Internal Data Imputation in Data Warehouse Dimensions

Yuzhao Yang, Fatma Abdelhedi, Jérôme Darmont, Franck Ravat, Olivier Teste

Internal Data Imputation in Data Warehouse Dimensions

Yuzhao Yang[1], Fatma Abdelhédi[3], Jérôme Darmont[2], Franck Ravat[1], and Olivier Teste[1]

[1] IRIT-CNRS (UMR 5505), Université de Toulouse, France
{Yuzhao.Yang, Franck.Ravat, Olivier.Teste}@irit.fr
[2] Université de Lyon, Lyon 2, UR ERIC, France
jerome.darmont@univ-lyon2.fr
[3] CBI[2] – TRIMANE, Paris, France
Fatma.Abdelhedi@trimane.fr

Abstract. Missing data occur commonly in data warehouses and may generate data usefulness problems. Thus, it is essential to address missing data to carry out a better analysis. There exists data imputation methods for missing data in fact tables, but not for dimension tables. Hence, we propose in this paper a data imputation method for data warehouse dimensions that is based on existing data and takes both intra- and inter-dimension relationships into account.

Keywords: Data warehouses · Data imputation · Dimensions

1 Introduction

Data warehouses (DWs) are widely used in companies and organizations to help building decision support systems. Data in DWs are usually modeled in a multidimensional way, which helps users consult and analyze aggregated data with On-Line Analytical Processing (OLAP). In a DW, there are non-NULL constraints on keys, but not always on the other attributes, so there may be missing data. Missing data may come from the DW's sources (operational data sources, data of other DWs) if not treated during the Extract-Transform-Load (ETL) process. We classify missing data in the DWs into factual missing data and dimensional missing data with respect to their occurrence in fact and dimension tables. Factual missing data are usually quantitative, making analysis results incomplete and preventing users from getting reliable aggregates. Dimensional missing data are usually qualitative, making aggregated data incomplete and making it hard to analyse them with respect to hierarchy levels. Therefore, it is significant to complete the missing data for the sake of a better data analysis.

Data imputation is the process of filling in missing data by plausible values based on information available in the dataset [5]. Imputation of missing data focuses on factual data, with statistic-based [11], K-Nearest Neighbour (KNN)-based [3], linear programming-based [2] and hybrid (KNN and constraint programming) [1] methods. There is no research about dimensional missing data.

However, dimensional data are mostly qualitative and there are methods for qualitative data imputation. Some methods replace missing values through business rules [4,9] or association rules [10,8]. Yet business rules are not always available in practice, and association rules require to define support and confidence thresholds, which is not always easy. External sources can also be employed, e.g., through crowdsourcing [6] or taking advantage of Web information [12]. Yet, suitable external sources may be difficult to find. Eventually, data imputation in DWs should consider the different structural elements in an OLAP systems, such as dimensions and hierarchies. As a result, we propose in this article an internal, i.e., based on existing data, data imputation method for dimensional missing data in DWs, by considering inter- and intra-dimension relationships.

The rest of the paper is organized as follow. In Section 2, we formalize the OLAP model. In Section 3, we detail our imputation method and provide the corresponding algorithms. In Section 4, we validate our proposal through a series of experiments. Finally, in Section 5, we conclude the paper.

2 Preliminaries

We introduce here the multidimensional DW concepts and notations used in this paper [7].

Definition 1. *A data warehouse, denoted DW, is defined as (N^{DW}, F^{DW}, D^{DW}, $Star^{DW}$), where N^{DW} is the data warehouse's name, $F^{DW} = \{F_1^{DW}, ..., F_m^{DW}\}$ is a set of facts, $D^{DW} = \{D_1^{DW}, ..., D_n^{DW}\}$ is a set of dimensions and $Star^{DW} : F^{DW} \rightarrow D^{DW}$ is a mapping associating each fact to its linked dimensions.*

Definition 2. *A dimension, denoted $D \in D^C$, is defined as (N^D, A^D, H^D, I^D), where N^D is the dimension's name, $A^D = \{a_1^D, ..., a_u^D\} \cup \{id^D\}$ is a set of attributes, where id^D represents the dimension's identifier. $H^D = \{H_1^D, ..., H_v^D\}$ is a set of hierarchies. $I^D = \{i_1^D, ..., i_e^D\}$ is a set of dimension instances. The value of instance i_e^D for attribute a_u^D is denoted as $i_e^D.a_u^D$.*

Definition 3. *A hierarchy of dimension D, denoted $H \in H^D$, is defined as ($N^H, Param^H$), where N^H is the hierarchy's name. $Param^H =< id^D, p_2^H, ..., p_v^H >$ is an ordered set of dimension attributes, called parameters, which set granularity levels along the dimensions: $\forall k \in [1...v], p_k^H \in A^D$. The case where p_1^H rolls up to p_2^H in H is denoted by $p_1^H \preceq_H p_2^H$. $Weak^H = Param^H \rightarrow (A^D - Param^H)$ is a mapping possibly associating each parameter with one or several weak attributes, which are also dimension attributes providing additional information. $Weak^H[p_x^H] = \{w_1^{p_x^H} ..., w_y^{p_x^H}\}$ is the weak attribute set for parameter p_x^H.*

3 Internal Data Imputation for Dimensions

Internal data imputation consists in replacing missing data in dimensions with the aid of existing data. Existing data imputation is convincible because we use

accurate data and not predictions or otherwise computed values. Imputation can be achieved through intra- and inter-dimensional relationships. Let us introduce these two types of data imputation.

Intra-dimensional Imputation Intra-dimension imputation relies on data from the same dimension. There are indeed functional dependencies between attributes in the same hierarchy. If an attribute is a parameter, its values depend on the values of lower-granularity parameters. Our intra-dimension imputation method is presented in Algorithm 1. We first check each parameter in hierarchies of the DW. If there exists missing data for this parameter (Lines 1-2), we search for an instance with value in a lower-granularity parameter and whose value exists (Lines 3-4). Then, we can then fill in the missing data with this value (Line 5).

Algorithm 1: Intra-dimension Imputation

1 **for** *each* $p_v^H \in Param^H$*, where* $H \in H^D, D \in D^{DW}$ **do**
2 **for** *each* $i_e^D \in I^D$*, where* $i_e^D.p_v^H$ *is null* **do**
3 **while** $p_{v_2}^H \in Param^H \wedge p_{v_2}^H \preceq_H p_v^H$ **do**
4 **if** $\exists i_{e_2}^D \in I^D, i_{e_2}^D.p_{v_2}^H = i_e^D.p_{v_2}^H \wedge i_{e_2}^D.p_v^H$ *is not null* **then**
5 $i_e^D.p_v^H \leftarrow i_{e_2}^D.p_{v_2}^H$
6 **for** *each* $i_{e_3}^D \in I^D$*, where* $i_{e_3}^D.w_y^{p_v^H}$ *is null,* $w_y^{p_v^H} \in Weak^H[p_v^H]$ **do**
7 **while** $p_{v_3}^H \in Param^H \wedge (p_{v_3}^H \preceq_H p_v^H \vee p_{v_3}^H = p_v^H))$ **do**
8 **if** $\exists i_{e_4}^D \in I^D, i_{e_4}^D.p_{v_3}^H = i_{e_3}^D.p_{v_3}^H \wedge i_{e_4}^D.p_v^H$ *is not null* **then**
9 $i_{e_3}^D.w_y^{p_v^H} \leftarrow i_{e_4}^D.w_y^{p_v^H}$

The value of a weak attribute depends on the values of its parameter. Then, for each weak attribute of the parameter we check, if there are missing data (Line 6), we search for the instance that has the same value of its parameter or a lower-granularity parameter whose value exists (Lines 7-8). The missing weak attribute data can then be supplied by this value (Line 9). It is important to note that, since the parameter sets of hierarchy are ordered sets, checking parameters is sequential (from the lowest-granularity to the highest-granularity parameter). This ensures that imputation is maximal, as the value of a higher-granularity parameter depends on its lower-granularity parameters.

Inter-dimensional Imputation In a DW, there may be attributes that are common to different dimensions. Therefore, we can replace missing data with such inter-dimensional common attributes. The main idea of inter-dimension imputation is similar to intra-dimension imputation's, except that instead of

searching for parameters in the same hierarchy, we search for common parameters of hierarchies in other dimensions (Algorithm 2, Lines 3-4 and 9-10). When performing the imputation of weak attributes, we must make sure that, in the searched dimension, the searched parameter is semantically identical with the parameter of the weak attribute to be completed; and that it bears a semantically identical weak attribute (Lines 10-11). We say "semantically identical" because in a DW, common attributes may be presented differently in different dimensions. Since in a DW, the designer would normally not use two vocabularies to describe a same entity, but may use the different prefixes or suffixes to distinguish the same entity in different dimensions, we must therefore use string similarity to match attribute names.

Algorithm 2: Inter-dimension Imputation

```
1  for each p_v^H ∈ Param^H, where H ∈ H^D, D ∈ D^DW do
2      for each i_e^D ∈ I^D, where i_e^D.p_v^H is null do
3          for each p_{v2}^{H2} ∈ Param^{H2}, where H2 ∈ H^{D2}, D2 ∈ D^DW ∧ D2 ≠ D do
4              if p_{v2}^{H2} ≃ p_v^H then
5                  while p_{v3}^{H2} ∈ Param^{H2} ∧ p_{v3}^{H2} ⪯_{H2} p_{v2}^{H2} do
6                      if ∃ i_{e2}^{D2} ∈ I^{D2}, i_{e2}^{D2}.p_{v3}^H = i_e^D.p_{v3}^H ∧ i_{e2}^{D2}.p_{v2}^{H2} is not null then
7                          i_e^D.p_v^H ← i_{e2}^{D2}.p_{v2}^{H2}
8      for each i_{e3}^D ∈ I^D, where w_y^{p_v^H} ∈ Weak^H[p_v^H], i_{e3}^D.w_y^{p_v^H} is null do
9          for each p_{v4}^{H3} ∈ H3, where H3 ∈ H^{D3}, D3 ∈ D^DW ∧ D3 ≠ D do
10             if p_{v4}^{H3} ≃ p_v^H ∧ ∃ w_{y2}^{p_{v4}^{H3}} ∈ Weak^{H3}[p_{v4}^{H3}], w_{y2}^{p_{v4}^{H3}} ≃ w_y^{p_v^H} then
11                 while p_{v4}^{H3} ∈ Param^{H3} ∧ (p_{v5}^{H3} ⪯_{H3} p_v^H ∨ p_{v5}^{H3} ≃ p_v^H) do
12                     if ∃ i_{e4}^{D3} ∈ I^{D3}, i_{e4}^{D3}.p_{v5}^{H3} = i_{e3}^D.p_{v5}^{H3} ∧ i_{e4}^{D3}.w_{y2}^{p_{v4}^{H3}} is not null
                           then
13                         i_{e3}^D.w_y^{p_v^H} ← i_{e4}^{D3}.w_{y2}^{p_{v4}^{H3}}
```

4 Experimental Assessment

We implement our algorithms[4] and conduct experiments with different datasets. Our code is developed in Python 3.7 and is executed on a Intel(R) Core(TM) i5-10210U 1.60 GHz CPU with a 16 GB RAM. Data are integrated in R-OLAP format with Oracle 11g.

[4] https://github.com/BI4PEOPLE/Internal-Data-Imputationin-Data-Warehouse-Dimensions/

4.1 Datasets and Experimental Method

Our experiments are based on one benchmark dataset and three real-world datasets. The TPC-H benchmark (**TPCH**) provides a relational schema[5] with 8 tables and a data generator we use to produce 100 MB of data. The first real-world dataset is a customer-centric dataset (**GlobalStore**) of a global super store[6]. It contains the order data of different customers and products. The second real world dataset is a regional sale dataset (**RegionalSales**) storing sales data for a company across US regions[7]. The third dataset (**GeoFrance**) contains information about French cities, departments and regions from the French government open data site[8]. We create a DW for each real-world dataset.

In our experiments, the parameter of the first granularity level in dimension hierarchies is the primary key of the dimension table. Its values are not repetitive, so weak attributes of the first granularity level and parameters of the second granularity level cannot be completed. Therefore, we generate missing data for parameters from the third granularity level of dimension hierarchies and for weak attributes from the second granularity level. Moreover, we apply different missing rates (1%, 5%, 10%, 20%, 30%, 40% and 50%). To generate a certain percentage of missing data for an attribute, we sort randomly all the tuples and remove attribute data of the first certain percentage of tuples. For each dataset, we carry out 20 tests and get the average imputation rate, accuracy and runtime. Imputation rate is the number of replaced values divided by the number of missing values. Accuracy is the number of correctly replaced values divided by the number of all replaced values.

4.2 Intra-dimensional Imputation Experiments

The datasets **TPCH**, **GlobalStore** and **RegionalSales** are employed in this experiment intra-dimensional imputation experiment. The imputation rate ranges between 61.73% and 100%; the accuracy between 97.08% and 100%; and missing rate between 1% and 50%.

Imputation Rate In Figure 1, imputation rates (X-axis) vary with respect to missing rates (Y-axis). We observe that, for dataset **TPCH**, the imputation rate is always 100%, while the imputation rates of the other datasets decrease when the missing rate increases. The imputation rate of **RegionalSales** is much lower than the two others. Since missing data are replaced by the tuple having the same value on a lower-granularity parameter, the imputation rate of an attribute depends on the ratio of the distinct values and the coefficient of variation of each distinct value of its lower-granilarity parameters. For example, in the dimension *Part* of **TPCH**, imputation rate and ratio are 0.125% and 0.027 for the second

[5] http://tpc.org/tpc_documents_current_versions/pdf/tpc-h_v2.18.0.pdf
[6] https://data.world/vikas-0731/global-super-store
[7] https://data.world/dataman-udit/us-regional-sales-data
[8] https://www.data.gouv.fr/fr/datasets/communes-de-france-base-des-codes-postaux/

granularity level parameter, respectively; while in dimension *StoreLocation* of **RegionalSales**, they are 62.13% and 1.1, respectively.

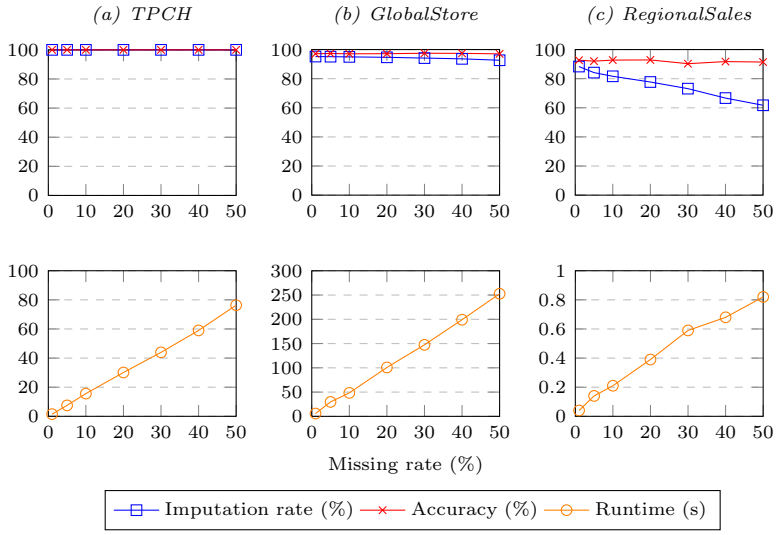

Fig. 1. Intra-dimensional imputation experiment results

Accuracy We can see in Figure 1 that the accuracy of **TPCH** is always 100%, while the accuracy of the two other datasets is always less than 100%. Since our imputation method is based on the hierarchy relationships, non-strict and incomplete hierarchies may impact accuracy. By analysing the data, we find that there are non-strict hierarchies in these datasets **GlobalStore** and **RegionalSales**, i.e., in the dimension *Customer* of **GlobalStore**, there are some tuples whose *City* values are the same, but they belong to different *State*s. There is a similar case in dimension *StoreLocation* of **RegionalSales**.

Runtime The evolution of runtime with respect to missing rate is linear (Figure 1), which is in line with the complexity of Algorithm 1, which is $O(n)$, where n is the missing rate.

4.3 Inter-dimensional Imputation Experiments

We use **TPCH** and **GeoFrance** in this inter-dimensional imputation experiment. There are two **TPCH** dimensions, *Customer* and *Supplier*, which have same geographical attributes. In **GeoFrance**, we create two dimensions and randomly divide the original data into two partitions with the same number of

tuples. Then, we load the each partition into one of the dimensions. The imputation rate ranges between 48.67% and 100%. The accuracy always remain at 100% with respect to missing rate.

Imputation Rate Yet again, the imputation rate of **TPCH** is always 100% (Figure 2), for the same same reason as intra-dimensional imputation. **GeoFrance**'s imputation rate is low when the missing rate is very low, then it increases with the missing rate. After analysing the data, we find that there is a tuple where the values of *RegionCode* and *RegioName* are originally missing. The lower-granularity parameter *DepartmentCode* being unique, *RegionCode* and *RegionName* missing data cannot be imputed. When the missing rate is low, the number of total missing data is low, too. Thus, the missing data in this tuple account for a large proportion of total missing data, which can explain why **GeoFrance**'s imputation rate is low when the missing rate is very low.

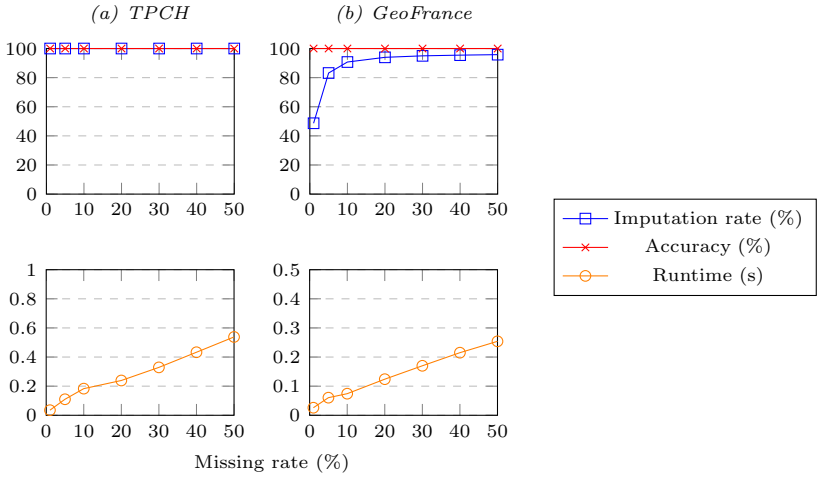

Fig. 2. Inter-dimensional imputation experiment results

Accuracy There is no incomplete nor non-strict hierarchy in **TPCH**'s and **GeoFrance**'s DWs. Hence, the accuracy for these two datasets is always 100% (Figure 2).

Runtime Again, the evolution of runtime with respect to missing rate is linear (Figure 2), which is in line with the complexity of Algorithm 2, which is $O(n)$, where n is the missing rate.

5 Conclusion and future work

In this article, we propose an internal data imputation method for dimensional missing data in DWs. Our method is based on the existing data found in both intra- and inter-dimensional relationships. We take in charge the imputation of both parameters and weak attributes. The solutions are formalized as algorithms and are actually implemented. Our method is validated by a series of experiments with the different percentages of missing data of the different attributes. However, not all missing data can be completed by the existing data, thus in the future we will also combine our method with web-based methods to achieve a better imputation

Acknowledgement

This research is funded by the French National Research Agency (ANR), project ANR-19-CE23-0005 BI4people (Business Intelligence for the people).

References

1. F. Amanzougarene, K. Zeitouni, and M. Chachoua. Predicting missing values in a data warehouse by combining constraint programming and knn. In *EDA*, 2014.
2. S. Bimonte, L. Ren, and N. Koueya. A linear programming-based framework for handling missing data in multi-granular data warehouses. *Data & Knowledge Engineering*, 128, 2020.
3. L. de S. Ribeiro, R. R. Goldschmidt, and M. C. Cavalcanti. Complementing data in the etl process. In *DaWaK*, pages 112–123, 2011.
4. W. Fan, L. Jianzhong, M. Shuai, T. Nan, and Y. Wenyuan. Towards certain fixes with editing rules and master data. *The VLDB Journal*, pages 173–184, 2010.
5. D. Li, J. Deogun, W. Spaulding, and B. Shuart. Towards missing data imputation: A study of fuzzy k-means clustering method. In *RSCTC*, pages 573–579, 2004.
6. C. Lofi, K. El Maarry, and W.-T. Balke. Skyline queries over incomplete data-error models for focused crowd-sourcing. In *Conceptual Modeling*, pages 298–312, 2013.
7. F. Ravat, O. Teste, R. Tournier, and G. Zurfluh. Algebraic and graphic languages for olap manipulations. *Inter. J. of Data Warehousing and Mining*, 4:17–46, 2008.
8. J.-J. Shen, C.-C. Chang, and Y.-C. Li. Combined association rules for dealing with missing values. *Journal of Information Science*, 33(4):468–480, 2007.
9. S. Song, A. Zhang, L. Chen, and J. Wang. Enriching data imputation with extensive similarity neighbors. *VLDB Endowment*, 8(11):1286–1297, 2015.
10. C.-H. Wu, C.-H. Wun, and H.-J. Chou. Using association rules for completing missing data. In *HIS*, pages 236–241, 2004.
11. X. Wu and D. Barbará. Modeling and imputation of large incomplete multidimensional datasets. In *DaWak*, pages 286–295, 2002.
12. M. Yakout, K. Ganjam, K. Chakrabarti, and S. Chaudhuri. Infogather: Entity augmentation and attribute discovery by holistic matching with web tables. In *SIGMOD*, page 97–108, 2012.