Graph Data Management: Techniques and Applications

Detailed Table of Contents

Foreword
Preface
Acknowledgment

Section I: Basic Challenges of Data Management in Graph Databases

Chapter 1
Graph Representation
  D. Dominguez-Sal, Universitat Politècnica de Catalunya, Spain
  V. Muntés-Mulero, Universitat Politècnica de Catalunya, Spain
  N. Martínez-Bazán, Universitat Politècnica de Catalunya, Spain
  J. Larriba-Pey, Universitat Politècnica de Catalunya, Spain

In this chapter, the authors review different graph implementation alternatives that have been proposed in the literature. The objective is to provide the readers with a broad set of alternatives to implement a graph, according to their needs. The authors pay special attention to the techniques that enable the management of large graphs. They also include a description of the most representative libraries available for representing graphs.

Chapter 2
The Graph Traversal Pattern
  Marko A. Rodriguez, AT&T Interactive, USA
  Peter Neubauer, Neo Technology, Sweden

A graph is a structure composed of a set of vertices (i.e. nodes, dots) connected to one another by a set of edges (i.e. links, lines). The concept of a graph has been around since the late 19th century, however, only in recent decades has there been a strong resurgence in both theoretical and applied graph research in mathematics, physics, and computer science. In applied computing, since the late 1960s, the interlinked table structure of the relational database has been the predominant information storage and retrieval model. With the growth of graph/network-based data and the need to efficiently process such data, new data management systems have been developed. In contrast to the index-intensive, set-theoretic operations of relational databases, graph databases make use of index-free, local traversals. This chapter discusses the graph traversal pattern and its use in computing.
Chapter 3
Data, Storage and Index Models for Graph Databases
Srinath Srinivasa, International Institute of Information Technology, India

Management of graph structured data has important applications in several areas. Queries on such data sets are based on structural properties of the graphs, in addition to values of attributes. Answering such queries pose significant challenges, as reasoning about structural properties across graphs are typically intractable problems. This chapter provides an overview of the challenges in designing databases over graph datasets. Different application areas that use graph databases, pose their own unique set of challenges, making the task of designing a generic graph-oriented DBMS still an elusive goal. The purpose of this chapter is to survey some of the piecemeal solutions that have been proposed to address specific challenges in graph data management and suggest an overall structure in which these different solutions can be meaningfully placed.

Chapter 4
An Overview of Graph Indexing and Querying Techniques
Sherif Sakr, National ICT Australia (NICTA), Australia
Ghazi Al-Naymat, University of Tabuk, Saudi Arabia

Recently, there has been a lot of interest in the application of graphs in different domains. Graphs have been widely used for data modeling in different application domains such as: chemical compounds, protein networks, social networks and semantic Web. Given a query graph, the task of retrieving related graphs as a result of the query from a large graph database is a key issue in any graph-based application. This has raised a crucial need for efficient graph indexing and querying techniques. This chapter provides an overview of different techniques for indexing and querying graph databases. An overview of several proposals of graph query language is also given. Finally, we provide a set of guidelines for future research directions.

Chapter 5
Behind Efficient Algorithms to Search in Graphs
Alfredo Ferro, Universita di Catania, Italy
Rosalba Giugno, Universita di Catania, Italy
Alfredo Pulvirenti, Universita di Catania, Italy
Dennis Shasha, New York University, USA

From biochemical applications to social networks, graphs represent data. Comparing graphs or searching for motifs on such data often reveals interesting and useful patterns. Most of the problems on graphs are known to be NP-complete. Because of the computational complexity of subgraph matching, reducing the candidate graphs or restricting the space in which to search for motifs is critical to achieving efficiency. Therefore, to optimize and engineer isomorphism algorithms, design indexing and suitable search methods for large graphs are the main directions investigated in the graph searching area. This chapter focuses on the key concepts underlying the existing algorithms. First it reviews the most known used algorithms to compare two algorithms and then it describes the algorithms to search on large graphs making emphasis on their application on biological area.
Chapter 6
A Survey of Relational Approaches for Graph Pattern Matching over Large Graphs

Jiefeng Cheng, The University of Hong Kong, Pokfulam Road, Hong Kong
Jeffrey Xu Yu, The Chinese University of Hong Kong, Shatin, NT, Hong Kong

Due to rapid growth of the Internet and new scientific/technological advances, there exist many new applications that model data as graphs, because graphs have sufficient expressiveness to model complicated structures. The dominance of graphs in real-world applications demands new graph processing techniques to access and analyze large graph datasets effectively and efficiently. Among those techniques, a graph pattern matching problem receives increasing attention, which is to find all patterns in a large data graph that match a user-given graph pattern. In this chapter, the authors review approaches to process such graph pattern queries with a framework of multi joins, which can be easily implemented in relational databases and requires no specialized native storage for graphs. We also discuss the top-k graph pattern matching problem.

Chapter 7
Labelling-Scheme-based Subgraph Query Processing on Graph Data

Hongzhi Wang, Harbin Institute of Technology, China
Jianzhong Li, Harbin Institute of Technology, China
Hong Gao, Harbin Institute of Technology, China

When data are modeled as graphs, many research issues arise. In particular, there are many new challenges in query processing on graph data. This chapter studies the problem of structural queries on graph data. A hash-based structural join algorithm, HGJoin, is first proposed to handle reachability queries on graph data. Then, it is extended to the algorithms to process structural queries in form of bipartite graphs. Finally, based on these algorithms, a strategy to process subgraph queries in form of general DAGs is proposed. It is notable that all the algorithms above can be slightly modified to process structural queries in form of general graphs.

Section II: Advanced Querying and Mining Aspects of Graph Databases

Chapter 8
G-hash: Towards Fast Kernel-based Similarity Search in Large Graph Databases

Xiaohong Wang, University of Kansas, USA
Jun Huan, University of Kansas, USA

Structured data such as graphs and networks have posed significant challenges to fundamental aspects of data management including efficient storage, indexing, and similarity search. With the fast accumulation of graph databases, similarity search in graph databases has emerged as an important research topic. Graph similarity search has applications in a wide range of domains including chemoinformatics, bioinformatics, sensor network management, social network management, and XML documents, among others. The objective of this chapter is to enable fast similarity search in large graph databases with graph kernel functions. In particular, we propose (i) a novel kernel-based similarity measurement and (ii) an efficient indexing structure for graph data management. In our method, we use a hash table to support
efficient storage and fast search of the extracted local features from graph data. Using the hash table, we have developed a graph kernel function to capture the intrinsic similarity of graphs and for fast similarity query processing. We have demonstrated the utility of the proposed methods using large chemical structure graph databases.

Chapter 9
TEDI: Efficient Shortest Path Query Answering on Graphs
Fang Wei, University of Freiburg, Germany

Efficient shortest path query answering in large graphs is enjoying a growing number of applications, such as ranked keyword search in databases, social networks, ontology reasoning and bioinformatics. A shortest path query on a graph finds the shortest path for the given source and target vertices in the graph. Current techniques for efficient evaluation of such queries are based on the pre-computation of compressed Breadth First Search trees of the graph. However, they suffer from drawbacks of scalability. To address these problems, this chapter describes TEDI, an indexing and query processing scheme for the shortest path query answering. TEDI is based on the tree decomposition methodology. The graph is first decomposed into a tree in which the node (a.k.a. bag) contains more than one vertex from the graph. The shortest paths are stored in such bags and these local paths together with the tree are the components of the index of the graph. Based on this index, a bottom-up operation can be executed to find the shortest path for any given source and target vertices. The experimental results show that TEDI offers orders-of-magnitude performance improvement over existing approaches on the index construction time, the index size and the query answering.

Chapter 10
Graph Mining Techniques: Focusing on discriminating between real and synthetic graphs
Ana Paula Appel, Federal University of Espirito Santo at São Mateus, Espirito Santo, Brazil
Christos Faloutsos, Carnegie Mellon University, USA
Caetano Traina Junior, University of São Paulo at São Carlos, São Paulo, Brazil

Graphs appear in several settings, like social networks, recommendation systems, computer communication networks, gene/protein biological networks, among others. A large amount of graph patterns, as well as graph generator models that mimic such patterns have been proposed over the last years. However, a deep and recurring question still remains: “What is a good pattern?” The answer is related to finding a pattern or a tool able to help distinguishing between actual real-world and fake graphs. Here we explore the ability of ShatterPlots, a simple and powerful algorithm to tease out patterns of real graphs, helping us to spot fake/masked graphs. The idea is to force a graph to reach a critical (“Shattering”) point, randomly deleting edges, and study its properties at that point.
Chapter 11
Matrix Decomposition-based Dimensionality Reduction on Graph Data
Hiroto Saigo, Kyushu Institute of Technology, Japan
Koji Tsuda, Computational Biology Research Center, National Institute of Advanced Industrial Science and Technology (AIST), Japan

Graph is a mathematical framework that allows us to represent and manage many real-world data such as relational data, multimedia data and biomedical data. When each data point is represented as a graph and we are given a number of graphs, a task is to extract a few common patterns that capture the property of each population. A frequent graph mining algorithm such as AGM, gSpan and Gaston can enumerate all the frequent patterns in graph data, however, the number of patterns grows exponentially, therefore it is essential to output only discriminative patterns. There are many existing researches on this topic, but this chapter focuses on the use of matrix decomposition techniques, and explains the two general cases where either i) no target label is available, or ii) target label is available for each data point. The resulting method is a branch and bound pattern mining algorithm with efficient pruning condition, and we evaluate its effectiveness on cheminformatics data.

Chapter 12
Clustering Vertices in Weighted Graphs
Derry Tanti Wijaya, Carnegie Mellon University, USA
Stephane Bressan, National University of Singapore, Singapore

Clustering is the unsupervised process of discovering natural clusters so that objects within the same cluster are similar and objects from different clusters are dissimilar. In clustering, if similarity relations between objects are represented as a simple, weighted graph where objects are vertices and similarities between objects are weights of edges; clustering reduces to the problem of graph clustering. A natural notion of graph clustering is the separation of sparsely connected dense sub graphs from each other based on the notion of intra-cluster density vs. inter-cluster sparseness. This chapter provides an overview of existing graph algorithms for clustering vertices in weighted graphs: Minimum Spanning Tree (MST) clustering, Markov clustering, and Star clustering. This includes the variants of Star clustering, MST clustering and Ricochet.

Chapter 13
Large Scale Graph Mining with MapReduce: Counting Triangles in Large Real Networks
Charalampos E. Tsourakakis, Carnegie Mellon University, USA

In recent years, a considerable amount of research has focused on the study of graph structures arising from technological, biological and sociological systems. Graphs are the tool of choice in modeling such systems since they are typically described as sets of pairwise interactions. Important examples of such datasets are the Internet, the Web, social networks, and large-scale information networks which reach the planetary scale, e.g., Facebook and LinkedIn. The necessity to process large datasets, including graphs, has led to a major shift towards distributed computing and parallel applications, especially in the recent years. MapReduce was developed by Google, one of the largest users of multiple processors computing in the world, for facilitating the development of scalable and fault tolerant applications. MapReduce has become the de facto standard for processing large scale datasets both in industry and academia. This Chapter presents state of the art work on large scale graph mining using MapReduce. We survey research work on an important graph mining problem, counting the number of triangles in large-real world networks. We present the most important applications related to the count of triangles and two families of algorithms, a spectral and a combinatorial one, which solve the problem efficiently.
Chapter 14
Graph Representation and Anonymization in Large Survey Rating Data
Xiaoxun Sun, University of Southern Queensland, Australia
Min Li, University of Southern Queensland, Australia

We study the challenges of protecting privacy of individuals in the large public survey rating data in this chapter. Recent study shows that personal information in supposedly anonymous movie rating records is de-identified. The survey rating data usually contains both ratings of sensitive and non-sensitive issues. The ratings of sensitive issues involve personal privacy. Even though the survey participants do not reveal any of their ratings, their survey records are potentially identifiable by using information from other public sources. None of the existing anonymisation principles can effectively prevent such breaches in large survey rating data sets. We tackle the problem by defining a principle called \((k, \varepsilon)\)-anonymity model to protect privacy. Intuitively, the principle requires that, for each transaction \(t\) in the given survey rating data \(T\), at least \((k - 1)\) other transactions in \(T\) must have ratings similar to \(t\), where the similarity is controlled by \(\varepsilon\). The \((k, \varepsilon)\)-anonymity model is formulated by its graphical representation and a specific graph-anonymisation problem is studied by adopting graph modification with graph theory. Various cases are analyzed and methods are developed to make the updated graph meet \((k, \varepsilon)\) requirements. The methods are applied to two real-life data sets to demonstrate their efficiency and practical utility.

Section III: Graph Database Applications in Various Domains

Chapter 15
Querying RDF Data
Faisal Alkhateeb, Yarmouk University, Jordan
Jérôme Euzenat, INRIA & LIG, France

The Resource Description Framework (RDF) is the W3C recommended data model for the representation of information about resources on the Web. Nowadays, more resources are annotated via RDF due to its simple data model, formal semantics, and a sound and complete inference mechanism. A query language that provides a range of querying paradigms is therefore needed. This chapter provides an introduction to the RDF language as well as surveys the languages that can be used for querying RDF graphs. Then it reviews some of the languages that can be used for querying RDF and provides a comparison between these query languages.

Chapter 16
On the Efficiency of Querying and Storing RDF Documents
Maria-Esther Vidal, Universidad Simón Bolívar, Venezuela
Amadís Martínez, Universidad Simón Bolívar and Universidad de Carabobo, Venezuela
Edna Ruckhaus, Universidad Simón Bolívar, Venezuela
Tomas Lampo, University of Maryland, USA
Javier Sierra, Universidad Simón Bolívar, Venezuela

In the context of the Semantic Web, different approaches have been defined to represent RDF documents, and the selected representation affects storage and time complexity of the RDF data recovery and query processing tasks. This chapter addresses the problem of efficiently querying and storing RDF documents, and presents an alternative representation of RDF data, Bhyper, which is based on hypergraphs.
Additionally, access and optimization techniques to efficiently execute queries with low cost, are defined on top of this hypergraph based representation. The chapter’s authors have empirically studied the performance of the Hyper based techniques, and their experimental results show that the proposed hypergraph based formalization reduces the RDF data access time as well as the space needed to store the Hyper structures, while the query execution time of state-of-the-art RDF engines can be sped up by up to two orders of magnitude.

Chapter 17
Graph applications in chemoinformatics and structural bioinformatics

Eleanor Joyce Gardiner, University of Sheffield, United Kingdom

The focus of this chapter is the uses of graph theory in chemoinformatics and in structural bioinformatics. There is a long history of chemical graph theory dating back to the 1860’s and Kekule’s structural theory. It is natural to regard the atoms of a molecule as nodes and the bonds as edges (2D representations) of a labeled graph (a molecular graph). This chapter will concentrate on the algorithms developed to exploit the computer representation of such graphs and their extensions in both two and three dimensions (where an edge represents the distance in 3D space between a pair of atoms), together with the algorithms developed to exploit them. The algorithms will generally be summarized rather than detailed. The methods were later extended to larger macromolecules (such as proteins); these will be covered in less detail.

Chapter 18
Business Process Graphs – Similarity Search and Matching

Remco Dijkman, Eindhoven University of Technology, The Netherlands
Marlon Dumas, University of Tartu, Estonia
Luciano García-Bañuelos, University of Tartu, Estonia

Organizations create collections of hundreds or even thousands of business process models to describe their operations. This chapter explains how graphs can be used as underlying formalism to develop techniques for managing such collections. To this end it defines the business process graph formalism. On this formalism it defines techniques for determining similarity of business process graphs. Such techniques can be used to quickly search through a collection of business process graphs to find the graph that is most relevant to a given query. These techniques can be used by tool builders that develop tools for managing large collections of business process models. The aim of the chapter is to provide an overview of the research area of using graphs to do similarity search and matching of business processes.

Chapter 19
A Graph-Based Approach for Semantic Process Model Discovery

Ahmed Gater, PRiSM Laboratory of Versailles University, France
Daniela Grigori, PRiSM Laboratory of Versailles University, France
Mokrane Bouzeghoub, PRISM Laboratory of Versailles University, France

One of the key tasks in the service oriented architecture that semantic web services aim to automate is the discovery of services that can fulfill the applications or user needs. OWL-S is one of the proposals for describing semantic metadata about web services, which is based on the OWL ontology language. Majority of current approaches for matching OWL-S processes take into account only the inputs/outputs service profile. This chapter argues that, in many situations the service matchmaking should take into account also the process model. We present matching techniques that operate on OWL-S process models.
and allow retrieving in a given repository, the processes most similar to the query. To do so, the chapter proposes to reduce the problem of process matching to a graph matching problem and to adapt existing algorithms for this purpose. It proposes a similarity measure used to rank the discovered services. This measure captures differences in process structure and semantic differences between input/outputs used in the processes.

**Chapter 20**

Shortest Path in Transportation Network and Weighted Subdivisions

*Radwa Elshawi, NICTA and University of Sydney, Australia*

*Joachim Gudmundsson, NICTA and University of Sydney, Australia*

The shortest path problem asks for a path between two given points such that the sum of its edges is minimized. The problem has a rich history and has been studied extensively since the 1950’s in many areas of computer science, among them network optimization, graph theory and computational geometry. This chapter considers two versions of the problem; the shortest path in a transportation network and the shortest path in a weighted subdivision, sometimes called a terrain.

**Compilation of References**

**About the Contributors**

**Index**