Graphs enriched by Cubes for OLAP on Bibliographic Networks

Wararat Jakawat

Université de Lyon (ERIC LYON 2), France
E-mail: wararat.jakawat@univ-lyon2.fr

Cécile Favre

Université de Lyon (ERIC LYON 2), France
E-mail: cecile.favre@univ-lyon2.fr

Sabine Loudcher

Université de Lyon (ERIC LYON 2), France
E-mail: sabine.loudcher@univ-lyon2.fr

Abstract: With the recent growth of bibliographic data, many research fields work on defining new techniques for bibliographic data analysis. In this context, data of interest could be represented as heterogeneous networks, in which there are multiple object and link types that have multidimensional attributes. In order to analyze information network in multidimensional way, OLAP (Online Analytical Processing) is an important tool. OLAP is effective for analysing classical data, however, it must be adapted for networked data by considering nodes and the interactions among nodes. In order to quickly analyse information, we propose graphs enriched by cubes. Each node and edge of the considered network are described by a cube. It allows greater multidimensional analysis possibilities as a user may gain insight within both network and cubes. Our proposal also solves the slowly changing problem in OLAP analysis. To illustrate our approach, we integrate three bibliographic databases. Then we implement our approach and we show results on a real data set. We perform the experimental studies of the efficiency of our proposal.

Keywords: OLAP, Bibliographic Networks, Data Cube, Graph database

Reference to this paper should be made as follows: xxxx (xxxx) 'xxxx', *xxxx*, Vol. x, No. x, pp.xxx–xxx.

Biographical notes: Wararat JAKAWAT is currently a PhD student in Computer Science at the University of Lyon, France. She is a member of the Decision Support Databases research group within the ERIC laboratory. She received her MSc degree in Computer Science from Prince of Songkla University, Thailand in 2010. After researching on the index techniques of XML documents, her current research interests now relate to OLAP on information networks.

Cécile FAVRE has been an associate professor at the University of Lyon in France since 2009. She is a member of the Decision Support Systems research group within the ERIC laboratory. She belongs to the Anthropology, Sociology and Political Science Faculty, where she is in charge of a Master in gender studies. After doing some research on the integration of data mining techniques into DBMSs and personalization within data warehouses, her current research interests now relate to Digital Humanities and are focusing on social network analysis, graph OLAP (On Line Analytical Processing), especially on the context of bibliographic data.

Sabine LOUDCHER is a Full Professor in Computer Science at the Department of Statistics and Computer Science of the University of Lyon, France. She received her PhD degree in Computer Science in 1996 and since 2015 she is a full professor. From 2003 to 2012, she was the Assistant Director of the ERIC laboratory. She carries out research on OLAP and Data Mining. She is more interested about data coming from documents or social networks. Her current work focuses on Graph OLAP, Text OLAP and Text Mining. She is involved in several projects specially in Digital Humanities with Geography researchers or political scientists.

1 Introduction

Over the last few years, information networks have been quickly increasing because of the popular use of Web, blogs

and various kinds of online databases. In general, networks can be homogeneous or heterogeneous. Homogeneous networks contain a single object type and a single link type such as friends networks, authors networks and movies networks. Links may include a label or a weight. Heterogeneous networks are composed of multiple object and link types. For example, an author-paper network is a heterogeneous network with two types of nodes (authors and papers) and three types of edges (written between authors and papers, co-author relationship and the last one relates papers written by the same author). As Zhao et al. (2011) said, a network can also be a multidimensional network with multiple node attributes and edge attributes.

We take the example of bibliographic data because a special feature of bibliographic data is that it can be seen as an information network. Multiple research fields are concerned with bibliographic data analysis because it can yield very rich and useful information. There are many types of analysis (Statistics, Data Mining, Graph Theory, OLAP -On Line Analytical Processing- analysis, etc.) to achieve different objectives in bibliometrics (relationship studying, ranking, community mining, etc.). Among these different types of analysis, OLAP can provide the flexibility for navigating into networks, for summarizing networks at different granularity levels and from different points of view. The ability of OLAP offers users to access networks in multidimensional ways. OLAP could be a good tool in order to have a more compact view of networked data.

The traditional OLAP was used to analyze structured data but with the rapid spread of information networks, it must be adapted to manage heterogeneous networks. It is called Graph OLAP. According to our recent survey about Graph OLAP, presented in Loudcher et al. (2015), a cube is often created for a graph, to provide multidimensional and multilevel view. A cell in the cube contains a graph snapshot. In some approaches, we regret that the slowly changing dimension problem is not taken into account (Waqas et al. (2015)). For example, an author, Y. Sun, published a paper when he was at Northeastern University then he published another paper when he was at university of Illinois. There are two publications of Y. Sun, one for each university. But from the author network, if the user does an OLAP operation like a Roll-Up in order to see the institutions network, these two papers will be counted for both universities, and it is an incorrect answer. In this case, networked data is non-summarizable: a higher level network cannot be computed solely from a set of networks that are at a lower level without access raw data. Based on the above notices, OLAP must be adapted to provide networked data by considering both data objects and the interactions among objects. To complete this idea, a framework analyzing various networks built from bibliographic data is introduced in Jakawat et al. (2016). We used the properties of graph theory and we presented a conceptual graph model for bibliographic networks. The content in the model comes from multiple bibliographic databases in a way that allows us to build several different networks such as co-authorships, institutions of authors and etc. In order to perform multidimensional network, we proposed graphs enriched by cubes. Each node or edge is weighted by an OLAP cube. It allows the user to quickly analyze information that has been summarized into cubes and by viewing the graph. It supports Graph OLAP operations such as informational and topological operations and it solves the slowly changing dimension problem. In the present paper, we extend our work and we want to:

- Present a formal definition of our multidimensional model and of graphs enriched by cubes.
- Introduce new measures in the multidimensional model. In our previous work, measures were only classical numeric measures. Now, we propose to add centrality measures (degree, betweenness and closeness) in order to explore the role of nodes in each networks. Centrality is important because it indicates which node occupies critical positions in the network.
- Evaluate our proposal on real data sets. We show examples to present how using our tool to analyze data and we study the performance of algorithms.

The remainder of this paper is organized as follows. Section 2 briefly present the basic concept of Graph OLAP and reviews the related work in the field of OLAP on bibliographic network. Section 3 presents our idea of graphs with cubes. We first explain the definitions. Next, we present a way to build graphs with cubes and a way to adapt OLAP to graphs. In section 4, we present the implementation of our proposal and we show the results on real data sets. Section 5 concludes this paper.

2 Graph OLAP

2.1 General definitions

The concept of Graph OLAP was first proposed by Chen et al. (2008) in a general framework for OLAP on information networks. Graph OLAP is a collection of network snapshots where each snapshot i has k informational attributes describing the snapshot and has a graph $G_i = (V_i, E_i)$. Such snapshots represent different sets of the same objects in real applications. Dimension and measure concepts, found in traditional OLAP domain, should be re-defined for Graph OLAP.

At first, there are actually two types of graph OLAP dimensions. The first one is an informational dimension, and it uses an informational attribute. These dimensions have two roles: organizing snapshots into groups based on different perspectives and granularity (each group corresponds to a cell in the OLAP cube) and controlling snapshot views but they do not touch the inside of any individual snapshot. For example, venue and time in author-paper network are two informational dimensions. We can look at the snapshot of each group e.g., (ICDM, all years) and (data mining area, 2010). The second type of dimension is a topological dimension coming from the attributes of topological elements. Topological dimensions operate on nodes and edges within individual networks. For instance,

authors network can be generalized by merging all authors of a same institution as one node and building a new graph at the institution level.

2.2 Related work

To the best of our knowledge, J. Han's team and his colleagues were among the first to investigate OLAP on information networks. Chen et al. (2008) presented the basic definitions of OLAP on information networks and introduced a general framework called Graph OLAP. While Qu et al. (2011) focused on an efficient topological OLAP, they presented two techniques (T-Distributiveness and T-Monotonicity) in order to achieve efficient query processing and cube materialization. Zhao et al. (2011) defined the concept of multidimensional networks to abstract the real networks and they introduced a new multidimensional model, called Graph Cube. They worked with structure-enriched aggregate networks and they proposed a new type of query for multidimensional networks, called crossboid query in contrast with traditional queries named cuboid query: a crossboid query can cross more than one cube in a squery, rather than a cuboid query is on a single cube. However, these researchers did not mention how to design model for heterogeneous networks.

According to J. Han team and his colleagues, only nodes are described by attributes. However, in reality, edges are always associated with attributes as well. For example, co-authorship network contains authors as vertices and collaboration relationship between two authors as edges. The relationships may be described by time or the papers they wrote together. To solve this problem, Zhang et al. (2012) and Wang et al. (2014) proposed models to deal with both node and edge attributes. Zhang et al. (2012) defined a new multidimensional network which contains attributes of nodes and links. Node attributes were defined as dimensions in a graph cube while edge attributes were defined as dimensions in a data cube. Their model can perform OLAP query from the inner data cube to the outer graph cube. While, Wang et al. (2014) proposed a new conceptual model with a hyper graph. Graph aggregation is performed on node and edge attributes. The aggregated graph is a multigraph, where several edges can be between two nodes. It allows users to see the different views.

The closest works to those of Han's team are those of Tian et al. (2008). They proposed new operations for summarizing graphs. The first one, called SNAP, can produce a summary graph by grouping homogeneous nodes. Moreover, users can control the different resolutions of summaries by a k-SNAP operation.

In a different way, Kaya and Alhajj (2014) developed three different information networks (authors, topics and venues) with a cube-based modeling method. In these networks, each node is represented by a data cube which is analyzed by OLAP operations.

2.3 Discussion

We sum up the related work into two remarks. The first remark is about the slowly changing dimension problem presented in Waqas et al. (2015). This happens when an object (a fact, a node, etc.) changes its contents over time and when this causes a change in the structure. For example, the author, Y. Sun, published a paper when he was at Northeastern University then he published another paper when he was at university of Illinois. To the best of our knowledge, the existing approaches in Graph OLAP are not resolving this issue.

The second remark is about the visualization of a multidimensional and multi-level view over graphs. For example, a cube, with a venue dimension and time dimension, can contain a cell for $(ICDE,\ 2008)$ and another one for $(DOLAP,\ 2008)$. In the first Graph OLAP approaches, in each cell there is a graph showing collaborations between authors for this venue and this year. Between two authors, we can see the collaborations only according to the venue and the year, we don't see a global view of the collaborations. Furthermore, Wang et al. (2014) proposed a graph with multiple edges. However, their approach summarizes a set of graphs with multiple edges and it is a complex task. In contrast, Zhang et al. (2012) used a single graph as input rather than a set of graphs. Kaya and Alhajj (2014) presented only three networks which each node is represented by a cube.
Thanks to the related work, we can say we want to:

- take into account the structure of the network in order to do topological OLAP operations and not only classical or informational OLAP operations.
- deal with heterogenous networks and not only homogeneous networks.
- consider both node and edge attributes.
- have a global view of the network with multidimensional information.
- take into account the slowly changing dimension problem.

To extend OLAP on information networks, this paper presents graphs enriched by cubes. The global idea is that each node or each edge is couple with a cube according to user's requirements. This graph model supports OLAP operations for analysis.

3 Graphs enriched by Cubes

In a previous work (Jakawat et al. (2016)), we introduced the overall process of graphs enriched by cubes. Building on this, each node or edge is weighted by an OLAP cube. We also presented the conceptual graph model to represent heterogeneous multidimensional networks as shown in Figure 1.
The conceptual graph model contains four types of nodes (author, paper, venue, keyword) and four types of edges

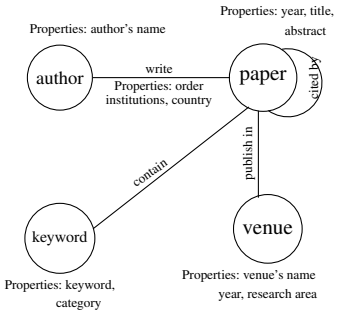

Figure 1: The conceptual graph model

among these nodes. Each node and edge are described by attributes. For instance, an edge represents the writing relationship between authors and a paper. This link is associated to order, institution and country attributes. Considering institution and country attributes, there are close to a dimension concept in traditional data warehouse. The attributes of a paper are the title, the year and the abstract. Year is an attributed dimension associated with time hierarchy. This model defines institution as an attribute on edge between author and paper to support query when authors change institutions. More details of other nodes and edges are presented in Figure 1.

Figure 2 illustrates a property graph capturing a bibliographic network. In reality, bibliographic data may have two problems. First, an entity concerns many different values in the same property. For example, author named Bin Yang works at Aalborg university and Fudan university in the same time (see Figure 2). Secondly, a property value is changing over time such as a change of institution. Look at Yzhou Sun in Figure 2, he published paper47 in 2009 when he was at university of Illinois (Urbaba-Champaign), whereas his other publications were published for Northeasten university. In order to keep this information correctly, we design institution as an edge property between author and paper. It is useful to track changes over time. We will use this bibliographic network as a running example throughout the following sections.

In this section, we extend our previous proposal with a formal definition of our model. More, we introduce a new measure in our model, graph-based measure, and we give the algorithms for computation.

3.1 Definitions

In this section, we consider an extending multidimensional structure for analyzing multidimensional networks by introducing graphs enriched by cubes.

3.1.1 Facts

In OLAP, the fact is the subject of the analysis. For example, interesting facts from our context can be the co-authorships or the production of authors. In our concept, we propose to view these facts by a network in order to face different information and to describe the interconnect among information. To analyze co-authorships, it is viewed as a network where a node is an author and an edge is the collaboration. Each co-authorship contains a cube showing the measures according to dimensions. A fact has cubes depending on what analysis is. If the considered fact is on the relationships, cubes are provided for edges. Otherwise, cubes are provided for nodes.

3.1.2 Dimensions

Dimensions provide the possible perspectives for the graph analysis. A dimension is derived from attributes of nodes and of edges. It is represented by the following definition.

Definition 1: (Dimension) A dimension D is defined by $D = (D^N, D^L, D^P)$ where,

- D^N is the dimension name.
- D^L is a set of level $L = \{l_1, l_2, ...l_n\}$ where L is a non-empty set of levels, and each l_n comes from an attribute of node or of edge that belongs to the same node or edge. Level names are unique.
- D^P is a partial pattern on the elements of D^L. It defines the pattern's topology or the constraints applied on the content of a graph. It is used to identify a graph that belongs to the dimension's level and that should be a generalized graph or a partial graph after roll up.

Dimension is usually structured into levels. We define a level of dimension as the following.

Definition 2: (Level) A level L is defined by $L = (L^N, L^A)$ where,

- L^N is the level name.
- L^A is an attribute value. It is derived from attributes of nodes or of edges.

A fact can be examined through the dimensions. Consider co-authorship for example, the dimensions are the time, the venue and the institution. Time and venue are defined to restrict on the content of graph. They are used as dimensions of cube. Institutions concern with an author. They are defined as topological dimension. The dimensions are defined with their respective levels: $\{year,\ all\}$; $\{venue's\ name, research\ area,\ all\}$; and $\{institution,\ country,\ all\}$, respectively.

3.1.3 Measures

In a multidimensional model, a measure is the basic unit of data that is placed in the multidimensional space be valued through the dimensions. A measure is identified as follows:

Definition 3: (Measure) A measure $M = (M^N, M^F, M^A)$ where,

- M^N is a measure name.
- $M^F : f(M^N) \rightarrow X$ could be a graph-specific function such the centrality algorithm or could be a function computing a numerical value.

Figure 2: A property graph for a bibliographic network

- M^A is the aggregation function.

To do analytics over graphs, multiple classification of graph measures were proposed in the literature. Here, we present a classification of graph measures, based on the type of the computation algorithm.

Numerical measures. They are extracted from the attributes of nodes or of edges. These measures are similar to the traditional measures such as the number of papers.

Graph-based measures. They could capture the properties of graphs and they are obtained by using graph algorithms. In this paper, we are interesting in the centrality of nodes within a graph. It determines the qualified status of a node e.g., how important an author is within the co-authorships network. There are many types of the centrality concept such as degree, betweenness and closeness.

- **Degree Centrality** is the simplest concept, which is defined as the number of incident links upon a node (Freeman (1998)). It is the number of nodes adjacent to a given node:

$$C_D(i) = \sum_{j}^{N} x_{ij}$$

where i is the given node, j represented all others nodes, N is the total number of nodes, and x is the adjacency matrix, in which x_{ij} is defined as 1 if the node i connected to the node j.

- **Betweenness Centrality** measures how often a given node sits between others. It relies on the identification of the shortest paths, and measures the number of them that passes through a given node. To faster computation, we use the algorithm described in Brandes (2001). This measure has been formalized as follows:

$$C_B(i) = \sum \frac{g_{jk}(i)}{g_{jk}}$$

where i is the given node, g_{jk} is the number of shortest paths between two nodes j and k, and $g_{jk}(i)$ is the number those paths go through node i.

- **Closeness Centrality** measures how many steps away from others one is in the network. It relies on the length of the paths from a node to all other nodes in the network, and it is defined as the inverse total length. Due to networks with disconnected components, Opsahl et al. (2010) rewrite the closeness equation as the sum of the inversed distances to all other nodes instead of the inversed of the sum of distances to all other nodes:

$$C_C(i) = \sum \frac{1}{d_{ij}}$$

where i is the given node, j is another node in the network, and d_{ij} is the shortest distance between these two nodes.

In the next two subsections, we give the details about the algorithm for computing graphs enriched by cubes and about the OLAP operations.

3.2 How to compute Graphs enriched by Cubes?

The graph enriched by cubes construction involves the two main algorithms: graph aggregation algorithm and cubes construction algorithm. There are different algorithms for building cubes according to measures.

3.2.1 Graph aggregation

As we said before, bibliographic data has two problems: many values in the same property and changing value over time. In order to support these two problems, we apply path in algorithm for computing the aggregated graph. To build a graph, we provide a set of paths in the preprocessing step. We give the definition of path as follows:

Definition 4: (Path) A path P is defined on the heterogeneous network, and is denoted by $V_1 \overset{E_1}{\rightarrow} V_2 \overset{E_2}{\rightarrow} \ldots \overset{E_m}{\rightarrow} V_q$. It defines a composite relation $E = E_1 \circ E_2 \ldots \circ E_m$ between nodes V_1 and V_q, where $\circ$ denotes the composition operator on edges.

To build a graph, we present an algorithm, BUILDGRAPH (Algorithm 1). It creates a graph $G' = (V',\ E')$ where $V' = \{(v_\alpha,\ P_\alpha)\}$, where $v_\alpha \in V$, $\alpha = 1,\ 2,\ \ldots,\ t$ and P_α is the set of paths of v_α and $E' = \{(e_{v_\beta - v_\gamma},\ P_{\beta-\gamma})\}$, where $v_\beta \in V$, $v_\gamma \in V$ and $P_{\beta-\gamma}$ is the set of paths of the edge $v_\beta - v_\gamma$.

In the first step, the algorithm starts with the user's requirements with a fact F, a measure M, a set of dimensions D. The user's requirements could be defined by exploiting meta-data. The meta-data is used in order to know the relationships between F, M, D, etc. The selected requirements induce the specific path. Path defines information on an heterogeneous multidimensional network $G = (V, E, A_V, A_E)$ where V is the set of vertices, E the set of edges, A_V and A_E respectively the set of attributes describing nodes and edges. Then, a set of paths P is created at line 1. To get these paths, user can filter data by limiting the attributes of nodes and of edges from paths. Subsequently, we traverse the set of paths. For each path, we add a new node v_p with its path to V', if there is no such value (line 4-5). Otherwise, we simply update a path id for the node v_p in V' (line 7). After the

loop, we compute E'. For each v_s in V', we compare the list of object's values with the adjacent v_s by using intersection operator. The considered object's values depend on meta-data. If the comparison result is not empty, we add a new edge $e_{v_s-v_r}$ with its paths to E'.

Figure 3 illustrates this algorithm. BUILDGRAPH takes as input the user's parameters. As previously, the fact can be the co-authorship, the measure is the number of papers written by two co-authors, the dimensions are the year, the venue and the keywords. In order to obtain the first graph (with authors as nodes and relationship between two authors as edges), a set of paths is generated like $author \overset{write}{\rightarrow} paper \overset{publish}{\rightarrow} venue$. In our example, there are 13 paths (Figure 3a). The next step is to compute a set of nodes. We get a list of authors with their paths (Figure 3b). Then, any two authors who wrote papers together, are added to a list of edges (Figure 3c). The number of papers on edges is computed by using intersection operators. For instance, J. Han published paper33, paper47, paper44 and etc. Y. Sun published paper44, paper10, paper47 and etc. A set of papers between them is computed by $\{paper\ 33,\ paper\ 47,\ paper\ 44, ...\} \cap \{paper\ 44,\ paper\ 10,\ paper\ 47, ...\} = \{paper44\ ,\ paper\ 47, ...\}$. Due to the need of edges cubes of co-authorship network, the output graph of co-authorship network is built by selecting a set of nodes from edges (Figure 3d). Authors who only write papers alone are not in the network. Papers written by only one author are not counted in this co-authorship network.

Algorithm 1 BUILDGRAPH

Input: An heterogeneous multidimensional network $G = (V,E,A_V,A_E)$, a fact F, a measure M, a set of dimensions D
Output: A graph $G' = (V',\ E')$ where $V' = \{(v_\alpha,\ P_\alpha)\}$ and $E' = \{(e_{v_\beta - v_\gamma},\ P_{\beta-\gamma})\}$

1: Generate a set of paths (P) according to G, F, M and D
2: $V' = \varnothing$
3: **for** each $p \in P$ **do**
4: **if** v_p not in V' **then**
5: $V' = V' + (v_p, \{p\})$
6: **else**
7: add p at node v_p in V'
8: **end if**
9: **end for**
10: $E' = \varnothing$
11: **for** each $s = 1$ to V'.size-1 **do**
12: $list_s$ = get the values of object according to P {the considered objects depend on meta-data, for instance papers for the authors}
13: **for** each $r = s+1$ to V'.size **do**
14: $list_r$ = get the values of object according to P
15: **if** $list_s \cap list_r \neq \phi$ **then**
16: $E' = E' + (e_{v_s-v_r},\ \{P_s + P_r\})$
17: **end if**
18: **end for**
19: **end for**
20: Return G'

After the creation of the graph, we present the algorithms in order to compute cubes in the following section.

3.2.2 Cube computation

There are different algorithms to build cubes according to the types of the measure. We present them in the following:

Numerical measure cubes. In algorithm, BUILDCUBESNUMBER (cf. algorithm 2), if the fact needs cubes on nodes, the algorithm scans through V'. Otherwise, it scans through E'. The measure's value is computed from each path of V' or E'.

Algorithm 2 BUILDCUBESNUMBER

Input: A graph G and G', a fact F, a measure M, a set of dimensions D
Output: A graph $G' = (V',E',C_{V'},C_{E'})$ where $C_{V'}$ and $C_{E'}$ are respectively the set of cubes enriching the nodes of V' and the edges of E'.

1: **if** F needs cubes on nodes {accroding to meta-data} **then**
2: **for** each v in V' **do**
3: Build the structure of C_v according to D
4: **for** each p in P_v **do**
5: Calculate the measure value of p for each cell of C_v
6: **end for**
7: **end for**
8: **end if**
9: **if** F needs cubes on edges {accroding to meta-data} **then**
10: **for** each e in E' **do**
11: Build the structure of C_e according to D
12: **for** each p in P_e **do**
13: Calculate the measure value of p for each cell in C_e
14: **end for**
15: **end for**
16: **end if**

Centrality measure cubes. If the measure is a graph-based measure, we need three algorithms in order to build the cubes when measures are the degree, the betweeness and the closeness. In this paper, we study this measure for nodes. In social network analysis, graph-based measures are used to understand and explain social phenomena. An essential tool for analysis of information networks is the centrality defined on the nodes of graph. Look at co-authorships network in Figure 4a, J. Han has 19 the number of edges, but we don't know the answer when we need to simple questions like what year is the best degree of J. Han? or how do the total degree from 2010 compare with the total degree from 2011?. In our proposal, the number of edges that J. Han has are provided to a cube according to dimensions in order to answer the above questions.

- Degree measure. BUILDCUBESDEGREE (cf. algorithm 3), first builds the structure of a cube for v

Figure 3: Computation of a co-authorships network

Figure 4: Roll up from the co-authorships network to the institutions network

(line 2). For each cell c of the cube, the algorithm gets a set of adjacent nodes of v from G' and they are kept into $listadd_v$. To get the number of degree for c, nodes which are not in c will be removed from $listadd_v$ (line 5-6).

- Betweenness measure. BUILDCUBESBETWEENESS (cf. algorithm 4) first starts with building the structure of a cube for v from its paths (line 2). Then, we traverse each cell c of the cube. For each c, we get a graph G'_c where $G'_c \subset G'$ and we compute the new shortest path between all pairs of nodes where a starting node and a ending node are not equal to v. Finally, the algorithm computes betweenness centrality C_B of c(line 6-8).

- Closeness measure. BUILDCUBECLOSENESS (cf. algorithm 5) first starts with building the srtucture of a cube for v from its paths (line 2). Subsequently, we travel each cell c of the cube. For each c, we get a graph G'_c where $G'_c \subset G'$ and we compute the new shortest paths from v to others. After that the algorithm computes closeness centrality C_C of c (line 6-8).

Algorithm 3 BUILDCUBESDEGREE

Input: A graph G' and G, and a set of dimensions D
Output: $C^{CD}_{V'}$ Degree centrality cubes of nodes

1: **for** each v in V' **do**
2: Build the structure of C^{CD}_v according to D
3: **for** each cell c in C^{CD}_v **do**
4: $listadd_v = \phi$
5: Get adjacent nodes of v in $listadd_v$
6: Remove nodes that are not in c from $listadd_v$
7: Put $listadd_v$.size in c
8: **end for**
9: **end for**
10: Return $C^{CD}_{V'}$

3.3 OLAP Operations

The classical OLAP, operations like Roll up, drill down, slice and dice are suppoted to explore data. We extend them to analyze graphs enriched by cubes. As we said before, two different types of operations are introduced in Graph OLAP Chen et al. (2008). The first one is an informational OLAP, and it uses informational attributes. This operation doesn't change the structure of the network. For example, venue and time are two informational attributes with their respective hierarchies $\{year, decade, all\}$ and $\{conference, area, all\}$. The second one, topological OLAP, implies a new structure of the network; if we do a topological

Algorithm 4 BUILDCUBESBETWEENESS

Input: A graph G' and G, and a set of dimensions D
Output: $C^{CB}_{V'}$ Betweenness centrality cubes of nodes

1: **for** each v in V' **do**
2: Build the structure C^{CB}_v according to D
3: **for** each cell c in C^{CB}_v **do**
4: Get a graph G'_c of c
5: Find all shortest paths SP between every pair of nodes PN in G'_c where both nodes in a pair are not equal to v
6: Betweenness centrality $C_B = 0$
7: **for** each pn in PN **do**
8: $C_B = C_B + \frac{Number\ of\ SP\ pass\ v}{Number\ of\ SP}$
9: **end for**
10: Add C_B to c
11: **end for**
12: **end for**
13: Return $C^{CB}_{V'}$

Algorithm 5 BUILDCUBESCLOSENESS

Input: A graph G' and G, and a set of dimensions D
Output:$C^{CC}_{V'}$ Closeness centrality cubes of nodes

1: **for** each v in V' **do**
2: Build the structure C^{CC}_v according to D
3: **for** each cell c in C^{CC}_v **do**
4: Get a graph G'_c of c
5: Find the shortest path SP from v to others
6: Closeness centrality $C_c = 0$
7: **for** each sp in SP **do**
8: $C_C = C_C + \frac{1}{length\ of\ sp}$
9: **end for**
10: Add C_C to c
11: **end for**
12: **end for**
13: Return $C^{CC}_{V'}$

Table 1 Comparision between the basic of graph OLAP and Graphs enriched by cubes

	Basic of graph OLAP concepts (Chen et al. (2008))	Concepts of Graphs enriched by Cubes
Main idea	A cube with graphs.	A graph with cubes.
Fact	Subject of analysis is viewed as a cube.	Subject of analysis is viewed as a graph.
Measure	Aggregated graph	Numerical measures Graph-based measures
Dimension	Informational and topological	Informational and topological
Aggregation function	Specific aggregation functions	Specific aggregation functions and support the slowly changing dimension
Informational Roll up OLAP operation	Overlay a set of graphs into a summarized graph	Perform on cubes
Topological Roll up OLAP operation	A new cube with aggregated graphs.	A new graph with smaller recalculated cubes.

Roll-Up, the network is generalized by merging some nodes. This operation uses topological attributes. In the author network, for instance, the hierarchy $\{institution, country, all\}$ associated with the node attribute author can be used for merging authors from a same institution into a generalized node.

Informational Roll up/Drill down. The Informational Roll up (IRollup) operation decreases the granularity for the specified dimension $d \in D$ of cubes by grouping measure value into the higher level (where $D_d^L = \{l_0, l_1, ..., l_n\}$ and D_d^P is defined for the constraints on the content of a graph). The informational Drill down (IDrilldown) operation increases the granularity by switching to the next lower level of the dimension. Derived granularities are defined as follows:

$$IRollup(G', d_l) := (G', d_{l_{i+1}})$$

$$IDrillDown(G', d_l) := (G', d_{l_{i-1}})$$

Topological Roll up/Drill down. The Topological Roll up (TRollup) operation generates the network at higher level. The Topological Drill Down (TDrilldown) operation generates the network at a lower level. Derived granularities are defined as follows:

$$TRollup(G', D) := (G^{rollup}, D)$$

$$TDrilldown(G', D) := (G^{drilldown}, D)$$

where G^{rollup} is higher level network of G' and $G^{drilldown}$ is a lower level network of G'

Slice. The slice operation filter the specified graph $g' \in G'$. It is defined as follows:

$$Slice(G', D) := (G'^{slice}, D)$$

where G'^{slice} is a sub graph of G'.

In graphs enriched by cubes, we can perform both informational and topological OLAP. Informational OLAP operations are classically done, so we don't give details in the present. The most difficult problem we have to solve is how to support topological OLAP operations over networks. This problem is even more difficult if we take into account the slowly changing dimension over time. A higher level of network cannot be computed from a lower level without accessing raw data. Networked data is often non-summarizable. The idea of keeping a set of paths into nodes in the previous algorithm allows us to solve this problem.
From Figure 4b shows an example of a topological roll-up of the co-authorship network to the institution network. While all authors of a same institution are merged as one node, edges are created when any two institutions published papers together. In case of many institutions of an author in the same time, the author is counted into all his institutions. After the roll-up, in the more generalized network, new cubes have to be computed. In our example, co-authorship network involves edge cubes, whereas institution networks needs both node and edge cubes. To build the institution network, we use both BUILDGRAPH and BUILCUBES algorithms. Before computing a set of nodes (line 2 in algorithm 1), we need to filter paths instead of generating a set of paths (line 1 in algorithm 1). We have to filter paths because all nodes of data set are collected in V', but some nodes may not be in co-authorship network (because some papers are written by only one author). The filter paths step is called when the previous network needs edge cubes. Then we compute a new set of nodes from line 2 in algorithm 1. Refer to example in figure 4b, nodes are grouped into institutions. For example, university of Illinois contains path6 and path7 because J. Han and P.S. Yu belong to this university.

Slice. Traditional slice operation selects one particular dimension from a given cube and provides a new sub-cube. In our context, slice operation can not be like the classical one. It should be adapted to graphs. The slice operation selects a part of the graph and provides a new sub-graph. For example, if a whole co-authorship network is too big to be comprehensive, the user can focus on a smaller subgraph more interesting to analyze information clearly.

3.4 Comparison between the basic of graph OLAP and Graphs enriched by cubes

In the Graph OLAP literature, Chen et al. (2008) is the principle concept of Graph OLAP. In this section, we propose a comparison between Chen et al. (2008)'s context and Graphs enriched by cubes in Table 1.

Chen et al. (2008) presented a graph to a cell. Building on that, a cube contains a set of graphs. On the contrary, graphs enriched by cubes presents a subject of analysis as a graph. Each node or edge is weighted by cube. Both these concepts support informational and topological dimension. There are the specific aggregation functions. However, graphs enriched by cubes supports slowly changing dimension. There are different ways for roll up on these dimensions. When a roll up is made on an informational dimension in Chen et al. (2008), a set of networks is explored to a summarized graph. In our proposal, a roll up is provided on cubes. It has en effect on the structure of graph. In contrast, a roll up on a topological dimension reorganizes the individual network for a more generlized view for Chen et al. (2008). Graphs enriched by cubes can perform this operation on a graph but not in the individual network.

4 Experiments

In this section, we first introduce the data set we used in our experiments. Then, we present how we have implemented our solution and we give some examples of analysis. Next, we do a comparison between our algorithm to build an aggregated graph with another algorithm and we speak about performances of all algorithms.

4.1 Data Sets

We get data from three bibliographic databases. First we use the well known database DBLP. But in order to complete data, we access on ACM and Microsoft Research databases for taking keywords, institutions and research areas. In theses three sources, we keep only three research areas (data mining, databases and information retrieval) and we pick only a few representative conferences for the three areas (PODS, EDBT, KDD, DOLAP, ASONAM, SIGIR and CIKM). At the end, we build a data set which contains 4,727 papers and 8,238 authors since 2009.

4.2 Implementation

The implementation has been done as follows:

- The ETL process is used to fulfill data into the model. The first step of the data pre-processing is data cleaning and validation. The way how data is treated depends on its type. Data is cleaned then validated in order to check its integrity. For example, if an institution's name is invalid, it is changed during the cleaning step. After cleaning, data is loaded into an unified structure with a graph model. Due to the association with the shortest paths of betweenness and closeness centrality, we prepare the possible paths for each pair of nodes in database.

- A new type of NoSQL databases, called graph databases, is used to implement our conceptual graph model. We choose Neo4j as a graph database because it is an open-source software, it supports the properties of our graph model and it provides a framework for graphs with massive scalability.

- Finally, an OLAP interface analysis is developed in Java and tested on a Mac OS X version 10.9.2 with Intel core i5 2.4 GHz and 8 GB of Ram. For graph visualization, we use GraphStream library because it is a library to model and analyze the dynamic graphs and it is an open source library. Although, Graphstream provides the algorithms for network centrality. However, its algorithms don't support a disconnected network. In our case, a network is composed of sub-networks.

4.3 Example of analysis

We first use the example of the co-authorships network introduced before. Figure 5 shows the co-authorships network in three areas since 2009. Each edge of this network has a cube. In order to reduce the graph, we filter edges in order to keep only edges with a number of papers over than 10.

For example, look at the edge between Iadh Ounis and Craig Macdonald; these authors published 29 papers together. We consider these papers through a cube with two dimensions. It could be interesting to have two ways of visualization. The first way is to focus on time, having the count of papers per year. Per each year, we get the count of papers by venues. The second way is to focus on the venue, having the count of papers per venue's name. For each venue, we get the count of papers by year. This cube shows that they wrote 9 papers together in 2013: three papers published in SIGIR and six papers published in CIKM.

Then, we do a topological roll up on this co-authorships network. Its next higher level is institutions network, we obtain the result as shown in figure 6. The institutions are filtered with a number of papers over than 10. For example, Microsoft Research Asia published 132 papers in three areas from 2009 to 2013. Look at the big number 1 in the figure, it means that Microsoft Research Asia has one collaboration with MDE-MS Key Lab of MCC in 2010 by publishing in the SIGIR conference. The big number 2 shows the number of papers written by several authors but all belonging to the same institution (Microsoft Research Asia).

Figure 7 shows a list of cubes in co-authorships network in three areas since 2009 when degree is defined as a measure. Jiawei Han has the highest degree. He appears relatively central. On the contrary, he is not the most central when measures are betweenness and closeness (see figure 8 and 9). The right part of figures 7, 8 and 9 show cubes of Jiawei Han in each year while measures are degree, betweenness and closeness respectively.

Figure 5: The co-authorships network (on three areas and all years) with a number of papers over than 10

Figure 6: The institution network (on three areas and all years) with a number of papers over than 10

Figure 7: Example of a Cube for the co-authorships network (on three areas and all years) when a measure is degree centrality

Table 2 Four Data Sets

Datasets	Number of Publications	Number of authors	Number of paths
D1	1,000	2,215	3,267
D2	2,000	3,794	6,476
D3	3,000	5,337	9,972
D4	4,000	7,040	13,369

Figures 8 and 9 show a list of cubes of co-authorships network in KDD conference when betweenness and closeness are selected as measures, respectively. For example, Jimeng Sun has the highest betweeness. He is an author that acts as a bridge to others through the shortest path in KDD conference (see figure 8). However, he does not have the highest closeness (see figure 9). Christos Faloutsos has the highest closeness. This is that he is close to other authors with the shortest paths in KDD conference more than others.

Furthermore, in the interface, the user can slice to consider only a sub-graph. There are several groups of authors in co-authorships network. Suppose that the user needs to consider only the interest group; with a slice operation, the user can select the sub co-authorship network. Finally, a roll-up operation is done on this sub-graph.

4.4 Performances

First, we compare our algorithm for building aggregated graphs with that of Beheshti (2012) because it is the most similar one. Their algorithm starts by scanning all paths to compute nodes. As a result, each node will be stored with its measures. Next, to compute edges, the algorithm first groups nodes according to their measure values. Each measure value contains its name and a set of nodes that associated with it. After that the algorithm travels each measure value to access a set of nodes. An edge is built by grouping any two nodes.

Regarding the complexity for building aggregated graphs by our approach and by Beheshti's approach, it can be split into two steps: the computation of nodes and the computation of edges. According Table2, the complexity of nodes computation for both approaches is $O(|P|)$ because both approaches have to scan all paths to get the different nodes. On the contrary, there is a difference for the edge computation. Our approach uses $O(|V'_f|^2)$, where V'_f is the number of generalized nodes. Whereas Beheshti approach uses $O(|P| + (|V_M| * |v_f|^2))$, where P is the number of paths, V_M is the number of measure values and v'_f is the number of generalized nodes in each measure value.

Figure 10: Running time of edge computation

Figure 11: Running time of cube computation with different measures

We also experiment the running time for the edge computation with two queries: the co-authorship network with the number of papers and the venue network with the number of authors. Two sets of paths are generated like $author - write - paper$ and $author - write - paper - publish - venue$ for query 1 and query 2, respectively. To better see the time complexity of the edge computation, we divide the data set into four data sets as shown in Table 2. Figure 10 compares the running time of query 1 and query 2 in four data sets and for both approaches. It shows that our approach has a better performance even if the number of nodes increases. Our approach uses less time and it scales linearly with respect to the number of nodes (V'_f).

Then, we study the performance of algorithms for creating cubes according to measures. Figure 11 shows the running time for the cubes creation according to the number of cubes. Two of them need to compute a classical measure and degree centrality. On the contrary, betweeness centrality and closeness centrality take much time if the number of cubes increases because they rely on the shortest paths.

graphystyleapalike

5 Conclusion

In this paper, we wanted to enhance decision support on networks by OLAP analysis on networked data. Therefore, we presented definitions for graphs enriched by cubes by mapping the concepts of fact, dimension and measure from the multidimensional model. The graphs enriched by cubes approach performs multidimensional views of an heterogeneous network rather than a set of graphs. Cubes are provided for nodes or edges according to the

Figure 8: Example of a Cube for the co-authorships network in KDD conference when a measure is betweenness centrality

Figure 9: Example of a Cube for the co-authorships network in KDD conference when a measure is closeness centrality

user's requirements (fact, measure, dimensions, etc.) We propose algorithms which, in addition, solve the slowly changing dimension problem in OLAP analysis in order to compute graphs enriched by cubes. Then we adapt the OLAP operations to graphs enriched by cubes. We show an implementation of our proposal and experiments with real data sets from three bibliographic databases.

In future, we would like to use another type of information networks in order to show that our approach can run with all heterogeneous information networks. Secondly, it could be interesting to consider text mining tools in order to enrich the model and the network by more attributes. Text mining tools can be useful for information extraction. So we will combine Graph OLAP and Text OLAP in order to handle all networked data.

References

Chen, C., Yan, X., Zhu, F., Han, J., and Yu, P. S. (2008), 'Graph OLAP: Towards online analytical processing on graphs' in *ICDM 2008: IEEE International Conference on Data Mining*, Pisa, Italy, pp. 103-112.

Brandes, M. (2001), 'A faster algorithm for betweenness centrality', *Scientometrics*, Vol.25 No.2, pp. 163-177.

Beheshti, S.M.R., Benatallah, B. and Motahari-Nezhad, H.R., 'A Framework and a Language for On-Line Analytical Processing on Graphs' in *WISE 2012: International Conference on Web Information Systems Engineering*, Paphos, Cyprus, pp.213–227.

Freeman, L.C. (1978), 'Centrality in social networks: Conceptual clarification', *Social Networks*, pp. 215-239.

GraphStream A Dynamic Graph Library. [online] http://graphstream-project.org/ (Accessed 30 March 2015).

Jakawat, W., Favre, C., and Loudcher, S. (2016), 'OLAP cube-based graph approach for bibliographic data' in *SOFSEM 2016: 42nd International Conference on Current Trends in Theory and Practice of Computer Science*, Harrachov, Czech Republic.

Kaya, M. and Alhajj, R. (2014), 'Development of multidimensional academic information networks with a novel data cube based modeling methord', *Information Sciences*, pp. 211-224.

Loudcher, S., Jakawat, W., Morales, E. P. S., and Favre, C. (2015), 'Combining olap and information networks for bibliographic data analysis: a survey', *Scientometrics*, Vol.103 No.2, pp. 471-487.

Neo4j Graph Database. [online] http://neo4j.com/ (Accessed 30 June 2014).

Opsahl, T., Agneessens, F., and Skvoretz, J. (2010), 'Node centrality in weighted networks: Generalizing degree and shortest paths', *Social Networks*, Vol 32, pp. 245-251.

Qu, Q., Zhu, F., Yan, X., Han, J., Yu, P. S., and Li, H. (2011), 'Efficient topological olap on information networks' in *DASFAA 2011: 16th International Conference on Database systems for advanced applications*, Vol. 1, Hong Kong, pp. 389-403.

Tian, Y., Hankins, R. A., and Patel, L. M. (2008), 'Efficient aggregation for graph summarization' in *SIGMOD 2008: ACM SIGMOD International Conference on Management of Data*, Vancouver, Canada, pp. 567-580.

Wang, Z., Fan, Q., Wang, H., K. L. Tan, K., Agrawal, D., and Abbadi, A. E. (2014), 'Pagrol: Parallel graph olap over large-scale attributed graphs' in *ICDE 2014: IEEE 30th International Conference on Data Engineering*, Chicago, IL, USA, pp. 496-507.

Waqas, A., Zimányi, E., and Wrembel, R. (2015), 'Temporal data warehouses: Logical models and querying' in *EDA 2015: Journées francophones sur les Entrepôts de Données et l'Analyse en Ligne*, Vol. RNTI-B-11, pp. 33-48.

Zhang, J., Hong, X., Peng, Z., and Li, Q. (2012), 'Nestedcube: Towards online analytical processing on information-enhanced multidimensional network' in *WAIM 2012: Web-Age Informational Management International Workshops*, Harba, China, pp. 128-139.

Zhao, P., Li, X., Xin, D., and Han, J. (2011), 'Graph cube: On warehousing and olap multidimensional networks' in *SIGMOD 2011: ACM SIGMOD International Conference on Management of Data*, Athens, Greece, pp. 853-864.