A SEMANTIC RESOURCE BASED APPROACH FOR STAR SCHEMAS MATCHING

Elhaj Elamin[1]Amer Alzaidi[2] and Jamel Feki[2]

[1]Department of Computer Science, Sudan University of Science and Technology, Khartoum, Sudan

[2]University of Jeddah, FCIT, Jeddah, Saudi Arabia

ABSTRACT

The hybrid approach is widely used in constructing data warehouse (DW) schemas. It relies on a complex process for matching two sets of multidimensional star schemas: schemas built from business requirements (BR-Star schemas) and schemas constructed on the organization data source (DS-Star schemas). Using a semantic resource during this matching helps solving heterogeneity problems. This paper suggests a semi-automatic approach for the construction of approved star schemas by matching DS-Star schemas with BR-Stars, and by using WordNet as a semantic resource for solving heterogeneity issues. This approach consists of three steps: i) Match DS-Stars with BR-Stars; ii) Involve the DW designer for approval of the matching process; and then iii) Generate approved star schemas. We have defined two Boolean functions and four semantic metrics for the matching process. We have developed a software prototype for testing our approach.

KEYWORDS

Star schemas matching, Hybrid approach, Data warehousing, Semantic resource.

1. INTRODUCTION

The data warehouse (DW) has been considered as a vital technology for modern decision support systems (DSS) for organizations. Indeed, the DW offers efficient capabilities for supporting decision-makers. Despite the valuable efforts and attempts from researchers devoted to developing DW approaches, several DW projects failed [1][2]. In fact, researchers agree that any successful attempt to develop a DW should consider two features namely i) the construction of the DW multidimensional schema by using a hybrid approach relying on user requirements and data sources; and ii) the use of semantic resource to overcome the heterogeneity issues complicating the DW construction process[3][6][9][12].

Furthermore, it is commonly admitted that the intervention of the decision-maker during the DW construction process greatly improves the quality of the DW schema. Obviously, a full automation of the design process may lead to inaccurate results; for instance, defining rules for extracting facts and dimensions automatically may be misleading, as real-world entities (tables, objects…) and relationships both may have common characteristics and therefore may play the role of fact or dimension. Therefore, it is helpful to let the decision-maker or the DW designer intervene for interpreting and evaluating the correctness of designed schemas[17].

Furthermore, it is widely admitted that the star schema represents the keystone structure for DW modelling. This is due to its simplicity, efficiency and its intuitive shape when compared to other

DOI : 10.5121/ijdms.2018.10602

multidimensional schemas (e.g., constellation or snowflake schemas).A multidimensional star schema has one fact modelling a business activity of interest for decision-makers and n (n>1) dimensions surrounding the fact [4].Relying the construction of the DW schema on a hybrid approach requires a matching process between BR-Stars(star schemas designed from business requirements) and DS-Stars(star schemas designed on the data source). Indeed, despite the agreement among DW community about the basic methods for constructing the DW, we reveal three issues in the literature review. First, there is no agreement about which technique is recommended for performing the matching process, if a technique could be adopted. Secondly, which kind of semantic resource should we if any? Thirdly, whether the approach considers BR first as in [7][9][10] or the DS first as in [5] or combines both BR and DS simultaneously [6][8]. These issues motivated us to investigate a new hybrid and semi-automatic approach focusing on the matching process of multidimensional star schemas; this approach aims to help DW designers to design star schemas closely related to BR and DS, and that are subject to low effort of adaptation/validation, therefore enhancing the DW quality and reducing costs. Our proposed approach differs from the literature approaches since:

It relies on a semantic resource; in fact, we have elected WordNet for the matching, this choice enables us to avoid building specific domain ontology; consequently, this will shorten the design/development time of the DW.

The matching process accepts as input two complementary sets of star schemas; this is to consider both BR and DS simultaneously and, therefore build star schemas closely related to users' requirements and the organization's data source.

This paper is organized as follows: Section 2 studies works related to the hybrid approaches for the DW design, as well as matching methods. In Section 3, we introduce our proposed approach for generating approved star schemas as result of matching DS-Stars and BR-Stars. Section 4 is dedicated to results presentation. Finally, Section 5 concludes this paper and enumerates some ongoing issues and related perspectives.

2. RELATED WORK

The DW construction process is complex, tedious and time-consuming [1]; these difficulties do not came from nonsense. Indeed, each step in this construction process consists of sub-steps, which can be carried out using various methods and/or techniques. Additionally, these steps should coined together in a systematic manner to produce a satisfying multidimensional model (i.e., schema). To shed light in this complexity, as an example, we cite the matching process between the DS and users' requirements; for this purpose, in [8][10] the authors base their work on a graph technique. In [5] authors use search patterns, whereas [7] uses three multidimensional normal forms (noted 1MNF , 2MNF and 3MNF) to define a set of Query/View/Transformation (QVT) relations for accomplishing the agreement between the multidimensional model obtained from user requirements ,and the DS. In not far away for the matching process, the proposed works vary in whether they first consider user requirements or DS structure (i.e., schema). As an example, in [10] [13] the authors first consider requirements then reconciling them against the DS; other works begin with considering first the DS and then the requirements [5]. Additionally, the solutions vary in the way the semantic resource had been applied; in this trend, the authors in [5] develop domain ontology in order to automate the generation of star schemas; authors in [9] use a global ontology. On behave of automating the DW construction process; existing contributions vary between generating the star schema manually or through a semi-automatic approach, while others try to automate the full process beginning from generating user requirements and DS models as well as the matching process that produces final star schemas. In

order to highlight the needs in generating DW star schemas, we give more attention to the approaches of the related works.
In [6], a hybrid-automated approach has been proposed where the authors use a graph to model the DS, as well as SQL queries for requirements representation. An important limitation in their work is that it relies on an expert for writing the queries. Additionally, their approach ignores the heterogeneity problems raised in the DW design activities. Hence, a basic component for constructing the DW is missing, namely a semantic resource.

Authors in [7] proposed a hybrid DW approach, they use two conceptual multidimensional models; on one hand, they first defined a conceptual multidimensional model for capturing user's requirements and, on the other hand, they used multidimensional normal forms to define a set of Query/View/Transformation relations. They reconcile the user requirements model with that of DS to ensure its correctness. The authors succeed in applying a systematic approach for developing the multidimensional model. However, in behave of requirements they concentrate on business goals related to DW users; nevertheless, there is no means for using semantic resource.
In [8], authors automatically generate a set of multidimensional schemas from the DB schema of the operational information system that satisfy user requirements expressed in terms of SQL queries. They used a multidimensional graph to store multidimensional information about the query. In their approach, SQL queries are accepted if they generate anon-empty set of multidimensional schemas. The drawback of this approach is that there is no template (i.e., style) for writing the queries. Additionally, the task of expressing user requirements as SQL queries needs a skilled person in SQL who knows precisely the DS schema.

A hybrid approach is proposed in [5]; its distinctive feature is an automatic analysis of the DS that leads to the design of the DW schema. Additionally, it belongs to the group of works supporting the multidimensional design from ontologies. However, the design process of the DW consumes a lot of effort and time because this approach relies on ontology as driving force for generating the multidimensional model. Moreover, the authors described their approach as a reengineering process; hence, the most choices in this situation as ontology language are UML (Unified Modelling Language) or ERD (Entity-Relationship diagram). The problem here is that generating ontology from these sources needs a heavy pre-process that delays the DW constructing. Additionally, the full automation of the DW schema generation in this work decreases the chance for intervening the DW designer to confirm the generated multidimensional elements.

An automatic hybrid approach is proposed in [9] where authors used algorithms for matching user requirements with a global ontology. However, in the matching process, their approach ignore the elements that do not fully match or are under a given threshold. As well, they assume the existence of a global ontology, whose construction complicates the design of the DW.

A nearest approach to ours is suggested in [13] where the authors use a Goal-Question-Metric technique for capturing users' requirements by means of interviews. The capturing goals are then aggregated and redefined in means of abstraction sheets; from these sheets, the star schema is generated. As well, they use the graph for constructing star schemas from the DS. Despite their efforts on using hybrid approach, their approach suffers from the following limitations: first, in behave of requirements, they use interview technique in capturing user requirements, but they do not define style for formalizing the goals; this may lead to goals expressed in divergent formats, which complicates the process of star schema construction. Secondly, in behave of generating star schemas from DS, authors dedicated an algorithm for exploring the E/R model. Therefore, mapping this model to a connectivity graph without using the reverse engineering process may result in poorly defined star schema elements. Additionally, the approach is manual and does not rely on a semantic resource.

Summing up, the DW development process needs additional investigation overcome these miscellaneous gaps. Table 1 summarizes the techniques used in hybrid approaches for the purposes of reconciling the DS with users' requirements, the degree of automation, and the use of a semantic resource.

Table 1. Comparison between proposed hybrid approaches.

Work reference	Matching Technique	Automation	Use of semantic resource
Romero, et al., 2006 [8]	Graph	Automatic	No
Romero, et al., 2010[5]	Search patterns	Automatic	Yes
Mazón, et al., 2007[7]	Multidimensional normal forms	No	No
Romero, et al., 2010[6]	Graph/SQL queries	Automatic	No
Thenmozhi, et al., 2012[9]	Algorithms	Automatic	Yes
Di Tria, et al., 2015 [10]	Graph	Automatic	Yes
Bonifati,et al., 2001[13]	Graph and Goal/Question metrics	No	No

3. PROPOSED APPROACH

Recent research works for constructing the DW greatly concentrate on two features: the first feature concerns the use of a hybrid approach; that means the design process considers both users' requirements and DS data model in order to produce a DW multidimensional schema. The second feature is to solve the heterogeneity problems by using a semantic resource. From our viewpoint, an important issue is how to apply these two features so that the resulting DW helps the organizations offering a reliable multidimensional model. We propose a matching approach for the alignment of star schemas issued from two complementary contributions in the literature: One dealing with the generation of star schemas from a relational DS model [16], and a second contribution tackling the construction of star schemas from BRs [14].Our matching approach relies on a semantic resource to overcome the heterogeneity between DS-Star schemas and BR-Star schemas. In the literature, ontology solves the problems of heterogeneity; however, domain ontology is not frequent especially for DWsing; in addition, its construction is neither easy nor rapid. Indeed, building domain ontology increases the time and the cost for developing the DW. As with our approach we aim to decrease the DW developing time, we have selected WordNet as a ready open source semantic resource for solving the heterogeneity between a DS-Star's elements and a BR-Star's elements. More precisely, our proposed approach consists of three steps: i) Matching DS-Star schemas with BR-Star schemas; ii) Involvement of the DW designer; and iii) Generation of approved star schemas. Figure 1 depicts the framework for our suggested approach.

Hereafter, we detail steps for generating approved star schemas.

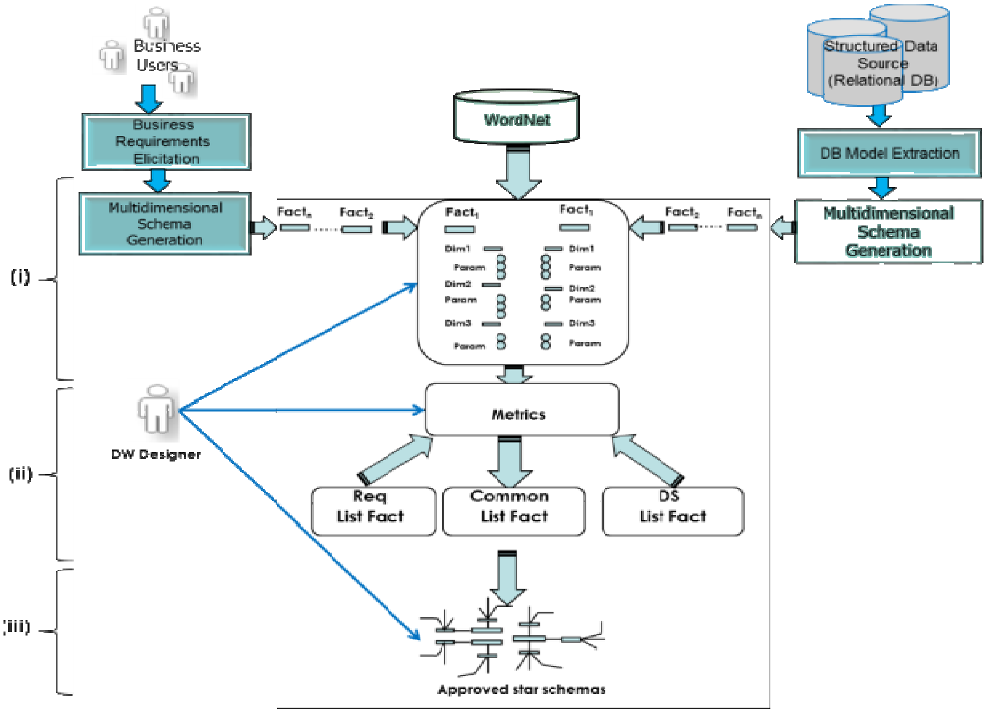


Figure 1. Proposed framework for generating approved star schemas from BRs and DS models

3.1. Matching DS-Star Schemas with BR-Star Schemas

Schema matching is a complex and time-consuming process [15]; it aims to match correctly DS-Star schema elements (fact, dimension, etc...) with BR-Star schema elements. Furthermore, we detect and suggest solutions for unmatched between schemas elements such as diversity in names of elements. Naturally, we optimize the matching step not by computing a Cartesian product of the two schemas' elements. So then, we identify 'similar schemas' (i.e., schemas analyzing the same business activity, having identical or synonym fact names, and common dimensions) and then we match each couple of similar schemas. To do so, we develop an algorithm called MatchStars, as well we define two Boolean functions for checking whether two names of fact or dimensions are identical/synonyms or not. Additionally, we define four semantic metrics to measure the similarity between two star schemas. To assist the DW designer solving the problem of unmatched elements, our approach allows him/her to intervene and matches them manually. It is worth mentioning that we have treated the extraction process of DS-Star schemas and BR-Star schemas in previous a work [17].

Hereafter we introduce the notation we use:

Notation

- *FName(S): A function that returns the Fact Name from a star schema S.*
- *SBF: A set composed of three lists:*
    - *List of fact names from BR-Stars*
    - *List of Dimension names in each fact from BR-Stars*
    - *List of parameter names in each dimension in each BR-Stars*
- *SDF: A set composed of three lists:*
    - *List of fact names from DS-Stars*
    - *List of Dimension names in each fact from DS -Stars*

  - *List of parameter names in each dimension in each DS -Stars*
- *SCF: A set composed of three lists:*
  - *List of all fact names Common to BR-Stars and DS-Stars.*
  - *List of Dimension names Common to BR-Stars and DS-Stars*
  - *List of parameter names Common to BR-Stars and DS-Stars.*

For the next illustrations, Figure 2 depicts an example of three simple DS-Stars schemas and three BR-Star schemas.

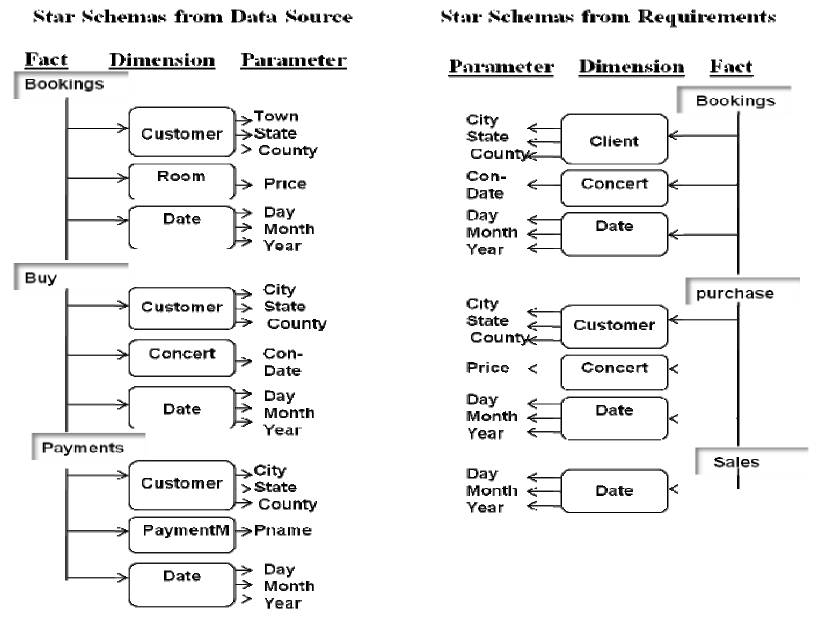


Figure 2. Example of threeBR-Star Schemas and three DS-Star Schemas.

Boolean Functions for Matching DS-Stars and BR-Stars

Hereafter we highlight for each of the two Boolean function Identical and Synonym, its objective, syntax, and then we give the result of its application on our running example of figure2. Note that the results of applying these functions will be append to the SCF set of lists.

1) *Function Identical (e_1, e_2):* returns *True* if the two-element names e_1 and e_2 are identical and *False* otherwise; e_1, e_2 may be fact names or dimension names. When true, this function means that the match of these elements makes sense.

Note that, we apply this function in two steps of the matching process; firstly on fact names, and secondly, on dimension names of identical facts having identical names. We append the common fact name and the common dimension names into SCF Fact list and Dimension list.

For our running example (Figure 2), the function Identical (Bookings, Bookings) is returns True; so then we reuse this function on the dimensions of the two Bookings facts in BR-Star and DS-Star. Identical (Date, Date) is also True, so we append the Bookings fact to the SCF Fact list as well as the common dimension Date in Dimension list to SCF.

2) *Function Synonym (e_1, e_2):* returns *True* if the two element names e_1 and e_2 are synonym, and *False* otherwise.

If *Synonym (e1, e2)* is True considering facts, we reuse it again on their dimensions. If we find synonym dimensions, we append the synonym fact name and the synonym dimension names of this fact to the *Fact list* and *Dimension list* in *SCF respectively*.

For our running example (in Figure 2), *Synonym (Buy, Purchase)* is True for the two facts; therefore we continue to look for the synonym dimensions between the facts *Buy* (in DS-Star) and *Purchase*(in BR-Star). Thus, since *Synonym (Customer, Client)* returns True for the two dimensions, we append the *Purchase* fact to the *Fact list* in *SCF* as well as the synonym *Client* dimension to the *Dimension list* in *SCF*.

Table 3 depicts the result of the *Identical* and *Synonym* functions on the example in Figure 2. Note that, so far, only the lists in ***SCF*** have elements (i.e. fact, dimension) while ***SDF*** and ***SBF*** remain empty.

Note that, the DW designer will be asked to treat the uncommon facts and uncommon dimensions later manually in the matching phase. It is worth mentioning that the uncommon dimensions have two cases: in one hand, the dimension name is loadable from the DS, but it is not considered by the business user in BR. On the other hand, the dimension name is not loadable with data from the DS, but it is required by the BR business.

Inorder to support the tractability of our approach, we highlight the main steps of the matching process in the MatchStars algorithm. We used the open library Rita that takes two words, it returns whether these two words are synonyms or not based on a threshold we have defined (less than 1). The threshold value 1 causes synonym words to be different (see table 4).

Algorithm ***MatchStars***

Aims: Matches the elements of BR-Star Schemas with the elements of DS-Star Schemas.

Input:

- S-BR: Set of star schemas from business requirements.
- S-DS: Set of star schemas from data source.
- fr: a fact in a star schema belonging to S-BR.
- fd: a fact in a star schema belonging to S-DS.
- DIM1: Set of dimensions of fr.
- DIM2: Set of dimensions of fd.

Output: SCF, SBF and SDF

```
Begin
Boolean flag= false;
String str="";
Integer i=1, j=1;
Foreach fact fr in S-BR do
  Foreach fact fd in S-DS do
      If (synonym (fr.name, fd.name) or Rita (fr.name,
        fd.name) <1)
      Foreach Dimension d1 in fr do
        Foreach Dimension d2 in fd do
        If (synonym (d1.name, dim2.name) or Rita
                (d1.name, dim2.name) <1)
                 SCF[i].Dimension=d1.name
                  Increment i;
        Endif
        Endfor
      Endfor
          Flag = true
      Str  = fr.name
    break
    Endif
         Else
         Flag = false
  If(rita(fr.name, fd.name)<1)
            SCF[j].Fact=fr.name
    Endfor
    If flag
         SCF[j].Fact=str
        Else
          SBF[j].Fact=fr.name
          SDF[j].Fact=fd.name
Increment j;
Endif
Endfor
End
```

Table 3 depicts the result of applying the MatchStars algorithm on examples in Figure 2 when different threshold values are taken.

Note that, in our example in Figure 2 we have equal number of facts for both BR-Stars and DS-Stars. Inorder to increase the capabilities of our approach, assume now that BR-Stars are limited to the first two facts (Bookings and Purchase) and the DS-Stars as it is (containing three facts). In this case we append the additional fact (Payments) in the DS-Stars to the SDF. Table 4 depicts the result for this case.

In the upcoming paragraphs we want to measure the matching between the DS-Stars and BR-Stars. Semantic metrics is appropriate tool for this measurement.

Metrics for Measuring the Matching Between DS-Stars and BR-Stars:

In order to measure the matching between DS-Stars and BR-Stars, we define four metrics namely: Common Fact/s (CF),Ratio of common fact/s (RCF), Common Dimension/s (CD)and Ratio of common dimension/s (RCD).

The objective of the first metric (CF) is to returns the number of common fact/s between DS-Stars and the BR-Stars. The syntax of this metric is as follows:

$$CF = |\,(S-BR)\cap(S-DS)|$$

Note that, S-BR represents the set of stars from business requirements; S-DS is the set of stars from data source.

We can calculate the ratio of common fact/s between the DS-Stars and BR-Stars by the second metric ***(RCF)***:

$$RCF = \frac{CF}{|(S-BR)\cup(S-DS)\,|}$$

Note that, the denominator in this metric means:

$$|(S-BR)|+|(S-DS)|-|(S-BR)\cap(S-DS)|$$

The common dimension/s (*CD*) between (DS-Stars) and (BR-Stars) (*CD*) can be defined by first determining the common dimension/s between each couple of the similar facts between (DS-Stars) and (BR-Stars), then, the union of these common dimensions will give the common dimensions. Note that, we need the number of common dimensions, so we use the cardinality concept. So the syntax of our third metric will be as follows:

$$CD = |\bigcup_{i=1}^{CF}(Dim_1(fr)_i \cap Dim_2(fd)_i)|$$

Here *CF* represents the number of common facts (metric 1); *$Dim_1(fr)$ and $Dim_2(fd)$* represents the set of dimensions of a fact inBR-Stars and the set of dimensions of a fact in DS -Stars respectively. To measure the ratio of common dimensions we suppose that there are *n* facts in BR-Stars, and *m* facts in DS-Stars; we define (*RCD*) metric as follows:

$$RCD = \frac{CD}{\sum_{i=1}^{n}|Dim_1(fr)_i| + \sum_{j=1}^{m}|Dim_2(fd)_j| - \sum_{i=1,j=1}^{CF}|Dim_1(fr)_i \cap Dim_2(fd)_j|}$$

In addition to the previous functions and metrics, and in order to enhance the capabilities of our system we define rules for empty fact and/or empty dimension. An empty fact (factless fact) means the fact in star schema may contain no measure/s; empty dimension means the dimension may contain no parameters. The following rules deals with such cases:

Rules for empty facts and/or empty dimensions

R1) For each fact f .in *BR-Stars* or in DS-*Stars* ifthe fact contains no measure (m), delete the fact name (*f*).

$$NofM\,(f) = 0$$

R2) For each parameter (*pm*) in each dimension (*dx*) in BR-Stars or in DS-Stars: if the number of parameters in the dimension=0, delete the dimension (*dx/dy*).

Finally, we have three sets of lists as in table 3; namely, the set of common facts (SCF), the set of business requirements fact (SBF), and the set of data source fact (SDF). Our consideration will devote to the set of common fact (SCF) lists, which will be used to be the nuclear of the approved star schemas. The other two set of lists will be matched manually by the DW designer. The result of the manual matching will be added to the set of common fact lists (SCF).Note that the final result of the matching process is a set of approved star schemas.

In the previous functions and metrics we use the semantic resource WordNet to overcome the heterogeneity in the matching process. In this subsection we detail the benefits of using the semantic resource.

Usage of a Semantic Resource

The matching of star schemas in the DW requires the usage of a semantic resource, the aim is to identify whether a name of a given concept such as fact or dimension is semantically equivalent to another or not. In our framework, we use the free and open source WordNet as a general semantic resource. The reason behind using WordNet is its simplicity; hence, we can decrease the total time for constructing the multidimensional model. The role of WordNet is obvious in the previous functions: *ENIdentical (e_1, e_2),ENSynonym (e_1, e_2).*

As well, we use Rita, which is free and open source library designed to be simple but have a powerful features [11]. For us, RiTa is suitable since it can be integrated with WordNet database, another reason is that RiTa can implement in java. RiTa works by taking two words under testing and check their resemblance by referencing the WordNet database and return the distance between the two words semantically; if the distance equals 1, this means that there is no relation between those words; if the distance is 0, this is indicate that the two words are synonym (have the same meaning). There is another variation as fraction for the variable distance, this fraction is in [0..1]. In our framework we use the value (less than 1) as threshold; so, whenever the distance between the two words is less than 1, this means that those two words are synonyms. The word can be in form of noun or verb. Table 9 depicts usefulness of WordNet to overcome the heterogeneity that may arise in the matching process.

Table 2. Fact/Dimension Synonym

The word	**Fact/dimension**	**The synonym**
Bookings	Fact	Reservation/Engagement
Payment	Fact	defrayment
Buy	Fact	purchase
Customer	Dimension	Client
City	Dimension	Metropolis

Note that, there will be six facts or more, thanks to WordNet, now the number of facts is only three, as the three facts are synonyms. This considerable redundancy may hinder the design of the multidimensional model. Indeed, the full automation of generating the approved star schemas may be inaccurate. The involvement of DW designer is helpful in this process; the second step in our approach is to intervene the DW designer.

3.2. Involvement of the DW Designer

Note that, our approach is semi-automatic; Hence, the DW designer can intervene to reflect the correctness of the generated multidimensional elements. His / Her role in this step is twofold: on one hand, the DW designer checks the *set of common facts* to confirm the generated elements in means of semantic confirmation (see Figure 1). When the number of facts is reasonable this task will not require much time and effort from the DW designer. On the other hand, the DW designer manually matches the elements in the *set of business Requirements fact list* with those elements in the *set of data source fact list* .The DW designer can play this role only when there exist one or more *business Requirements facts* not matching any of the *data source facts,* or vice versa. Generally, this operation could be very simplified when an appropriate semantic resource is used; i.e., a semantic resource that correctly describes the business domain will produce better results than a general resource such as WordNet. The result of the manual matching may concern facts, dimensions and parameters; the DW designer adds these elements to the *set of common fact list;* finally, the combination is forming the approved star schemas.

3.3. Generating approved star schemas

The final output of our framework is a set of approved star schemas; these schemas should satisfy what the decision maker's needs, as well as, loaded correctly with data from the operational data source. Moreover, to enhance schema validity, we defined multidimensional constraints such as avoid empty facts, as well as empty dimensions.

The sources of our approved star schemas are three sets namely the *set of common facts (SCF)*that contains the common facts and its consequences as dimensions and parameters in both BR-Star schemas and the DS-Star schemas. The second set is the *set of business requirements facts (SBF)* which includes the facts and dimensions found only in BR-Star schemas. The third set is *the set of data source facts (SDF)* that contains all facts with their dimensions found only in the DS-Star schemas. Note that, in the matching process there will be two cases: the first one happens when all elements of BR-Star schemas and the elements of DS-Star schemas are common. Consequently, the approved star schemas encompass the elements in the *set of common fact list.* In the second case, there will be unmatched element(s) in *business requirements facts* list or in *data source facts list.* Here, *the* DW designer performs a manual matching process. We have tested our prototype with various examples of star schemas, to generate resulted approved star schemas.

4. RESULTS

The result of this paper is approved star schemas which represent the conceptual model for the DW. To construct this model, we made a matching process between the star schemas generated from business requirements in our previous work [14] and the star schemas issued from the data source [16]. We use star schema in the matching process since it has a simple structure and powerful features, for our knowledge, our work is first one to elaborate semantic metrics and use star schema for matching purpose. To enhance our results we used WordNet as semantic resource, the aim behind using WordNet is its simplicity as free an open source dictionary and hence we can shorten the period of constructing the multidimensional model semantically. In order to increase the approvability of our model, our approach gives the DW designer the chance to intervene to confirm the acceptance of the concepts (facts, dimensions, etc) in the matching process. The generated results (see Figures 3, 4 in the appendix) show the capabilities of our approach for generating the approved star schemas. Our defined functions and metrics covered the various cases that may a raised in the matching process.

5. CONCLUSION AND FUTURE WORK

Recently, researchers in the DW design area agree that the hybrid approach and semantic resource considered as mandatory factors for designing the DW. But still there is no agreement about how to generate the multidimensional model for the DW. In this paper, we generate approved star schemas representing a multidimensional model by applying a matching process between the star schemas issued from the business requirements and the star schemas generated from the data source. For our knowledge, few works used the star schemas for the matching process. Our approach encompasses three steps: the first step is Matching DS-Star schemas with BR-Star schemas; the second step is involvement of the DW designer; the last step is the generation of approved star schemas. Our contribution consists in defining two Boolean functions for the matching process and four metrics for measuring its correctness. Moreover, we design *MatchStars* algorithm showing the main steps of our approach. In order to produce accurate approved star schemas, our approach allows the DW designer to intervene in the matching process, as well as in the confirmation of the approved star schemas. To complete the picture, we used WordNet as a semantic resource, the generated results shows the usefulness of WordNet to reduce the redundancy of the multidimensional elements. We have used java NetBeans to build our prototype; additionally, we use RiTa Library to be integrated with WordNet.

As a further work, we will complete the designing of the DW by expanding our conceptual model to involve the logical and the physical models. As well, we will use domain specific semantic resource to minimize the amount of human intervention.

REFERENCES

[1] F. Bargui, H. Ben-Abdallah, and J. Feki. "A natural language-based approach for a semi-automatic data mart design and ETL generation." Journal of Decision Systems (2016): 1-36.

[2] G.Matteo and S. Rizzi. "A methodological framework for data warehouse design." Proceedings of the 1stACM international workshop on Data warehousing and OLAP. ACM, 1998.

[3] M. Thenmozhiand K. Vivekanandan. "An Ontological Approach to Handle Multidimensional Schema Evolution for Data Warehouse." International Journal of Database Management Systems 6.3 (2014): 33.

[4] Ch. Adamson. Star Schema: The Complete Reference. US: McGraw-Hill Osborne Media, 2010

[5] O. Romero and A. Abelló. "A framework for multidimensional design of data warehouses from ontologies." *Data & Knowledge Engineering* 69.11 (2010): 1138-1157.

[6] O. Romero and A. Abelló. "Automatic validation of requirements to support multidimensional design." Data & Knowledge Engineering 69.9 (2010): 917-942.

[7] J. Mazón, J. Trujillo, and J. Lichtenberger. "Reconciling requirement-driven data warehouses with data sources via multidimensional normal forms." Data &Knowledge Engineering 63.3 (2007): 725-751

[8] O. Romero and A. Abelló "Multidimensional design by examples." InternationalConference on Data Warehousing and Knowledge Discovery. Springer Berlin Heidelberg, 2006.

[9] M. Thenmozhi and K. Vivekanandan. "An ontology based hybrid approach to derive multidimensional schema for data warehouse." International Journal of Computer Applications 54.8 (2012).

[10] F. Di Tria, E. Lefons, and F. Tangorra. "Academic Data Warehouse Design Using a Hybrid Methodology." Computer Science & Information Systems 12.1

[11] https://rednoise.org/rita/index.php , Visited 20/3/2017.

[12] M. Thenmozhi and K. Vivekanandan. "A tool for data warehouse multidimensional schema design using ontology." Int. J. Comput. Sci. Issues (IJCSI) 10.2 (2013): 161-168.

[13] A.Bonifati, F.Angela, S. Ceri, A.Fuggetta and S. Paraboschi."Designing data marts for data warehouses." ACM transactions onsoftware engineering and methodology 10.4 (2001): 452-483.

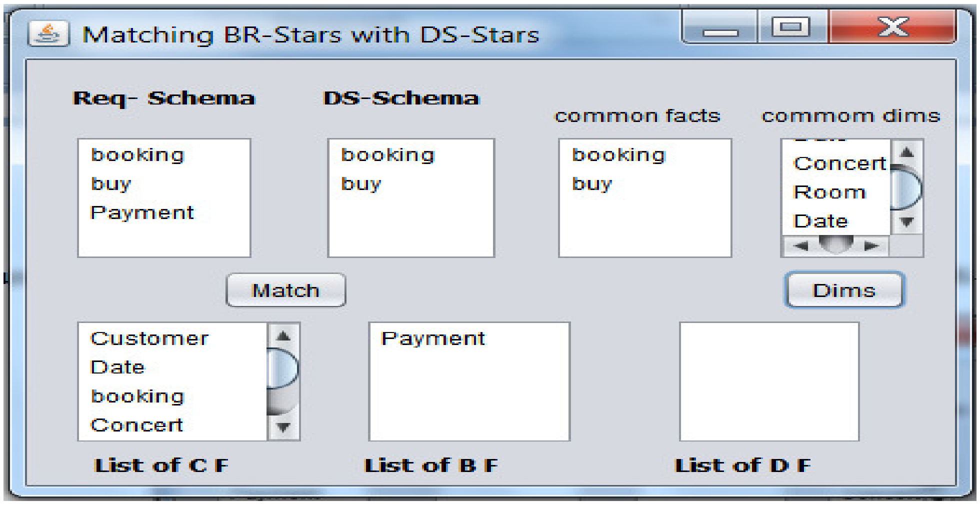
[14] E.Elamin, S. Alshomrani, and J. Feki. "SSReq: A method for designing Star Schemas from decisional requirements." Communication, Control, Computing andElectronics Engineering (ICCCCEE), 2017 International Conference on. IEEE, 2017.

[15] HH. Do, S. Melnik, and E. Rahm. "Comparison of schema matching evaluations." Net. ObjectDays: International Conference on Object-Oriented and Internet-Based Technologies, Concepts, and Applications for a Networked World. Springer, Berlin, Heidelberg, 2002

[16] E.Elamin, A. Altalhi, and J. Feki. “Heuristic Based Approach for Automating Multidimensional Schemas Construction”. *International Journal of Computer and Information Technology* (ISSN: 2279–0764) Volume. 2017 November.

[17] E.Elamin. “A hybrid Semi Automatic Approach for the Design of Data Warehouse Conceptual Model”. Ph.D, Sudan University of Science and Technology, Collage of Graduate Studies. Sudan.October 2018.

APPENDIX



Req- Schema: booking, buy, Payment

DS-Schema: booking, buy

common facts: booking, buy

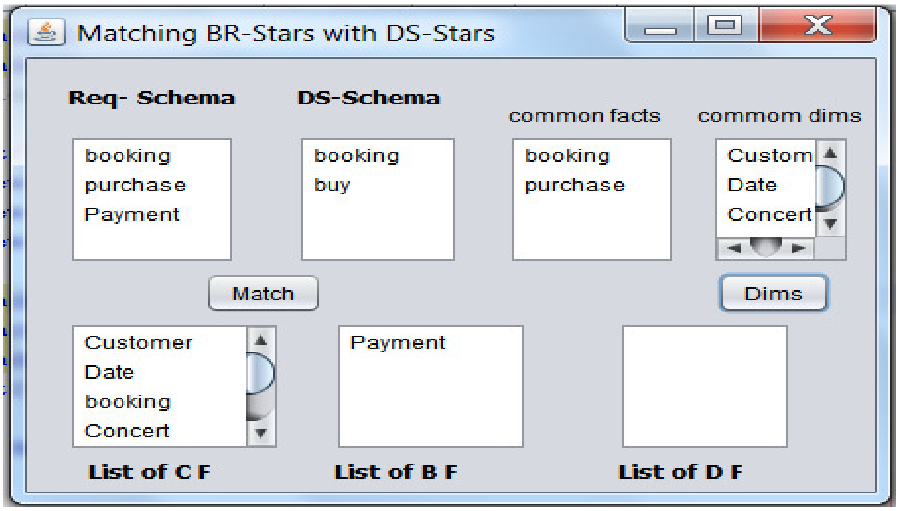
commom dims: Concert, Room, Date

List of C F: Customer, Date, booking, Concert

List of B F: Payment

List of D F:

Figure 3. Our system’s result for matching BR-Star Schemas with DS-Stars Schemas of Figure2.



Req- Schema: booking, purchase, Payment

DS-Schema: booking, buy

common facts: booking, purchase

commom dims: Custom, Date, Concert

List of C F: Customer, Date, booking, Concert

List of B F: Payment

List of D F:

Figure 4. Our system’s result for matching BR-Star Schemas with DS-Stars Schemas having synonyms.

Table 3. Result of the application of the Boolean functions on example in Figure 2

SCF: Set of Common Facts Lists			SDF: Set of DS Facts Lists			SBF: Set of BR Facts Lists		
Fact	Dimension	Parameters	Fact	Dimensions	Parameters	Fact	Dimensions	Parameters
Bookings	Date	-	-	-	-	-	-	-
	Client							
Purchase	Customer	-	-			-		
	Concert							
	Date							

Table 4. Result of the application of MatchStars algorithm on example in Figure2 with different values for threshold.

SCF: Set of Common Facts Lists			SDF: Set of DS Facts Lists			SBF: Set of BR Facts Lists			Threshold
Fact	Dimension	Parameters	Fact	Dimension	Parameters	Fact	Dimension	Parameter	
Bookings	Date	-	Payments	Customer	-	Sales	Date	-	< 1
				Payment					
	Client			Date					
Purchase	Customer	-	-			-			
	Concert								
	Date								
Bookings	Date	-	Payments	Customer	-	Sales	Date	-	1
				Payment					
	Client	-		Date					
			Purchase	Customer	-	-			
				Concert					
				Date					

Table 5. Result generated after applying MatchStars algorithm for example 2 having BR-Stars limited the first two facts (BOOKINGS and PURCHASE)

SCF: Set of Common Facts Lists			SDF: Set of DS Facts Lists			SBF: Set of BR Facts Lists		
Fact	Dimension	Parameters	Fact	Dimension	Parameters	Fact	Dimension	Parameter
Bookings	Date	-	Payments	Customer	-		-	-
	Client			PaymentM				
	Room			Date				
	Concert							
Purchase	Customer	-	-			-		
	Concert							
	Date							