International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 07
March 2018

Top-k Dominating Queries on Incomplete Data

[1] J Chandrasekhar. M
M.Sc (Computer Science)
Besant theosophical college,madanapalli.

[2] . D.Venkata Siva Reddy
Head dept .of CS
Besant theosophical college,madanapalli

ABSTRACT — The top-k doming (TKD) query returns k objects that exceed the maximum number of objects in a given dataset. It combines the advantages of skype and top k queries, and plays an important role in many decision support applications. There is incomplete data in a wide range of real data sets, due to hardware failure, privacy preservation, data loss, etc. In this paper, for the first time, we conduct a systematic study of TKD queries on incomplete data, which include data that contains some missing dimension values (dimensions). We formalize this problem and suggest a set of effective algorithms to answer TKD queries on incomplete data. Our methods use some new techniques, such as high point pruning, point pruning, and partial evaluation, to enhance query efficiency. Intensive experimental evaluation using real and industrial data sets demonstrates the effectiveness of advanced exploration and evolution methods for the performance of our submitted algorithms.

Indexing - query dominance on top k, incomplete data, processing requests, relationship dominance, algorithm

INTRODUCTION

In the development of a set of d-dimensional objects, the top controlled query k (TKD) classifies the objects o in S based on the number of objects in the controlled S, and returns the k objects from S that dominate the maximum number. Here, object o controls another object o0, ifo is not worse than o0 in all dimensions, which is better than o0 in at least one dimension. Since the TKD query defines the most important things in an intuitive way, it is a powerful decision-making tool for classifying things in many real-life applications. Take the MovieLens sample set from the Movie Suggestion System (http://www.imdb.com/) as an example. MovieLens includes a set of movies with ratings from audiences, where each movie is represented as a multidimensional object with each dimension corresponding to an audience [1, 5] rating. The higher rating usually refers to a better recognition. For example, if we take two o1 = (5, 3, 4) and o2 = (3, 3, 2), we understand that there are three masses that get o1 and o2, where the first audience (o1) and o2 5 and 3 respectively, the second audience (delete the second dimension) registers O1 and O2 as 3, and the third audience (WRT the third dimension) record O1 and O2 at 4 and 2 respectively. Hence, among the three fans, both the first and third audiences believe that O1 is better than O2, and the second audience believes that it is both good. According to dominance hegemony, it can be concluded that o1 controls o2, which means no higher o2 public rates

From o1. Thus, if a film takes control of many other films, it is very likely that the film is fairly common. Note that the MovieLens data set is used extensively in many previous works, including [1], [2]. Intuitively, the TKD query can identify the most popular films of filmmakers. Because of the large application base, the TKD application received a lot of attention from the database community [3], [4], [5], [6], [7], [8], [9]. However, we would like to emphasize that the current work associated with this query focuses solely on incomplete data or data. In the real movie proposal system, it is very common for ratings from some users to be missing, because the user tends to evaluate only the films he knows. As a result, each movie is referred to as a multidimensional object with some blank dimensions (that is, incomplete). Therefore, the set of film ratings is incomplete. As shown in Figure 1, since the a2 audience watches the last three films m2, m3, m4 and not the first m1, Schindler (1993) list, a2 only displays m2, m3, and m4. The data is incomplete and the query for incomplete data has become more important recently. It also stimulated much effort in the database community, including the incomplete data model [10], [11], [12], query evaluation [13], indexing [14], [15] , [2], [16], search for similarities [17], retrieving higher than [18], [19], etc. Here, we would like to point out the difference between incomplete data and unconfirmed data. In this research, as shown in Figure 1, we are dealing with a data object with a missing value (s) as an incomplete data object, following the model entered in [1] which requires zero prior knowledge of the missing dimension value (s). On the other hand, for unconfirmed data, uncertainty is usually expressed from the value (s) of the missing data in terms of probability or derived from some probability distributions. Typically, these probabilities are specified in an original data set. As we have clearly stated in [20], "the case of missing persons is a state of non-existence" of lost data, and its loss means "a fixed state, not a single or potential state". Thus, the incomplete data model (on which our work depends) and the probabilistic concept (used by uncertain data) are two approaches to dealing with lost data.

An intuitive way to support TKD query on incomplete data is to make wise comparisons between a whole set of data to obtain the degree of each object o, the number of objects controlled by o, and to return k objects with the highest scores.

This approach is obviously inefficient because of the very large size of the candidate group and the high cost of the calculation on the basis of brute force.

Disadvantages of existing system:

In the real movie proposal system, it is very common for ratings from some users to be missing, because the user tends to evaluate only the films he knows. As a result, each movie is referred to as a multidimensional object with some blank dimensions (that is, incomplete). Therefore, the set of film ratings is incomplete.

Although a TKD query on incomplete data or data is well studied, processing the TKD query for incomplete data remains a major challenge. This is because the existing techniques can not be applied to handle the TKD query on incomplete data efficiently.

In traditional and unconfirmed databases do not directly apply to incomplete data.

R-tree / aR-treecould is not created on incomplete data directly, because the MBRsof tree does not exist because of the missing dimension values for the data elements.

Proposed System:

We would like to refer to the difference between incomplete data and unconfirmed data.

In this paper, we deal with a data object with a missing value (s) as an incomplete data object, following the submitted form, which requires prior knowledge of the missing value (s).

On the other hand, for unconfirmed data, uncertainty is usually expressed from the value (s) of the missing data in terms of probability or derived from some probability distributions. Typically, these probabilities are specified in an original data set.

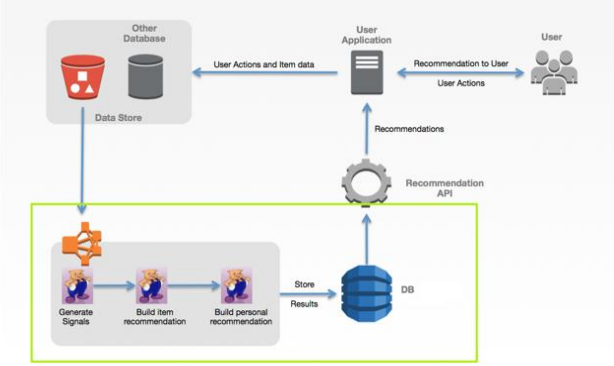
The incomplete data model (on which our work depends) and the probabilistic concept (used by uncertain data) are two approaches to dealing with missing data. It should be noted that, compared to the uncertain data model, the incomplete data model has a significant advantage, ie, it does not require any presumption regarding data association or prior knowledge.

In this paper, we consider an incomplete data set where some objects are missing attribute values in some dimensions, and the TKD query processing problem is examined on incomplete data. Specifically, the TKD query in the full data displays k objects that override the maximum number of objects from an incomplete data set.

Advantages of the proposed system:

To my knowledge, this is the first attempt to troubleshoot a TKD query on incomplete data.We formalize the TKD query problem in the context of incomplete data. To our knowledge, there is no prior work on this problem.We propose efficient algorithms to handle TKD queries on incomplete data, using many new inferences.We offer a flexible alignment strategy with an effective way to select the appropriate number of boxes to reduce the area of the bitmap index for I BIG.We conduct extensive experiments using real and synthetic datasets to demonstrate the effectiveness of inference on the advanced technologies and performance of our proposed algorithms

SYSTEM ARCHITECTURE:



CONCLUSIONS In this research, we investigate the problem of a TKD query on incomplete data where some dimension values are missing. To address this effectively, we first propose the ESB and UBB algorithms, which use new technologies (ie, local Skyband technology).Pruning Top Score Curve) to reduce the search area. In order to further reduce the cost of calculating points, we present a BIG algorithm, which uses higher point pruning, bitmap trim and bitto-based bitmap operations that improves index calculation and increases query performance accordingly. Moreover, in order to effectively exchange space, we propose the IBIG algorithm using bitmap compression technology and binning strategy above BIG, and develop a method to select the appropriate number of boxes. Large experimental results on both real and aggregate datasets create efficient and effective inference and algorithms provided. In the future, we will study how to improve theTKDqueryoverincompletedata quality.

Author Details

Chandrasekhar. M

D.Venkata Siva Reddy