International Journal of Engineering Research in Computer Science and Engineering (IJERCSE) Vol 3, Issue 4, April 2016

Resolve the Classification Problem over Encrypted data Using K-Nearest Neighbour

[1]C Deepika, [2]S Deepak, [3]Jestina Thomas, [4]Akansha Singh
[1][2][3][4]SRM University, Chennai, TamilNadu, India
[1]c.deepika1@gmail.com,[2] sdeepak94@gmail.com, [3]jestinamini@gmail.com,[4]akansha0594@gmail.com

Abstract: **Data mining is widely used in numerous areas such as banking, medicine, scientific research and various other government applications. Data Classification is most extensively used task in data mining applications. The rise of various issues in privacy has led to several solutions to this problem. However, with the increased fame of cloud computing, users can now outsource their data onto the cloud in an encrypted form, as well as perform the required mining tasks on the cloud. Since the data is in an encrypted form on the cloud, existing privacy-preserving classification techniques are not suitable. In this paper, we focus on a viable technique to perform data classification using k-NN classifier over the encrypted data and secure the confidentiality of data, privacy of the user's input query and also hide the data access patterns. We also analyze the efficiency of the proposed protocol empirically.**

Index Items: **Data mining, attribute based encryption, k-nn classification.**

I. INTRODUCTION

Data mining is a popular field that connects the world of databases and artificial intelligence. Data mining helps mine useful information from the vast amount of data present. A lot of organisations today have moved from storing and accessing the data locally to a more cost effective and flexible cloud platform. Organisations often delegate their computational operations to the cloud in addition to the data. Cloud provides reduced data management costs, reduced data storage overhead, and improved service. But these advantages come at a price of higher privacy risks[1] and vulnerabilities. The cloud service providers or adversary may learn meaningful information or access sensitive data and use it for their profits.

One straightforward way to protect the confidentiality of outsourced data from the cloud as well as from unauthorized users involves the data owner encrypting the data before it is outsourced .This paper uses Attribute Based Encryption[2] (ABE), a public key encryption where secret key of a user and the ciphertext are dependent on the attributes. Performing data mining tasks on encrypted data without decrypting would be difficult. To give a higher security to the data stored on the cloud, this paper discusses on the techniques to perform data mining over encrypted data[3] (DMED). The privacy requirements for DMED problem include (i) confidentiality of the encrypted data, (ii) confidentiality of the user's query and (iii) data access patterns.

The outsourced data uses k-NN model [4] that focuses on providing classification, rather than training. Hence, it is assumed that the parameters for the algorithm have been adequately chosen through initial study. Prior work on preserving privacy in such a model uses computation over encrypted data. They assume that a single trusted data owner encrypts her data before sending it to a cloud provider. Then users can submit queries in encrypted form to the system for k-NN classification. In this setting, we would like to keep data private from the cloud provider and users, and keep queries private from the cloud party and the data owner. In some existing works, the users share the secret (often symmetric) data encryption keys. In others, the user interacts with the data owner to derive the encryption for a query without revealing it.

Existing work on Privacy-Preserving Data Mining[5] cannot solve the DMED problem. In this paper, we propose new methods to solve the DMED problem assuming the encrypted data are outsourced onto cloud. We focus on the classification problem in detail since it is a widely used data mining task. This paper focuses on the execution of k-nearest neighbour classification method over encrypted data in the cloud environment.

II. RELATED WORK

Initially, a fully homomorphic cryptosystem[6] could solve the DMED problem as it allows arbitrary functions over encrypted data to be executed by third party without ever decrypting them. However, techniques as such

are very expensive and their practical applications usage have not yet been explored.

The existing secret sharing techniques such as Shamir's scheme[7] can be used to develop a PPkNN protocol in SMC[8]. However, our work require only two parties whereas solutions based on the secret sharing schemes require three parties or more. For example, Sharemind[7], a familiar SMC framework assumes that there are three participating parties.

A. Privacy Preserving Data Mining

Privacy-Preserving Data Mining (PPDM) was first introduced by Lindell and Pinkas. It extracts/derives knowledge about data without compromising the privacy of the data. The existing PPDM techniques are basically categorised as: (i) data perturbation and (ii) data distribution. The first data perturbation technique was to build a decision-tree classifier[9] proposed by Agarwal and Srikant. Data perturbation techniques perturb the individual data records by adding random noise in such a way that the distribution of the perturbed data look very different from that of the actual data. However, they are not applicable over semantically secure encrypted data. Due to the addition of statistical noises they don't produce accurate data mining results. The first data distribution technique, decision tree classifier under the two-party setting assumes that data were distributed between them horizontally or vertically. The PPkNN problem cannot be solved using the data distribution techniques as the data in our case is not distributed in plaintext among multiple parties but encrypted.

B. K- nearest neighbour classification

Existing privacy-preserving classification techniques are not sufficient and applicable in this work. K nearest neighbour classification identifies the top k closest records to the query input record at the database. Given a specific threshold t, to find all records in the dataset whose distances with the query lie either below or above t. This however reveals other information and causes issues: (i) Reveals the intermediate k-nearest neighbours to the user. (ii) It is very difficult to find the majority class label among these neighbours even if we know the k-nearest neighbours. (iii)Data access patterns can be leaked. Useful and sensitive information about users' data items can be obtained by simply observing the data access patterns. To solve this problem we use privacy preserving KNN[10] (PPkNN) classification with secure k-nearest neighbour query protocol being the first stage in it.

In the outsourced k-NN system, a trusted data owner again performs data encryption, using a symmetric scheme with a secret matrix transformation as a key. However, users do not share this key. To derive a query encryption without revealing the query, they interact with the data owner instead. This requires a data owner to always remain online for all queries. Also, data tuple encryption (being a matrix transformation) is deterministic, but query encryption is not due to randomness introduced during the query encryption protocol. The encryption scheme is designed to preserve distance, so a cloud uses distances computed from encrypted data tuples and queries provided to execute k-NN. In this system's trust model, the data owner is trusted while the users and the cloud providers are semi-honest.

C. Paillier Cryptosystem

The Paillier cryptosystem[11] is a probabilistic public-key encryption scheme with security on based on the Decisional Composite Residuosity Assumption. It is also an additive homomorphic cryptosystem. Let Epk be the encryption function with public key pk ie. (N, g), where N is a product of two large primes of similar bit length and g is a generator in $Z* N2$. Also, Dsk is the decryption function with secret key sk. The Paillier cryptosystem shows the following properties for plaintexts a and b:

1) Homomorphic Addition

$Dsk(Epk(a+b)) = Dsk(Epk(a) * Epk(b) \bmod N2)$;

2) Homomorphic Multiplying

$Dsk(Epk(a * b)) = Dsk(Epk(a)b \bmod N2)$;

3) Semantic Security - The encryption scheme is semantically secure.

For a given a set of ciphertexts, an adversary cannot obtain any additional information about the plaintext.

III. PROPOSED SYSTEM

A. Privacy preserving primitives:

Here we mention a number of generic sub-protocols that is used in our proposed k-NN classifier being constructed. All of the below protocols are considered under two-party semi-honest model.

- Secure Multiplication (SM): This protocol considers P1 with input (Epk(a),Epk(b)) and outputs Epk(a * b) to P1, where a and b are not known to P1 and P2. During this process, no information regarding a and b is revealed to P1 and P2.
- Secure Squared Euclidean Distance (SSED): In this, P1 with input (Epk(X),Epk(Y))and P2 with sk securely compute the encryption of squared

Euclidean distance between vectors X and Y .The output Epk(|X − Y |2) will be known only to P1.

- Secure Minimum (SMIN): P1 holds private input (u′, v′) and P2 holds sk, where u=([u],Epk(su)) and v'= ([v],Epk(sv)). The goal of SMIN is for P1 and P2 to compute together the encryptions of the individual bits of minimum number between u and v. The output ([min(u, v)],Epk(smin(u,v))) is known only to P1. No information regarding the contents of u, v, su, and sv is revealed to P1 and P2.
- Secure Minimum out of n Numbers (SMINn): We consider P1 with n encrypted vectors ([d1], . . . , [dn]) including their respective encrypted secrets and P2 with sk. They compute pk(smin(d1,...,dn)). At the end of this protocol, the output ([min(d1, . dn)],Epk((smin(d1,...,dn))) is known only to P1. During SMINn, no information regarding any of di's and their secrets is revealed to P1 and P2.
- Secure Frequency (SF): Here P1 with private input (hEpk(c1), . . .Epk(cw)i, hEpk(c′ 1), . . . ,Epk(c′ k)i) and P2 securely compute the encryption of thefrequency of cj , denoted by f(cj), in the list hc′ 1, . . . , c′ki, for 1 ≤ j ≤ w. The output hEpk(f(c1)), . . . Epk(f(cw))i will be known onlyto P1.

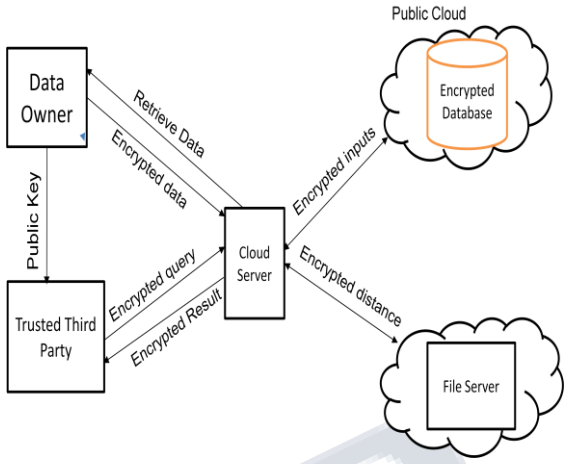
B. System Model:

In our work, we consider a model of cloud system, which consists of three main entities. Data owner, Cloud Service Provider, Third Party.

Data Owner: Entity that stores large amount of encrypted data on the cloud.

Cloud Service Provider: is an entity that provides services for data storage and computational resources dynamically to the data owner and third party respectively.

Third Party: Entity which wishes to classify his data by sending a query using his data stored in the cloud.



In fig, the data owner first encrypts his data using Attribute based Encryption(ABE) and outsources the encrypted database into the cloud servers via the cloud service provider. Once the data moves to the cloud he loses his control over it, this lack of control on data raises privacy issues in the cloud. The Cloud Service Provider provides two clouds C1 and C2, one which stores encrypted data, and one which has access to the associated decryption function that acts as a decryption.

The proposed PPkNN protocol primarily consists of the following two stages:

Stage 1: Secure Retrieval of k-Nearest Neighbors (SRkNN):

- User initially sends his query q (in encrypted form) to C1 in this stage.
- Next step is thatthe clouds, C1 and C2 involve in a set of sub protocols to securely retrieve (in encrypted form) the class labels corresponding to k-nearest neighbors of the input query q.
- At the end of this step, C1 alone knows the encrypted class labels of k-nearest neighbors.

Stage 2: Secure Computation of Majority Class :

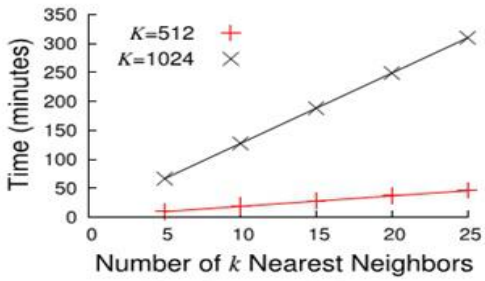
- C1 and C2 jointly compute the class label with a majority voting among query q's k-nearest neighbors.
- At the end of this stage, only user knows the class label corresponding to his input query record q.

The encrypted data from the data owner is stored on C1. The third party sends a query on which classification

is to be performed to the cloud in an encrypted manner. The K nearest neighbors of the query sent by the third party is computed using the SSED and S-MIN, S-MINn protocols. C1 then sends the k- nearest neighbors of the data to the C2. C2 takes the k nearest neighbors, computes the majority classifier for them using Secure Frequency protocol and assign it to the third party's data. The classifier is sent to the third party in an encrypted manner, which the third party decrypts on receiving the secret key from data owner.

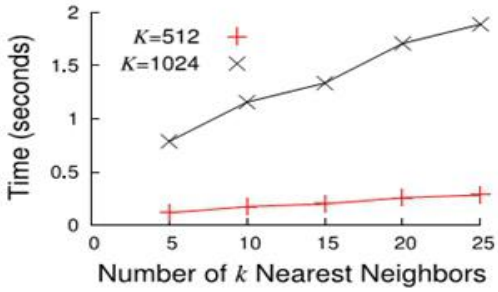
IV. PERFORMANCE ANALYSIS

Here, we discuss the performance of our kNN classifier over encrypted data under different parameter settings. We used the Paillier cryptosystem as the underlying additive homomorphic encryption scheme. The performance of our kNN classifier is checked under various different parameters.



(a) Total cost of Stage 1

First, we calculate the computation costs of Stage 1 in kNN over encrypted data for varying number of k-nearest neighbors. The Paillier encryption key size K is either set to 512 or 1024 bits. The results are shown in Figure (a). For K=512 bits, the computation cost at Stage 1 slightly increased when k is increased. On the other hand, when K=1024bits, the computation cost of Stage 1 significantly increased when k is also increased. We observe that the cost of Stage 1 grows in a linear manner with k. We now evaluate the computation cost at Stage2 for varying k and K.



(b) Total cost of Stage 2

When K is 512 bits, the computation time at Stage 2 to generate the final class label resultant to the input query varied from 0.118 to 0.285 seconds when k was increased. On the other hand, if K was set 1024 bits, Stage 2 took 0.789 and 1.89 seconds when k is increased by same amount. The low computation costs of Stage 2 were due to Secure Frequency primitive which incurs significantly less computations than SMINn at Stage 1.A similar analysis can be observed for other values of k and K. It is clear that the computation cost of Stage 1 is significantly higher than that of Stage 2 in kNN. We also observe that the stage 1 of the kNN classifier algorithm takes more than 95% of the total computation time.

V. CONCLUSION AND FUTURE WORK:

In conclusion, we analyzed that Privacy and security issues in the cloud are preventing companies from utilizing the tremendous advantages that the cloud offers . Therefore, due to the rise of various privacy issues on the data that are outsourced , we introduced the concept of privacy preserving k-Nearest neighbor classification protocol, where the third party want to cooperatively compute the k nearest neighbours to a query without enlightening their private inputs to the other party. In this paper, we proposed two novel PPkNN protocols over encrypted data in the cloud. The first protocol, which acts as a basic solution, leaks some information to the cloud and the Secure Retrieval of k-Nearest Neighbors. On the other hand, our second protocol is fully secure, that is, it protects the privacy of the data, user's input query, and also hides the data access patterns by Secure Computation of Majority Class.

One possible extension to the current work is to explore alternative ways of developing efficient SMIN algorithm. The empirical results clearly showed that they are only theoretically implemented and not practically. Encryption is not only the way of protecting the privacy of the data, but there are a variety of other techniques. For future work, we aim to improve the algorithm to not reveal the intermediate neighbourhood information, thus reducing the potential information leakage. Also, as this work is focused on horizontally partitioned data, another area of future work would be extending it to vertically partitioned data.

REFERENCES:

[1]. S. De Capitani di Vimercati, S. Foresti, and P. Samarati. Managing and accessing data in the cloud: Privacy risks and approaches. In 7th International Conference on Risk and Security of Internet and Systems (CRiSIS), pages 1–9, 2012.

[2]. Goyal, Vipul, et al. "Attribute-based encryption for fine-grained access control of encrypted data." Proceedings of the 13th ACM conference on Computer and communications security. Acm, 2006.

[3]. Aggarwal, Charu C., and S. Yu Philip. A general survey of privacy-preserving data mining models and algorithms. Springer US, 2008.

[4]. Peterson, Leif E. "K-nearest neighbor." Scholarpedia 4.2 (2009): 1883.

[5]. Lindell, Yehuda, and Benny Pinkas. "Privacy preserving data mining." Advances in Cryptology—CRYPTO 2000. Springer Berlin Heidelberg, 2000.

[6]. Gentry, Craig. A fully homomorphic encryption scheme. Diss. Stanford University, 2009.

[7]. Shamir, Adi. "How to share a secret." Communications of the ACM 22.11 (1979): 612- 613.

[8]. A. Ben-David, N. Nisan, and B. Pinkas. Fairplaymp - a system for secure multi-party computation. In ACM CCS, October 2008.

[9]. Bogdanov, Dan, Sven Laur, and Jan Willemson. "Sharemind: A framework for fast privacy-preserving computations." Computer Security-ESORICS 2008. Springer Berlin Heidelberg, 2008. 192-206.

[10]. Agrawal, Rakesh, and Ramakrishnan Srikant. "Privacy-preserving data mining." ACM Sigmod Record. Vol. 29. No. 2. ACM, 2000.

[11]. Nishide, Takashi, and Kouichi Sakurai. "Distributed paillier cryptosystem without trusted dealer." Information Security Applications. Springer Berlin Heidelberg, 2010. 44-60.