RESEARCH ARTICLE OPEN ACCESS

Enhanced Cloud-Based Approximate Nearest Neighbour Search on High -Dimensional Data

CH. Nikitha Reddy,P.V.Shilohini Angel,P. Hrithika Malkan,V. Nikitha , Mr.K. Anil Kumar

Information Technology, Guru Nanak Institutions Technical Campus, Ibrahimpatnam

ABSTRACT

As one fundamental data-mining problem, ANN (approximate nearest neighbour search) is widely used in many industries, including computer vision, information retrieval, and recommendation systems. LSH (Local sensitive hashing) is one of the most popular hash-based approaches to solve ANN problems. However, the efficiency of operating LSH needs to be improved! as the operations of LSH often involve resource-consuming matrix operations and high-dimensional large-scale datasets. Meanwhile; for resource-constrained deices, this problem becomes more serious. One way to handle this problem is to, outsource the heavy computing of high-dimensional, large-scale data to cloud servers. However, when a cloud server responsible for computing tasks is untrustworthy, some security issues may arise. In this study, we; proposed a cloud server-aided LSH scheme and the application model.

Keywords: ANN (approximate nearest neighbour search) LSH (Local sensitive hashing)

I. INTRODUCTION

This As illustrated in the computations of ANN (approximate nearest neighbour) search on high-dimensional dataset are basic in manly applications. The purpose of ANN is to find the approximate nearest neighbour vectors in a given dataset and because of the better performance of ANN on high dimensional large-scale data than other methods; ANN has been applied in various high-dimensionallarge-scale fields, including product recommendation image retrieval clustering etc. However, existing algorithms for ANN can be efficient when the data dimensionality is small even if the data dimensionality is large. As the data dimension increases, these exact algorithms may even be less efficient than brute force linear scanning due to dimensionality disaster. This mechanism aims to enable resource-constrained devices to securely and efficiently perform the ANN task while preserving the privacy of the client's information and providing a lightweight verification mechanism with a high level of certainty. By addressing these issues, we aim to enhance the applicability and performance of ANN in various high dimensional large-scale fields,promoting.

II. EXISTING SYSTEM

ANN (approximately near nearest neighbor search) is widely used in many industries including the computer vision, information retrieval, and recommendation system. LSH (Local sensitive hashing) is one of the most popular hash-based approaches to solving ANN problems. However, the efficiency of operating LSH needs to be improved, as the operations of LSH often involve the resource-consuming matrix operations and high-dimensional large-scale datasets. Meanwhile, for resource-constrained devices, this particular problem becomes way more serious. One of the ways to handle this particular problem is to outsource the heavy computing of the high-dimensional large-scale data to cloud servers. However, when a cloud server that is responsible for computing the tasks is untrustworthy, some of the security issues may arise. However, existing algorithms for ANN can be efficient when the data dimensionality is small even if the data dimensionality is large.

A. LITERATURE SURVEY:

Title: Efficient locality-sensitive hashing over high-dimensional streaming data, Neural Comput. Appl. Author: H. Wang, C. Yang, X. Zhang, and X. Gao.

Description:

Approximate nearest neighbour (ANN) search in high-dimensional spaces is fundamental in many applications. Locality-sensitive hashing (LSH) is a well-known methodology to solve the ANN problem. Existing LSH-based ANN solutions typically employ a large number of individual indexes that are optimized for searching efficiency and to find the results quality. Firstly, we use the write-friendly LSM-trees to store the LSH projections to facilitate efficient updates. Secondly, we develop a novel estimation scheme to estimate the number of required LSH functions, with which the disk storage and access costs are effectively reduced. Third, we exploit both the collision number and the projection distance to improve the efficiency of candidate selection, improving the search performance with theoretical guarantees on the result quality. Experiments on four real-world datasets show that our proposal outperforms the state-of-the-art schemes.

Title: Similarity search in high dimensions via hashing, Author: A. Gionis, P. Indyk, and R. Motwani.

Description:

The nearest- or near-neighbour query problems arise in a large variety of database and genome databases. Unfortunately, all known techniques for solving this problem fall prey to the "curse of dimensionality." That is, the data structures scale poorly with data dimensionality; in fact, when the number of dimensions exceeds 10 to 20, searching in k-d trees and related structures involve the inspection of a large fraction of the database, thereby doing no better than brute-force linear search. In this paper, we examine a novel scheme for approximate similarity search based on hashing. The basic idea is to hash the points. A similarity search problem involves a collection of objects (e.g., documents, images) that are characterized by a collection of relevant features and represented as points in a high-dimensional attribute space; given queries in the form of points in this space, we are required to find the nearest (most similar) object to the query. The problem is of major importance to a variety of applications; some examples are: data compression databases, data mining information retrieval, image and video databases, machine learning pattern recognition, and, statistics and data analysis. Typically, the features of the objects of interest are represented as points in and a distance metric is used to measure similarity of objects.

Title: Novel secure outsourcing of modular inversion for arbitrary and variable modulus.
Author: C. Tian, J. Yu, H. Zhang, H. Xue, C. Wang, and K. Ren.

Description:

In cryptography and algorithmic number theory, modular inversion is viewed as one of the most common and time-consuming operations. It is hard to be directly accomplished on resource-constrained clients (e.g., mobile devices and IC cards) since modular inversion involves a great number of operations on large numbers in practice. To address the above problem, this paper proposes a novel unimodular matrix transformation technique to realize secure outsourcing of modular inversion. First, to the best of our knowledge, it is the first secure outsourcing computation algorithm that supports arbitrary and variable moduli, which eliminates the restriction in previous work that the protected modulus has to be a fixed composite number. Second, our algorithm is based on the single untrusted program model, which avoids the non-collusion assumption between multiple servers. Third, for each given instance of modular inversion, it only needs one round of interaction between the client and the cloud server, and enables the client to verify the correctness of the results returned from the cloud server with the (optimal) probability. At last, two important and helpful applications of our algorithms, the outsourced implementations of the key generation of the RSA algorithm and the Chinese Reminder Theorem are given.

III. PROPOSED SYSTEM

In this paper, we proposed a cloud server-aided LSH scheme and the application model. This scheme can perform the LSH efficiently with the help of a cloud server and guarantee the privacy of the client's information.

- To identify the improper behavior of the cloud server, we also provide a verification method to check the results returned from the cloud server.
- Meanwhile, for the implementation of this scheme on resource-constrained devices, we proposed a model for the real application of this scheme. To verify the efficiency and correctness of the proposed scheme, theoretical analysis and experiments are conducted.

A. SYSTEM ARCHITECTURE :

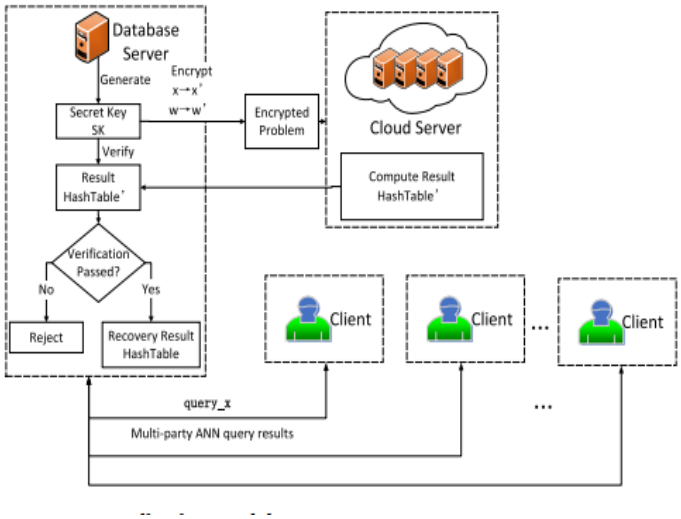

Fig. 1. In the process of outsourcing the LSH algorithm to the cloud server

Description:

In This Project the main reason is The proposed application model of outsourced LSH is shown in Fig. 1. In the process of outsourcing the LSH algorithm to the cloud server, the database server will firstly generate a Secret Key (SK). Then, using the SK to obscure the database server's data, from in which x is the raw data, w is the project matrix. Then, the obscured data will transfer to the cloud server. In turn, the cloud server will calculate multiple Hash Table by multiple hash functions, and then return them to the database server. The database server will verify the correctness of the Hash Table through the proposed verification mechanism. After passing the verification, the database server will recover the final Hash Table through SK. Otherwise, the database server will reject the Hash Table.

B. SYSTEM MODULES:

Modules:

1.Login (User)
2. Upload
3. Cloud
4.CSP

1. User:

In this module, we design the windows for the project. These windows are used for secure login for all users. To connect with the server user must give their username and password then only they can be able to connect to the server. If the user already exists directly and can log into the server else user must register their details such as username, password, and email ID, into the server. The server will create an account for the entire user to maintain the upload and download rate. The name will be set as the user ID. Logging in is usually used to enter a specific page.

2. Upload:
This is the first module is users can register and log in With the help of CSP approval. After login, there is an option to Upload. In that Upload bar, we can upload data by submitting that button we can see the data is stored in the Database and data is converted into Hash values and generating one secret key.
3. CSP :
This is the third module of this project. In this module, CSP will manage the user Request which means when the user requests a product with his price the Cloud Service Provider will decide whether the status of the user request is Approved or Not.
4. Cloud:
This is the fourth module in this project. This module also has login only and this module will show System Requirements and we will see the users in this module and stored in the Database. This is the final module in this project.

EXPLANATION:

Entity-Relationship Model (ERM) is an abstract and conceptual representation of data. Entity-relationship modeling is a database modeling method, used to produce a type of conceptual schema or semantic data model of a system, often a relational database.

IV. CONCLUSION

In this paper, we proposed a secure and verifiable scheme to outsource the classical LSH algorithm for the ANN query problems on resource-constrained devices. The scheme avoids the leakage of input data by obscuring the client data. At the same time, the correctness of the computing results of the cloud server can be verified with 100% possibility through the proposed verification mechanism. To verify the efficiency and correctness of our scheme, we conducted experiments based on the CIFAR-10 dataset and described the experimental results. With the rise of machine learning, such as individual phones and edge devices, need to do complex machine learning operations. The performance of personal devices and edge devices is getting higher and higher, and the volume and dimension of processed data are also increasing. Therefore cloud-aided machine learning is an idea to solve this problem. We can outsource part of the complex computations of machine learning algorithms to enable resource-constrained devices to complete complex machine learning tasks based on ensuring the privacy of client's data. This not only requires cloud servers to be able to run part of machine learning algorithms' computations but also needs to run on encrypted data.

V. REFERENCES

[1] H. Wang, C. Yang, X. Zhang, and X. Gao, ''Efficient locality- sensitive hashing over high-dimensional streaming data,'' Neural Comput. Appl., vol. 35, no. 5, pp. 3753–3766, Feb. 2023, doi: 10.1007/s00521-020-05336-1.

[2] J. Suchal and P. Návrat, ''Full text search engine as scalable k-nearest neighbor recommendation system,'' in Artificial Intelligence in Theory and Practice III (IFIP Advances in Information and Communication Technology), vol. 331. AICT, 2010, pp. 165–173, doi: 10.1007/978-3-642-15286-3_16.

[3] G. Giacinto, ''A nearest-neighbor approach to relevance feedback in content based image retrieval,'' in Proc. 6th ACM Int. Conf. Image Video Retr. New York, NY, USA: ACM Press, Jul. 2007, pp. 456–463, doi: 10.1145/1282280.1282347.

[4] T. Liu, C. Rosenberg, and H. Rowley, ''Clustering billions of images with large scale nearest neighbor search,'' in Proc. IEEE Workshop Appl. Comput. Vis. (WACV), Feb. 2007, p. 28, doi: 10.1109/WACV.2007.18.

[5] J. L. Bentley, ''Multidimensional binary search trees used for associative searching,'' Commun. ACM, vol. 18, no. 9, pp. 509–517, Sep. 1975, doi: 10.1145/361002.361007.

[6] K. L. Cheung and A. W.-C. Fu, ''Enhanced nearest neighbour search on the R-tree,'' ACM SIGMOD Rec., vol. 27, no. 3, pp. 16–21, Sep. 1998, doi: 10.1145/290593.290596.

[7] R. Weber, H. J. Schek, and S. Blott, ''A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces,'' in Proc. 24rd Int. Conf. Very Large Data Bases, 1998, pp. 194–205.

[8] O. Jafari and P. Nagarkar, ''Experimental analysis of locality sensitive hashing techniques for high-dimensional approximate nearest neighbor searches,'' in Proc. Australas.

Database Conf., 2021, pp. 62–73, doi: 10.1007/978-3-030-69377-0_6.

[9] K. Zhao, H. Lu, and J. Mei, ‘‘Locality preserving hashing,’’ in Proc. AAAI Conf. Artif. Intell., vol. 28, no. 1, Jun. 2014, doi: 10.1609/aaai.v28i1.9133.

[10] A. Gionis, P. Indyk, and R. Motwani, ‘‘Similarity search in high dimensions via hashing,’’ in Proc. 25th Int. Conf. Very Large Data Bases, 1999, pp. 518–529, Accessed: Oct. 8, 2022. [Online]. Available: http://dl.acm.org/citation.cfm?id=645925.671516

[11] Y. Tao, K. Yi, C. Sheng, and P. Kalnis, ‘‘Efficient and accurate nearest neighbor and closest pair search in high-dimensional space,’’ ACM Trans. Database Syst., vol

[12] W. Liu, H. Wang, Y. Zhang, W. Wang, and L. Qin, ‘‘I-LSH: I/O efficient c-approximate nearest neighbor search in high-dimensional space,’’ in Proc. IEEE 35th Int. Conf. Data Eng. (ICDE), Apr. 2019, pp. 1670–1673, doi: 10.1109/ICDE.2019.00169.