa Abdulqahar Jihad, **"Development DSS for huge data based cleaning techniques"**,2011, Master thesis\ computer science\ college of computer\ Anbar university.

9- Mosud Y. Olumoye, COURSE "**Data Mining/ Data Warehousing**"\ Module 1: Concepts of Data Mining\ Unit 5: Data Preparation and Preprocessing, national open university of Nigeria, Rashmoye Publications (2009).

10-Leul Woldu Asegehgn, "**The Application Of Data Mining In Crime Prevention: The Case Of Oromia Police Commission**", Master Thesis (July 2003), Addis Ababa University\School Of Graduate Studies.

مرحلة قبل المعالجة لاكتشاف المعرفة

مرتضى محمد حمد ، بناز انور قادر

كلية الحاسوب ، جامعة الانبار ، رمادي ، العراق

الملخص:

مرحلة قبل المعالجة للبيانات تعرف أيضاً بمرحلة (تهيئة البيانات) وهي مرحلة أساسية لتحليل البيانات واكتشاف المعرفة. عند وجود معلومات غير متعلقة بالموضوع وفائضة أو بيانات مشوشة وغير موثوقة ، فان عملية اكتشاف المعرفة خلال مراحل التحليل والتنقيب سوف تكون صعبة ومعقدة. لذلك نعتبر مرحلة قبل المعالجة للبيانات خطوة مهمة لعملية اكتشاف المعرفة وذو تأثير مهم على دقة التنبؤ. بصورة أساسية، بينما كل حقل خاص في الجدول يحتاج نوع خاص من المعالجة لكل خوارزمية، لذا فان اختيار خوارزمية قبل المعالجة يعتمد على نوع قاعدة البيانات المستخدمة. في هذا البحث قمنا باختيار وتوضيح تقنيتين مختلفتين من تقنيات قبل المعالجة للبيانات والتي هي (التناسق و التقليل أو التخفيض) معتمداً على مستودع البيانات الخاص بالتسويق والذي يحتوي على حقول غير متناسقة وقيود متكررة. في هذا البحث قمنا أيضاً باقتراح وبناء خو ارزميتين جديدة إحداها للتقليل تسمى (خوارزمية إزالة التكرار) و الأخرى للتناسق تسمى (خوارزمية تحليل عدم التناسق) محققا بذلك أحسن الانجازات لمجاميع البيانات. في هذا البحث قمنا بتطبيق وتنفيذ الخوارزميتين المقترحتين على مستودع البيانات مستخدماً (لغة البرمجة #C) و(ملف Microsoft Access) وقد حصلنا على مستودع بيانات نظيفة ذو حقول منسقة وخالية من القيود المكررة وجاهزة لتهيئة بيانات ذو جودة عالية كإدخال لخوارزميات عملية تنقيب البيانات أو أي طريقة تحليلية والتي تؤثر على نوعية المعرفة المكتشفة خلال عملية تنقيب البيانات.