ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 20, April 2016

K-Nearest Neighbor Classification over Semantically Secure Encrypted Relational Data

Dr.V.Selvi[1] *and Ms.K.Rajalakshmi*[2]
[1]*Assistant Professor and* [2]*M.Phil Scholar*
[1]Selvigiri.s@gmail.com and [2]rajikutty718@gmail.com
Department of Computer Science
Mother Teresa Women's University
Kodaikanal, Tamil nadu, India.

---***---

Abstract-Data mining has wide variety of real time application in many fields such as financial, telecommunication, biological, and among government agencies. Classification is the one of the main tasks in data mining. For the past few years, due to the increment in various privacy problems, many conceptual and feasible solutions to the classification problem have been proposed under different certainty prototype. With the increment of distributed computing users have an opportunity to offload the data and processing the cloud, in an encrypted form. The data in the distributed are in encrypted form, existing privacy preserving classification systems are not relevant. To perform privacy preserving k-NN classification over encrypted data. The recommended protocol preserves the privacy of data, protect the user query, and hide the access mode.

Keywords-Security, K-NN Classifier, Outsourced databases, Encrypted.

I. INTRODUCTION

Recently, the cloud computing paradigm has revolutionized the organizations' way of operating their data, particularly in the way they store, access and process data. As an emerging computing paradigm, cloud computing attracts many organizations to consider seriously regarding cloud potential in terms of its cost-efficiency, flexibility, And offload of administrative overhead. Most often, organizations delegate their computational operations in addition to their data to the cloud. Despite the tremendous advantages that the cloud offers, privacy and security issues in the cloud are preventing companies to utilize those advantages. When data are highly sensitive, the data need to be encrypted before outsourcing to the cloud. However, when data are encrypted, irrespective of the underlying encryption scheme, performing any data mining tasks becomes very challenging without ever decrypting the data.

The data owner outsources his/her database and DBMS functionalities (e.g., KNN query) to an external service provider which manages the data on behalf of the data owner where only trusted users are allowed to query the hosted data at the service provider. By outsourcing data to an untrusted server, many security issues arise, such as data privacy (protecting the confidentiality of the data from the server as well as from query issuer). To achieve data privacy, data owner is required to use data Anonymization models (e.g., k-anonymity) or cryptographic (e.g., encryption and data perturbation) techniques over his/her data before outsourcing them to the server. Encryption is a traditional technique used to protect the confidentiality of sensitive data such as medical records. Due to data encryption, the process of query evaluation over encrypted data becomes challenging. Along this direction, various techniques have been proposed for processing range and aggregation queries over encrypted data.

Data mining over encrypted data (denoted by DMED) on a cloud also needs to protect a client's record when the record is a part of a data mining process. However, cloud can also abstract useful and sensitive information about the outsource data items by observing the data access patterns even if the data are encrypted. Therefore, the privacy/security requirements of the DMED problem on a cloud are of three types: (1) privacy of the encrypted data, (2) privacy of a user's query record, and (3) hiding data access patterns.

Specifically, we focus on the classification problem since it is one of the most common data mining tasks. Because each classification technique has their own advantage, to be concrete, this paper concentrates on executing the k-nearest neighbor classification method over encrypted data in the cloud computing environment.

II. RELATED WORK AND BACKGROUND

Due to space limitations, here we briefly review the existing related work and provide some definitions as a background please refer to technical, report for a more elaborated related work and background.

It is possible to use the existing secret sharing techniques in SMC, such as Shamir's scheme, to develop a PPkNN protocol. However, our work is different from the secret sharing based solution in the following aspect. Solutions based on the secret sharing schemes require at least three parties where as our work require only two parties.

For example, the constructions based on share mind, a well-known SMC framework which is based on the secret sharing scheme, assumes that the number of participating parties is three. This paper work is orthogonal to share mind and other secret sharing based schemes.

A. Privacy Preserving Mining of Association Rules

A framework for mining association rules from transactions consisting of categorical items, where the data has been randomized to preserve privacy of individual transactions. While it is feasible to recover association rules and preserve privacy using a straightforward uniform" randomization, the discovered rules can unfortunately be exploited to and privacy breaches. We analyze the nature of privacy breaches and propose a class of randomization operators that are much more elective than uniform randomization in limiting the breaches. We derive formulae for an unbiased support estimator and its variance, which allow us to recover item set supports from randomized datasets, and show how to incorporate these formulae in to mining algorithms. Finally, we present experimental results that validate the algorithm by applying it on a real data set.

B. Query Processing over Encrypted Data

Introduced new security primitives, namely secure minimum (SMIN), secure minimum out of n numbers (SMINn), secure frequency (SF), and proposed new solutions for them. Second, the work in not provide any formal security analysis of the underlying sub-protocols. On the other hand, this paper provides formal security proofs of the underlying sub-protocols as well as the protocol under the semi-honest model.

III. LITERATURE SURVEY

P. Williams, R. Sion, and B. Carbunar

"Building castles out of mud: practical access pattern privacy and correctness on untrusted storage"

We introduce a new practical mechanism for remote data storage with efficient access pattern privacy and correctness. A storage client can deploy this mechanism to issue encrypted reads, writes, and inserts to a potentially curious and malicious storage service provider, without revealing information or access patterns. The provider is unable to establish any correlation between successive accesses, or even to distinguish between a read and a write. Moreover, the client is provided with strong correctness assurances for its operations illicit provider behavior does not go undetected. We built a first practical system -- orders of magnitude faster than existing implementations that can execute over several queries per second on 1Tbyte+ databases with full computational privacy and correctness.

Y. Lindell and B. Pinkas

"Privacy preserving data mining"

In this paper, we address the issue of privacy preserving data mining. Specifically, we consider a scenario in which two parties owning confidential databases wish to run a data mining algorithm on the union of their databases, without revealing any unnecessary information. Our work is motivated by the need to both protect privileged information and enable its use for research or other purposes. The above problem is a specific example of secure multi-party computation and as such, can be solved using known generic protocols. However, data mining algorithms are typically complex and, furthermore, the input usually consists of massive data sets. The generic protocols in such a case are of no practical use and therefore more efficient protocols are required. We focus on the problem of decision tree learning with the popular ID3 algorithm. Our protocol is considerably more efficient than generic solutions and demands both very few rounds of communication and reasonable bandwidth.

R. J. Bayardo and R. Agrawal

"Data privacy through optimal k-anonymization"

Data de-identification reconciles the demand for release of data for research purposes and the demand for privacy from individuals. This paper proposes and evaluates an optimization algorithm for the powerful de-identification procedure known as k-anonymization. A k- anonymized dataset has the property that each record is indistinguishable from at least k-1 others. Even simple restrictions of optimized k-anonymity are NP-hard, leading to significant computational challenges. We present a new approach to exploring the space of possible anonymizations that tames the combinatory of the problem, and develop data-management strategies to reduce reliance on expensive operations such as sorting. Through experiments on real census data, we show the resulting algorithm can find optimal -anonymizations under two representative cost measures and a wide range of k. We also show that the algorithm can produce good anonymizations in circumstances where the input data or input parameters preclude finding an optimal solution in reasonable time. Finally, we use the algorithm to explore the effects of different coding approaches and problem variations on anonymization quality and performance. To our knowledge, this is the first result demonstrating optimal k-anonymization of a non-trivial dataset under a general model of the problem.

IV. ALGORITHM AND TECHNIQUE USED

A.Privacy-Preserving Primitives

Here we present a set of generic sub-protocols that will be used in constructing our proposed k-NN protocol. All of the below protocols are considered under two-client semi-honest setting. In particular, we consider the presence of two semi honest clients P1 and P2 such that the Palliser's secret key s_k is known only to P2 whereas i_k is public.

1) *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.
2) *Secure Squared Euclidean Distance (SSED)*-In this protocol, p1 with input (E_{pk} (x), E_{pk}(y)) and p2 with s_k securely compute the encryption of squared Euclidean distance between vectors x and y. Here x and y are m dimensional vectors where E_{pk} (x)= {E_{pk}(x1),…,E_{pk} (x_m)} and E_{pk}(y)= {E_{pk}(y1),…,E_{pk} (y_m) }. The output Epk (|x-y|2) will be known only to p1.
3) *Secure Minimum (SMIN)* **-** In this protocol, P1holds private input (u′, v′) and P2 holds s_k, where u′ = ([u], Epk (s_u)) and v′ = ([v], Eik(s_v)). Here s_u (resp., s_v) denotes the secret associated with u(resp., v). The goal of SMIN is for P1 and P2 to jointly Here we present a set of generic sub-protocols that will be used in constructing our proposed k-NN protocol .All of the below protocols are considered under two-clients semi-honest setting. In particular, we consider the presence of two semi honest clients P1 and P2 such that the Palliser's secret key s_k is known only to P2 whereas ik is public.
4) *Secure Minimum out of n Numbers (SMINn)* **-** In this protocol, we consider P1 with n encrypted vector's ([d1], [dn]) along with their corresponding encrypted secrets and P2 with sk. Here [dp] = hE_{ik} (dp,1), . . . ,E_{ik}(dp,l)i where dp,1 anddi,l are the most and least significant bits of integer irrespectively, for 1 ≤ p≤ n. The secretor dp is given by sdi. P1 and P2 jointly compute [min (d1. . . dn)]. In addition, they compute E_{pk} (smin (d1,..., dn)). At the end of this protocol ,the output([min(d1,...,dn)],E_{pk}(smin(d1,...,dn)))is known only to P1. During SMINn, no information regarding any of dp's and their secrets is revealed to P1 and P2.
5) *Secure Bit-OR (SBOR)* - P1 with input (E_{pk} (o1), E_{pk} (o2)), and p2 securely compute E_{pk}(o1 v o2), where o1 and o2 are two bits. The output E_{pk} (o1 vo2) is known only to P1.
6) *Secure Frequency (SF)* - Here P1 with private input (hEik(c1), . . .Eik(cw)p, hE_{ik}(c′1), . . . ,E_{ik}(c'kp)p)and P2 securely compute the encryption of the frequency of cq , denoted by f(cq), in the listhc′1, . . . , c′kp, for 1 ≤ q≤ w. Here we explicitly assume that cq 's are unique and c′p ∈{ c1, . . . , cw}, for 1 ≤ p≤ k. The output Eik (f (c1)), . . . ,E_{ik}(f (cw))p will be known only to P1. During the SF protocol, no data regarding c′p, cq, and f (cq) is revealed to P1 andP2, for 1 ≤ p≤ k and 1 ≤ q≤ w.

B.Secure Minimum (SMIN)

In this protocol, P1holds private input (u′, v′) and P2 holds s_k, where u′ = ([u], E_{pk}(s_u)) and v′ = ([v],Eik(sv)). Here su (resp., sv) denotes the secret associated with u(resp., v). The goal of SMIN is for P1 and P2 to jointly Here we present a set of generic sub-protocols that will be used in constructing our proposed k-NN protocol .All of the below protocols are considered under two-clients semi-honest setting. In particular, we consider the presence of two semi honest clients P1 and P2 such that the Palliser's secret key s_k is known only to P2 whereas ik is public.

```
*Algorithm 1* SMIN(u′, v′) → [min(u, v)],$E_{pk}$(smin(u,v))
Require: P1 has u′ = ([u],$E_{pk}(s_u)$) and v′ = ([v],$E_{pk}(s_v)$), where
0 ≤ u, v < 2l; P2 has $s_k$
1: P1:
  (a) Randomly choose the functionality F
  (b) **for** i = 1 to l **do**:
        • $E_{pk}(u_i * v_i) \leftarrow SM(E_{pk}(u_i),E_{pk}(v_i))$
        • $T_i \leftarrow E_{pk}(u_i \times v_i)$
        • $Hi \leftarrow H^{ri}_{i-1*}T_i$; $r_i$ £R $Z_N$ and H0 = $E_{pk}(0)$
        • $\Phi i \leftarrow E_{pk}(-1) * H_i$
if F : u > v then:
   – $W_i \leftarrow E_{pk}(u_i) * E_{pk}(u_i * v_i)N-1$
   – $\Gamma_i \leftarrow E_{pk}(v_i - u_i) * E_{pk}(\hat{} r_i)$; ˆri £$_R$ $Z_N$
else
   – $W_i \leftarrow E_{pk}(v_i) * Epk(ui * vi)N-1$
   – $\Gamma_i \leftarrow E_{pk}(u_i - vi) *E_{pk}(\Phi r_i)$; ˆ$r_i$ £$_R$ $Z_N$
        • $L_i \leftarrow W_i *\Phi^{r'i}{}_i$ ; $r'_i$ £$_R$ $Z_N$
(c). if F : u > v then: $\delta \leftarrow E_{pk}(s_v - s_u) * E_{pk}(^{-}r)$
else
        $\delta \leftarrow E_{pk}(s_u - s_v)*E_{pk}(^{-}r)$, where ¯r £$_R$ $Z_N$
  (d). $\Gamma' \leftarrow \pi 1(\Gamma)$ and $L' \leftarrow \pi 2(L)$
  (e). Send δ, Γ′ and L′ to P2
2: P2:
  (a). Receive δ, Γ′ and L′ from P1
  (b). Decryption: $Mi \leftarrow D_{sk}(L'i)$, for 1 ≤ i ≤ l
  (c). if ∍ j such that Mj = 1 then α ← 1
else α ← 0
   (d). if α = 0 then:
        • $M'_i \leftarrow E_{pk}(0)$, for 1 ≤ i ≤ l
        • $\delta' \leftarrow E_{pk}(0)$
else
        • $M'_i \leftarrow \Gamma'_i * r^N$, where r £$_R$ $Z_N$ and is
```

```
                different for $1 \leq i \leq l$
            • $\delta' \leftarrow \delta * r^N_\delta$ , where $r £_R Z_N$
  (e). Send $M'$,$E_{pk}(\alpha)$ and $\delta'$ to P1
3: P1:
  (a). Receive $M'$,$E_{pk}(\alpha)$ and $\delta'$ from P2
  (b). $M \leftarrow \pi^{-1}{}_1 (M')$ and $\Theta \leftarrow \delta' * E_{pk}(\alpha)N-^{-}r$
  (c). $\lambda_i \leftarrow M_i * E_{pk}(\alpha)^{N-\hat{}ri}$ , for $1 \leq i \leq l$
  (d). if F : u > v then:
            • $E_{pk}(smin(u,v)) \leftarrow E_{pk}(s_u) * \Theta$
            • $E_{pk}(min(u, v)i) \leftarrow E_{pk}(u_i) * \lambda_i$, for $1 \leq i \leq l$
else
            • $E_{pk}(smin(u,v)) \leftarrow E_{pk}(s_v) * \Theta$ _
            • $E_{pk}(min(u, v)i) \leftarrow E_{pk}(v_i) * \lambda i$, for $1 \leq i \leq l$
```

once. Also, if _j = $E_{pk}(0)$, then index j is the position at which the bits of u and v differ first (starting from the most significant bit position).

Now, depending on F, P1 creates two encrypted, vectors W and □ as follows, for $1 \leq i \leq l$:

- If F : u > v, compute
  $W_i = E_{pk}(u_i * (1 - v_i))$
  $\Gamma_i = E_{pk}(v_i - u_i) * E_{pk}(\hat{}r_i) = E_{pk}(v_i - u_i + \hat{}r_i)$
- If F : v > u, compute
  $W_i = E_{pk}(v_i * (1 - u_i))$
  $\Gamma_i = E_{pk}(u_i - v_i) * E_{pk}(\hat{}r_i) = E_{pk}(u_i - v_i + \hat{}r_i)$

Where $\hat{}r_i$ is a random number (hereafter denoted by $£_R$) in Z_N. The observation is that
if F: u > v, then
$W_i = E_{pk}(1)$ iff $u_i > v_i$, and $W_i = E_{pk}(0)$ otherwise.
Similarly, when F: v > u, we have $W_i = E_{pk}(1)$ iff $v_i > u_i$, and $W_i = E_{pk}(0)$ otherwise.

Also, depending of F, Γi stores the encryption of the randomized difference between u_i and v_i which will be used in later computations.

TABLE 1
P1 chooses F as v > u where u = 55 and v = 58

u	v	W_i	Γ_i	G_i	H_i	Φ_i	L_i	$\Gamma^{i`}$	L'_i	M_i	λ_i	min i
1	1	0	r	0	0	-1	r	1+r	r	r	0	1
1	1	0	r	0	0	-1	r	r	r	r	0	1
0	1	1	-1+r	1	1	0	1	1+r	r	r	-1	0
1	0	0	1+r	1	r	r	r	-1+r	r	r	1	1
1	1	0	r	0	r	r	r	r	1	1	0	1
1	0	0	1+r	1	r	r	r	r	r	r	1	1

All column values are in encrypted form except Mi column. Also, r $£_R$ Z_N is different for each row and column.

If F: u > v, then min (u, v) i = (1 − α) * ui + α * vi always holds, for1 ≤ i ≤ l. On the other hand, if F: v > u, then min (u, v) i = α * u_i + (1 − α) * v_i always holds. Similar conclusions can be drawn for smin (u, v). We emphasize that using similar formulations one can also design a SMAX protocol to compute [max (u, v)] and E_{pk} (s_{max} (u, v)). Also, we stress that there can be multiple secrets of u and v that can be fed as input (in encrypted form) to SMIN and SMAX. For example, let s^1_u and s^2_u (resp., s^1_v and s^2_v) be two secrets associated with u (resp., v). Then the SMIN protocol takes ([u],$E_{pk}(s^1_u)$,$E_{pk}(s^2_u)$) and ([v],$E_{pk}(s^1_v)$,$E_{pk}(s^2_v)$)as P1's input and outputs [min(u, v)],E_{pk}(s1min(u,v))and E_{pk}(smin(u,v)) to P1.

c. Secure Minimum out of n Numbers (SMINn)

In this protocol, we consider P1 with n encrypted vector's ([d1], [d_n]) along with their corresponding encrypted secrets and P2 with s_k. Here[d_p] = h$E_{ik}(d_p,1)$, . . . ,E_{ik}(dp,l)i where dp,1 anddi,l are the most and least significant bits of integer irrespectively, for 1 ≤ p≤ n. The secretor dp is given by s_{di} . P1 and P2 jointly compute [min(d1, . . . , dn)]. In addition, they compute E_{pk}(smin(d1,...,dn)). At the end of this protocol ,the output ([min(d1, . . . , dn)],E_{pk}(smin(d1,...,dn)))is known only to P1. During SMINn, no information regarding any of dp's and their secrets is revealed to P1 and P2.

Algorithm 2 SMINn(([d1],E_{pk}(sd1)), .., ([dn],E_{pk}(sdn))) → ([dmin],E_{pk}(sdmin))

```
Require: P1 has (([d1],$E_{pk}$(sd1 )), . . . , ([dn],$E_{pk}$(sdn)));
P2 has $s_k$
1: P1:
  (a) $[d'_i] \leftarrow [d_i]$ and s′
        $i \leftarrow E_{pk}(sdi)$, for $1 \leq i \leq n$
  (b) num ← n
2: for i = 1 to [log2 n]:
  (a) for $1 \leq j \leq$ [num/2]:
   • if i = 1 then:
      – $([d'_{2j}-1], s'_{2j}-1) \leftarrow SMIN(x, y)$, where
        $x = ([d'_{2j}-1], s'_{2j}-1)$ and $y = ([d'_{2j}], s'_{2j})$
     – $[d'_{2j}] \leftarrow 0$ and $s'_{2j} \leftarrow 0$
else
    – $([d'_{2i}(j-1)+1], s'_{2i}(j-1)+1) \leftarrow SMIN(x,y)$,
where $x = ([d'_{2i}(j-1)+1], s'_{2i}(j-1)+1)$ and
          $y = ([d'_{2ij-1}], s'_{2ij-1})$
   – $[d'_{2ij-1]} \leftarrow 0$ and $s'_{2ij-1} \leftarrow 0$
  (b) num ← [num/2 ]
3: P1: $[d_{min}] \leftarrow [d'_1]$ and $E_{pk}(sd_{min}) \leftarrow s'_1$
```

D. Secure Frequency (SF)

Here P1 with private input (hE_{ik}(c1),...E_{ik}(c_w)p, hE_{ik}(c′1), . . . ,E_{ik} (c′k)p)and P2 securely compute the encryption of the frequency of cq , denoted by f(cq), in the listhc′1, . . . , c′kp, for 1 ≤ q≤ w. Here we explicitly assume that cq's are unique and c′p ∈{c1,..., cw}, for 1 ≤ p≤ k. The output E_{ik}(f(c1)),...,$E_{ik(}$f(cw))p will be known only to P1.

During the SF protocol, no data regarding $c'p$, cq, and $f(cq)$ is revealed to P1 andP2, for $1 \leq p \leq k$ and $1 \leq q \leq w$.

V. *PROPOSED METHODS*

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

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

- In this stage, User initially sends his query q (in encrypted form) to C1.
- After this, C1 and C2 involve in a set of sub-protocols to securely retrieve (in encrypted form) the class
- Labels corresponding to the k-nearest neighbors of the input query q.
- At the end of this step, encrypted class labels of k-nearest neighbors are known only to C1.

Algorithm 4 PPkNN(D′, q) → c_q

```
•   Require: C1 has D′ and _; C2 has $s_k$; Bob has q
•   1: Bob:
      (a) Compute Epk(qj), for $1 \leq j \leq m$
      (b) Send Epk(q) = h$E_{pk}$(q1), . . . ,$E_{pk}$(qm)i to C1
•   2: C1 and C2:
    (a) C1 receives $E_{pk}(q)$ from Bob
    (b) for i = 1 to n do:
              _$E_{pk}(d_i) \leftarrow$ SSED($E_{pk}(q)$,$E_{pk}$(ti))
             _[di] ← SBD($E_{pk}$(di))
•   3: for s = 1 to k do:
    (a) C1 and C2:
        ([dmin],Epk(I),Epk(c′))←SMINn(1, . . . , _n),
    where
          _i =([di],$E_{pk}$(Iti),Epk(ti,m+1))
          $E_{pk}(c's) \leftarrow E_{pk}(c')$
    (b) C1:_ ← $E_{pk}$(I)N−1
        for i = 1 to n do:
              – τi ← $E_{pk}(i)$ δ
              – $\tau'_i \leftarrow$ _ $r_i$
•   i , where $r_i$ £$_R$ $Z_N$
•   β← π(τ ′); send _ to C2

   (c) C2: _′i ← $D_{sk}$(_i), for $1 \leq i \leq n$
• Compute U′, for $1 \leq i \leq n$:
        – if _′i = 0, then U′
  i = $E_{pk}(1)$– otherwise, U′
  i = $E_{pk}(0)$
• Send U′ to C1
(d). C1: V ← τ−1(U′)
(e). C1 and C2, for $1 \leq i \leq n$ and 1 ≤  ≤ l:
   $E_{pk}$($d_i$,) ← SBOR($V_i$,Epk($d_i$,))
4: SCMCk($E_{pk}(c'_1)$, . . . ,$E_{pk}$(c′k))
```

Stage 2: Secure Computation of Majority Class ($SCMC_k$):

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

Algorithm 5 SCMCk(E_{pk}(c′)1), . . ,$E_{pk}(c_k)$) → c_q

```
•   Require: h$E_{pk}$(c1), . . . ,$E_{pk}(c_w)$i, h$E_{pk}$(c′1), . . .
    ,$E_{pk}$(c′k)i
•   are known only to C1; $s_k$ is known only to C2
•   1: C1 and C2:
     (a) h$E_{pk}$(f(c1)), . . . ,$E_{pk}$(f(cw))i ← SF(^,^′),
    where φ = h$E_{pk}$(c1), . . . ,$E_{pk}$(cw)i, ^′ =
          h$E_{pk}$(c′1), . . . , $E_{pk}$(c′k)i
    (b) for i = 1 to w do:
    • [f($c_i$)] ← SBD($E_{pk}(f(c_i))$)
(c) ([fmax],$E_{pk}(c_q)$) ← SMAXw(1, . . ,w),
    where
     i = ([f($c_i$)],$E_{pk}(c_i)$), for $1 \leq i \leq w$
•   2: C1:
   (a) $\gamma_q \leftarrow E_{pk}(c_q) * E_{pk}(r_q)$,
 where $r_q$ $\psi_R$ $Z_N$
    (b) Send q to C2 and $r_q$ to Bob
•   3: C2:
    (a)Receive q from C1
    (b) $\gamma'_q \leftarrow D_{sk}(q)$; send $\gamma'_q$ to Bob
•   4: Bob:
    (a) Receive rq from C1 and ′q from C2
    (b) cq ← $\gamma'_q - r_q$ mod N
```

Mathematical Model

- Let S is the Whole System Consist of
- S= {Q, PPKNN, D’, SRKNN, SCMCK, PPP}.
- Where Q is set of query entered by user.
- Q={q1, q2, q3,…..qn}.
- D’ = Encrypted Data set.
- PPKNN = process as privacy-preserving k-NN.
- SRKNN = Secure Retrieval of k-Nearest Neighbors.
- SCMCK = Secure Computation of Majority Class.

VI.*CONCLUSION AND FUTURE WORK*

To protect user privacy, various privacy-preserving classification techniques have been proposed over the past decade. The existing techniques are not applicable to outsourced database environments where the data resides in encrypted form on a third-party server. This paper proposed a novel privacy-preserving k-NN classification protocol over encrypted data in the cloud. Our protocol protects the confidentiality of the data, user’s input query, and hides the data access patterns. We also evaluated the performance of our protocol under different parameter settings. Since improving the efficiency of SMINn is an important first step for improving the performance of our PPkNN protocol; we plan to investigate alternative and more efficient solutions to the SMINn problem.In our future work. Also, we will investigate and extend our research to other classification algorithms.

REFERENCES

[1] P. Mell and T. Grance, “The nist definition of cloud computing (draft),” *NIST special publication*, vol. 800, p. 145, 2011.

[2] S. De Capitani di Vimercati, S. Foresti, and P. Samarati, “Managing and accessing data in the cloud: Privacy risks And approaches,” in *CRiSIS*, pp. 1 –9, 2012.

[3] P. Williams, R. Sion, and B. Carbunar, “Building castles out of mud: practical access pattern privacy and Correctness on untrusted storage,” in *ACM CCS*, pp. 139– 148, 2008.

[4] P. Paillier, “Public key cryptosystems based on composite Degree residuosity classes,” in *Eurocrypt*, pp. 223–238, 1999.

[5] B. K. Samanthula, Y. Elmehdwi, and W. Jiang, “k-nearest Neighbor classification over semantically secure Encrypted relational data.” eprint arXiv: 1403.5001, 2014.

[6] C. Gentry, “Fully homomorphic encryption using ideal lattices,”in *ACM STOC*, pp. 169–178, 2009.

[7] C. Gentry and S. Halevi, “Implementing gentry’s fully homomorphic Encryption scheme,” in *EUROCRYPT*, Pp. 129–148, Springer, 2011.

[8] A. Shamir, “How to share a secret,” *Commun. ACM*, vol. 22,pp. 612–613, Nov. 1979.

[9] D. Bogdanov, S. Lear, and J. Willemson, “Sharemind: A framework for fast privacy-preserving computations,” In *ESORICS*,Pp. 192–206, Springer, 2008.

[10] R. Agrawal and R. Srikant, “Privacy-preserving data mining,”in *ACM Sigmod Record*, vol. 29, Pp. 439–450, ACM, 2000.

[11] Y. Lindell and B. Pinkas, “Privacy preserving data mining,” in *Advances in Cryptology (CRYPTO)*, Pp. 36–54, Springer, 2000.

[12] P. Zhang, Y. Tong, S. Tang, and D. Yang, “Privacy preserving naive Bayes classification,” *ADMA*, Pp. 744–752, 2005.

[13] A. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke, “Privacy preserving mining of association rules,” *Information Systems*, vol. 29, no. 4, pp. 343–364, 2004.

[14] R. J. Bayardo and R. Agrawal, “Data privacy through optimal K-anonymization,” in *IEEE ICDE*, pp. 217–228, 2005.

[15] H. Hu, J. Xu, C. Ren, and B. Choi, “Processing private queries Over untrusted data cloud through privacy Homomorphism,”In IEEE ICDE, pp. 601–612, 2011.