

DOCTORAL THESIS

Estimation techniques for advanced database applications

Peng, Yun

Date of Award:
2013

[Link to publication](#)

General rights

Copyright and intellectual property rights for the publications made accessible in HKBU Scholars are retained by the authors and/or other copyright owners. In addition to the restrictions prescribed by the Copyright Ordinance of Hong Kong, all users and readers must also observe the following terms of use:

- Users may download and print one copy of any publication from HKBU Scholars for the purpose of private study or research
- Users cannot further distribute the material or use it for any profit-making activity or commercial gain
- To share publications in HKBU Scholars with others, users are welcome to freely distribute the permanent URL assigned to the publication

Estimation Techniques for Advanced Database Applications

Yun PENG

**A thesis submitted in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy**

Principal Supervisor: Assistant Professor Byron CHOI

Hong Kong Baptist University

August 2013

Abstract

Database systems have been ubiquitously used as fundamental facilities to manage a large amount of data efficiently. Nowadays, the data that needs to be managed by database systems is growing explosively. For example, the number of users of Twitter has exceeded 500 millions and the total number of tweets sent has exceeded 170 billions; the number of users of Facebook has exceeded 1.15 billions and the number of friend connections has exceeded 150 billions. How to efficiently manage and analyze data of such scale is a crucial task of database systems. Among many other solutions, estimation techniques have been proved successful to address some of these problems. In this thesis, we study the applications of estimation techniques in important database problems, including three graph database problems and one relational database problem. Specifically, regarding graph databases, we study selectivity estimation of twig queries on cyclic graphs, authentication of outsourced subgraph similarity search and optimal graph index prediction. Regarding relational databases, we study the classical view update problem.

Firstly, we study the selectivity estimation of twig queries on cyclic graphs. Similar to relational database, selectivity estimation plays a crucial role in query optimization of graph database. However, determining the optimal query evaluation plan (QEP) for a graph query is more challenging than its relational counterpart. This is because of the complexity of graph structure analysis. For example, given a twig query against a cyclic graph, computing the cost of a QEP may need to enumerate all the matchings between the subtrees of the twig query and the graph. This is potentially costly as the graph may contain an exponential number of matchings. Many works have been proposed to study the selectivity estimation. However, most of them focus on either the cyclic graphs or

the twig queries but not both. In the first part of this thesis (Chapter 3), we propose a novel histogram-based method to support selectivity estimation of twig queries on cyclic graphs.

Secondly, we study the estimation in outsourced subgraph similarity search. Subgraph similarity search itself is *estimation-like*, which returns the graphs that have *approximate* substructures with the query graph. This query has been used in a wide range of applications including bioinformatics, chem-informatics, Web topology, etc. Recently, due to the complexity of subgraph similarity search and the explosive growth of graph data, the data owner of graph database is more appealing to outsource their data to third-party service providers, which will process the query on behalf of the data owner. However, the service provider may not be trustable and therefore it is required to return an authentication structure to the user for authenticating the correctness of query results. Since the manipulation of the authentication structure takes the major overhead of outsourced subgraph similarity search, in the second part (Chapter 4) of this thesis, we propose an estimation-based method to optimize its processing.

Thirdly, estimation is also crucial in the optimal graph index prediction. Recently, an ample body of graph indexes have been proposed to optimize query processing on graphs. However, the performances of such indexes may vary greatly as verified from our experiments with a large number of random and scale-free graphs. In particular, our preliminary experiments show that the runtime of 1,000 random queries on an index, even on the same graph, can often exhibit large variances. Specifically, the mean and standard deviation of `2-hop labeling` are 14.1 seconds and 4.2 and those of `prime labeling` are 11.6 seconds and 59.7, respectively. Moreover, the runtime is often skewed and has a long tail at large values. Therefore, it is desired to predict the optimal index on a graph. However, designing an exact performance model is a daunting task since the structures of graph indexes are often complex and ad-hoc. In the third part (Chapter 5) of this thesis, we apply statistical distributions to estimate the query performance. Then, the classical data mining techniques are applied to predict the optimal index.

Finally, we study the application of estimation techniques in the classical view update

problem, where the updates specified by the user on the view need to be translated to the updates on the source database, such that the new view derived from the updated source database is consistent with the user's expectation. The state-of-the-art view update analysis methods often involve two interleaving stages: side-effect determination and update translation. It is well-known that the translation problem is NP-complete. Therefore, it is desirable to develop a method that can efficiently estimate the side effects, such that the view updates having side effects can be filtered before they are passed to the costly translation. In this forth part (Chapter 6) of this thesis, we develop a data-oriented side-effect estimation technique to support such view update analysis.

The works proposed in this thesis verify that the estimation techniques are useful in various major components in database systems.

Keywords: Selectivity estimation, Subgraph similarity search, Graph index prediction, Side-effect estimation, Graph database, Outsourced database, Relational database

Table of Contents

Declaration	i
Abstract	ii
Acknowledgements	v
Table of Contents	vi
List of Tables	xi
List of Figures	xii
Chapter 1 Introduction	1
1.1 Case Study: Query Optimization	1
1.2 Graph Database	4
1.2.1 Graph Data	4
1.2.2 Graph Query	7
1.3 Estimation Techniques for Graph Database	10
1.3.1 Selectivity Estimation of Twig Queries on Cyclic Graphs	11
1.3.2 Estimation for Outsourced Subgraph Similarity Search	13
1.3.3 Query Time Estimation for Optimal Index Prediction	16
1.4 Estimation Techniques for Relational Database	18
1.4.1 Side-effect Estimation for View Updates	18
1.5 Thesis Organization	20

Chapter 2	Related Work	21
2.1	Selectivity Estimation on Graphs	21
2.1.1	Graph-based Approach	21
2.1.2	Relational-based Approach	24
2.1.3	Alternative Representation of Graphs	26
2.2	Outsourced Subgraph Similarity Search	27
2.3	Optimal Graph Index Prediction	33
2.3.1	Review of Works on Reachability Query	33
2.3.2	Review of Works on Optimal Graph Index Prediction	35
2.4	View Update Problem	37
Chapter 3	Selectivity Estimation of Twig Queries on Cyclic Graphs	43
3.1	Introduction	43
3.2	Definitions and Preliminaries	47
3.2.1	Data Model and Twig Queries	47
3.2.2	Consecutive Ones Property	48
3.3	Overview	48
3.4	Representation of Cyclic Graphs	49
3.4.1	The Original Prime Labeling	50
3.4.2	Prime Labeling for Cyclic Graphs	50
3.4.3	Matrix Representation of Cyclic Graphs	52
3.5	Matrix transformations	53
3.5.1	Transforming to C1P Matrix	54
3.5.2	Optimizing Matrix Transformation	56
3.6	Selectivity Estimation	58
3.6.1	Two-dimensional Histograms	59
3.6.2	The Overall Estimation Algorithm	60
3.6.3	Estimation Details with Histograms	65
3.7	Experimental Evaluation	69
3.7.1	Experiments on overall performance	71

3.7.2	Experiments on optimizations	73
3.7.3	Indirect Comparison with XSketch and TreeSketch	76
3.8	Conclusions	77

Chapter 4 Authenticated Subgraph Similarity Search in Outsourced Graph

	Databases	78
4.1	Introduction	78
4.2	Background and Problem Statement	82
4.2.1	Backgrounds	82
4.2.2	Problem Formulation	85
4.2.3	Query Paradigm and Overview of Our Method	87
4.3	Metric Based Filtering	88
4.4	Graph Metric Tree	91
4.4.1	GMTree Structure	92
4.4.2	Subgraph Similarity Search on GMTree	94
4.5	Authentication with GMTree	95
4.5.1	Signing GMTree	95
4.5.2	Definition of Verification Objects	97
4.5.3	\mathcal{VO} Construction	98
4.5.4	Cost Model of VO Size	101
4.5.5	Authentication Algorithm	105
4.6	Optimization Problems	108
4.6.1	Authenticated MCS Computation	109
4.6.2	Pivot Selection	113
4.7	Experimental Evaluation	119
4.7.1	Comparison with the Baseline and Grafil*	121
4.7.2	Authenticated Query Overhead	122
4.7.3	Detailed Performances	123
4.7.4	Effectiveness of Optimizations on GMTree	127
4.8	Conclusions	129

Chapter 5	Spectral Decomposition for Optimal Graph Index Prediction	130
5.1	Introduction	130
5.2	Background to Graph Spectral Decomposition	133
5.3	Problem Formulation	134
5.4	Performance Metric	135
5.5	Spectral Similarity of Graphs	137
5.5.1	Unifying the Dimensionalities of Graphs	137
5.5.2	Permutation of Vertex ID	138
5.5.3	Spectral Similarity Between Graphs	139
5.6	Prediction Algorithm	140
5.6.1	Clustering Algorithm	141
5.6.2	Prediction Algorithm	142
5.7	Experimental Evaluation	143
5.7.1	Experimental Setup	143
5.7.2	Experiments on Distribution Fittings	144
5.7.3	Prediction Accuracies	145
5.8	Conclusions	148
Chapter 6	Side-effect Estimation: A Practical Support to the View Updates of Relational Databases	149
6.1	Introduction	149
6.2	Preliminaries and Problem Statement	154
6.3	Quality of Side-effect Detector	155
6.3.1	Analysis with KL Divergence	156
6.3.2	Revised Notion of Errors	161
6.4	Join Cardinality Summary (JCard)	162
6.4.1	Terminologies and Notations	162
6.4.2	JCard Definition	165
6.5	Side-effect Estimation with JCard	168
6.5.1	Estimation Algorithm	169

6.5.2	update_equiv_class	172
6.5.3	Analysis of Source of Estimation Errors	173
6.6	Deletions and Replacements	175
6.6.1	Deletions	175
6.6.2	Replacements	176
6.7	Projection	177
6.7.1	Filling in Missing Attributes	177
6.7.2	Extension of JCard and Its Algorithm	179
6.8	Optimization Problems in JCard	180
6.8.1	Candidate Tuples Selection	180
6.8.2	Optimal Join Tree Selection	184
6.9	Experimental Evaluation	185
6.10	Conclusions	198
Chapter 7	Conclusions and Future Work	199
7.1	Summary and Contributions	199
7.2	Future Work	202
Bibliography		206