International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 04 Issue 02
February 2017

Fast neighbor search with keywords

A. ANISH
DEPARTMENT OF CSE
TKR COLLEGE OF ENGINEERING AND TECHNOLOGY

Mr. M. Naveen Kumar
ASSISTANT PROFESSOR
DEPARTMENT OF CSE
TKR COLLEGE OF ENGINEERING AND TECHNOLOGY

ABSTRACTS:

Keyword-based search in text-rich multi-dimensional datasets facilitates many novel applications and tools. In this paper, we consider objects that are tagged with keywords and are embedded in a vector space. For these datasets, we study queries that ask for the tightest groups of points satisfying a given set of keywords. We propose a novel method called ProMiSH (Projection and Multi Scale Hashing) that uses random projection and hash-based index structures, and achieves high scalability and speedup. We present an exact and an approximate version of the algorithm. Our experimental results on real and synthetic datasets show that ProMiSH has up to 60 times of speedup over state-of-the-art tree-based techniques.

INTRODUCTION

Objects (e.g., images, chemical compounds, documents, or experts in collaborative networks) are often characterized by a collection of relevant features, and are commonly represented as points in a multi-dimensional feature space. For example, images are represented using color feature vectors, and usually have descriptive text information (e.g., tags or keywords) associated with them. In this paper, we consider multi-dimensional datasets where each data point has a set of keywords. The presence of keywords in feature space allows for the development of new tools to query and explore these multi-dimensional datasets. In this paper, we study nearest keyword set (referred to as NKS) queries on text-rich multi-dimensional datasets. An NKS query is a set of user-provided keywords, and the result of the query may include k sets of data points each of which contains all the query keywords and forms one of the top-k tightest cluster in the multi-dimensional space. Fig. 1 illustrates an NKS query over a set of 2-dimensional data points. Each point is tagged with a set of keywords. For a query Q = fa; b; cg, the set of points f7; 8; 9g contains all the query keywords fa; b; cg and forms the tightest cluster compared with any other set of points covering all the query keywords. Therefore, the set f7; 8; 9g is the top-1 result for the query Q. NKS queries are useful for many applications, such as photo-sharing in social networks, graph pattern search, geolocation search in

GIS systems1 [1], [2], and so on. The following are a few examples.

Client Server

Over view:

With the varied topic in existence in the fields of computers, Client Server is one, which has generated more heat than light, and also more hype than reality. This technology has acquired a certain critical mass attention with its dedication conferences and magazines. Major computer vendors such as IBM and DEC, have declared that Client Servers is their main future market. A survey of DBMS magazine reveled that 76% of its readers were actively looking at the client server solution. The growth in the client server development tools from $200 million in 1992 to more than $1.2 billion in 1996.

What is a Client Server

Two prominent systems in existence are client server and file server systems. It is essential to distinguish between client servers and file server systems. Both provide shared network access to data but the comparison dens there! The file server simply provides a remote disk drive that can be accessed by LAN applications on a file by file basis. The client server offers full relational database services such as SQL-Access, Record modifying, Insert, Delete with full relational integrity backup/ restore performance for high volume of transactions, etc. the client server middleware provides a flexible interface between client and server, who does what, when and to whom.

IMPLEMENTATION

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

The implementation stage involves careful planning, investigation of the existing system and it's constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

MODULES DESCRIPTION:

In this project , Nearest Keyword Set Search in Multi-dimensional Datasets have following modules.

Multi-dimensional data

Nearest Keyword

Indexing

Hashing.

ALGORITHMS:-

ProMiSH:-

results on real and synthetic datasets show that ProMiSH has up to 60 times of speedup over state-of-the-art tree-based techniques. ProMiSH(Projection and Multi Scale Hashing) that uses random projection and hash-based index structures, and achieves high scalability and speedup. ProMiSH (short for Projection and Multi-Scale Hashing) to enable fast processing for NKS queries. In particular, we develop an exact ProMiSH (referred to as ProMiSH-E) that always retrieves the optimal top-k results, and an approximate ProMiSH (referred to as ProMiSHA) that is more efficient in terms of

International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 04 Issue 02
February 2017

time and space, and is able to obtain near-optimal results in practice. ProMiSH-E uses a set of hashtables and inverted indexes to perform a localized search. The hashing technique is inspired by Locality Sensitive Hashing (LSH).

Euclidean Distance:-

The Euclidean distance or Euclidean metric is the "ordinary" (i.e. straight-line) distance between two points in Euclidean space. With this distance, Euclidean space becomes a metric space. The associated norm is called the Euclidean norm. Older literature refers to the metric as Pythagorean metric.

Since Euclidean space with dot product is an inner product space,

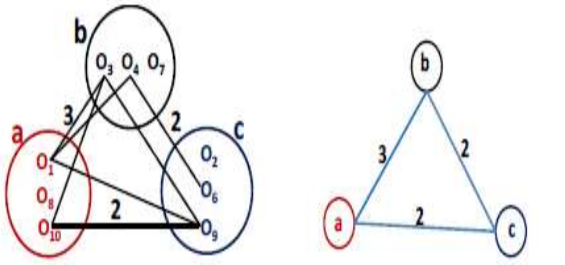
we have

$\|O1z-O2z\|2=|(O1-O2)z|$

$< \|z\|2. \ \|O1-O2\|2$

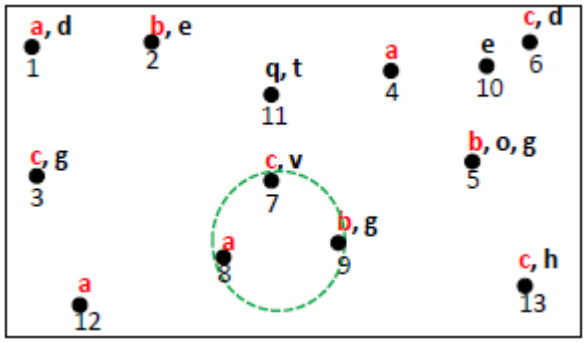
$=\|O1-O2\|2$, since $\|z\|2=1$

$$E(x,y) = \sqrt{\sum_{i=0}^{n}(x_i - y_i)^2}$$

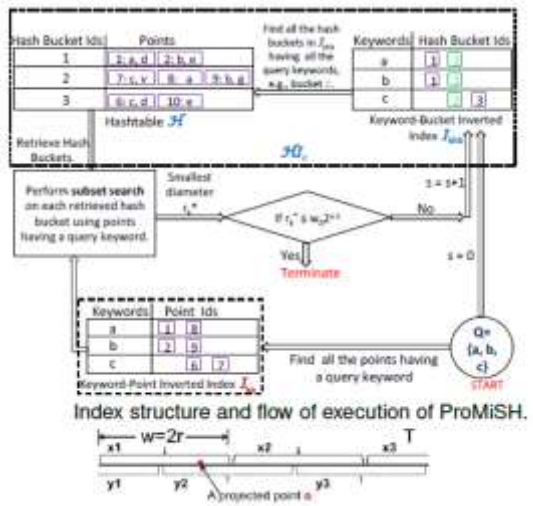


(a) Pairwise inner joins (b) A graph representation

Pruning is a technique in machine learning that reduces the size of decision trees by removing sections of the tree that provide little power to classify instances. Pruning reduces the complexity of the final classifier, and hence improves predictive accuracy by the reduction of overfitting.



ARCHITECTURE DIAGRAMS:



Index structure and flow of execution of ProMiSH.

PRUNING ALGORITHM:-

SYSTEM STUDY

FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- ECONOMICAL FEASIBILITY
- TECHNICAL FEASIBILITY
- SOCIAL FEASIBILITY

SYSTEM TESTING

TESTING METHODOLOGIES

The following are the Testing Methodologies:

- **Unit Testing.**
- **Integration Testing.**
- **User Acceptance Testing.**
- **Output Testing.**
- **Validation Testing.**

CONCLUSIONS

In this paper, we proposed solutions to the problem of top-k nearest keyword set search in multi-dimensional datasets. We proposed a novel index called ProMiSH based on random projections and hashing. Based on this index, we developed ProMiSH-E that finds an optimal subset of points and ProMiSH-A that searches near-optimal results with better efficiency. Our empirical results show that ProMiSH is faster than state-of-the-art tree-based techniques, with multiple orders of magnitude performance improvement. Moreover, our techniques scale well with both real and synthetic datasets. Ranking functions. In the future, we plan to explore other scoring schemes for ranking the result sets. In one scheme, we may assign weights to the keywords of a point by using techniques like tf-idf. Then, each group of points can be scored based on distance between points and weights of keywords. Furthermore, the criteria of a result containing all the keywords can be relaxed to generate results having only a subset of the query keywords. Disk extension. We plan to explore the extension of ProMiSH to disk. ProMiSH-E sequentially reads only required buckets from Ikp to find points containing at least one query keyword. Therefore, Ikp can be stored on disk using a directory-file structure. We can create a directory for Ikp. Each bucket of Ikp will be stored in a separate file named after its key in the directory. Moreover, ProMiSH-E sequentially probes HI data structures starting at the smallest scale to generate the candidate point ids for the subset search, and it reads only required buckets from the hash table and the inverted

index of a HI structure. Therefore, all the hash tables and the inverted indexes of HI can again be stored using a similar directory file structure as Ikp, and all the points in the dataset can be indexed into a B+-Tree [36] using their ids and stored on the disk. In this way, subset search can retrieve the points from the disk using B+-Tree for exploring the final set of results

REFERENCES

[1] W. Li and C. X. Chen, "Efficient data modeling and querying system for multi-dimensional spatial data," in GIS, 2008, pp. 58:1–58:4.

[2] D. Zhang, B. C. Ooi, and A. K. H. Tung, "Locating mapped resources in web 2.0," in ICDE, 2010, pp. 521–532.

[3] V. Singh, S. Venkatesha, and A. K. Singh, "Geo-clustering of images with missing geotags," in GRC, 2010, pp. 420–425.

[4] V. Singh, A. Bhattacharya, and A. K. Singh, "Querying spatial patterns," in EDBT, 2010, pp. 418–429.

[5] J. Bourgain, "On lipschitz embedding of finite metric spaces in Hilbert space," Israel J. Math., vol. 52, pp. 46–52, 1985.

[6] H. He and A. K. Singh, "Graphrank: Statistical modeling and mining of significant subgraphs in the feature space," in ICDM, 2006, pp. 885–890.

[7] X. Cao, G. Cong, C. S. Jensen, and B. C. Ooi, "Collective spatial keyword querying," in SIGMOD, 2011.

[8] C. Long, R. C.-W. Wong, K. Wang, and A. W.-C. Fu, "Collective spatial keyword queries: a distance owner-driven approach," in SIGMOD, 2013.

[9] D. Zhang, Y. M. Chee, A. Mondal, A. K. H. Tung, and M. Kitsuregawa, "Keyword search in spatial databases: Towards searching by document," in ICDE, 2009, pp. 688–699.

[10] M. Datar, N. Immorlica, P. Indyk, and V. S. Mirrokni, "Locality-sensitive hashing scheme based on p-stable distributions," in SCG, 2004.

[11] Y. Zhou, X. Xie, C. Wang, Y. Gong, and W.-Y. Ma, "Hybrid index structures for location-based web search," in CIKM, 2005.

[12] R. Hariharan, B. Hore, C. Li, and S. Mehrotra, "Processing spatialkeyword (SK) queries in geographic information retrieval (GIR) systems," in SSDBM, 2007.

[13] S. Vaid, C. B. Jones, H. Joho, and M. Sanderson, "Spatio-textual indexing for geographical search on the web," in SSTD, 2005.

[14] A. Khodaei, C. Shahabi, and C. Li, "Hybrid indexing and seamless ranking of spatial and textual features of web documents," in DEXA, 2010, pp. 450–466.

[15] A. Guttman, "R-trees: A dynamic index structure for spatial searching," in ACM SIGMOD, 1984, pp. 47–57.