Data & Knowledge Engineering 92 (2014) 1–19

Contents lists available at ScienceDirect

Data & Knowledge Engineering

journal homepage: www.elsevier.com/locate/datak

Editorial

A framework for ranking uncertain distributed database

Yousry M. AbdulAzeem [*], Ali I. ElDesouky, Hesham A. Ali

Computer Engineering and Systems Department, Faculty of Engineering, Mansoura University, Egypt

ARTICLE INFO

Article history:
Received 4 September 2012
Received in revised form 25 November 2013
Accepted 24 May 2014
Available online 13 June 2014

Keywords:
Uncertainty
Distributed databases
Database applications
Ranking
Threshold

ABSTRACT

Distribution and uncertainty are considered as the most important design issues in database applications nowadays. A lot of ranking or top-*k* query processing techniques are introduced to solve the problems of communication cost and centralized processing. On the other hand, many techniques are also developed for modeling and managing uncertain databases. Although these techniques were efficient, they didn't deal with distributed data uncertainty. This paper proposes a framework that deals with both data distribution and uncertainty based on ranking queries. Within the proposed framework, communication and computation-efficient algorithms are investigated for retrieving the top-*k* tuples from distributed sites. The main objective of these algorithms is to reduce the communication rounds utilized and amount of data transmitted while achieving efficient ranking. Experimental results show that both proposed techniques have a great impact in reducing communication cost. Both techniques are efficient but in different situations. The first one is efficient in the case of low number of sites while the other achieves better performance at higher number of sites.

1. Introduction

Distributed data processing has been a major field in nowadays applications. Many applications collect data from distributed sites and process it to gain an overall result. Content Distribution Network (CDN) [1,2], sensor networks [3–6], multimedia database [7], information retrieval from geographically separated data centers [8–10], network monitoring over distributed logs [11] and data extracted from a set of data streams [12,13] are all examples of distributed data processing applications. In these applications, processing sites' data at a centralized server is a very expensive task. Both the large amount of data transfer and network delay make it hard to process these applications in a centralized manner [1]. Usually, in most of these applications, user does not need to process all data in the system. Instead only a fraction of data is returned to user according to his interest. This is often done by using ranking queries.

Ranking or top-*k* queries return only the highest ranked *k* tuples according to a user-defined scoring function. Many techniques have been designed to rank distributed database systems introducing many efficient algorithms [3,7–9,11,12]. But, none of these techniques have dealt with distributed uncertain data. However in most of the applications nowadays, data is fuzzy or uncertain. Examples of these applications are distributed sensor networks with imprecise measurements [14,15] and multiple data sources for information integration based on fuzzy similarity scores [16,17].

Ranking query in uncertain database system is finding the highest ranked *k* tuples over all possible instances of the database. These instances are called possible worlds. Each possible world is a certain relation on which any query can be evaluated. The problem is how to combine query result from all the possible worlds without materializing them all. Ranking in distributed uncertain database has two steps, first, ranking tuples at each site, second, getting the overall top-*k* list. Processing this query will consume both time and communication.

[*] Corresponding author. Tel.: +20 1015176039.
E-mail address: yousry@mans.edu.eg (Y.M. AbdulAzeem).

http://dx.doi.org/10.1016/j.datak.2014.05.001
0169-023X/© 2014 Elsevier B.V. All rights reserved.

The motivation for this work is obvious through the study and analysis of the Shipboard Automated Meteorological and Oceanographic System (SAMOS) project [18]. This project aims to improve the quality of meteorological and near-surface oceanographic observations collected in-situ on research vessels (R/Vs) and select volunteer observing ships (VOS). It is a form of data logging system that continuously records navigational (ship position, course, speed, and heading), meteorological (wind speed, air temperature, pressure, moisture, rainfall, and radiation) and surface oceanographic (sea temperature and salinity) parameters while the vessel is at sea.

In the SAMOS project, it can be observed that: (1) Data is naturally distributed because ships and vessels are at geographically separated locations. (2) Ambiguity, errors, imprecise readings and uncertainty are present in the data collected due to measurement conditions and multiple readings. (3) A large amount of data needs to be processed continuously. Answering a query such as "What are the most highly temperature regions during last week?" is a case of uncertain distributed data ranking.

Many recent approaches have dealt with ranking centralized uncertain database [19–22], others with ranking distributed certain database [1,2,4–6,11,13]. However, there are only two approaches that have dealt with ranking in distributed uncertain systems [15,23]. These approaches have had some drawbacks, such as multiple communication phases, extensive calculation of certain values (e,g. Sufficient Boundary in [15]) or lack of data updates monitoring. This paper proposes a new framework that will help in overcoming these drawbacks. The framework utilizes only one or two rounds of communication in the proposed algorithms. So, the amount of transmitted data is minimized for achieving efficient distributed uncertain database ranking.

The rest of this paper is organized as follows. Section 2 briefly overviews the background and the most related work to the proposed framework. Section 3 formally defines the used data model and formulates the distributed uncertain database ranking problem. Section 4 presents the proposed framework and its layers. Section 5 discusses the details of the proposed algorithms. Section 6 validates the proposed algorithms and evaluates their efficiency and performance by a comprehensive experimental study. Finally, the paper is concluded in Section 7.

2. Background

Uncertain data modeling and query processing have received great attention in database applications nowadays. Many models describing uncertain data have been introduced recently in literature. Systems such as, *Trio* [24,25], *MayBMS* [26,27], *MystiQ* [16] and *Orion* [28,29] are examples of systems developed recently providing a variety of uncertain data modeling and processing techniques. *Trio* is a DBMS that adapts tuple-level uncertainty model based on an uncertain relational model called *ULDBs* [30]. *MayBMS* is another uncertain DBMS that adapts possible world semantics. It uses the probabilistic world-set algebra [31]. *MystiQ* is a system that adapts efficient query processing techniques for large probabilistic databases. *Orion* presents query processing and indexing techniques to manage uncertainty over continuous intervals.

Other systems are based on possibility theory such as that proposed by Bosc et al. [32,33]. They presented a framework to deal with the so-called generalized yes–no queries. In this framework, processing obeys three steps scheme: (1) — Pre-processing in order to eliminate the unnecessary attributes. (2) — Evaluation of the query to get the result. (3) — Post-processing aimed at computing the final possibility and certainty [34].

In uncertain database environment, there are many types of queries in two main categories; queries over one-dimensional and queries over multi-dimensional uncertain data [35,36]. Aggregate, Join and Top-*k* queries are one-dimensional queries. Range, Nearest neighbor, Skyline, and Similarity join queries are multi-dimensional queries. Top-*k* query is the query type utilized in the proposed framework. Uncertain database models and the main semantics of the top-k query in uncertain database are discussed in the next two sub-sections.

2.1. Uncertain database models

Many models have been presented to describe uncertain data. An uncertain database can be seen as a discrete probability space with a set of possible instances. These instances are called Possible Worlds (PWs). Each world is simply a set of tuples with certain values chosen. These tuples or values are responsible for specifying the probability of the world. Correlations are considered while specifying each world. For example, two mutually exclusive tuples cannot appear in the same world [24,37].

Consider uncertain database D with number of tuples N. Each tuple t has probability p. D can be spread over a set of PWs $W = w_1$, w_2,.., w_n. The probability of any PW w to appear is $p(w) = \prod_{t \in w} p(t)$, and the total probability of all PWs is: $P(W) = \sum_{w \in W} p(w) = 1$.

The structure of PWs makes it easy to map uncertain data into conventional data which is the most important advantage of PWs model. Uncertain databases can be classified into two main categories; attribute-level and tuple-level uncertainty. Many describing models were introduced inheriting their characteristics from these categories. These categories model database at different levels of uncertainty. In attribute-level, a tuple exists with uncertainty in its attributes, while in tuple-level, the existence of the tuple is uncertain.

2.1.1. Attribute level uncertainty model

In this model, uncertain relation is a table with N tuples. Each tuple has one attribute with uncertain value together with other certain attributes. The uncertain attribute obeys a probability density function describing its value distribution. A possible world in this model is instantiated by taking one independent value for each tuple's uncertain attribute according to the distribution. This model