Towards NoSQL Graph Data Warehouse for Big Social Data Analysis

Hajer Akid and Mounir Ben Ayed

Research Groups in Intelligent Machines, University of Sfax
National School of Engineers (ENIS), BP 1173, Sfax, 3038, Tunisia,
Faculty of Science of Sfax, BP 1171, Sfax, 3000, Tunisia
akid.hajer@gmail.com
mounir.benayed@ieee.org

Abstract. Big Data generated from social networking sites is the crude oil of this century. Data warehousing and analysing social actions and interactions can help corporations to capture opinions, suggest friends, recommend products and services and make intelligent decisions that improve customer loyalty. However, traditional data warehouses built on relational databases are unable to handle this massive amount of data. As an alternative, NoSQL (Not only Structured Query Language) databases are gaining popularity when building Big Data Warehouses. The current state of the art of proposed NoSQL data warehouses is captured and discussed in this paper. The paper will also focus on the opportunities and challenges of using NoSQL graph databases for storing and querying Big Social Data and how graph theory can help to mine information from these data warehouses.

Keywords: Big Data; Social graph; Decision Support Systems; NoSQL Data Warehouses; Graph-NoSQL Data Warehouse; Graph Theory; Big Social Data Analysis.

1 Introduction

Social network is a term used to describe web-based services that allow individuals to construct a public or semi-public profile within a bounded system, articulate a list of other users with whom they share a connection, and view and traverse their list of connections and those made by others within the system[10]. During the last decade, the number of shared activities and exchanged opinions about products and services on social networks has grown exponentially. Therefore, many researchers and organizations are looking for making sense of this river of massive data to make better strategic decisions. Since data warehouses are accepted as the heart of decision support systems, an emerging topic is building Big Data Warehouses able to handle such amount of data. Traditionally, data warehouses are mainly built using R-OLAP (Relational-OLAP) approach based on relational databases. Despite their maturity, these databases are facing many challenges [22] [20] such as scalability and concurrency issues,high cost of the

join operation and inability to analyse unstructured data. These problems make it difficult to store and manage big data effectively and enterprises are therefore turning to NoSQL database technology. Even if NoSQL data models and solutions have been well introduced, compared and discussed [14] [3] [11] [12], no survey on proposed approaches to build NoSQL data warehouses is available. Hence, this paper intends to highlight the most important features of proposed logical models and make comparison between them to show advantages and drawbacks of each model. Further, and more importantly, Big Data has been mainly tied to Hadoop, MongoDB, and Cassandra. However, we believe that NoSQL graph databases are going to have a great impact on the technology landscape due to the re-emergence of interest in storing and analysing graph data. An important aspect of social networks is the connectivity of its nodes. Hence, we will introduce later a new V to describe Big Social Data. Storing and analysing these connections can give more informations about people's interests than their names and occupations. Unlike other database management systems, which require us to infer connections between entities using contrived properties such as foreign keys, or out-of-band processing like map-reduce, relationships are first-class citizens of the graph data model[24]. Although NoSQL graph databases have been recently introduced, the interest towards graphs is not new. Indeed, graph Theory, and more generally a graph,has been around for nearly 300 years and it is considered as one of the best and clearest way to represent and detect the structure defined by the way a relation connects different individuals of a social group[19]. In addition, many authors [15] [2] [27] proved that graph theory is a powerful and very useful in social networks analysis. Therefore, building NoSQL graph data warehouse and taking advantage of graph theory would be an efficient way to store and mine Big Social Data.

The reminder of this paper is organized as follows. Section 2 explains how the common V's of Big Data are not enough to characterize Big Social Data. In Section 3, we discuss related works. In section 4, we argue the efficiency of building NoSQL graph data warehouse for Big Social Data Analysis. Our general approach is proposed in section 5. Finally, section 6 concludes the paper and presents future works.

2 Big Social Data

The term Big Data is an evolving term and it has been defined in several ways in the literature. According to Gartner, Big data is high-volume, high-velocity and/or high-variety information assets that demand cost-effective, innovative forms of information processing that enable enhanced insight, decision making and process automation. IDC, International Data Corporation, defines Big Data technologies as a new generation of technologies and architectures designed to economically extract value from very large volumes of a wide variety of data by enabling high-velocity capture, discovery and/or analysis. As for the Mckinsey Global Institute (MGI), Big Data refers to datasets whose size is beyond the ability of typical database software tools to capture, store, manage and analyze.

However, since the appearance of the web 2.0 construct, the web has become highly connected. Thus, we believe that these definitions and the common 3 V's of Big Data are not enough to describe Big Data generated from social networking sites. We propose to add another V which is "Very-Connected". Very Connected data is data whose informations are related and where value is hidden in connections. between things.

3 Related Works

Big Data analysis has become an emerging area. It aims to extract hidden values, provide suggestions and help decision makers. Many methods, architectures and tools for big data analysis must be established to meet the challenges of this phenomenon. Thus, new data warehouse and database technologies are introduced to address this problem. In particular, NoSQL are emerging as a new generation of databases. In this section, we discuss some related works in building Big Data Warehouses using NoSQL databases in order to introduce our approach. We will also focus on how graphs are the most suitable when dealing with connected data.

3.1 NoSQL data models

Recent years have seen a meteoric adoption of NoSQL, a family of data storage that emerged as an alternative to relational databases. There are four types of NoSQL data stores which vary by their data model.

Key-Value stores These systems are often considered as the most simple type of NoSQL stores. They store objects in the form of key-value pair. The key is the index used to fetch the data. All the key-value stores support insert, delete, and lookup operations and provide scalability through key distribution over nodes[3]. Yet, the queries supported by such systems are very simple since they can only cover the key. The users have to make efforts and extract the desired value form the block of returned data.

Over the past few years, many key-value databases have appeared such as: DynamoDB, LevelDB, GenieDB, RocksDB, Berkeley DB, Oracle NOSQL Database, Azure Table Storage, Riak, Redis and Voldemort.

Column stores Column family stores were motivated by the success of Googles BigTable. Unlike relational databases, a column-oriented database stores data by column rather than by row. Indeed, relational model has a static schema that requires the definition of the same columns for the several rows. However, the use of a column oriented database allows to define different columns for each line which avoids columns having NULL values. Column family databases are made up of column, super column, column family.

When you have volume and variety of big data, you should use a column-oriented databases such as: HBase, Cassandra, Hypertable, Amazon SimpleDB and BigTable.

Document stores In document stores, value associated with a unique key is a document which have a different schema to support more complex data forms. The structure of the documents and their parts is represented in JSON (JavaScript Object Notation), BSON (Binary JSON) or XML (eXtensible Markup Language) formats.

Document databases are most useful when you have to produce a large amount of data and they need to be dynamically assembled from elements that change frequently. MongoDB, SimpleDB, CouchDB and Couchbase Server are some examples of NoSQL databases using this model.

Graph stores Is a type of NoSQL database that uses graph theory to the storage of information about the relationships between entries such as the relationships between people in social networks or between items and attributes in recommendation engines. A graph database is essentially a collection of nodes and edges connecting each other through relations. Each node represents an entity such as a person, customer, businesses, accounts, or any other item and each edge represents a connection or relationship between two nodes. In a graph database, a node is defined by an unique identifier, a set of outgoing or incoming edges and a set of properties expressed as key/value pairs. Each edge is defined by an unique identifier, a starting and ending node and a set of properties.

Past surveys on NoSQL have done a great job of describing their applications, advantages and challenges. In particular, many authors [21] [25] [16] gave an overview on the multiple commercial and open-source implementations of NoSQL databases for helping users to "Use the right tool for the job". They have shown the need to choose the suitable NoSQL data model after evaluating data. In addition, some authors [17] [27], focused on the comparison between relational and NoSQL databases based on their features and their tools. Many criteria such as data model, scalability, Big Data handling and ability of use for data warehousing were considered.

3.2 Graph databases

Graph databases are the best way to model, store, and manage connected data. They are almost used in three major groups of applications which are chemical and biological data, social networks, and the web [26]. Far from being a recent data handling development, graphs ans graph theory were actually around for nearly 300 years and can be attributed to Leonhard Euler. Yet, it is only in the past few years that graph theory started to be used on information management. In that time, graph databases have helped solve important problems in the areas of social networking, master data management, geospatial, recommendations, and more [24].

Many surveys have discussed the opportunities behind graph databases. For instance, in[1] the author encouraged the use of graph data models in real life applications where interconnectivity is a key feature and concentrated on data structures, query languages, and integrity constraints. Also, some others made

a comparison between graph databases and relational databases[26] based on objective and subjective measures. The objective measures include processing speed, disk space requirements and scalability. Subjective measures include maturity, ease of programming and security. Authors found that graph database did better than the relational at the structural type queries and full text character searches. In other work the author presented three major groups of applications for graph data which are chemical and biological data, social networks, and the web. Moreover, in an other paper [23], authors found that the use of relational databases in highly connected data applications such as web, computer networks, geographical structure is inefficient. Thereby he proposed Graph Database Management System to convert a relational database to graph database. This idea can be revisited taking advantage of the current graph NoSQL tools ans solutions.

As both of NoSQL surveys have evaluated its performance away from the field of decision support systems (DSS), this paper will focus more on the approaches proposed to implement Big Data warehouses using NoSQL.

3.3 NoSQL Columnar Warehouses

To build columnar big data warehouses, an indirect approach was proposed [18] to convert in two phases the multidimensional model of a data warehouse to HBase which is column-oriented database management system similar to Big Table. The first phase is to transform the relational schema of the data warehouse into HBase schema based on the data model of HBase. In the second phase, relationships between two schemas are expressed as a set of nested schema mappings. Another indirect approach [13] helped to implement three-dimensional data model, which uses the version dimension of HBase to store the values of a data item over time. Results showed that the performance was improved by the use of the third dimension of HBase. Despite their ability to build columnar NoSQL warehouses, these approaches are limited to the transformations of a logical model (Relational model) to another logical representation (NoSQL data model) and do not respect the constraints described by the conceptual level. This problem gave birth to another generation of approaches. First, a benchmark called CNSSB (Columnar NoSQL Star Schema Benchmark) was proposed for the columnar NoSQL data warehouse [9]. Yet, authors didn't focus on the mapping process. In a further work [8], they proposed three approaches called NLA, DLA,and DLA-CF to convert the multidimensional model to NoSQL data model. The NLA allows the mapping with a normalized approach. The DLA proposes transforming the data conceptual model into a model based on a large structure (table) called BigFactTable. The DLA-CF proposes transforming the data conceptual model into columnar NoSQL logical model. Results showed that DLA and DLA-CF are more efficient than NLA when queries requires to join dimensions. Other authors [5] implemented two decision support systems based on column-oriented NoSQL and document-oriented NoSQL using respectively HBase and MongoDB. Results have shown the performance of the columnar

NoSQL data warehouse when loading data generated by the benchmark TPC-DS.

3.4 NoSQL Document Warehouses

Document-oriented NoSQL attracted other researches. In [7], authors evaluated the performance, scalability and fault-tolerance of using MongoDB which is a document-oriented NoSQL database with Hadoop for scientific data analytic. Results shown important a great interest of using MongoDB for storage and Hadoop for analysis. However authors didn't give any detail on the transformation process. This latter has been the aim of a recent work[5]. A direct approach was proposed to build document NoSQL warehouse using MongoDB. Comparing this latter to column NoSQL warehouse has shown the performance of the document NoSQL warehouse when querying the database. Also, other authors [4] three approaches called MLD0, MLD1 and MLD2. The MLD0 approach transform every star schema to a collection of documents. This latter stores fact's measures and dimension's attributes as key-value pairs. The first approach doesn't use the concept of embedded documents. The MLD1 approach transforms every star schema to a collection of documents where measures are stored in an embedded document and the attributes that belong to the same dimension are also stored as embedded documents. The MLD2 approach store the star schema in a collection of documents using references instead the concept of embedded document to refer the documents of the corresponding dimensions. Results showed that MLD0 and MLD1 are more efficient than MLD2.

4 NoSQL Graph Data Warehouse: opportunities and challenges

Nowadays, social networks services have explosively grown. Social networks are often modeled as large graphs which contains a huge number of interconnected nodes. Building Graph NoSQL social warehouses requires to store the multidimensional data into graph-oriented NoSQL database. The transformation process have to express the concepts of multidimensional model in terms of nodes and edges. Building such a data warehouse can help to navigate on the collected data for supporting decision makers and make recommendations. However, while On-line analytical processing (OLAP) is a powerful primitive for structured data analysis, it faces major challenges in manipulating interconnecting data. Thus, authors [6] proposed in recent work a new data warehousing model, namely Social Graph Cube to support OLAP technologies on multidimensional social networks. Other challenges are the absence of a standard query language and the variety of data models. Research and industry will certainly influence the future direction of NoSQL graph database.

In summarize, both of direct and indirect approaches enabled the built of big data warehouses. Yet, only the direct approaches have considered the conceptual multidimensional model during the transformation process. The main drawback

of the direct approaches hierarchies didn't be conserved after the transformation process. Here we can mention that the concepts of embedded documents and super columns can be of great interest to store hierarchies. Also, we noticed that only column data models and document data models were chosen as target models.

5 NoSQL Graph Data Warehouse: A novel Approach

We propose in this section a new architecture for the construction of a graph data warehouse for the storage and analysis of Big Social Data. As this category of database management systems is adapted to the complexity of the problem, not volume, we find it necessary to use first a document-oriented NoSQL DBMS for the storage of the information shared on the social web. Than we will store in graph NoSQL database the nodes and links that we want to analyse. Although the approaches proposed in the literature only allow to migrate from a relational database management system to another NoSQL database management system, we will propose a new Intra-NoSQL approaches to migrate from NoSQL to NoSQL. This is a recursive approach. The loading of data will be automatic using the ETL process. Finally, we will focuse ont the analysis and extraction of relevant informations using the graph algorithms known by their maturity (in-depth courses, courses in widths, etc.).

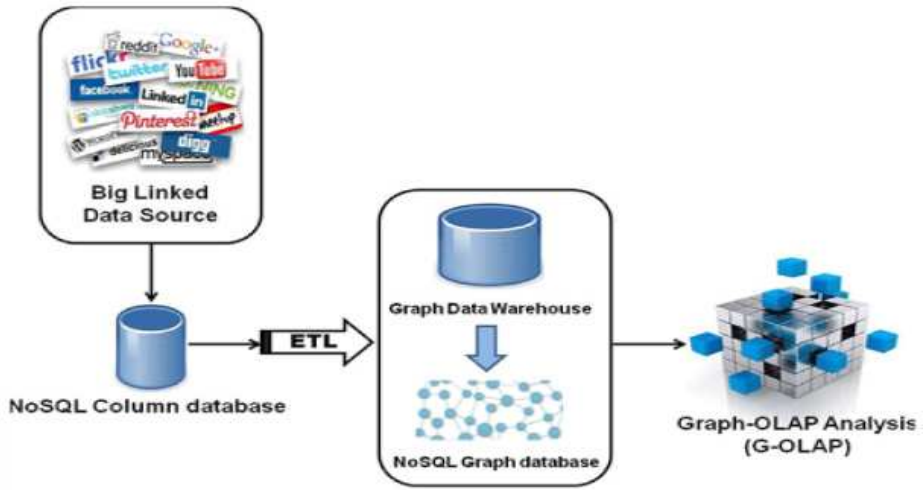

Fig. 1. Building NoSQL Graph Data Warehouse

6 Conclusion

This paper gave an overall summary of the current state of big data warehouses. We focused almost on the process of transformation that convert multidimensional model to NoSQL data model. Moreover, we discussed the opportunities and challenges behind the implementation of graph NoSQL warehouses in the context of social networking. Indeed, the use of Graph NoSQL databases to data warehouse linked data such as social data is of great interest.Therefore we proposed a novel approach to build NoSQL Graph Data Warehouse to storage and mine Big Social Data.

References

1. Angles, R., Gutierrez, C.: Survey of graph database models. ACM Computing Surveys (CSUR) 40(1), 1 (2008)
2. Cartwright, D., Harary, F.: A graph theoretic approach to the investigation of system-environment relationships. Journal of Mathematical Sociology 5(1), 87–111 (1977)
3. Cattell, R.: Scalable sql and nosql data stores. Acm Sigmod Record 39(4), 12–27 (2011)
4. Chevalier, M., El Malki, M., Kopliku, A., Teste, O., Tournier, R.: Implantation not only sql des bases de données multidimensionnelles. In: Colloque VSST (2015)
5. Chevalier, M., El Malki, M., Kopliku, A., Teste, O., Tournier, R.: Implementing multidimensional data warehouses into nosql. In: 17th International Conference on Enterprise Information Systems (ICEIS15), Spain (2015)
6. De Virgilio, R., Maccioni, A., Torlone, R.: R2g: a tool for migrating relations to graphs. In: EDBT. pp. 640–643 (2014)
7. Dede, E., Govindaraju, M., Gunter, D., Canon, R.S., Ramakrishnan, L.: Performance evaluation of a mongodb and hadoop platform for scientific data analysis. In: Proceedings of the 4th ACM workshop on Scientific cloud computing. pp. 13–20. ACM (2013)
8. Dehdouh, K., Bentayeb, F., Boussaid, O., Kabachi, N.: Using the column oriented nosql model for implementing big data warehouses. In: Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA). p. 469. The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp) (2015)
9. Dehdouh, K., Boussaid, O., Bentayeb, F.: Columnar nosql star schema benchmark. In: International Conference on Model and Data Engineering. pp. 281–288. Springer (2014)
10. Ellison, N.B., et al.: Social network sites: Definition, history, and scholarship. Journal of Computer-Mediated Communication 13(1), 210–230 (2007)
11. Gajendran, S.K.: A survey on nosql databases. University of Illinois (2012)
12. Grolinger, K., Higashino, W.A., Tiwari, A., Capretz, M.A.: Data management in cloud environments: Nosql and newsql data stores. Journal of Cloud Computing: Advances, Systems and Applications 2(1), 1 (2013)
13. Han, D., Stroulia, E.: A three-dimensional data model in hbase for large time-series dataset analysis. In: 2012 IEEE 6th International Workshop on the Maintenance and Evolution of Service-Oriented and Cloud-Based Systems (MESOCA). pp. 47–56. IEEE (2012)

14. Han, J., Haihong, E., Le, G., Du, J.: Survey on nosql database. In: Pervasive computing and applications (ICPCA), 2011 6th international conference on. pp. 363–366. IEEE (2011)
15. Harary, F., Norman, R.Z.: Graph theory as a mathematical model in social science (1953)
16. Hecht, R., Jablonski, S.: Nosql evaluation. In: International conference on cloud and service computing. pp. 336–41. IEEE (2011)
17. Jatana, N., Puri, S., Ahuja, M., Kathuria, I., Gosain, D.: A survey and comparison of relational and non-relational database. International Journal of Engineering Research & Technology 1(6) (2012)
18. Li, C.: Transforming relational database into hbase: A case study. In: 2010 IEEE International Conference on Software Engineering and Service Sciences. pp. 683–687. IEEE (2010)
19. Martino, F., Spoto, A.: Social network analysis: A brief theoretical review and further perspectives in the study of information technology. PsychNology Journal 4(1), 53–86 (2006)
20. Mohamed, M.A., Altrafi, O.G., Ismail, M.O.: Relational vs. nosql databases: A survey. International Journal of Computer and Information Technology 3(03), 598–601 (2014)
21. Moniruzzaman, A., Hossain, S.A.: Nosql database: New era of databases for big data analytics-classification, characteristics and comparison. arXiv preprint arXiv:1307.0191 (2013)
22. Nayak, A., Poriya, A., Poojary, D.: Type of nosql databases and its comparison with relational databases. International Journal of Applied Information Systems 5(4), 16–19 (2013)
23. Prasanth, N., Arul, K.: Converting employee relational database into graph database. Middle-East Journal of Scientific Research 22(11), 1618–1621 (2014)
24. Robinson, I., Webber, J., Eifrem, E.: Graph Databases: New Opportunities for Connected Data. " O'Reilly Media, Inc." (2015)
25. Tudorica, B.G., Bucur, C.: A comparison between several nosql databases with comments and notes. In: 2011 RoEduNet International Conference 10th Edition: Networking in Education and Research. pp. 1–5. IEEE (2011)
26. Vicknair, C., Macias, M., Zhao, Z., Nan, X., Chen, Y., Wilkins, D.: A comparison of a graph database and a relational database: a data provenance perspective. In: Proceedings of the 48th annual Southeast regional conference. p. 42. ACM (2010)
27. Wasserman, S., Faust, K.: Social network analysis: Methods and applications, vol. 8. Cambridge university press (1994)