The Multidimensional Patterns of Linked Open Data

Dihia Lanasri[1], Selma Khouri[1], Roaya Saidoune[1], Kamila Boudoukha[1], Ladjel Bellatreche[2]
[1] ESI, Algiers
Algeria
[2] LIAS/ISAE-ENSMA, Poitiers
France

ABSTRACT: *Data and information resources required for companies are norally available and are extracted from the internal information resources which are loaded in the dataware house of the companies. In the current society, the data that companies are looking for is no longer limited to the internal data but also incorporates the external sources including Linked Open Data. Normally these sources are composed of two components of dataset and data left behind the people while exploiting these data by SPARQL query logs. Considering the breadcrumbs in the process of DW construction represents a big challenge due to the volume, variety and the expertise of users who produce them. In this paper, we used Crumble Cube a comprehensive tool for identification multidimensional patterns from LOD breadcrumbs. It offers designers mechanisms to manage investigate and visualize these breadcrumbs before integrating them into a target data warehouse.*

Keywords: Multidimensional Patterns, Linked Open Data, Open Data, Smart Data, Data Management

Received: 29 May 2019, Revised 10 September 2019, Accepted 19 September 2019

DOI: 10.6025/jisr/2019/10/4/119-123

1. Introduction

Industries normally do not end the information integration from the local resources in to their own managed data warehouses to obtain enhanced value. Local sources are usually insufficient to let these industries being more comprehensive in naming intelligent decisions. In this context, they are obliged to include smart external resources to enrich their internal sources to get advanced insights. Under the impulse of the openworld data the enrichment technique has who do not benefit in terms of the added value of designing several constantly evolving digital entities such as Knowledge graphs and data repositories. (9). This enrichment is usually performed by resourcing to external sources such as sensors, social network messages, and Linked open data. These sources are usually composed of two components namely the datasets and second the data left behind the people while exploring datasets. LOD is one of the valuable external sources. Their principles allow the naïve and expert people accessing and retrieving the data stored and published through SPARQL, endpoints reflecting the needs. Various initiatives such as USEWOD and LSQ collected these query logs and that may contain valuable multidimensional patterns that include facts, dimensions, and hierarchies. Face to this situation these crumbs have to be studied to change the hidden data into smart multidimensional data which refines the existing data warehouse.

In the last few years, a couple of studies proposed a materialized integration of the first component of LOD in the data ware house design. [4, 5] where multidimensional models are derived from the LOD datasets. In these proposals, the LOD datasets undergo the same DS processes performed for internal data. Other studies analysed the LOD dataset via standard LPA operational using tools such as SPARQLytics. [7]. These studies ignore the second component of LOD: the query logs. Integrating these crumbs in the DW construction is not an easy task, because they are large, contain hidden smart multidimensional data, do not have the same structure and format as the internal data sources. To facilitate this integration these crumbs have to be prepared in terms of cleaning, exploration, investigation [3] and visualization in order to extract smart multidimensional data. Once prepared they will be integrated into the DW. In this study we focus on the investigation of the query-logs for discovering MDP [1], the enrichment process of the MDP in an existing DS is not supported by our tool, it requires schema matching and integration techniques that have been widely addressed in many studies.

In this current exercise we propose a tool Cumber4Cube for augmenting an existing through automate discovering of MDP from the LOD query logs and creating their associated conceptual multidimensional graphs than can be exploited by designers to investigate the hidden data in the query logs. This work is organized as below. The section 2 presents the architecture of Crumbe4cubes. Section 4 presents our demonstration and section 4 concludes our work.

2. System Architecture Overview

Crumbs4Cube generates a conceptual multidimensional graph based on the Fact/dimension dichotomy. Our applications adopts a three tiered architecture on the MVC framework.

I) Data Layers

In this data layers, Jena TDB triple stored is used to persist the RDF graphs. LOD are defined using RDF standard presenting data in the form of RDF graphs. Each RDF statement is a directed labelled graph and takes the form <s, p, o, g> such that in the graph label g, subject s has the predicated (i.e property) p and the value of that property is the object o. Sparql queries are defined for matching a defined subgraphs of triples <s', p', o'> in the queried RDF graph. For instance, *Example* 1 illustrated a Sparql query of scholarly data LOD used in our demonstration.

```
SELECT DISTINCT  ?pred  ?author_url  ?author_name

WHERE { <uri/conquer-query>  bibo:authorList  ?authorList.  ?authorList  ?pred
?author_url ?author_url foaf :name  ?author_name}
```

II) Business Layer

The different modules of Crubms4Cube (Figure 1) are developed using JAVA and the Jena API (ARQ and Core Libraries) as follows.

(1) Clean log-queries

It consists in syntactic and semantic cleaning of SPARQL queries in order to keep valid SELECT queries using REGEX, HTML parser (to decode queries) and ARQ JENA (SPARQL parser).

(2) MDP Exploration

It is the core of our process. It consists in constructing the star graphs from the cleaned queries using three identified scenarios.

Scenario 1: Aggregate Queries. It starts by identifying the aggregate queries among the set of valid queries. In such queries, the measure of facts are already aggregated via functions such as COUNT, SUM etc and analysed by some dimensions identified in the GROUR BY clause, the identification of facts and dimensions in consequently straightforward. Aggregate queries are of the form:

```
Select    ?var1   ?var2 ...?varn    AggFct1(?varx1) ...AggFct(?varxm)}

Where {triple  patterns  <S, P, O>}  Group  By  ?var1,..., ?varn
```

Scenario 2: Star Graphs. Unlike the first scenario, this scenario investigated the interactions between queries. The MDP [6] are

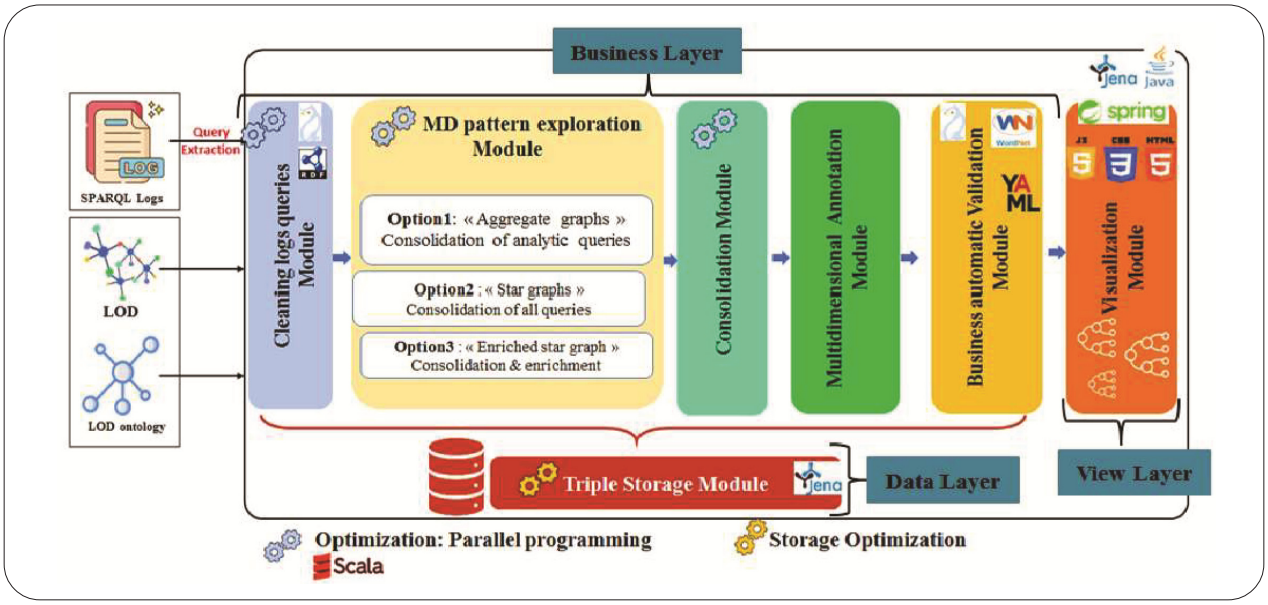

Figure 1. Crumbs4Cube architecture

identified from the consolidation of the different graphs generated from the all valid queries.

Scenario 3: Enriched Star Graphs. It extends scenario 2 by enriching the obtained MDP with new concepts identified from the LOD dataset.

For **Scenario 1**, the following steps are performed for each query. ***(a) Aggregate query construction*:** For identifying the variables, in each triple <?*S*, *P*, ?*O*> of the query (*like ?pred or ?author_url* in the query of *Example* 1 given above), we defined BP Enriched function that analyses the queries by: (i) adding a triple <?*S*, *rdf:type,* ?*type*> to receive ?type reflecting the concept of ?S (same processes is achieved for variables ?O) (ii) adding a triple <?*P*, *rdf:range,* ?*type*> for identifying the *Datatype* of property *P*, ***(b) Query Execution*:** Executed the query enriched by the cited triples on the SPARQL endpoint and retrieves the results. ***(c) Graph Construction*:** From the obtained results, the process constructs the multidimensional graph of each query where the attributes on which aggregate functions are applied are considered as measures. The concepts of these attributes are considered as facts, the concepts associated to the Group by are considered as dimensions or dimension attributes according to their type. These MDP are constructed as graphs where the vertex is the fact and dimensions & attributes are the related nodes.

For **scenarios 2 and 3**, the following steps are performed: ***(a) Transform Select to Construct Query*:** After enriching the queries using BP Enriched function one main step that is specific to scenarios 2 and 3 is the construction of a graph reflecting the triple patterns of each query, using a Construct query. ***(b) Query Extraction*:** It executes the Construct queries on the SPARQL endpoint then retrieves the results as an RDF graph. ***(c) Graph alleviation*:** It consists in lightening graphs by removing triples containing generic properties including rdf:type, rdf:label etc and generic classes like owl: Thing that are not relevant for MDP. ***(d) Consolidation*:** It consists of two main phases. ***(i) Consolidation by vertex*:** The obtained graphs sharing the same vertex are consolidated. ***(ii) Consolidation by nodes*:** For each non-vertext node in a given graph, if it exists as a vertex of *G*. All the following steps are applied for all scenarios.

(3) Graphs Alleviation

A valid multidimensional graph has at least a fact with a dimension and a measure. Non valid graphs are rejected.

(4) Multidimensional Association

It allows annotating the multidimensional graphs by adding triple of the Class, annotated, Annotation such as a Class represents the node of the graph, and Annotation. $\in$ {*Fact, Dimension, Leveldimension, Factattribute, Dimension Attributes, Measure*}.

(5) Graph Enrichment
Following scenario 3, this module looks deeper into the interactions between the query logs and the LOD dataset. The process starts exploring new properties and concepts that enrich the obtained multidimentional graph annotated. For each fact/dimension/level (*S*) of the consolidated graphs, we look for its related triples <*S*, *P*, *O*> in the LOD knowledge base. For each resources (*O*): if *P* is datatype property, *O* is considered as an attribute. If *P* is a relationship, *O* is considered as a dimension or level.)

(6) Authentic Business Validation
It allow as an automatic validation using a domain ontology if available (or atleast Wordnet Ontology) using WS4J library. Automatic validation rules are checked defined in [8].

III) View Layer
Represents the interfaces of Crumbs4Cube that (a) help the designer discovering MDP from LOD. (b) calculate the quality metrics identified (saved in YAML files) of the multidimentional model obtained. This layer is developed in HTML/CSS/Javascript (JS) graphs are represented with JS amCharts4 Library. Scalar language is used for optimization issues, Apache Maven and Github for versioning management. The source code of our tool Crumbs4Cube is available at https://github.com/SaidouneROuaya/cibeQE.

3. Demonstration Overview

For the demonstration of Crumbs4Cube we simulate the process of exploiting LOD query logs using Scholarlydata Logs for business validation we used the Scholarly Data Ontology. Log files contain 5485082 queries. This demonstration was performed on a machine OS Windows 10, 128GB of RAM, Processor Intel ® Xeon ® CPU E5-2673 V 4 @2.30 GHz.

Scenario 1: A small rate (1.54%) of aggregate queries have been obtained from the set of valid queries, which impacted the rate of MDP discovered. This encouraged us to consider the second scenario.

Scenario 2: The resulting rate of valid queries in the logs is about 63.24% Crumbs4Cube allows the designer to launch the process of generating a multidimensional graphs from the treated queries. For this scenario and after validation 26 multidimensional graphs have been discovered from the logs.

Scenario 3: After enriching the multidimensional graph with Scholarly Data dataset we obtained new results that show new patterns discovered from the dataset (Figure 3 for the concept Workshop Events). The tool also provides different metrics for

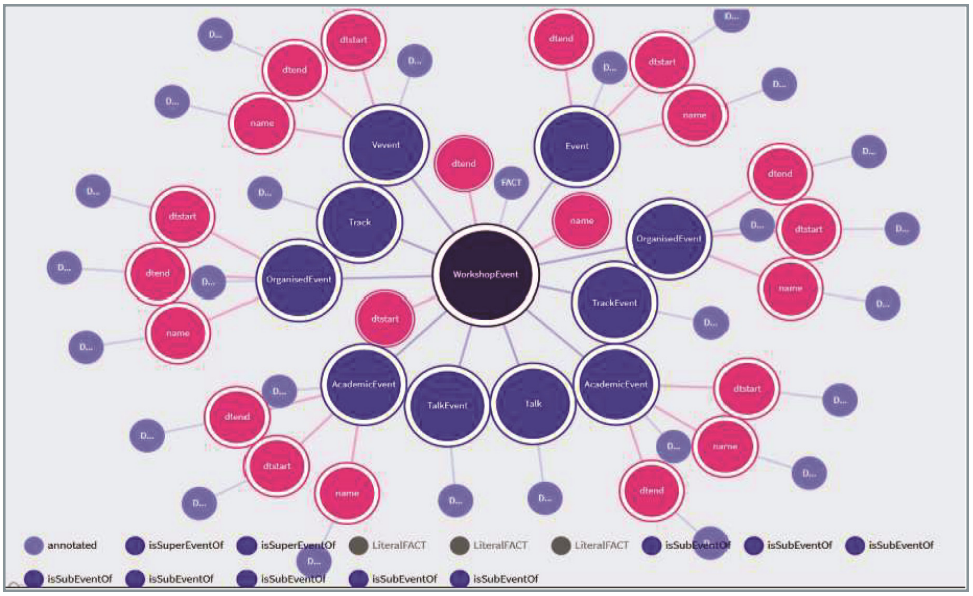

Figure 3. WorkshopEvents Multidimensional Graph

all scenarios used to evaluate the quality of the discovered MDP, they are used to help the BI designer to validate the obtained MD graphs. A demonstration video is hosted at https://youtube/4Z1X12gB4.

4. Conclusion

IN this paper we presented a new tool Crumbs4Cube that discovers MDP from LOD query-logs exploration. Crumbs4Cube is developed to assist the designer through different modules and allows extracting useful multidimensional insights that are difficult to explore directly from the dataset without prior knowledge.

References

[1] Khouri, S., Lanasri, D., Saidoune, R., Boudoukha, K., Bellatreche, L. (2019). Loglinc: Log queries of linked open data investigator for cube design. DEXA.

[2] Lehmann, J. (2015). Dbpedia - A large scale, multilingual knowledge base extracted from wikipedia. *Semantic Web* 6 (2) 167-195.

[3] Morton, K., Balazinska, M., Grossman, D., Mackinlay, J. (2014). Support the data enthusiast: Challenges for next generation data-analysis systems. *In*: Proceedings of the VLDB Endowment 7 (6) 453-456.

[4] Nebot, V., Berlanga, R. (2016). Statistically-driven generation of multidimensional analytical schemes from linked data. *Knowledge-Based Systems*, 110, 15-29.

[5] Rizzi, S., Gallinucci, E., Abello, M. G. A., Romero, O. (2016). Towards exploratory olap on linked data. *In*: SEBD. 86-93.

[6] Romero, O., Abello, A. (2010). Automatic validation of requirements to support multidimensional design. *Data & Knowledge Engineering*, 69 (9) 917-942.

[7] Rudolf, M., Voigt, H., Lehner, W. (2017). Sparqlytics: multidimensional analytics for rdf. Datenbanksysteme fur Business, Technologie and Web (BTW 2017).

[8] Salem, A., Ben-Abdallah, H. (2015). The design of valid multidimensional star schemas assisted by repair solutions. *VJCS* 2 (3) 169-179.

[9] Wang, X., Carey, M. J. (2019). An IDEA: An ingestion framework for data enrichment in asterixdb. CoRR abs/1902.08271.