CLASSIFICATION OF SECURE ENCRYPTED RELATIONALDATA IN CLOUD COMPUTING

S.Narayanan[1], Naushin Ghani.V[2], Sangeetha.P[3], Manimegalai.K[4], Vishali.P[5]

[1]*Assistant Professor, B.Tech Information Technology, Valliammai Engineering College, Kattankulathur, Kancheepuram District - India*

[2,3,4,5]*Final Year Students, Department of Information Technology, Valliammai Engineering College, Kattankulathur, Kancheepuram District - India*

Abstract

Due to the increasing popularity of cloud computing, organisations have the choice to outsource their large encrypted data content along as well as data mining operations to cloud the environment. Outsourcing data to such a third party cloud environment can compromise the data security as cloud operations and data mining tasks cannot carry out computations without decrypting the data. Hence, already present privacy-preserving data mining techniques are not efficient to address the security and confidentiality problems. In the base paper, a k-NN classification algorithm over secure data under a semi-honest model was developed using a Paillier cryptosystem for public key encryption. The usage of public key cryptosystems has security issues during data transfer in the cloud. In this proposed work, we focus on solving the k-NN problem over secure encrypted data by proposing a privacy preserving k-nearest neighbour classification on encrypted information in the cloud using private key for encryption and decryption based on the symmetric AES cryptographic algorithm under the secure multiparty computations for creating a complete homomorphic encryption (CHE) scheme which results in the reduction of space requirement and processing time. Also, we aim to apply the same PPk-NN classification over encrypted images. The proposed protocol hides the input query and data access patterns of the users and also preserves the confidentiality of text and image data.Finally, we present a practical analysis of the efficiency and security performance of our proposed protocol for application in a Life insurance firm where the clients are classified according to their risk-level.

Keywords: *Data Mining, PPk-NN, Semi-Honest Model , Individual Key, Symmetric Homomorphic Encryption, AES Algorithm,CHE, Less Space and Time.*

----------***----------

1. INTRODUCTION

In today's computerized world, security is a major concern in all sectors of the workforce. Many organizations, due to extensive need for storage and resources, choose to outsource their databases to the third party server. But due to concerns about confidentiality and security issues, they have to think twice before outsourcing their data to the cloud. [2]

Generally, data is encrypted before sending to the cloud. But cloud operations such as data computations and data mining tasks cannot be performed over such encrypted data[3][6]. Hence, data has to be decrypted by the third party cloud before such data mining or cloud operations are performed. Also, during processing queries, the cloud can also obtain confidential and private information about the data by analysing the data access pattern while user query and information are encrypted[10]. Therefore, from the observations ,a privacy preserving query processing needs to assure aconfidential setup for the encrypted data in cloud and encrypt the user's query while keeping data access patterns hidden. [4][7]

In this paper, we mainly aim to avoid such pre-decryption of data before performing the data mining classification tasks. We propose a novel system for the semi-honest model[1] which makes use of SMIN (Secure Minimum), SF (Secure Frequency), SMC(Secure Multiparty Computation), $SMIN_n$(Secure Minimum out of n numbers) protocols[1][3] to avoid leakage of intermediate data computation results. We present a PPkNN(Privacy-Preserving k Nearest Neighbor) classification method under the homomorphic encryption scheme which aims to preserve the encrypted format of data during mining operations by prohibiting decryption activities by a third party evaluator.

Existing systems under semi-honest model only provide a Somewhat Homomorphic Encryption(SHE) due to the public key encryption system. In this paper, we use symmetric AES algorithm[5] for the private key generation that creates a Fully Homomorphic Encryption (FHE)[4][7], hence providing additional security. Also, not much work has been done on preserving the encryption of images as much as textual data records[5]. Hence, we present a reliable means of preserving encryption of images during data mining classification.

2. RELATED WORK & BACKGROUND

Here, we first present a light study of the existing secure k-nearest neighbour methods. Then, we present the security modelused in this paper along with the Complete

homomorphic encryption scheme with symmetric key cryptosystem and secret image sharing techniques.

Fully Homomorphic Encryption scheme

The view of privacy homomorphism was first introduced by Rivest, Adleman and Dertouzos in 1978 [11].Generally, many encryption techniques possess either a multiplicative property or additive homomorphic property used in certain areas. A fully homomorphic encryption scheme which can compute any calculation of arbitary nature over encrypted data was presented only in Gentry's paper in 2009[12]. Gentry starts from a Partial Homomorphic Encryption(PHE) that handles only a finite number of computations performed homomorphically on cipher texts.

Based on Gentry's work, a fully homomorphic scheme [13] over integers appeared at 2010. The primary attraction of the work is that it has a conceptual ease; as inall operations are performed over the integer values instead of lattice values. But, the public-key size was very large. Coron et al [15] presented a way to minimize size of the public key to Õ(7 □), by saving only a smaller range of the public key. The public key was lowered to Õ(3.5 □) as preset in [16] by using a decryption algorithm based on probability .

Paillier Cryptosystem:

Pascal Paillier proposed a new cryptosystem[14] in the year 1999. This method is based on composite residuosity classes.The evaluation of these classesis is said to be too complex. It is a cryptosystems using asymmetric algorithm based on probability and makes use of additive homomorphic properties, where the product of two ciphertexts will be decrypted based on the total of their plaintexts.

Image sharing Techniques

The latest techniques in instant communication require new methods of protecting, storing and transmitting important data and involves security strategies such as encryption watermarks ,digital signatures.Shamir[17] presented a novel method ofsecret sharing in 1979.The secret image sharing [5]investigates the integrity and security of shared secret images and proposes new methods for polynomial secret image sharing. Normal security systems for sharing secret images assume that all pixels are independent. However, this assumption may not be always correct. Advancements in secret image sharing means increasingly sophisticated ways of producing shares. Enhancing the security of traditional secret image sharing has 2 parts –1)Encoding and 2)decoding. All computations are to be done in the Galois Field GF(N) which is a finite field with values in the range of [0, N] and pixel depth is 1.The proposed method attempts to prevent dealers from knowing the position of any pixel. A user who doesn't know the pixel positions cannot formulate a relationship basis to identify the original pixel value when the amount of shares is less than the critical value.Firstly, the entire image is encoded using an AES encryption system along with a key. Secondly, the secret image sharing method is adjusted from prime value 251 to GF(256) to attain a lossless recovery. Thirdly, the shared key is assigned to some temporary keys by the Shamir secret sharing method [18]. Finally, the temporary keys and images are combined to produce a complete set of shares. This algorithm is effectively more secure compared to traditional secret image sharing.

3. PROBLEM STATEMENT

In a two-party communication,one party A will encrypt the database D using a secret key E_{sk}containing *n* records and *m* attributes in bulk and outsource the encrypted database D' to a third party cloud for storage and evaluation. The second party B will want to extract the data by sending the query q_u for retrieving a specific record containing text and images from the database $D'_{(r, i)}$.But during the retrieval process,the third party cloud decrypts the data in order to perform the k-NN computation based on the attribute classifiers m+1.Hence,we aim to preserve the encrypted format of the database by applying Privacy Preserving k-NN that prevents such decryption. Also, encrypted images can be decrypted by mapping the individual pixel positions of the images.So, preserving the privacy of encrypted images is done using fault tolerance key.

Therefore,

$E_{sk}(D_{(r,i)}) \rightarrow \mathbf{D'}$

$q_u \rightarrow D'$

4. SEMI-HONEST MODEL

Assume a situation where the parties involved in communication sends its inputs to a trusted third party that calculates the functional component ,a arbitrarily random process that designates and maps n inputs to n outputs. This trusted third party gives both the sender and receiver its output , so we would prefer to perform computations without this trusted party.So a semi-honest party is used where all the protocols are followed by the trusted party only with the exception that it keeps all its intermediate computational results to itself. The semi-honest party while tossing a fair coin,will only toss a fair coin and perform or reveal anything else.A separate protocol securely calculates what a semi-honest party can obtain after participating in the computations, and also from its inputs and outputs.

Use an encoding of 0's and 1's Alice chooses a random encoding of a random bit *b* and sends Bob the one-way function (or more exactly bit commitment) of the bit.Bob sends a random bit *c* to Alice , Alice reveals the commitment to *b* The common random bit is *b+c.*

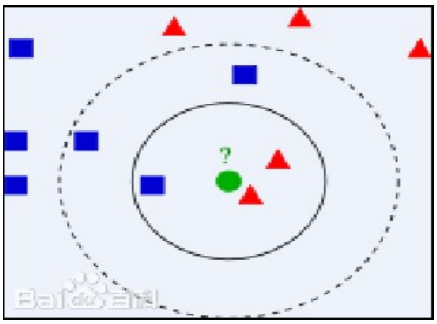
We considered in classes, the functionality in which A outputs a random bit and B outputs nothing andcannot learn A's output bit. We showed a protocol that is clearly insecure: A chooses a random bit and outputs it to B (B ignores the bit). This protocol is insecure since B learns A's output. We could have proved this protocol to be secure if we used a security definition that compares the view of a corrupt B in the real execution to the output of a simulator that only gets B's input and output: The simulator should generate a transcript that contains a single random bit sent from A to B. This demonstrates that a security definition

that does not take into account the joint distribution of both parties is insufficient. Note that the protocol cannot be proved to be secure according to the security definition that looks compares the joint definition: In the real execution the bit sent to B is identical to the bit output by A. In the simulation the simulator does not have access to A's output and, therefore, the value that it generates for B's transcript would be independent of A's output.

5. PROPOSED SYSTEM

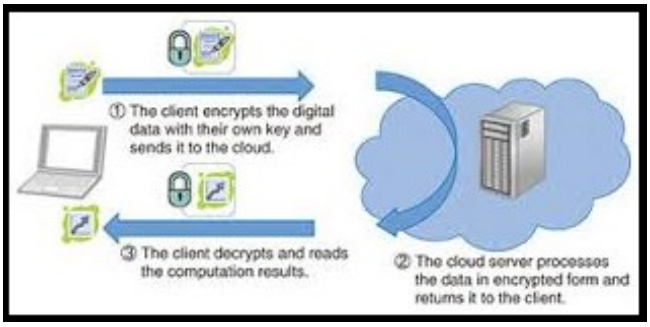
In this system, we propose a k-NN classification over semantically secure encrypted data in the cloud environment. A privacy preserving k-NN is used to classify the data based on the user's query(which is also preserved in encrypted format) and also to provide security to the encrypted data. This PPkNN classification solves the DMED(data mining over encrypted data) problems such as user data confidentiality, privacy of the user's query and encrypting the access patterns. encryption .We have based the security premises on a Semi-honest model where the revelation of the interparty computations is prevented.The secure privacy-preserving kNN is built using various sub-protocols, namely secure multiplication (SM), secure minimum out of numbers (SMINn), secure Bit Decomposition(SBD), Secure Shortest Euclidean distance(SSED) for performing secure PPkNN, along with formal security analysis and proof under the semi-honest model[1].

We also propose a new concept of image encryption in the cloud environment by using the safe secret image sharing with fault tolerance key. In this process, an advanced method of secret image sharing is performed compared to the traditional method where independent neighboring image's pixel values are mapped secretly where a co-efficient is required in sharing the images to preserve the pixel values. To achieve this , a fault tolerant key along with AES cryptographic technique is used for implementing safe image sharing under the semi-honest homomorphic model.



The database encryption and user query encryption is based on a (FHE)Complete/fully homomorphic encryption scheme that is achieved using a private key AES(Advanced Encryption Scheme), a block cipher algorithm is used for the symmetric key. This scheme is used for evaluation of expressions with encrypted operators, handling memory operation (such as assignment and lookup operations) and execution of loop done under a secure environment.

In our setting , let us consider two users Bob and Alice that make use of two semi-honest cloud service providers C1 and C2.Bob outsources his encrypted database D' to one cloud server C1 and the secret key to C2.For added security,after outsourcing Bob cannot involve in any future computations.Now,Alice would want to retrieve a record from the database in a secure manner. Hence,we implement our goal to classify users' records in D' using PPkNN protocol for secure retrieval.



6. PRIVACY-PRESERVING PRIMITIVES

We have presented a privacy reserving k-NN that is based on semi-honest model and fully homomorphic encryption(FHE). The semi-honest model is applied over two parties (P1 and P2)communication scheme. Here,We apply a symmetric private key AES cryptosystem where a single secret key S is used for the purpose of encryption and decryption and only when the secret key is shared to the decryption party P2, the data can be decrypted.

6.1.1 AES Encryption

AES (Advanced Encryption Standard) is a symmetric algorithm which is used for encryption. It was designed to be efficient for both hardware and software.It has a various block lengths such as 128 bits, 192 bits and 256 bits.

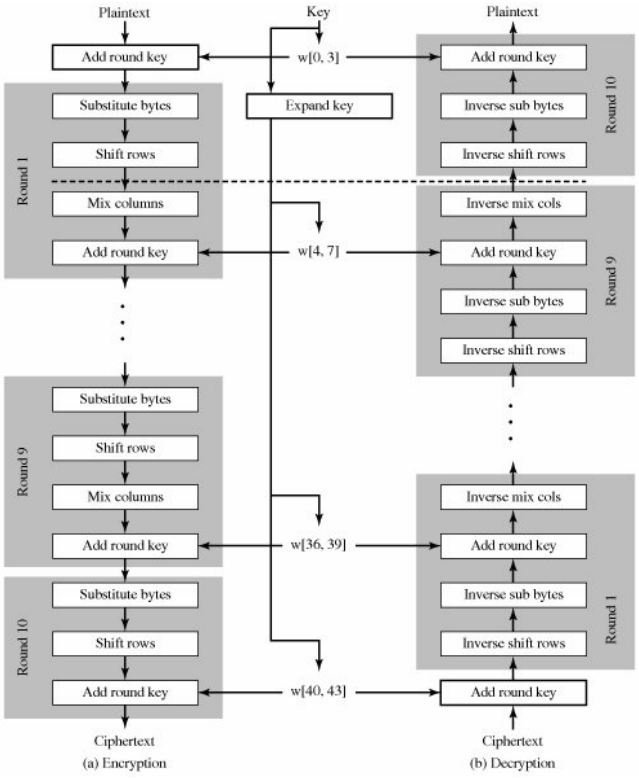
	KEY LENGTH (Nk words)	BLOCK SIZE (Nb words)	ROUNDS(Nr)
AES 128	4	4	10
AES 196	6	4	12
AES 256	8	4	14

Steps in AES algorithm:

- Convert the plain text and given cipher key into the state table using the ASCII table and the state table has 4x4 square matrix of bytes. The key that is provided as input expanded into the array of 32-bit words (a_0 $a_{1.....}a_{n)}$ this key would be used as a round key for each round.
- Perform the XOR operation for the state table and cipher key and you will get another state array of the 4x4 matrix.
- Key expansion:

- It takes as input as four words(16 bytes) key and produces a linear array of 44 words(176 bytes) and these will be used for the 10 rounds of the cipher.
- The key is copied to first four words of the expanded key. The remainder will be filled in four words at a time.Each added word will depend on the immediately preceding word.In these cases, a simple XOR function is used.
- After that, we have to take the RCON table first column to XOR with the previous result.

To understand the PPkNN classification better we have given a sample dataset containing life insurance details of clients with their medical history into to identify which risk level they belong to certification of an insurance policy scheme.



- Like this, the operation gets continues till the end of the operation.
- Every 4x4 matrix will be taken for the next round key for the encryption process.

- After the expansion of the cipher key for the first round there will be 4 rounds:

They are
- SUBBYTES
- SHIFTROWS
- MIX COLUMNS
- ADD ROUNDKEY

- Substitute Bytes: Uses an S-Box to perform a byte-by-byte substitution of this block.
- Shiftrows: A simple permutation i.e we have to shift the 2nd row by one left byte and vice versa for the 3rd and 4th byte
- Mixcolumns: A substitution that makes a use of arithmetic over GF.
- AddRoundKey: A simple bitwise XOR operation of the current block with the portion of the end key.
- The above four rounds will be continued for 9 rounds and the last round is the 10th round that will not have a mix column step.
- Finally, the 4x4 square matrix will consider the encrypted data or cipher data.

Consider the following plain text that has to be encrypted using the given key.
PLAINTEXT:0123456789abcdeffedcba9876543210
KEY : 0f1571c947d9e8590cb7add6af7f6798

Step 1: The plain text is changed into the 4x4 square matrix state table and the key will also change to the array of bytes.There will occur an XOR operation and produces the cipher state table for the next round.

For example
01=0000 0001
0f=0000 1111

0000 1110=0e

01	89	fe	76
23	Ab	dc	54
45	Cd	ba	32
67	Ef	98	10

⊕

0f	47	0c	af
15	d9	b7	7f
71	e8	ad	67
c9	59	d6	98

⇩

0e	Ce	f2	d9
36	72	6b	2b
34	25	17	55
ae	b6	4e	88

The XOR function will be performed on all the bytes in the state table and the produced state table will be applied for further process.

Step 2: Round 1 (Sub Bytes)
In this round, it has four steps to be performed to provide more security.for that, the next step would be substitute bytes using the S-BOX. After substituting the bytes from the s-box the resultant matrix would be:

ab	8b	89	35
05	40	7f	f1
18	3f	f0	fc
e4	4e	2f	c4

Step 3: Shifting Rows
In this the above matrix will get the shifting process i.e 1st row will not be shifted.But the 2nd row is shifted towards the left by one byte likewise the 3rd will be shifted by two bytes and the 4th row will be shifted by three bytes.The bytes which are goes beyond the MSB will occupy the empty place in the respective row.BeforeShifting,

b9	94	57	75
e4	8e	16	51
47	20	9a	3f
c5	d6	f5	3b

After shifting,

ab	8b	89	35
05	40	7f	f1
18	3f	f0	Fc
e4	4e	2f	c4

Step 3: Mix Columns

The forward mix column transformation will be considered as Mix columns which operate on each column individually.Each byte of a column will be mapped into a new value that is a function of all four bytes in that column.The matrix multiplication has the following state

02	03	01	01
01	02	03	01
01	01	02	03
03	01	01	02

After multiplying each column with the abovemultiplication table there will occur a resultant matrix to be XOR with the round key which has to be moved for the next round

02	03	01	01		ab		b9
01	02	03	01	*	40	⇨	e4
01	01	02	03		f0		47
03	01	01	02		c4		c5

Like this matrix multiplication will be performed on all the columns and the result matrix will be:

ab	8b	89	35
40	7f	f1	05
f0	Fc	18	3f
c4	e4	4e	2f

6.1.2 Privacy Preserving K-NN

After the encryption has taken place, the encrypted databaseD' will be outsourced to cloud.when a cloud user would wish to classify and download the data, the k-NN algorithm is used.But for the cloud to perform this operation,
it has to decrypt the data.We propose a method to avoid such decryption under a Fully Homomorphic encryption scheme by using a Privacy Preserving K-NN algorithm.The secure k-NN takes place in two stages,
Stage I: Secure retrieval of k-NN.
Stage II: Secure calculation of the Majority Classes

6.1.2. Stage I

Alice sends her query to C1.After which, both C1 and C2 involve in securely retrieving the class labels k nearest neighbor value of the user query q by using certain sub-protocols such.

Secure squared Euclidian distance metric(SSED)
This protocol[19] does not take the square root of the value unlike Euclidean Distance metric hence provides faster computations during privacy preserving classification where the database records are classified according to the attributes. For every data object, P1 and P2 calculate the SSED with a minimum k neighbor value.

The Secure Shortest Euclidean distance (SSED) protocol is used to determine the relative distance between the nearest neighbors.

The Euclidean distance is calculated using the formula,
$dist((y, z), (m, n)) = \sqrt{(y - m)^2 + (z - n)^2}$

wherex,y denotes the user query parameters which are the criteria for classification.

Secure bit-decomposition (SBD)
P1 has input data and P2 securely computes the encryptions of the individual bits of the input data. The output computed is known only to P1.

Step1: generation of Bytecode.
Step2:swapping of the byte code values(rearrangement).
Step3: Cross multiplication of rearranged values.
Step4: encryption using the secret key.
Step5: further computations.

Decryption occurs in a similar manner except the steps take place in reverse order.

After the Shortest distance is calculated the Secure Bit-Decomposition(SBD) protocol is used to provide encryption for the classified data.

Secure minimum out of n numbers ($SMIN_n$)
In $SMIN_n$, we consider a party P1 with encrypted vectors m from d_1 to $d_n$along with their corresponding encrypted data and P2.During SMINn,any information relating to $d_i$is not revealedto a third party .Only relevant information is revealed even to P1 and P2.

6.1.2. Stage II

In stage II, C1 and C2 both compute the classifier label by using majority ranking among the k value of query q which only Alice knows. This stage computes class labels by using SMAXn protocol which is used to classify the query record to one of the w classes based on the query inputs.

Secure multiplication (SM)
Secure Minimum considers P1 with an input $E_s(e)$ and $E_s(f)$ and provides output $E_s(e*f)$ to P1, where e and f are unknown to P1 and P2. During this process, any detailsabout the variables is not revealed to P1 andP2.

Secure minimum(SMIN)
In Secure Minimum, P1 contains sensitive input(y,z)and P2 holds the secret key S. The main aim of this protocol is for P1 and P2 to calculate the encryption values of the individual bits of the minimum number of y and z jointly. In addition, they compute the encrypted S_{min}value between y and z which will be known only to P1.
The two stages can be explained with an example as follows,

Client id	Client name	image	Father name	Address	phone	Heart disease	smoker	Risk class
#$%!@&	+_=@#$	------	*&%#":@	+=1#@$%	*&^%#	&^%_+	@#$%_	-=#$$@!
-=/*$%	$$%##	5&^	+_##1@#	/+*^2%$#	(%^##&*#	(&$#)#$	&^%!@~#$	~!#$^&^##

User Query=client id :LICA010,Heart disease=7,Smoker=3.

Y classification has w=2 classes(ATTRIBUTE VALUES).

1.Assume the value of parameter k which is equal to the number of nearest neighbors in the database.
Let us consider k as 3.
2. Compute the distance between values in the query instance and all the data records i.e training samples.
Using the query instance values (3,7),we calculate the shortest square Euclidean distance which is more quick to evaluate.

X1=heart disease	X2=smoker	SQUARE DISTANCE TO QUERY INSTANCE.
7	7	$(7-3)^2+(7-7)^2=16$
7	4	$(7-3)^2+(4-7)^2=25$
3	5	$(3-3)^2+(5-7)^2=4$
1	3	$(1-3)^2+(3-7)^2=20$

3.Evaluate the distance and calculate the values of the nearest neighbors which depends on the k parameter value.

X 1	X 2	SQUARE DISTANCE TO QUERY INSTANCE	RANK THE MIN DISTANCE	INCLUSION IN THE 3-NEAREST NEIGHBORS?
7	7	16	3	Yes
7	4	25	4	No
3	5	4	1	Yes
1	3	20	2	Yes

4.Obtain the classification Ythat the evaluated nearest neighbors belong to.
5.Use the average value of the majority category to which the nearest neighbours belong to as a prediction strategy for the query instance record.

Here the values that are within the k neighbor value are classified to sub-standard life insurance whereas the ones that are outside the neighbor value are credited with a standard life insurance scheme.

X 1	X 2	SQUARE DISTANCE TO QUERY INSTANCE(3,7)	RANK MIN DISTANCE	IS IT INCLUDED IN 3-NEAREST NEIGHBOR	Y=CATEGORY OF NEAREST NEIGHBOR
7	7	16	3	yes	substandard
7	4	25	4	no	standard
3	5	4	1	yes	substandard
1	3	20	2	yes	substandard

7. SIMULATION RESULTS

We discuss the application developed using the PPkNN protocol for a life insurance scheme and record the obtained results. We used the Private symmetric key cryptosystem(AES) for creating the underlying fully homomorphic encryption scheme and implemented the proposed PPkNN protocol for a Life Insurance Policy scheme with the dataset containing over1000 records and more than 8 attributes and use of 4 classifiers.
The life insurance scheme was applied such that the client attributes containing the medical details such as height, weight, cholesterol,bloodgroup, heart disease and smoking were analyzed on a mark based scale and the clients were classified based on their medical details as to which class of insurance they belong to.
The 4 classes were the risk levels available such as super preferred, preferred, standard and substandard.

The clients with excellent health were given super preferred class containing highest insurance benefits, the ones with above average health were placed in the preferred class, average health clients were in standard class and the ones with very poor health were classified into sub-standard class or were not able to avail insurance at all.

We evaluated and analyzed the efficiency the PPkNN separately in two stages using J2EE and My SQL 5.0. The application was developed using Windows 07 having PENTIUM IV 2.6 GHz processor and 512 MB DD RAM with 40GB hard disk.

7.1 Cost-performance Analysis of PPkNN

The experiment was conducted in two stages. Stage I consists of executing PPkNN using SMINn where k value is varied from 5 to 25 and key size K=32 bytes. The cost of the stage I is directly proportional to the value of k .The cost increases 5%-10% for the increase in k value.

In stage II, the computation cost to generate the final class label w varies from 0.789sec to 1.89 sec when k value is varied between 5 and 25.The low computation time for stage II is due to the use of SMAXn.

The computation cost of stage I is relatively higher compared to stage II in PPkNN, where stage I takes up almost 99% of the computation time.

7.2 Improvement of PPkNN Performance Efficiency

In order to improve the efficiency of our PPkNN protocol, an offline phase is allocated for pre-computed encryptions of random numbers denoted as $PPkNN_o$.It is technically seen that $PPkNN_o$ is 33% faster that PkNN without the offline computation phase.

As operations on each record take place individually, another method to increase the efficiency is to consider parallelizing the stage I computations($PPkNN_p$). Computing the stage I operations for each record parallel reduces computation time effectively.

In case, the user operates a resource constraint device such as a cell phone.Then the cost ofencrypting the user query takes up to 17 milliseconds which is found to be very ideal.

8. CONCLUSIONS AND FUTURE WORK

Organizations can confidently outsourcetheir data over tocloud without fearof security threats such as illegal access or information theft. The existing techniques for secure data storage promise safety and integrity but are not applicable for data that is outsourced to third party environments where data has to remain secure and encrypted on an external server. This paper showed a new method to perform PPk-NN classification protocol over semantically encrypted data that is present in the cloud environment. Our protocol safe guards the input query confidentiality and also covers the patterns followed during data access.Thereby, eliminating the remnant crumbsand traces left behind during dataretrievaland cloud operations. We also evaluated the performance of our protocol using a real time application for both data ad images using secret image sharing the technique. Since the efficiency of the proposed system mainly is based on theSMINnprotocol, we strategize to further analyze different and more effective answers to the SMINn problem in detail in further studies and work along with ways for improving encrypted image sharing and classification techniques by efficiently managing computation cost during pixel evaluation.

REFERENCES

[1] k-Nearest Neighbor Classification over Semantically Secure Encrypted Relational Data Bharath K. Samanthula, Member, IEEE, Yousef Elmehdwi , and Wei Jiang, Member, IEEE.
[2]Homomorphic Symmetric key Encryption, Network of the future,2013, Fourth National Conference, Pohang.
[3] Secure k-Nearest Neighbor Query over Encrypted Data in Outsourced Environments, Yousef Elmehdwi, Bharath K. Samanthula and Wei Jiang July 19, 2013, Technical Report Department of Computer Science, Missouri S&T

[4] Fully Homomorphic Encryption Scheme with Symmetric Keys ,Itisharma, university college of engineering
[5] Safe Secret Image Sharing with Fault Tolerance Key IJCSNS International Journal of Computer Science and Network Security, VOL.11 No.9, September 2011 , Wen-Pinn Fang Yuanpei University, Taiwan, R.O.C.
[6] Practical Techniques for Searches on Encrypted Data, Dawn Xiaodong Song David Wagner Adrian Perrig, Dawnsong, daw, perrig, University of California, Berkeley
[7] Fully Homomorphic Encryption scheme with Symmetric Keys with Application to Private Data Processing in Clouds, C P Gupta, and Iti Sharma , Department of Computer Sciences and Engineering.
[8] S. De Capitani di Vimercati, S. Foresti, and P. Samarati. Managing and accessing data in the cloud: Privacy risks and approaches. In 2012 7th International Conference on Risk and Security of Intersnet and Systems (CRiSIS), pages 1–9. IEEE, 2012.
[9] Finding Minimum Weight Connected Dominating Set in Stochastic Graph basedOn Learning Automata, Peter Mell,Timothy Grance,2008.
[10] A Private Database Search with Sub linear Query Time, Keith B. Frikken and Boyang Li, 2012
[11] R. Rivest, L. Adleman, and M. Dertouzos. On data banks and privacy homomorphisms: In Foundations of Secure Computation, pages 169-180, 1978.
[12] C. Gentry. Fully homomorphic encryption using ideal lattices. In Proc. of STOC, pages 169-178,
[13] M. van Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan.Fully homomorphic encryption over the integers. In Proc. of Eurocrypt, volume 6110 of LNCS, pages 24-43. Springer, 2010.
[14] P. Paillier, “Public-key cryptosystems based on composite degree residuosity classes”, Proc of EUROCRYPT-99, Springer, pp 223–238, 1999.
[15] J.-S. Coron, A. Mandal, D. Naccache, and M. Tibouchi.Fully homomorphic encryption over the integers with shorter public-keys. In Advances in Cryptology - Proc. CRYPTO 2011, volume 6841 of Lecture Notes in Computer Science. Springer, 2011
[16] D. Stehle and R. Steinfeld. Faster Fully Homomorphic Encryption. Cryptology ePrint Archive: Report 2010.
[17] P. Paillier, “Public-key cryptosystems based on composite degree residuosity classes”, Proc of EUROCRYPT-99, Springer, pp 223–238, 1999.
[18] A. Shamir, “How to share a secret,” Communication of the ACM, Vol. 22, no. 11, pp. 612-613, 1979.
[19] Secure Two-Party Computation of Squared Euclidean Distances in the Presence ofMaliciousAdversaries,Marc Mouffron, Frederic Rousseau, Huafei Zhu.