International Journal of Research in Engineering, Science and Management
Volume-1, Issue-5, May 2018
www.ijresm.com

Privacy Preserving Top-k Spatial Keyword in Cloud Computing Using Biometric Identification

H. S. Mangala[1], B. N. Veerappa[2]

[1]*M. Tech. Student, Department of Computer Science & Engineering, UBDTCOE, Devanagere, India*
[2]*Associate Professor, Department of Computer Science & Engineering, UBDTCOE, Devanagere, India*

***Abstract*—With the rapid development of location-based services in mobile Internet, spatial keyword queries have been widely employed in various real-life applications in recent years. To realize the great flexibility and cost savings, more and more data owners are motivated to outsource their spatio-textual data services to the cloud. However, directly outsourcing such services to the untrusted cloud may arise serious privacy concerns. Existing studies primarily focus on the design of privacy-preserving schemes for either spatial or keyword queries, and they cannot be applied to solve the privacy preserving spatial keyword query problem. To address this problem, we present a novel privacy-preserving top-k spatial keyword query scheme. Biometric identification is a reliable and convenient way of identifying individuals. The widespread adoption of biometric identification requires solid privacy protection against possible misuse, loss, or theft of biometric data. Existing techniques for privacy-preserving biometric identification primarily rely on conventional cryptographic primitives.**

Index Terms— Biometric identification, Cloud computing, Location-based Services, Privacy, Spatial keyword query

I. Introduction

With the increasing popularity of location-based services in mobile Internet, spatial keyword queries have drawn growing interest from both the industrial and academic communities in recent years. Given a set of spatiotextual objects (e.g., points of interest) and a query with a location and a set of keywords, a top-k spatial keyword query finds k objects that are most relevant to the query in terms of both spatial proximity and textual relevancy which has been widely used in real- life applications such as Google Maps and Foursquare. To realize the great flexibility and cost savings, more and more data owners are motivated to outsource their data services (including data, indices, querying algorithms, etc.) to the cloud. For example, Yelp1, acting as a data owner, outsources its data and services to the cloud provided by Amazon 2 for saving cost. However, directly outsourcing such service to cloud may arise serious privacy concerns. On one hand, the spatiotextual data may involve some private data objects whose locations or textual descriptions cannot be learned by any third parties including the cloud provider. Moreover, it requires human and financial resources to collect the spatiotextual objects, which can be regarded as business secrets to competitors, and it prohibits any unauthorized parties to grab the data. On the other hand, if the locations and the query keywords in the spatial keyword queries of data users are acquired illegally by the untrusted third parties, the travel habits or the query manners will be analyzed or even utilized by some potential attackers. Thus, it is of great significance to study the privacy-preserving scheme for top- k spatial keyword queries in outsourced environments.

Biometric identification is a task of searching biometric collections to find the best match for a biometric trait, and further checking whether these two traits are from the same individual. Due to the universality, uniqueness and permanence of biometric data, biometric identification has been increasingly adopted as an extremely reliable way of identifying and authenticating individuals in various systems. It is believed to be a promising replacement to conventional identification approaches which are based on passwords, identification cards, etc. Despite the proliferation of biometric identification, there are also increasing concerns on its associated privacy and legal issues since biometric data is highly sensitive and is impossible to be revoked and replaced once leaked. Appropriate security and privacy protection mechanism shall be in place to defend against deliberate or inadvertent disclosure or misuse of biometric data. Ideally, a biometric identification process shall not disclose any sensitive information than the result about whether or not a given biometric trail can be identified. If the storage server is not trustworthy, it shall also assure that sensitive biometric data is not disclosed to the untrusted server.

However, realizing such a privacy-preserving biometric identification is challenging considering the requirements of practical systems on security, efficiency and system scalability. In particular, identification of each given biometric trait involves search over the entire biometric database, i.e., the biometric trait shall be compared with all the items stored in the database one by one to identify the most similar one.

In practical large-scale applications, this kind of search may introduce a tremendous burden to the system considering that both the database size and the number of simultaneous identification requests can be large. Several solutions have been proposed for privacy-preserving biometric identification, nevertheless, most of them suffer from the above efficiency and scalability issues. In this paper, we address this open problem by incorporating the computing power of cloud. Considering an organization owns a large biometric database and a client associated with a piece of candidate biometric trait, we design our scheme with the following idea: the database owner first encrypts the whole database and uploads it to the cloud; when a client with a candidate biometric trait needs to be identified, the trait will be encrypted and sent to the cloud; cloud servers

then execute most of the operations pertaining to identification process over the ciphertexts and return the index of the matching ciphtertext to the owner, with which the final result can be efficiently decrypted. During these processes, cloud servers and the client learn no privacy data from biometric database even if they collude with each other. Through offloading most computation tasks to the cloud, our scheme makes the real-time computational/communication complexity for both database owner and clients minimal. The system is highly scalable thanks to the parallel processing power of the cloud.

Back-propagation is an effective method for learning neural networks and has been widely used in various applications. The accuracy of the learning result, despite other facts, is highly affected by the volume of high quality data used for learning. As compared to learning with only local data set, collaborative learning improves the learning accuracy by incorporating more data sets into the learning process the participating parties carry out learning not only on their own data sets, but also on others' data sets. With the recent remarkable growth of new computing infrastructures such as Cloud Computing, it has been more convenient than ever for users across the Internet, who may not even know each other, to conduct joint/collaborative learning through the shared infrastructure. Despite the potential benefits, one crucial issue pertaining to the Internet-wide collaborative neural network learning is the protection of data privacy for each participant. In particular, the participants from different trust domains may not want to disclose their private data sets, which may contain privacy or proprietary information, to anybody else. In applications such as healthcare, disclosure of sensitive data, e.g., protected health information, is not only a privacy issue but of legal concerns according to the privacy rules such as Health Insurance Probability and Accountability Act(HIPAA). In order to embrace the Internet wide collaborative learning, it is imperative to provide a solution that allows the participants, who lack mutual trust, to conduct neural network learning jointly without disclosing their respective private data sets. Preferably, the solution shall be efficient and scalable enough to support an arbitrary number of participants, each possessing arbitrarily partitioned data sets.

Challenges, theoretically, secure multi-party computation (SMC) can be used to solve problems of this kind. But the extremely high computation and communication complexity of SMC, due to the circuit size, usually makes it far from practical even in the two-party case. In order to provide practical solutions for privacy preserving back-propagation neural (BPN) network learning, three main challenges need to be met simultaneously:1) To protect each participant's private dataset and intermediate results generated during the BPN network learning process, it requires secure computation of various operations, e.g. addition, scalar product and the nonlinear sigmoid function, which are needed by the BPN network algorithm; 2) To ensure the practicality of the proposed solution, the computation/communication cost introduced to each participant shall be affordable. In order to accommodate a large range of collaborative learning, the proposed solution shall consider system scalability. In particular, it shall be able to support an arbitrary number of participants without introducing tremendous computation/communication costs to each participant. 3) For collaborative training, the training data sets may be owned by different parties and partitioned in arbitrary ways rather than a single way of partition.

Existing studies primarily focus on the design of privacy preserving schemes for either spatial or keyword queries. They cannot be applied to solve the privacy-preserving top- k spatial keyword query problem. Even though the spatial keyword queries are performed by simply combining such separate schemes, available queries cannot be provided due to the efficiency and validity, since both text relevancy and spatial proximity are exploited for search space pruning and results ranking. Therefore, it calls for effective methods to efficiently process privacy-preserving top-k spatial keyword queries. To this end, we first define the problem of the top- k spatial keyword query over outsourced spatio-textual data in cloud. We then present a brand new scheme for achieving privacy-preserving top-k spatial keyword queries (PkSKQ). Specifically, in our scheme, a secure index based on existing tree-based index [1] is built to facilitate PkSKQ. In this index, to achieve a unified encryption, the spatial and textual data (i.e., coordinates and keyword weights) are converted into vectors and encrypted by an enhanced version of Asymmetric Scalar-product-Preserving Encryption (ASPE), namely ASPE with Noise (ASPEN). We prove that ASPEN is resilient to chosen-plaintext attack and knownplaintext attack. To search with the secure index, a basic operation is to compute the similarity between a query point and a tree node in the secure index. However, since the coordinates and keywords of the query point and the tree node are encrypted, we cannot directly compute such similarity. To solve this problem, we develop two techniques, anchor-based position determination and position distinguished trapdoor generation. In particular, by adding auxiliary points into the query point and each tree node, the anchor-based position determination method allows to determine the positional relation between them under encryption; and for the position-distinguished trapdoor generation method, to facilitate the similarity computations between the query point and tree nodes, it generates query vectors corresponding to all the possible positional relations between the query point and tree nodes in the trapdoor generation process. In this way, the similarity computations between the query point and tree nodes can be performed without privacy breaches. Furthermore, to satisfy the performance requirements for handling large-scale spatio-textual data, we propose a keyword-based secure pruning method to improve query performance during query processing. In this pruning method, encrypted bloom filters are devised for pruning the tree nodes which do not contain query keywords. To summarize, our contributions are as follows: To the best of our knowledge, this is the first attempt to define and solve the privacy-preserving top-k spatial keyword query problem in untrusted cloud environments. We propose a new privacy-preserving scheme for top-k spatial keyword queries, which is referred to as PkSKQ. In particular, we devise a secure index to facilitate the privacy-preserving top-k query, where spatial and textual data are encrypted in a unified way using ASPEN. To search with the secure index, we propose two techniques, anchor-based position determination and position-

distinguished trapdoor generation, for the similarity computations between the query point and tree nodes under encryption.

- We further propose a keyword-based secure pruning method to improve query performance on large-scale spatio-textual data during query processing.
- Thorough analysis shows the validity and security of our scheme, where it is proven to be resilient to chosen-plaintext attack and known-plaintext attack. Extensive experimental results on real datasets further demonstrate our scheme can achieve high efficiency and good scalability. The rest of the paper is organized as follows. Section 2 first describes the privacy-preserving top-k spatial keyword query problem over outsourced data in cloud, and then introduces our encryption method. Section 3 presents our privacy-preserving top-k spatial keyword query scheme. In Section 4, we analyse the validity and security of the proposed scheme. In experimental evaluation is presented. Section 6 reviews related works on spatial keyword queries and privacy-preserving schemes for spatial and keyword queries.

II. Literature Survey

Barni et al. [1] proposed a privacy-preserving fingerprint authentication scheme, which is purely based on homomorphic encryption. Instead of computing the global minimum value between candidate fingerprint and those in database as our scheme did, their work outputs all matched indexes for a specific threshold. However, due to linearly increasing cost, their scheme is limited by the size of database and number of requests submitted at the same time. For a 5MB database and one request, it takes 16 seconds and uses 9.11MB bandwidth for the identification. In our scheme, an identification request over 10GB can be achieved by 4.31 seconds.

A.M. Bazen and S. H. Gerez [4], a practical privacy-preserving face identification system (SCiFI) proposed by Osadchy et. al. which is a component-based face identification protocol. By utilizing a secure Hamming distance and secure minimum algorithms that based on the optimized additive homomorphic encryption and oblivious transfer, this technique is suitable for two entities to compare their few number of biometric data online. Nevertheless, for conducting identification over a relative large database, the linearly increasing computational cost still becomes the limitation of the scheme. As the authors mentioned in their work, this scheme needs 31 seconds to perform an online identification for a database of 100 records (about 11KB), while no bandwidth consumption is reported. Therefore, this scheme is far away from practical secure biometric identification for large database with multiple simultaneous requests.

Huang et al.[2] proposed a novel protocol that greatly improve the identification efficiency without disclosing any private biometric data. In their protocol, by using an improved Euclidean-Distance protocol based on the ciphertexts packing techniques and a novel garbled circuits design for minimum searching, a single identification over 1GB database can be processed in 18 seconds with 7.6MB bandwidth cost. However, their work is a client leading system, which needs the transmission of whole encrypted database to from the sever side to the client side for each identification. With the growth of database size and simultaneous requests number, a linearly increasing cost in both computation and communication will be introduced to the system. For example, when 100 identification requests submitted simultaneously for 5GB database, the system can has a bandwidth consuming over 3.5GB and a computational cost over 90 seconds after the large data transmission, which is obviously impractical to handle.

Y. Huang, L. Malka, D. Evans, and J. Katz [2] requires powerful hardware on every client, which greatly increase the cost of system deployment and limit the usage scenarios. For instance, the equipment used to collect a candidate biometric trait(fingerprint, faces, etc) in many scenarios are always lightweight and has high mobility, the high hardware requirement makes can hardly be adopted in these scenarios.

W. K. Wong, D. W. Cheung, B. Kao, and N. Mamoulis.[3], Wong et.al. Introduced a secure kNN computation scheme on encrypted databases. This scheme enables untrusted server to compare the distance between different points over ciphertexts, which can be partially used to construct privacy-preserving biometric identification schemes. However, there are two assumptions for the attack model in this scheme, which can lead to the reveal of the whole database once any of them is failed: 1) the attacker cannot disguise as client and submit requests to the database; 2) the attacker cannot access any encrypted data and its corresponding plaintext. To enhance the security level of previous privacy-preserving biometric identification schemes, our scheme omits these assumptions by introducing randomness to every data in the database and can handle more powerful attackers.

III. Privacy-Preserving Top-K Spatial Keyword Query Scheme

In this section, we first give an overview of the privacy preserving top-k spatial keyword query (PkSKQ) scheme, and then describe this scheme in details.

A. Overview

As the system model described in Section-II, the data owner first builds a regular IR-tree for indexing spatio-textual data. To protect the privacy of the IR-tree, it is encrypted by a unified encryption method. In particular, for the entries (i.e., MBRs and objects) in the IR-tree nodes, the coordinates and weights of keywords are converted into one vector and encrypted using ASPEN, while the parent-children relationships are preserved. The details of the unified encryption for the secure index are presented in Section-B.

To enable the similarity computations between the query point and MBRs under encryption, some auxiliary points need to be generated by the data owner before oursourcing the index. Specifically, to search with the IR-tree, as a basic operation, the computation of the distance between a query point and an MBR relies on the positional relation between them, which cannot be directly determined under encryption. To this end, an anchor-based position determination method is proposed to help determine the relations, where auxiliary points, called anchors, are generated for each MBR and outsourced as a part of index by the data owner.

B. Unified Encryption for Secure Index Construction

To enable top-k spatial keyword queries over encrypted spatio-textual data in the cloud, we design a unified encryption over the spatio-textual data to facilitate the computations of the similarities between query points and entries in the tree nodes of the secure index. To achieve the unified encryption, for the IR-tree, the coordinates and the keyword weights of MBRs and objects are transformed into one vector and encrypted using ASPEN, while the parent-children relationships are not encrypted. Exploiting the scalar-product preserving property of ASPEN, the similarities can be computed in the way of the scalar product between vectors. Particularly, for MBRs and objects, the inverted files are first transformed and extended into textual vectors of the same length to facilitate the computation of the textual relevancy.

C. Keyword-based Secure Pruning

Our scheme can already achieve effective spatial keyword queries without privacy breaches. However, since the current scheme cannot avoid visiting the nodes of the index (MBRs or objects) which do not contain any query keywords, before retrieving the top-k results, all the nodes need to be accessed.

A bloom filter is an efficient and storage-saving methodology to store information about the existence of an item in a dataset. It is an m-bit array B initialized to zero, and requests a set of _ independent hash functions Fi (•). Each function produces a uniformly distributed bit location in the range [0; m − 1] for a given item. To map an item T to the filter, calculate b1 = F1(T); b2 = F2(T); :::b_ = F_ (T), and set B[bi] = 1, where i [1; _]. To check whether T is in the dataset, bi (i from 1 to_) are calculated and bits B[bi] are examined. If all are 1, the item probably exists; otherwise, it does not exist. Accordingly, for the data user, he/she first sends the query keywords {qk1; qk2; :::; qkm} to the data owner to acquire the mapped bloom filters. These filters are sent back as tuples,

Tu1 =⟨F1(qk1); F2(qk1):::F_ (qk1)⟩
Tu2 =⟨F1(qk2); F2(qk2):::F_ (qk2)⟩
:::
Tum =⟨F1(qkm); F2(qkm); ::::; F_ (qkm)⟩;

each of which is corresponding to one query keyword. The secret keys K and p are also acquired by the user from the data owner. For security reasons, each Fi(qkj) is encrypted as EnK(Fi(qkj)), where $1 \le i \le \_$ and $1 \le j \le m$. Note that the corresponding relation Node(•) between bits of Yz and nodes of the index I is known by the data user.

To perform the pruning, the data user first sends all the encrypted bloom filters EnK(Fi(qkj)) ($1 \le i \le \_$, $1 \le j \le m$) as a sequence to the cloud. Once the cloud receives them, it retrieves the encrypted identifiers EnK(z). If EnK(Fi(qkj)) is equal to EnK(z), the cloud sends back its corresponding string Enp(Yz) to the data user. Then, the data user decrypts the received strings Enp(Yz) as Yz. For each tuple Tuj of filters, the data user calculates Y j = YF1(qkj)&YF2(qkj)&:::&YF_ (qkj) ($1 \le j \le m$) and bY = Y 1|Y 2|:::|Y m. Here, & and | represent the conjunction operation and disjunction operation in binary system respectively. After that, the data user examines which bits in bY are equal to 1 to produce a candidate node list. According to the corresponding relation between bits of Yz and nodes, if bY [i] is 1, the corresponding node identifier Node (i) is added to the candidate list LN. Finally, the data user sends LN to the cloud. The keyword-based secure pruning method is summarized in Algorithm-1.

Algorithm-1: Keyword-based Secure Pruning Method

Cloud Server:
Input: encrypted sequence of bloom filters EnK(Fi(qkj)) (1 _ i _ _ , 1 _ j _ m);
Output: a sequence of bit strings BS;
Procedure:
1: generate a queue BS;
2: for j from 1 to m do
3: for i from 1 to _ do
4: if EnK(Fi(qkj)) == EnK(z) then
5: BS.Enqueue(Enp(Yz));
return BS and send to Data User;

Data User:
Input: BS, secret keys p;
Output: the candidate node list LN;

Procedure:
1: Decrypt each Enp(Yz) in BS as Yz;
2: Y j 1
3: bY 0
4: for j from 1 to m do
5: for i from 1 to _ do
6: Y j = Y j&YFi(qkj);
7: bY = bY jY j ;
8: for each bY [i] do
9: if bY [i] == 1 then
10:
LN.Insert (Node(i));
return LN and send to Cloud Server;

D. Query Processing of PkSKQ

To retrieve the top-k objects, we summarize the overall query processing of the PkSKQ scheme. Through the keyword-based secure pruning method, the data user can first compute the candidate node list LN. Together with LN, the data user sends the query vectors (Qi:v and Q:a) to the cloud. To perform PkSKQ, the cloud server first checks whether the next visiting node is in the candidate list. If so, the corresponding data vectors of the node are read into memory. If the visiting node is an MBR, the cloud then performs the anchor-based position determination to choose the corresponding query vector and calculates the similarity between the MBR and the query. If the visiting node is an object, the cloud just calculates the similarity between the object and the query. Finally, MBRs or objects with their similarities are enqueued into a priority queue, and the best first algorithm goes on performing. The PkSKQ scheme is detailed in Algorithm-2.

Algorithm-2: Query Processing of PkSKQ
Input: the secure index I, encrypted query vectors fQi:vg and

Q:a, the number of requested objects k; Output: the list of query results LR; Procedure:
1: receive candidate node list LN returned in Algorithm 1
2: generate a priority queue PQ;
3: PQ.Enqueue(I:root,0);
4: while not PQ.Isempty() do
5: Element PQ.Dequeue();
6: if Element is an object then
7: if Element in LN then
8: if not PQ.Isempty() and
9: SP(Element:v;Q8:v) > PQ.Front().Key then
10: PQ.Enqueue(Element,SP(Element:v;Q8:v));
11: else
12: LR.Insert(Element);
13: if LR.size == k then
14: break;
15: else if Element is a leaf node then
16: for each object O in leaf node Element do
17: if O in LN then
18: PQ.Enqueue(O, SP(O:v;Q8:v));
19: else
20: for each child node e of non-leaf node Element do
21: if e in LN then
22: for i 2 to 5 do
23: hi PD(e:An1; e:Ani; Q:a);
24: H h2jh3jh4jh5;
25: if H is in Table 1 then
26: choose the corresponding Qi:v;
27: PQ.Enqueue(e, SP(e:v;Qi:v));
return LR;

IV. Proposed Scheme

In this section, we first outline the main idea of our cloud based efficient privacy-preserving biometric identification scheme, and then present the design of its three stages. Finally we analyze the security of our proposed scheme.

A. Scheme Overview

In our design, we decompose our system into three stages: biometric database processing (Stage 1), privacy- preserving FingerCode comparison (Stage 2) and final result generation (Stage 3). In Stage 1, by generating the secret keys based on the selected parameter, the database owner encrypts the entire database and uploads it to the cloud server with the corresponding Index. In our scheme, this is one-time cost and can be considered as a preparation stage. In Stage 2, the owner creates the credential for the client's candidate Finger Code and submits it to cloud servers; the servers then find out the ciphertext of Finger Code that has the minimum Euclidean distance with the candidate one from the encrypted database. Different form the existing works, which use cryptographic tools to securely compute the exact Euclidean distances and compare them with each other, we observe that *computing the exact distance is not necessary in the identification process*. Instead, our proposed scheme just computes the comparison result of the Euclidean distances from the candidate finger code for any two encrypted finger codes in the database, i.e., we just obtain the result of which Finger Code in the database is more similar to the candidate without calculating their actual distances. In Stage 3, on receiving the result (i.e., the index of the matching Finger Code) from cloud server, the database owner generates the final identification result with two more simple operations: computing the exact Euclidean distance between the candidate Finger Code and the one returned by cloud; checking the Euclidean distance with the threshold.

Problem Statement: In this paper, we aim at enabling multiple parties to jointly conduct BPN network learning without revealing their private data. The input data sets owned by the parties can be arbitrarily partitioned. The computational and communicational costs on each party shall be practically efficient and the system shall be scalable. Specifically, we consider a 3-layer neural network for simplicity but it can be easily extended to multi-layer neural networks.

V. Conclusion

We have studied the privacy-preserving top- *k* spatial keyword query problem in untrusted cloud environments. We proposed a privacy-preserving top-*k* spatial keyword query scheme. In this scheme, we built a secure index based on a state- of-the-art index for spatiotextual data (i.e., IR-tree), where the spatial and textual data are encrypted in a unified way. To search with such index, we developed anchor-based position determination and position-distinguished trapdoor generation for the similarity computations between query points and tree nodes under encryption. To improve query performance on large scale datasets, we further propose a keyword-based secure pruning method to reduce the I/O cost. Privacy-preserving biometric identification offers the promise of identifying individuals based on private biometric trait without disclosing privacy biometric data. We proposed the first cloud based privacy-preserving biometric identification scheme. By securely leveraging the power of cloud computing, we enable a biometric database owner to securely move most biometric identification operations to cloud servers, which guarantees the performance of the system even in case of large databases and a relatively large number of simultaneous requests.

References

[1] M. Barni, T. Bianchi, D. Catalano, M. D. Raimondo, R. D. Labati, P. Failla, D. Fiore, R. Lazzeretti, V. Piuri, F. Scotti, and A. Piva, "Privacy Preserving Fingercode Authentication," in *Proceeding MM & Sec '10 Proceedings of the 12th ACM workshop on Multimedia and security*, pp. 231-240, 2010.

[2] Y. Huang, L. Malka, D. Evans, and J. Katz, "Efficient privacy-preserving biometric identification," in *18th Network and Distributed System Security Conference (NDSS 2011)*, 6-9, February 2011.

[3] W. K. Wong, D. W. Cheung, B. Kao and N. Mamoulis, "Secure kNN Computation on Encrypted Databases," in *Proceeding SIGMOD '09 Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data*, pp. 139–152, 2009.

[4] A. M. Bazen and S. H. Gerez, "Systematic methods for the computation of the directional fields and singular points of fingerprints," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 905-919, Jul 2002.