HMGraph OLAP: a Novel Framework for Multi-dimensional Heterogeneous Network Analysis

Mu Yin
School of Computer Science
Beijing University of Posts and Telecommunications
Beijing 100876, China
ym0513@126.com

Bin Wu
School of Computer Science
Beijing University of Posts and Telecommunications
Beijing 100876, China
wubin@bupt.edu.cn

Zengfeng Zeng
School of Computer Science
Beijing University of Posts and Telecommunications
Beijing 100876, China
gdfeng@126.com

ABSTRACT

As information continues to grow at an explosive rate, more and more heterogeneous network data sources are coming into being. While OLAP (On-Line Analytical Processing) techniques have been proven effective for analyzing and mining structured data, unfortunately, to our best knowledge, there are no OLAP tools available that are able to analyze multi-dimensional heterogeneous networks from different perspectives and with multiple granularities. Therefore, we have developed a novel HMGraph OLAP (Heterogeneous and Multi-dimensional Graph OLAP) framework for the purpose of providing more dimensions and operations to mine multi-dimensional heterogeneous information network. After information dimensions and topological dimensions, we have been the first to propose entity dimensions, which represent an important dimension for heterogeneous network analysis. On the basis of this notion, we designed HMGraph OLAP operations named *Rotate* and *Stretch* for entity dimensions, which are able to mine relationships between different entities. We then proposed the HMGraph Cube, which is an efficient data warehousing model for HMGraph OLAP. In addition, through comparison with common strategies, we have shown that the optimizations we have proposed deliver better performance. Finally, we have implemented a HMGraph OLAP prototype, LiterMiner, which has proven effective for the analysis of multi-dimensional heterogeneous networks.

Categories and Subject Descriptors

E.1 [**Data Structures**]: Graphs and networks; H.2.7 [**Database Administration**]: Data warehouse and repository

General Terms

Algorithms, Management, Performance

DOLAP'12, November 2, 2012, Maui, Hawaii, USA.

Keywords

Data warehouse, Graph OLAP, Graph Cube, Multi-dimensional heterogeneous network

1. INTRODUCTION

In this paper, we explore design and management problems pertaining to multi-dimensional heterogeneous network analysis from the novel perspective of entities dimensions. On this basis, we propose a series of optimizations to improve the performance of the prototype.

1.1 Traditional OLAP and Graph OLAP

The concept of online analytical processing (OLAP) was first proposed by E.F.Codd, the father of the relational data-base. In computing, OLAP presents us with a means of swiftly answering multi-dimensional analytical queries[3]. OLAP has succeeded online transaction processing (OLTP) as a popular research topic in the data processing and database field. OLAP is part of the broader category of business intelligence, which also encompasses relational reporting and data mining. OLAP tools enable users to interactively analyze multi-dimensional data from multiple perspectives. OLAP consists of three basic analytical operations: roll-up, drill-down, and slicing and dicing. Consolidation (Roll-up) involves the aggregation of data that can be accumulated and computed in one or more dimensions. In contrast, drill-down is a technique that allows users to navigate through the details. Slicing and dicing is a feature whereby users can take out a specific set of data from the cube and view the slices from different viewpoints. Users can navigate through different dimensions and multiple hierarchies via the above operations. The core of an OLAP system is an OLAP cube[9]. It consists of numeric facts called measures, which are categorized by dimensions. The cube meta data is typically created from a star schema or snowflake schema of tables in a relational database. Measures are derived from the records in the fact table and dimensions are derived from the dimension tables.

However, in recent years, more and more graph data have come into being, such as bibliographic network or social network. Unfortunately, traditional OLAP cannot satisfy analysis on graph, because they do not consider the relationships between attributes. For this reason, some researchers have developed Graph OLAP frameworks [4] [7], which present multi-dimensional and multi-level views over graphs. The emphasis of Graph OLAP tends to be on the analysis of homogeneous networks, with less consideration being given to

heterogeneous networks. Nevertheless, networked data often contains different types of nodes and complex relationships in the real world. The exploration of massive amounts of heterogeneous network information may disclose a great deal of valuable, in-depth information about relationships. It is generally known that each individual play different role in different social network and the relationships between individuals change constantly. For example, in campaigns, Hilary was a ferocious opponent to Obama, but now she is a crucial member for the Obama administration. A user may be interested in finding social network like that, for this reason, we should propose measures to analyze heterogeneous networks and get meaningful aggregate networks from them. On the other hand, the proposed Graph OLAP frameworks only look into two different dimensions and they are not well-equipped to handle heterogeneous networks. We will explain why topological dimension and information dimension are more appropriate for homogeneous network analysis in Section 2. Furthermore, not all networks are applicable to roll-up or drill-down operation. For instance, author may belong to more than two affiliations in co-author network and we don't know how to observe them at high level of granularity. These problems have motivated us to explore the topic of multi-dimensional heterogeneous network analysis and develop the LiterMiner prototype.

1.2 Contributions

Our major contributions are summarized as follows :

- We define the concept of *Entity Dimensions*, which is a good complement for multi-dimensional heterogeneous network, because topological dimensions and information dimensions are less involved with entities and relations.
- Based on the notion of entity dimensions, we have designed novel operations named *Rotate* and *Stretch*. *Rotate* operations can convert between entities and relations, *Stretch* operations can mine implicit relationships between different entities.
- We have proposed the HMGraph OLAP data warehousing model, which can perform multi-dimensional and multi-granularity analysis. In particular, it presents a flexible choice for the analysis of different topics. Furthermore, we have proposed the HMGraph Cube, enabling the OLAP tool to be efficient in the aggregation of heterogeneous networks with multidimensional attributes.
- We have made use of a novel partial materialization strategy to implement the HMGraph Cube. Compared with common strategies, we have demonstrated that the strategy we have proposed delivers better performance. We then proposed the bitmap index, which is proper for HMGraph OLAP.

The rest of this paper is set as follows: Section 2 introduces framework and concepts of HMGraph OLAP; Section 3 presents the warehouse and graph cube for HMGraph OLAP; Section 4 proposes a novel partial materialization strategy and bitmap index measure; Section 5 introduces experiments and section 6 presents demonstration; Section 7 describes related work; Section 8 presents the final conclusion of this paper and outlines future work.

2. HMGRAPH OLAP FRAMEWORK

Social networks are often heterogeneous networks involving multiple typed objects and multiple typed links denoting different relations. Heterogeneous information networks are not only more aligned with the real world, but also contain a wealth of information, and therefore have the potential to provide us with more accurate and implicit knowledge. To be more accurate and vivid in our description of the model of HMGraph OLAP and the related concepts, we will give a description of bibliographic information in this paper as a background. In addition to having multi-dimensional characteristics, bibliographic information also contains a wealth of information networks, such as the co-author network, the citation network, and HMGraph OLAP technology scenarios. Bibliographic information includes titles, authors, affiliations, keywords, publication data and other entities, which are connected to form a multi-dimensional heterogeneous network. Despite research attention on Graph OLAP, and efficient topological algorithm design, a much more fundamental issue concerning the design of the heterogeneous organization infrastructure has not been addressed. Therefore, we present the general framework of HMGraph OLAP.

Definition 1 (Multi-dimensional network) : A multi-dimensional network is defined as an undirected network $N = (V, E, V_A, V_R)$ where node type set $V = \{V_1, V_2, \cdots, V_n\}$ and edge type set $E = \{E_{ij} | E_{ij} = V_i \times V_j, 1 \leq i, j \leq n\}$. V_i represents a specific type of nodes and E_{ij} denotes a type of connections between two nodes. Note that, if $n > 1$, network N is a heterogeneous network. Set $V_A = \{V_{A1}, V_{A2}, \cdots, V_{An}\}$ is a series of attributes of node V_i. $V_{Ei} = \{V_{Ei1}, V_{Ei2}, \cdots, V_{Eik}\}$ is a value set and V_{Eik} denoted as value of V_{Ei}. The relation set $V_R = \{V_{Rij}\}$ is dimension set of V_i and V_{Rij} denotes dimension type.

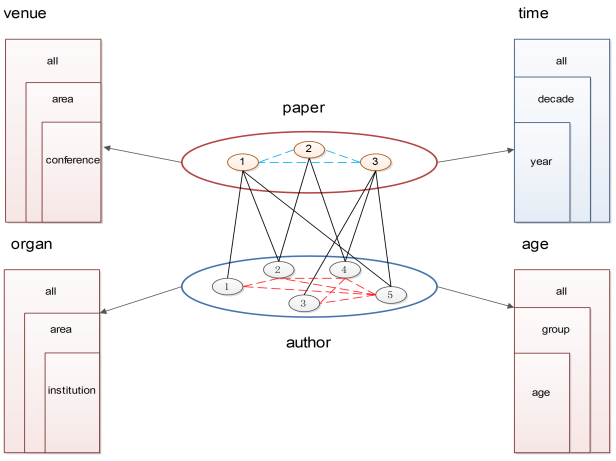


Figure 1: Bibliographic Networks

For instance, a toy bibliographic heterogeneous network is shown in Figure 1, the sample network consists of several objects; including *Paper*, *Author*, *Organ*, *Time*, *Venue* and *Age*. Obviously, this graph has two types of nodes: red node denotes paper and gray node denotes author. There are three types of edges: solid line represents relation between authors and papers which are written by them, whereas the red dashed line denotes co-author relationships and the blue one links papers written by the same person. Each entity of the network contains a set of multi-dimensional attributes

describing them. For example, we have organ and age attribute for the author node, and we used them for describing the characteristics of authors. As a multi-dimensional network, node v can be an individual author, associated with attributes: *Author Name*, *Affiliation* and *Number of Papers Published*, node v can be a director, with attributes: *Film* and *Award* as well. Evidently, edges between two nodes have different meanings. In the co-author network, the relationships can be summarized as writing and written-by, which are always co-occurrence relationships in some other scenarios.

Figure 1 shows important concept of dimension as well, which is used to construct a cuboid lattice and enable user to analyze network at multiple granularity. The dimensions of our bibliographic network include:

- *Venue: conference* $\rightarrow$ *area* $\rightarrow$ *all*,
- *Time: year* $\rightarrow$ *decade* $\rightarrow$ *all*,
- *Organ: institution* $\rightarrow$ *area* $\rightarrow$ *all*,
- *Age: age* $\rightarrow$ *age group* $\rightarrow$ *all*;

Actually, there is more than one type of dimension in Figure 1. First, let us look at their definition.

Definition 2 (Information Dimensions) : .The set of information attributes $ID = \{I_1, I_2, \cdots, I_n\}$ are called information dimensions of HMGraph OLAP (abbr. I-OLAP).

Definition 3 (Topological Dimensions) : .The set of dimensions attributes of topological element $TD = \{T_1, T_2, \cdots, T_n\}$ are called topological dimensions of HMGraph O-LAP (abbr. T-OLAP).

Topological dimensions and information dimensions in the graph OLAP scenario are proposed by [4]. Every year or every conference, we may have a new bibliographic network describing the collaboration patterns among researchers, each of them can be viewed as a snapshot of the overall bibliographic network in a bigger context. The snapshot presents specific view and granularity over networks. Actually, the role of the two dimensions is to organize snapshots into groups based on different perspectives and granularity. Information dimensions control what snapshots are to be viewed, but do not touch the inside of any single snapshot. Typically, time is one of information dimensions we find in data warehouses allowing comparisons of different periods. Topological dimensions can be used to group one kind of nodes from the same institution into a generalized node, and the new graph that results will depict interactions among these groups as a whole, which summarizes the original network and hides specific details.

However, these two dimensions are used to analyze homogeneous networks, and tend to be too simplistic for the analysis of heterogeneous networks. Let us examine an example. There are some students (S) and professors (P) working in a specific field. For any two of them, if they coauthor one paper, then we add a link between them. Obviously, this coauthor network has three different types of links, including: S-S, S-P, and P-P. Nevertheless, traditional graph OLAP tools can't distinguish their different, and they just sum up the respective weights of link. A question then raise: "Can we observe the co-operation of the students between two schools?" This knowledge involves mining information from heterogeneous networks. To give a positive answer to this problem, we propose the concept of entity dimensions and employ relation meta path.

Definition 4 (Entity Dimensions) : The set of entity attributes $ED = \{ED_1, ED_2, \cdots, ED_n\}$ are called entity dimensions of HMGraph OLAP. As showed in Figure 2 and Figure 3, the novel operations of entity dimension are summarized as follows.

- **Rotate:** Given an information network $N = (V, E)$, where node type set $V = \{V_1, V_2, \cdots, V_n\}$ and edge type set $E = \{E_{ij} | E_{ij} = V_i \times V_j, 1 \leq i, j \leq n\}$. For any type of node, there is a node set $V_i = \{v_1, v_2, \cdots, v_n\}$. In addition, for any type of edge, there is also an edge set $E_{ij} = \{e_{ij1}, e_{ij2}, \cdots, e_{ijn}\}$. In order to make this definition clearly, let us first examine homogeneous network, for any edge e_{ij}, we change it to node v_{ij}. Correspondingly, if node v_i has more than two edges, then we change it to a link between its two edges. Assume that v_i has two edges, e_{ij} and e_{ik}, then change node v_i to edge e_{jik}. As shown in Figure 2. The graph describes collaboration among individual authors. Nodes present person working in a specific field. If two persons coauthor one paper, then add a link between them. After rotate operation, the graph changes to a paper network where node is paper. If paper is written by the same person, then add a link between two papers.

- **Stretch:** Given an information network $N = (V', E')$, where node set $V'_i = \{v_1, v_2, \cdots, v_n\}$ and edge set $E' = \{e_{ij} | e_{ij} = v_i \times v_j, 1 \leq i, j \leq n\}$. Then we change relationships to entities. For each edge e_{ij}, we change them to v_{ij} and add links between the new generated nodes and their original endpoint. Note that stretch operation is only proper for homogeneous networks. As shown in Figure 3, the co-author network is changed to a heterogeneous network with two different kinds of nodes and the relationship is changed to writing and written-by.

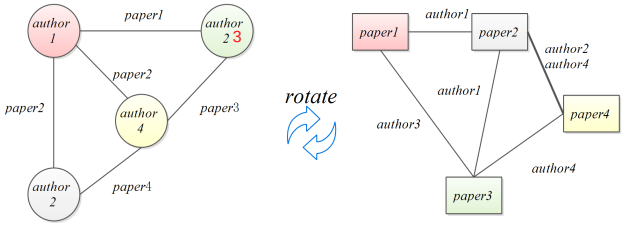


Figure 2: Rotate Operation for HMGraph OLAP

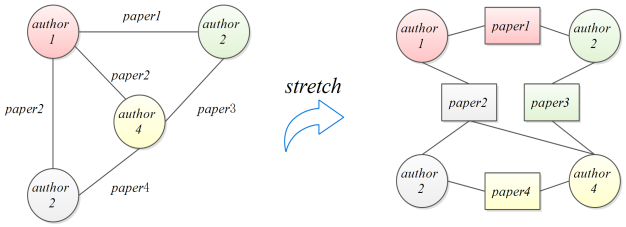


Figure 3: Stretch Operation for HMGraph OLAP

Traditional graph OLAP operation view graph through different concept hierarchies, whereas operations of entity dimension lay emphasis on mining implicit knowledge of entities. For instance, with regard to IMDB [1] data set, user could be interested in question like that "what is the network structure about cooperation of a group of actors?" Through entity dimension operations, we will get abundant information, including: an opposite play, similar type of film, cooperation with same director, rather than simple weights of cooperation. Furthermore, user may discover competitive relation between actors or find similarity of them. It can be seen that entity dimension operations enriched our analysis method greatly.

The role of entity dimensions is to make a distinction between homogeneous networks and heterogeneous networks. On the one hand, entity dimensions are provided to convert between entities and relations and mine implicit relationships. On the other hand, the characteristics of different types of networks can be analyzed to achieve the purpose of mining information by selecting different entities. Different from the operations on topological dimensions which can lead to generate a new kind of node, entity dimensions operations always change the meaning of edge and number of nodes in the network.

With regard to the above question: "How can we get the subgraph accurately if we need to view the cooperation relationship of authors or find the implicit advisor-advisee relationships between specific authors?" Evidently, T-OLAP and I-OLAP operations cannot handle these demands, because they are only appropriate for organize homogeneous networks on different perspectives and granularity. This motivates us to propose the concept of relation meta path.

Definition 5 (Relation Meta Path): A meta relation path P defined on the graph model $N = \{V, E\}$, V is the set of entities and E is the set of relations. P is denoted in the form of $V_1 \overset{E_i}{\rightarrow} V_2 \overset{E_j}{\rightarrow} \cdots \overset{E_m}{\rightarrow} V_n$,which defines a composite relations between V_1 and V_n. For instance, relation meta path $P(author, author) = author \overset{writing}{\rightarrow} paper \overset{written-by}{\rightarrow} author$ in coauthor network, and relation meta path $P(paper, paer) = paper \overset{cite}{\rightarrow} paper \overset{cited-by}{\rightarrow} paper$ in citation network.

A relation meta path is a path consisting of a sequence of relations defined between different types of entities[13][14]. In heterogeneous networks, two objects can be connected via different relations. For example, two authors can be connected via the "author-paper-author" path, the "author-paper-keyword-paper-author" path, and so on. Formally, these paths are called relation meta paths. The length of a relation meta path is the number of relations in a relation meta path. Given a relation meta path, we can extract a unique network from a heterogeneous network by performing entity dimensions operations.

3. HMGRAPH OLAP MODELS

In this section, we propose a useful approach to model multi-dimensional heterogeneous network and how to construct our graph cube.

Data warehouses are a primary means for a consolidated view on the data and frequently a first step in integrating decision support systems. Above all, data warehouses are used for analyzing data online, giving the possibility to aggregate and compare data along dimensions. In order to accurately reflect network understandable and easily extendable data warehouse schema, we pay attention to the star structure, which is dominant in data warehouses. Based on a bibliographic network warehouse environment, we discuss the involved concepts.

[1]http://www.imdb.com

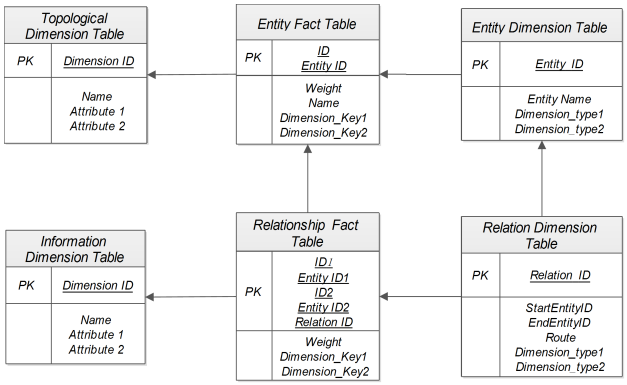


Figure 4: Heterogeneous Network Schema

3.1 Warehouse Model

The heterogeneous network warehouse model is different from the traditional OLAP model in that the main objective is to organize dimensions and graph operations of heterogeneous networks in a way that makes it easy to perform multi-dimensional and multi-level analysis. Furthermore, another aim is to reduce redundant storage and achieve better maintainability. As shown in Figure 4, we have proposed a heterogeneous network schema. To extend the model to fit heterogeneous network analysis and entity dimension operations, we have proposed two fact tables and four dimension tables. Entity fact tables include all entity types and the primary key is Entity ID. Entity type ID is a foreign key to link entity dimension tables. Here, entity dimension tables describe all of the dimension types of entities in the graph. Relation fact tables include all relation types and the primary key is Entity ID and Entity type ID. Relation ID links to relation dimension table to describe all of the dimension types of relations. With this warehouse model, algorithms and operations are easy to implement and multi-dimensional and multi-granularity analysis can be performed. In particular, this warehouse model is flexible and can be used for the analysis of different topics.

3.2 HMGraph Cube

Traditional OLAP on relational databases is based on the hypercube lattice structure[6][2]. Each node of the lattice structure represents a possible view. Each node is labeled with the set of dimensions in the "group by" list for that view[1]. To supports OLAP queries effectively on large multi-dimensional networks, [17] introduce Graph Cube, a new data warehousing model. Graph Cube focuses on OLAP inside a single large graph. Furthermore, a set of aggregated networks with varying size and resolution is examined in the lens of multi-dimensional analysis. However, Graph Cube is only feasible for homogeneous network analysis and is less involved with how to index cuboid.

With regard to the graph model presented in Definition 1, HMGraph Cube includes all aggregate graphs of the

heterogeneous network at specific dimensions. Each aggregate graph is called a cuboid. Unlike traditional data cubes, each vertex represents a unique aggregation graph in a graph cube. Given a multi-dimensional network N with n dimensions,there are 2^n cuboids in the graph cube. Supposing that there are two levels in a graph, Article and keyword, as shown in Figure 5, then we will have a graph cube lattice, in which each node is a cuboid in the HMGraph Cube generated from multi-dimensional heterogeneous network. The edges in the lattice depict the parent-child relationship between two cuboids. Every aggregate graph of a network can be examined and analyzed in this way.

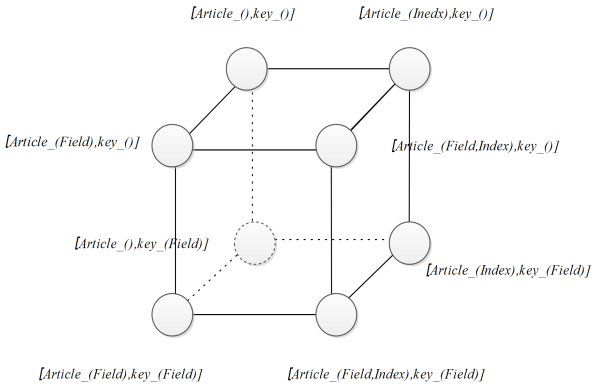


Figure 5: HMGraph Cube

4. OPTIMIZATIONS

In this section, we present an approach to index graph and we propose a novel materialization strategy. The most crucial issue with the optimization is a cost reduction of the warehousing processes to minimum. There have been several algorithms invented for cube implementation in the context of relational data[6][11], whereas these algorithms unable to get good performance in network data context. To perform efficient OLAP analysis on information networks, we discuss the optimizations about HMGraph OLAP.

4.1 Bitmap Indexing

Traditional studies on Graph OLAP do not mention how to index graphs. However, indexing is a way to improve the efficiency of operations and queries. Therefore,we propose a bitmap index that is suited to HMGraph OLAP. Firstly, we built a priority binary tree which depicts the entity types and dimension types of the graph, as is shown in Figure 6. There are three characteristics in the priority binary tree. (1) the root node is entity type, (2) every left child node is a dimension type node and every right node is an entity type node, (3) For the dimension node, the nearest ancestor node represents its entity type.

According to the above conditions, we can ascertain that there are 6 entity types and 12 dimension types in the binary tree. Note that the node ID represents the bit of a complete binary tree, so we should empty the bit position even if there is no node in the current bit. The result is that entity type ID is odd and entity dimension type id is even. Secondly, we should pre-order traversal on the priority binary tree, the sequence obtained as a global priority sequence, labeled node ID, as shown in table 1. Then, we can generate the bitmap index according to priority binary tree and global priority sequence, and unsigned binary integer should be used for the bitmap index of cuboid.

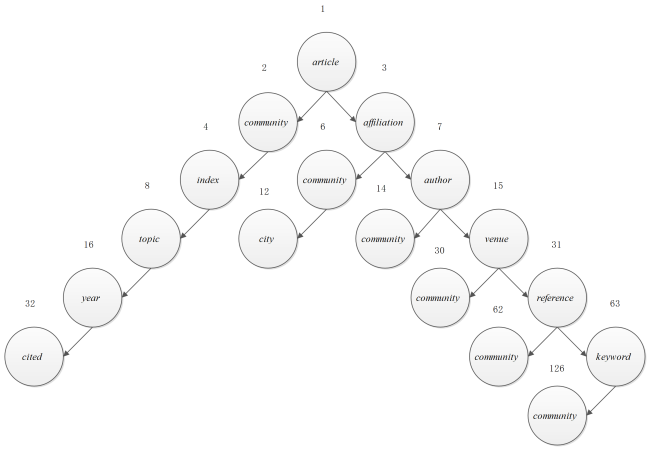


Figure 6: Priority Binary Tree

For instance, Figure 7 represents an author-keyword operation on topological dimension. Every topological dimension index can be divided into n parts and n is the number of entity types of the current network G. The index is divided into two parts in Figure 7, which describe article and keyword entity respectively. Each part can be divided into two sections, the first part is representative of the entity type of operations and the second part is representative of the dimension which the aggregate functions work on. Finally, we select the reversed order of this index in order to ensure that the highest level is 1 and the length of the index is determined. In addition, the reversed order makes the subsequent decoding more convenient.

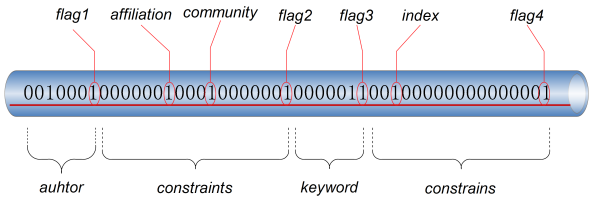


Figure 7: bitmap index for author-keyword operation

4.2 Materialization strategy

Materialization of a traditional data cube is a way to compute and store multi-dimensional aggregates so that multi-dimensional analysis can be performed efficiently[10]. In order to implement a graph cube, we need to compute the aggregate networks of different cuboids of multi-dimensional network. [17] adopt a greedy algorithm for partial materialization on the graph cube. However, the time complexity of the greedy algorithm is high when network N is a high-dimensional heterogeneous network. So we propose a novel materialization strategy.

Table 1: Entity Dimension Table

Entity Type	Dimension Type	Entity ID	priority
article	community,index,topic,year,cited	1	1
affiliation	community,city	3	2
author	community	7	3
conferences/journals	community	15	4
reference	community	31	5
keyword	community	63	6

The response time is our largest concern when performing queries and other operations. However, due to space limitations, we cannot physically materialize the whole cube. Therefore,we have proposed a novel strategy for partial materialization, which can be summarized as follows: (1) We materialize the current entity dimension network. (2) If the current network has child cuboids, we materialize them. (3) We also materialize the sub-graph of the current network. Experiments show that this approach delivers better response performance than a non-materialization strategy as well as acceptable memory consumption.

5. EXPERIMENTS

5.1 Data Sets

In this section, we present the major experimental results of our proposed optimization strategy and operations. We performed experiments on real world data sets taken from the Science Citation Index[2], which is regarded as the most authoritative academic search platform and our evaluation is conducted from both effectiveness and efficiency perspectives. We extracted major entities from the data set, including: *Title*, *Author*, *Year*, *Affiliation*, *Publication* and *Keyword*. The details of data sets are shown in Table 2. With these entities, we can automatically construct a heterogeneous network.

Table 2: DataSets

Data Set	Node	Edge
1	44	177
2	209	1631
3	540	7875
4	1049	34422
5	4999	119667

5.2 Efficiency Comparison

We selected a partial-materialization strategy, and saved the generated sub-cuboids during the operation process. In our experiment, we took the Co-author Network as our scenario.Selecting the topological dimension, we drilled down from the top network on a layer-by-layer basis. There were four layers in this operation in total.

Figure (a) shows the running time comparison for the two strategies as the number of vertices of the original network increases. The partial materialization problem in the graph cube scenario is still NP-complete [17], so we also concern memory consumption as shown in Figure (b). Experiments show that partial materialization can achieve better response performance than no materialization strategy and memory consumption is relatively high. If we select full materialize strategy, we will achieve the best response time performance. However, pre-computing and storing every aggregate network is not feasible for large multi-dimensional networks analysis, in that we have 2^n aggregate networks to materialize and the memory consumed could be excessively large.

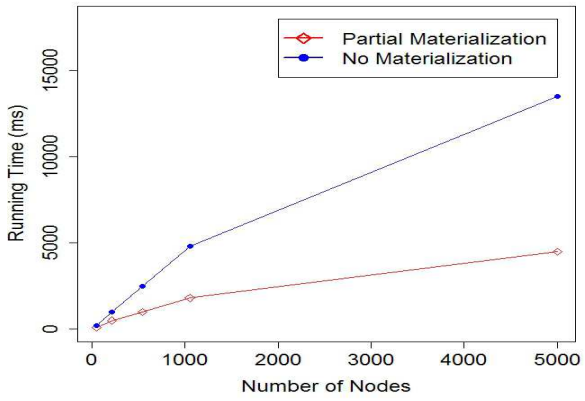
We perform other experiments on the other SCI data sets and build a heterogeneous network with 500 papers, 953 authors, 1,255 keywords, 5,715 references, 274 affiliations and 116 conferences\journals. To measure the performance of our operations and strategies, we compare the running time of them. Firstly, we extract coauthor network from the raw graph. Then, we compare rotate and stretch operation in a different order of magnitude. As shown in Figure (c), when the number of nodes is increasing, both of the running time showed a linear growth and stretch operation needs longer time than rotate operation. In addition, we want to know the influence of the length of relation meta path. We perform this experiment by extracting subgraph form the raw graph above. Then, we select relation meta path $P = keyword \rightarrow paper \rightarrow keyword$ and relation meta path $P = keyword \rightarrow author \rightarrow paper \rightarrow author \rightarrow keyword$ and start varying the network size. Figure (d) shows that time consumed is greatly influenced by the length of relation meta path. In particular, when added to 2 in length, the time consumption doubling.

6. ABOUT THE DEMONSTRATION

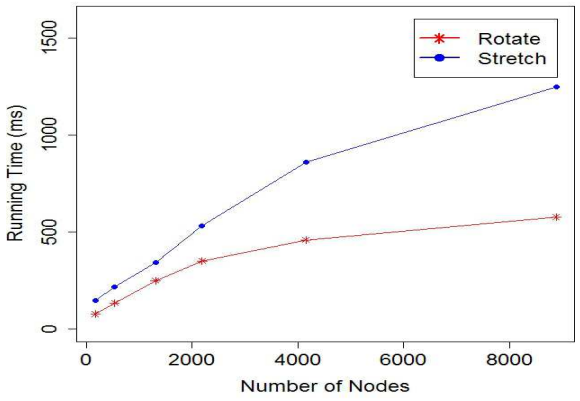
The LiterMiner is currently being implemented in Java 2 sdk 1.6 under Microsoft Windows XP with a 3GHz Pentium IV CPU and 2GB main memory. We have conducted extensive experiments on the most up-to-date data sets from the Science Citation Index and The Engineering Index. Experiments show that at most times, on graph data sets of about 5,000 vertices and 100,000 edges, the operation can be performed in about 3 seconds.

We have extracted 567 papers on data mining (DM) since the beginning of 2012. The data set was downloaded from the Science Citation Index in April, 2012. Then, we took the scenario of keywords co-occurrence network by the relation meta path $P = keyword \rightarrow paper \rightarrow keyword$. Figure 7 demonstrates roll-up, drill-down and rotate operations. Firstly, we rolled-up the keyword co-occurrence network to the topic network on the topological dimension and then

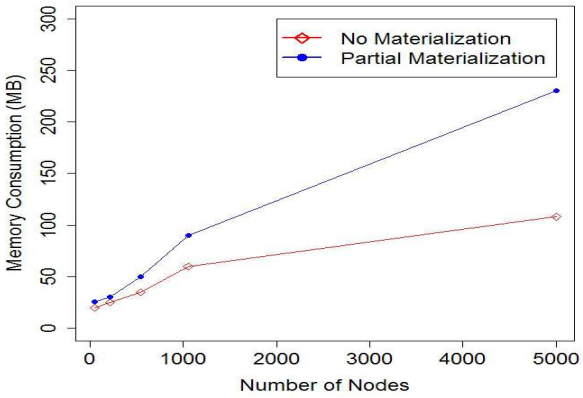
[2]http://science.thomosonreuters.com



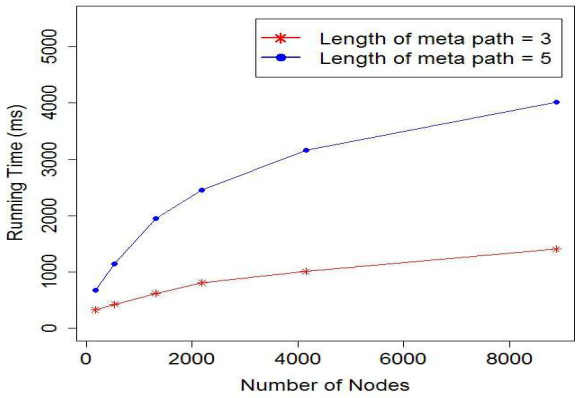
(a) Time Consumption



(c) Rotate vs. Stretch



(b) Memory Consumption



(d) Relation Meta Path

drilled-down by affiliation on information dimension. Finally, we rotated the network to convert keywords and papers.

7. RELATED WORK

Online analytical processing is an important concept in data mining. Although OLAP for the traditional form of spreadsheet data has been extensively studied, there are few studies on OLAP for graphs. As a result, a wealth of information hidden in such bibliographic information networks has gone largely unexplored. In2007, the concept of Link OLAP [16] was proposed, which extended entity-oriented analysis to a link-oriented analysis. In addition, Link OLAP can provide superior solutions in specific analysis scenarios. Being based on complex network visualization and OLAP technology, link-oriented analysis breaks the monotony of the two dimensional form expression of the traditional OLAP system. However, it only focuses on link analysis, and has not proposed operations and models of Link OLAP. In the same year, there was an interesting study that put graphs in a multi-dimensional and multi-level OLAP framework in [4] [5], trying to introduce online analytical processing technology into complex network analysis. Graph OLAP presents a multi-dimensional and multi-level view over graphs. It looks into different semantics of OLAP operations, and classifies the framework into

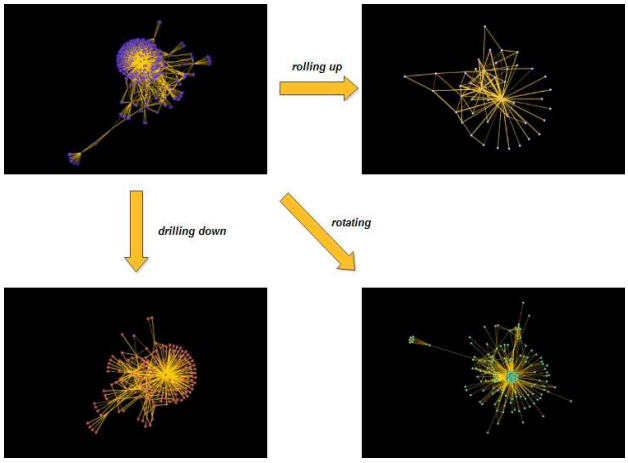


Figure 8: Operations of HMGraph OLAP

two major sub-cases: informational OLAP and topological OLAP. We can see that the initial study did not consider the Graph OLAP data warehouse model and algorithm design. Subsequent research added to the contents of these studies. Some researchers proposed Graph Cube [17], a new data warehousing model that supports OLAP queries effec-

tively on large multi-dimensional networks. Besides traditional cuboid queries, a new class of OLAP queries, crossboid, was introduced,being of unique use in multidimensional networks. Some researchers proposed a framework for efficient OLAP [12] on information networks with a focus on the most interesting kind, the topological OLAP, which incurs topological changes in the underlying networks, proposing two techniques in a constraint-pushing framework, T-Distributiveness and T-Monotonicity. To support efficient graph OLAP operations on information networks, some researchers proposed novel prototypes for Graph OLAP. Of these, BibNetMiner [15] is worth mentioning, and is designed for sophisticated information network mining on such bibliographic databases. In addition, BibNetMiner contains several attractive functions, including clustering, ranking and profiling of conferences and authors based on the research sub-fields. However, it is unable to provide basic application flexibility. For instance, when users need to change their focus from one topic to another, it has to generate a new network for analysis. Similar situations occur in Graph OLAPer [8]. It is hard to extract topics from the heterogeneous network. To address these issues, InfoNetOLAPer [7] came into being, providing topic-oriented, integrated, and multi-dimensional organizational solutions for Information networks. Unfortunately, it is unable to analyze heterogeneous networks directly, such as bibliographic networks and social media networks. Moreover, different semantic meanings behind entities and paths are not taken into consideration.

8. CONCLUSION

Graph OLAP is a new research field that has emerged in recent years which focuses on multi-dimensional and multi-granularity analysis on graphs. In this paper, we have proposed a HMGraph OLAP framework, which includes a warehouse model and HMGraph Cube. Furthermore, we have proposed entity dimensions for the very first time, which are provided for conversions between entities and relations and the mining of implicit relationships. With regard to efficient implementation, we have detailed a partial materialization strategy herein. Experiments have demonstrated the efficiency of our proposed optimizations and the sound performance of our prototype.

As for future studies, there are numerous directions that we would like to pursue in regard to this topic. These include: Improving the current algorithm and optimizing the design of HMGraph OLAP to enhance the performance of the prototype system. Furthermore, we would also like to focus on the granularity of HMGraph OLAP, including granularity size selection and granularity storage. More rational division of granularity will lead to significant enhancements in the query algorithm and storage efficiency. Furthermore, in order to meet the requirements of large-scale networks in actual applications, we will propose a distributed framework for HMGraph OLAP and apply this system into a real-world scenario to test its scalability, effectiveness and efficiency.

9. ACKNOWLEDGMENTS

The authors would like to thank Wanting Wen for her valuable comments and suggestions.

This study was supported by the National Science Foundation of China (No.60905025, 90924029, 61074128).

10. REFERENCES

[1] K. S. Beyer and R. Ramakrishnan. Bottom-up computation of sparse and iceberg cubes. In *SIGMOD*, pages 359–370, 199.

[2] D. Burdick, A. Doan, R. Ramakrishnan, and S. Vaithyanathan. Olap over imprecise data with domain constraints. In *VLDB*, pages 39–50, 2007.

[3] S. Chaudhuri and U. Dayal. An overview of data warehousing and olap technology. 26(1):65–74, March 1997.

[4] C. Chen, X. Yan, Z. Feida, J. Han, and P. S.Yu. Graph olap: Towards online analytical processing on graphs. In *ICDM'08*, pages 103–112, Dec 2008.

[5] C. Chen, X. Yan, Z. Feida, J. Han, and P. S. Yu. Grapholap: a multi-dimentional framework for graph data analysis. *Knowledge and Information System*, 21(1):41–63, 2009.

[6] V. Harinarayan, A. Rajaraman, and J. D. Ullman. Implementing data cubes efficiently. In *SIGMOD'96*, volume 25, pages 205–216, June 1996.

[7] C. Li, P. S.Yu, L. Zhao, Y. Xie, and W. Lin. Infonetolaper : Integrating infonetwarehouse and infonetcube with infonetolap. In *VLDB'11*, volume 4, pages 1422–1425, 2011.

[8] C. Li, L. Zhao, C. Tang, Y. Chen, J. Li, X. Zhao, and X. Liu. Modeling, design and implementation of graph olaping. *Journal of Software*, 22(2):258–268, 2011.

[9] X. Li, J. Han, and H. Gonzalez. High-dimensional olap: a minimal cubing approach. In *VLDB*, pages 528–539, 2004.

[10] X. Li, J. Han, and H. Gonzalez. High-dimensional olap: A minimal cubing approach. In *VLDB'04*, volume 30, pages 528–539, 2004.

[11] K. Morfonios, S. Konakas, Y. Ioannidis, and N. Kotsis. Rolap implementations of the data cube. In *ACM Computing Surveys*, volume 4, 2007.

[12] Q. Qu, F. Zhu, X. Yan, J. Han, P. S. Yu, and H. Li. Efficient topological olap on information networks. *DATABASE SYSTEMS FOR ADVANCED APPLICATIONS*, pages 389–403, 2011.

[13] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu. Pathsim: Meta path-based top-k similarity search in heterogeneous information. In *VLDB'11*, 2011.

[14] Y. Sun, B. Norick, J. Han, X. Yan, P. S. Yu, and X. Yu. Integrating meta-path selection with user-guided object clustering in heterogeneous information networks. In *KDD'12*, 2012.

[15] Y. Sun, T. Wu, Z. Yin, H. Cheng, J. Han, and X. Yin. Bibnetminer: Mining bibliographic information networks. In *SIGMOD'08*, pages 1341–1344, June 2008.

[16] W. wei. Complex network virtualization and link olap. 2007.

[17] P. Zhao, X. Li, D. Xin, and J. Han. Graph cube: On warehousing and olap multidimensional networks. In *SIGMOD'11*, pages 12–16, 2011.