PP-TAN: a Privacy Preserving Multi-party Tree Augmented Naive Bayes Classifier

Maria E. Skarkala*†, Manolis Maragoudakis‡§, Stefanos Gritzalis¶‖ and Lilian Mitrou***
*Department of Information and Communication Systems Engineering
University of the Aegean, Karlovasi, Samos 83200, Greece
‡ Department of Informatics, Ionian University, Corfu, Greece
¶Department of Digital Systems, University of Piraeus, Greece
Email: †mes@aegean.gr, §mmarag@ionio.gr, ‖sgritz@unipi.gr, **l.mitrou@aegean.gr

***Abstract*—The rapid growth of Information and Communication Technologies emerges deep concerns on how data mining techniques and intelligent systems parse, analyze and manage enormous amount of data. Due to sensitive information contained within, data can be exploited by potential aggressors. Previous research has shown the most accurate approach to acquire knowledge from data while simultaneously preserving privacy is the exploitation of cryptography. In this paper we introduce an extension of a privacy preserving data mining algorithm designed and developed for both horizontally and vertically partitioned databases. The proposed algorithm exploits the multi-candidate election schema and its capabilities to build a privacy preserving Tree Augmented Naive Bayesian classifier. Security analysis and experimental results ensure the preservation of private data throughout mining processes.**

***Index Terms*—Data mining, Distributed databases, Privacy preserving, Paillier cryptosystem, Homomorphic encryption, Tree augmented naive bayes**

I. Introduction

Databases containing private data (medical, social, financial, etc) distributed across several parties are exploited for the discovery of useful patterns. Voluminous data disseminated daily due to the rapid spread of Internet and Information and Communication Technologies. The uncontrollable growth, storage, retrieval and processing of data, has led to the decrease of useful information contained within statistical databases.

The General Data Protection Regulation (GDPR) [1] recognizes the need to facilitate the free flow of data, and promotes the protection of personal data. Information and Communication systems should be designed so that data protection is taken into consideration, in order to meet the requirements of the Regulation. Partitioned databases owners demand their privacy to be preserved as data mining techniques are applied to further analyze their private data [2], [3]. For example, some companies wish to collaborate in order to extract knowledge on market trends, on the premise their sensitive data will not be disclosed, mainly for competitiveness reasons.

Distributed databases can be either horizontally [4]–[7] or vertically partitioned [8], [9]. In horizontally partitioned databases, each party holds a different set of records but a unified set of attributes. On the contrary, each vertically partitioned database has different set of attributes for the same recordset [10], [11].

Various privacy preserving data mining techniques, such as randomization, perturbation and k-anonymity, have been proposed in the literature aiming to efface possible disclosure of sensitive information to expectant aggressors, while data mining processes are applied. Many data encryption techniques are based on the idea of Yao [12]. An extension of Yao's idea introduced by Goldreich [13] is also widely exploited. The basic idea is *"the computation of a function that accepts as input some data is secure if at the end of the calculation process neither party knows anything but their own personal data, which constitute one of the inputs, and the final results"*.

In a secure multi party computation protocol, a set of parties wish to jointly compute a function given their private data as input. A Trusted Third Party (Miner), collects the private data from all involved parties and performs all necessary calculations. The Miner forwards the final results to each party while providing privacy required in distributed environments. This protocol is secure only if neither party nor the Miner learn anything more than the output [14].

In this paper, we extend a privacy preserving data mining technique proposed by Skarkala et al. [15], which exploits the multi-candidate election schema [16] and aims to extract global information not only from horizontally but additionally from vertically partitioned statistical databases. For that purpose, a robust privacy preserving version of a Tree Augmented Naive Bayesian classifier was implemented. The homomorphic primitive, first proposed by Yang et al. [17] is exploited from Paillier cryptosystem [18] to preserve privacy. Based on this primitive, the Miner is unable to identify the original data included in the sharing databases or the database owners identity. The protocol allows the performance of all necessary operations only when at least three parties are connected with the Miner, making communication among them unfeasible.

The paper is structured as follows. A brief review of privacy preserving techniques is introduced in the next section. The theoretical background of the current proposal is presented in Section 3, while in Section 4 we describe the proposed protocol and its requirements. In the next sections, both types of database partition are evaluated, and the confrontation of possible threats to the proposed protocol follows. The basic

978-1-7281-6445-8/20/$31.00

conclusions of our work are presented at the last section.

II. Related Work

Privacy in data mining aims to prevent leakage of sensitive data without undermining the extracted knowledge produced by the application of the data mining process [19].

Privacy preserving data mining algorithms can be categorized in five segments [9], i.e. apportionment of data, modification of data, data mining algorithm, type of data and technique for preserving privacy. Qi and Zong [20] illustrate evaluation criteria and review privacy protection technologies in data mining. Malik et al. [21] also discuss the evaluation parameters and the trade off between privacy and utility. The authors in [22] define different parameters to quantify the trade-off between privacy and information loss in order to create a framework for evaluating privacy preserving data mining algorithms. Bertino et al. [23] identify a set of criteria, such as privacy level, hiding failure, data quality and complexity, to evaluate the effectiveness of privacy preserving data mining algorithms.

Most existing privacy preserving data mining methodologies can be classified into two main categories [19]; methodologies that protect the input data in the mining process, and methodologies that protect the final data mining results. In the first approach, techniques are applied to the input data in order to hide any sensitive information and safely distribute the data to other parties. The goal is to generate accurate data mining results in a distributed environment. In the second approach, the applied techniques prohibit the disclosure of sensitive knowledge derived through the application of data mining algorithms.

One major question that should be answered in every methodology is "*Do the results themselves violate privacy?*" as defined by Kantarcioglu, Jin and Clifton [24]. The authors propose a model for privacy implication of the learned classifier, and within this model they study possible ways in which the classifier can be used by an attacker to compromise privacy. However, they do not provide a solution that prevents an attacker from accessing the data mining results and thus violate privacy. Scardapane et al [25] consider the analysis of medical data distributed in multiple parties. Such environments may apply privacy protocols that forbid to disclose their local data to a centralized location.

Randomization and cryptography are the most widely studied privacy preservation techniques. Huai et al. [26] constructed differentially private protocols for distributed data used to extract knowledge through Naive Bayes learning techniques. Liu et al. [27] proposed an approach where multiplicative perturbations are applied on the data for introducing noise. These techniques however do not assure the quality of the final results. Also, the authors in their privacy analysis did not take into account prior knowledge.

Vaidya et al. [28] focus on generating privacy preserving results instead of sharing secure data sets. They apply differential privacy to develop a Naive Bayes classifier provided as a cloud service. However, those techniques only focus on publishing useful results and not sanitized data that can be shared.

Randomization is used in association rules [29] and decision trees [30] for vertically and horizontally partitioned databases respectively. Although this method is efficient, results to inaccurate outcomes. Kargupta et al. [31] reveal that randomization techniques may compromise the privacy and special attacks can result to the reconstruction of the original data, as they point out that additive noise can be easily filtered out. For example, Zhang et al [32] proposed a randomization technique that combines data transformation and data hiding, exploiting a privacy preserving modified Naive Bayes classifier to predict the class values on the distorted data.

On the other hand, cryptographic techniques are more secure providing accurate results, but in many cases they lack efficiency. Cryptography is applied in models [9], [33] which most of them are based on the idea of Yao [12], and an extension proposed by Goldreich [13], who studied the secure multi-party computation problem.

Several privacy preserving techniques that have been proposed in the literature encrypt the data within horizontally partitioned databases for building Decision Trees [34], [35], Naive Bayesian classifiers [5], [17], [36], [37], and Association Discovery Rules [4]. This technique was also applied to vertically partitioned databases to create Association Rules [6], [33] and Naive Bayesian classifiers [36], [38]. Tassa [39] proposed a protocol for secure mining of Association Rules for horizontally partitioned databases. The author presented the leverages of the proposal over existing protocols [4].

The method proposed by Goethals et al. [40] is both simple and secure. The key idea behind the protocol is to use a homomorphic encryption system such as the Paillier cryptosystem.

Keshavamurthy et al. [41] propose as well a secure multi party approach to compute the aggregate class for vertically partitioned data using the Naive Bayes classifier. Many researches [5], [17], [37], [42] utilize Naive Bayes classification because of its simplicity and straightforward approach. Simplified Bayesian Networks have also been used for data mining processes by either applying the Tree Augmented Naive Bayes [43] or K2 algorithm [6].

A similar proposal [43] to ours, uses an algebraic technique to perturb the original data. On the other hand, our protocol uses cryptographic techniques, which can assure privacy and result to accurate outcomes.

III. Background

A. Classification of nominal and numeric attributes

Classification aims to predict the value of an attribute by estimating the probabilities from a training set. The calculation of the probabilities differs for numeric and nominal attributes.

For nominal attribute X, with values $x_1,\ldots,x_r$, the probability for each value is $P(X = x_k|u_j) = n_j/n$, where n is the total number of training instances for which $V = u_j$ and n_j is the number of the ones that have $X = x_k$. The conditional probability that an instance belongs to a certain class c is

calculated by (1), where n_{ac} is the number of instances with class value c and attribute value a, while n_a is the number of instances with attribute value a.

$$P(C = c|A = a) = \frac{P(C = c \cap A = a)}{P(A = a)} = \frac{n_{ac}}{n_a} \quad (1)$$

For a numeric attribute, the mean μ and variance σ^2 parameters are calculated for each class and each numeric attribute. The probability that an instance is of class u_j, denoted as $P(X = x'|u_j)$, can be estimated by substituting $x = x'$ in the probability density equation. The conditional probability of a class given the instance is calculated for all classes and the class with the highest relative probability is chosen as the class of this instance. In order to compute the mean μ for a class value, these local sums are added together and divided by the total number of instances having that same class. Since each party knows the classification of the training instances, it can subtract the appropriate mean μ from an instance having class value y, square the value, and sum all such values together. The global sum divided by the global number of instances having the same class y gives the required variance.

The normal probability distribution is computed in (2), where x is a random variable, μ is the mean of the distribution and σ is the standard deviation (σ^2 is the variance), π is approximately 3.14159 and e is approximately 2.71828.

$$P(x) = \frac{1}{\sigma * sqrt(2\pi)} * e^{\frac{-(x-\mu)^2}{2\sigma^2}} \quad (2)$$

B. Tree-Augmented Naive Bayesian Classifier

Traditional Naive Bayes, a classifier that supports both nominal and numeric attribute values, computes the conditional probability of each attribute A_i given the class C. Bayes theorem is applied to compute the probability of class C given a specific instance vector $< A_1.....A_n >$, where n is the total number of attributes. However, this classifier is based on a sometimes unrealistic assumption, leading to poor prediction outcomes in some domains [44], since it does not take into consideration any prior knowledge on the class variable C. The classifier assumes that all attributes are independent given the value of C, a restrictive and oversimplified assumption. The performance of such classifiers can be improved by effacing this assumption.

An interesting variation of Bayesian networks, is the Tree-Augmented Naive Bayesian (TAN) classifier [45]. This classifier, unlike the Naive Bayesian, is not based on the unrealistic assumption that attributes are independent. The classifier allows the existence of additional edges between attributes that represent the correlation among them. In a TAN network the class C has no parents and each attribute A_i has as parents the class and at most one other attribute A_j. In an augmented structure, an edge from attribute A_i to A_j implies that the influence of A_i on the assessment of the class also depends on the value of A_j. TAN removes any independence assumptions, improving classical Naive Bayes classifier and behaves more robust regarding classification since it combines the initial structure of the Naive Bayes algorithm with prior knowledge (if available) or obtained knowledge about the correlation of input features via a training approach.

C. Homomorphic primitive

A tool widely used in the literature is the homomorphic primitive, whereby the result of encrypting two messages is equal to the sum of the two messages separately encrypted (3).

$$E(M1 \otimes M2) = E(M1) \otimes E(M2) \quad (3)$$

This primitive was first used in the work of Yang et.al. [17] in order to build a privacy preserving data mining model in a distributed environment.

D. Paillier cryptosystem

The Paillier algorithm exploits the additive homomorphic primitive [36] to achieve anonymity and unlinkability between parties and personal data. In the proposed algorithm, a random variable M, which is computed by the Miner and used to confront any chosen-plaintext attacks, is delivered to each party encrypted with their own public key and used for encrypting the transmitted messages.

More specifically, if a party j wishes to send to the Miner the frequency i, then he encrypts the message with the Miner's public key. When at least three parties have sent their data to the Miner, the homomorphic primitive is applied to calculate the total frequencies of each possible attribute value in relation to each class value by decrypting the messages received all at once. The Miner cannot associate the frequencies obtained with the original records and link the data to their owners, since the decryption process occurs only after the participation of minimum three parties.

IV. Protocol Description

The objective of the current work is to develop a secure protocol by exploiting efficient encryption that satisfies the essential security and design requirements. Global and accurate information is extracted using Tree Augmented Naive Bayesian classification algorithm [46], while privacy is preserved. Encryption processes are applied to a client-server (party-Miner) environment ensuring that any transmitted message is not accessible by unauthorized parties, in a fully distributed environment.

The Miner's objective is to generate the classification model by collecting, from at least three parties, the frequencies of each attribute value in relation to each class value of the horizontally or vertically partitioned database. In vertically partitioned databases we assume that every participant is aware of the class value of each record. Attributes can have either nominal (Fig. 1) or numeric (Fig. 2) values, including binary data. The frequencies are encrypted using Paillier cryptosystem, which exploits the homomorphic primitive, ensuring sensitive data remain secret. The only data flow is only between each party and the Miner.

for $c_1 \dots c_m$ class values **do**
  for $a_1 \dots a_i$ attribute values **do**
    for each party $1 \dots n$ **do**
      1. compute # instances f_{im} with class value m and attribute value i
      2. compute # instances f_m^n with class value m
    end for each
    Miner computes using homomorphic primitive:
    1.

$$E(f_{mi}^1 \otimes f_{mi}^2 \otimes \cdots \otimes f_{mi}^n) = E(f_{mi}^1) \otimes E(f_{mi}^2) \otimes \cdots \otimes E(f_{mi}^n)$$

    2.

$$E(c_m^1 \otimes c_m^2 \otimes \cdots \otimes c_m^n) = E(c_m^1) \otimes E(c_m^2) \otimes \cdots \otimes E(c_m^n)$$

  end for
  Miner computes:

$$P_{im} = \frac{E(f_{mi}^1 \otimes f_{mi}^2 \otimes \cdots \otimes f_{mi}^n)}{E(c_m^1 \otimes c_m^2 \otimes \cdots \otimes c_m^n)}$$

end for

Fig. 1. Protocol for Nominal Attribute Values

As mentioned, the current work is an extension of a previous research [15] and is based on the study by Mangos et al. [47]. Thus, among the features used and the requirements to be met, they spring up from their quotations.

A. Security requirements

The GDPR [1] requires appropriate measures to implement the data protection principles and safeguard individual rights. The Privacy by Design approach ensures privacy and data protection are taken into consideration at the design phase of any system and throughout the entire lifecycle. This approach highly impacted the implementation of the current proposal, and all proper measures were operated in all phases, as described in this section.

Each party in a distributed environment can be considered either semi-honest or malicious. Semi-honest parties are curious to learn more information, but they follow the protocol specifications. On the other hand, malicious parties can be categorized to internal and external. An internal adversary deviates from the protocol by sending specific inputs, with the purpose to recognise other parties private data. An external adversary will impersonate a legal party and then behave as an internal. In the current protocol we consider both adversary types.

The Miner and each party are mutually authenticated, by sending their digital signatures, assuming they were signed by a Certification Authority (CA), in order to confront such behaviors. Thus, only authorized parties can participate to the protocol and connect with the literal Miner.

for $c_1 \dots c_m$ class values **do**
  for party $1 \dots n$ **do**
    1. compute # instances f_m with class value c_m
    2. compute sum of instances s_m^n with c_m
  end for
  Miner computes using homomorphic primitive:
  1. Total sum s_m

$$E(s_m^1 \otimes s_m^2 \otimes \cdots \otimes s_m^n) = E(s_m^1) \otimes E(s_m^2) \otimes \cdots \otimes E(s_m^n)$$

  2. Total # instances N_m

$$E(f_m^1 \otimes f_m^2 \otimes \cdots \otimes f_m^n) = E(f_m^1) \otimes E(f_m^2) \otimes \cdots \otimes E(f_m^n)$$

  3. Mean

$$\mu_m = \frac{s_m}{N_m}$$

end for
for $c_1 \dots c_m$ class values **do**
  for party $1 \dots n$ **do**
    for each instance y **do**
      1.

$$u_{mn}^i = x_{mn}^i - \mu_m$$

      2.

$$u_{mn}^i = \sum_y (u_{mn}^2)$$

    end for each
  end for
  Miner compute variance:
  1.

$$u_m = E(u_m^1 \otimes u_m^2 \otimes \cdots \otimes u_m^n) = E(u_m^1) \otimes E(u_m^2) \otimes \cdots \otimes E(u_m^n)$$

  2.

$$\sigma_m^2 = u_m * \frac{1}{N_m - 1}$$

end for

Fig. 2. Protocol for Numeric Attribute Values

If confidentiality, anonymity and unlinkability are fulfilled, then privacy is preserved. Asymmetric encryption ensures that all transmitted data between one party and the Miner are encrypted, and only the party to whom the message is intended for can decrypt it. Anonymity and un-linkability can be achieved as the Miner, through the homomorphic primitive, cannot identify the inputs that each party submits. Sensitive data remain secret and the identity of each party is not revealed. Integrity mechanisms are implemented in case any active attacker tries to modify the transmitted messages, and cause variations to the final results or disclosure sensitive data. In every transmitted message, an SHA-1 digest is concatenated assuring any altered message can be detected.

B. Protocol analysis

The current work presents a protocol which utilizes the Paillier cryptosystem that follows the homomorphic model through which privacy preservation and mining processes are combined in a fully distributed environment. Our approach is based on the classical homomorphic election model and particularly on an extension for supporting the multi-candidate election scheme, where each party has k-out-of-1 selections [16].

The digital signature scheme is exploited in order both the Miner and each party to be mutually authenticated, given that each one possesses a key pair. The public keys are created during the generation phase of Paillier cryptosystem, and later exchanged in order all transmitted messages are encrypted.

The Miner is considered a trusted third party who regroups all data send by the participants of the protocol. The Miner, after obtaining the encrypted data from at least three participants, exploits the homomorphic primitive provided by Paillier cryptosystem and applies the Tree Augmented Naive Bayesian classifier to find the correlation among the attributes and the network structure that represents them. The final results will be sent later to each participant who contributed to the creation of the mining model. The final results represent the frequencies of each possible value of all attributes of horizontally or vertically partitioned databases in relation to each class value. In case the databases are vertically partitioned we assume that every participant is aware of the class value of each database record.

The proposed protocol is divided into six phases and carried out for horizontally and vertically partitioned databases.

1) Key generation: The Miner's encryption key pair (S_{pu} and S_{pr}) is generated through Paillier cryptosystem (key generation phase), and an RSA key pair (S_{Dpu} and S_{Dpr}) of 1024 bit using MD5 hash function to create the digital signature. Every party follows the same procedure (C_{pu} / C_{pr}, and C_{Dpu} / C_{Dpr}). We assume that the RSA keys are signed by a Certificate Authority and each side is able to obtain the public key of the other side. After exchanging the public keys, all transmitted messages are encrypted. A random value M is generated, which will be sent in later phase encrypted to every party and will be used during the encryption of personal data.

2) Mutual authentication: During the authentication phase, when a participant requests connection to the Miner, sends the public key C_{pu} and the digital signature encrypting the C_{pu} key with the private key C_{Dpr}. The Miner decrypts the digital signature with the participants public key C_{Dpu} and creates a digest of the message. In return, if the Miner verifies the party's identity, sends his public key S_{pu} and digital signature encrypted with the private key S_{Dpr}. Now the participant is able to verify that a connection with the legal Miner is accomplished. The purpose of this phase is to prevent any unauthorized access and participation to the protocol.

3) Data collection: Next, the forwarding of the random value M encrypted with the public key C_{pu} of each party, takes place. The phase of collecting the data from each party starts from the Miner's side. The Miner requests the frequencies for attribute A_i. Each party encrypts, with the Miner's public key S_{pu}, the frequency of each value for this specific attribute in relation to every class value. In case the databases are horizontally partitioned the party sends all the possible attribute values. In case the databases are vertically partitioned and the party does not possess A_i then returns zero. The frequencies of each attribute value in relation to each possible class value are the only sensitive data send by each participant.

4) Classifier Initialization: After the collection of the encrypted frequencies that correspond to attribute A_i, the Miner proceeds to their decryption applying the homomorphic primitive. The frequencies are decrypted all at once and the Miner receives the overall distributions of each possible value of A_i related to each value of C. The Miner requires the frequencies for the next attribute A_{i+1} and this process continues for all attributes A_n, where n is the total number of attributes of the horizontally partitioned databases or the sum of each participants number of attributes for the vertically partitioned databases. These procedures are necessary for the Miner to initialize the TAN classifier.

5) TAN classifier creation: When at least three participants are connected with the Miner and participate in the execution of the protocol, the classifier is created.

6) Final results: The Miner after the completion of the above phases, delivers the final results of the mining process encrypted to all participants using the public key C_{pu} of each party.

V. Evaluation

By examining the main procedures of the proposed protocol, our aim is to evaluate it in terms of computational cost and security. The main procedures of the protocol are the classifier's initialization, the data collection from the Miner, the TAN classifier creation given these data and the delivery of the final results from the Miner to each party. For that purpose, three different scenarios were established.

In each scenario, all three parties participate with their own data contained within horizontally or vertically partitioned databases. The purpose of these scenarios is to be evaluated and compare the performance of the protocol given different number of records and attributes. The data was accrued from UC Irvine Machine Learning Repository real dataset [48], and tailored for each scenario. The training set size was selected between 1000 records, 2000 records and 5000 records. All results in Table I, tabulate the mean time to complete each of the proposed protocol procedure, calculated in milliseconds (ms), on a modest PC equipped with Intel i5 2.4Ghz, 4GB of RAM.

The overall execution time of all the main procedures of the protocol is determined mainly by the collection of the data, which is proportional to the number of attributes. By comparing both experiment cases, we can conclude that the partition of the databases affects mainly the data collection process when the number of instances is growing.

A. Key establishment

The mean key generation time and the mean authentication time were measured by collecting measurements from 50 runs performed for one party and the Miner. The key generation phase includes the encryption key pair generation and the creation of the RSA digital signature. We assume that each participant knows the Miner's S_{Dpu} key and the Miner is aware of all public keys C_{Dpu} of the parties involved in the mining process. A participant requires 479 ms and 122 ms to create the encryption key pair and the digital signature, respectively. The Miner requires 433 ms for the encryption key pair, 108 ms to create the digital signature, and 43 ms to generate the random variable M used by Paillier cryptosystem during encryption and decryption processes. From the results we can conclude that the asymmetric encryption algorithm is efficient in terms of creation and establishment of keys. The authentication time represents the time needed by the Miner and each party to be mutually authenticated, by sending their public keys and their digital signatures. From the measurements we calculate that 24 ms are needed for the mutual authentication.

B. Experiments: horizontally partitioned databases

The experiments carried out for horizontally partitioned databases, were evaluated using the following three scenarios:

Scenario 1. Each database has 50 records and 5 attributes

Scenario 2. Each database has 100 records and 5 attributes

Scenario 3. Each database has 100 records and 10 attributes

From the results, we can conclude that the mean time to initialize the classifier is low, but when the number of attributes is increased, the mean time is affected. However, when the number of instances is growing, the mean initialization time is slightly raised. The same conclusions also apply when the data are collected by the Miner. This process however has high execution time, as each party has to send all their data to the Miner. On the other hand, the mean time for the creation of the TAN model is increased when the number of instances is growing unlike the mean time increment when the attributes number is doubled. The mean time to send the final results to each party, is affected by both the number of attributes and instances in a database.

C. Experiments: vertically partitioned databases

For vertically partitioned databases, the experiments that took place were evaluated using the following three scenarios:

Scenario 1. Each database has 50 records and 3 attributes

Scenario 2. Each database has 100 records and 3 attributes

Scenario 3. Each database has 100 records 6 attributes

In each scenario we assume that all parties are aware of the class C. For the first scenario all the mean times needed for the main procedures are slightly higher in comparison to the first scenario using horizontally partitioned databases. Data collection phase is almost doubled, due to the partition of the data. When the number of instances is getting larger, the creation of TAN classifier and the collection of data are mainly increased, similar to horizontally partitioned databases. On the other hand, when the number of attributes is increased for vertically partitioned databases, the data collection phase requires more time in relation to the second scenario, but less time compared to horizontally partitioned databases. The mean initialization time of the classifier is fairly increased and almost doubled in relation to horizontally partitioned databases. The TAN classifier creation requires less time for the second scenario, but the mean time is higher than the corresponding scenario for horizontally partitioned databases. However, for vertically partitioned databases, the mean delivery time of the final results is mainly affected when the number of attributes is doubled.

D. Cryptosystem performance

Because of the different number of characters involved in the messages being exchanged during the execution of the protocol we collected from all the above executed scenarios all the encryption and decryption mean times. Our measurements showed that the average time to encrypt a message is equal to 51.5 ms. Similar results obtained about the measurement of the mean decryption time. The average decryption time that resulted is 67 ms. As the mean times are low, we can conclude that the Paillier cryptosystem is efficient.

E. Classifier evaluation

Variables Recall and Precision were calculated in order to examine the mining model created by the Miner. Variable Recall is the percentage of records categorized with the correct class in relation to the number of all records with this class. Variable Precision is the percentage of records that have truly a certain class over all the records that were categorized with this class. Three set of data, with different number of instances, were used as training sets (1000 records, 2000 records and 5000 records). The databases contained 14 attributes and come from a real dataset [48]. As test set, 100 records (10% of the training records) were used, which were kept off the training phase. Table II presents the results of the TAN classifier evaluation. Naive Bayes classifier evaluation is also demonstrated for comparison of both classifiers and the results are presented in Table III.

VI. Threat model

In distributed environments, some attacks depend on whether one [7], [17] or more Miners [37] are involved, and whether personal data are exchanged among two [6], [29], [34] or many parties [4], [5], [30], [33]. In a protocol with only one Miner the final results can be revealed to him, but in case more than one Miners exist, the protocol is vulnerable to collusion attacks [43]. When parties exchange data directly to each other, in the two party model, each party can easily discover the other party's private data, but in the many parties model, a malicious one can modify the data, given as input. In a distributed environment, this can be disastrous if $n-1$ users collaborate. To prevent these behaviors in the proposed protocol, participants cannot communicate to each other, and the number of parties involved must be at least three in order to prevent any probing attacks. Data is transmitted only between

TABLE I
MAIN PROCEDURES COMPARISON FOR EACH SCENARIO

Procedure	1st horizontal	1st vertical	2nd horizontal	2nd vertical	3rd horizontal	3rd vertical
Data collection	31777	58939	35502	59764	94793	89073
Classifier initialization	13	57	16	56	30	64
TAN classifier creation	39	52	117	118	68	110
Final results	2407	3411	3744	3592	4455	6076

TABLE II
TAN CLASSIFIER EVALUATION RESULTS

Records	1000		2000		5000	
Correct	54		55		56	
Incorrect	46		45		44	
Class value	≤ 50	> 50	≤ 50	> 50	≤ 50	> 50
Recall	0.42	0.63	0.52	0.6	0.54	0.6
Precision	0.48	0.57	0.73	0.38	0.73	0.39

TABLE III
NAIVE BAYES CLASSIFIER EVALUATION RESULTS

Records	1000		2000		5000	
Correct	49		49		50	
Incorrect	51		51		50	
Class value	≤ 50	> 50	≤ 50	> 50	≤ 50	> 50
Recall	0.42	0.54	0.48	0.52	0.50	0.8
Precision	0.43	0.53	0.77	0.23	0.47	0.2

the Miner and each party, so collusion attacks are confronted. The case in which parties collaborate outside the protocol is not considered in the present work.

The proposed protocol provides mutual authentication, thus participants with no permission to connect to the Miner are not able to participate to the protocol and authorized ones are connected with the literal server. Possible man-in-the-middle attacks are faced by exploiting the digital signatures of each side, signed by a Certification Authority. Eavesdropping attacks are confronted as the asymmetric encryption fulfills the requirement of confidentiality.

The parties involved usually are considered to be mutually mistrustful. If a party does not deviate from a protocol it is considered semi-honest, but in case it tries to discover other party's data, it is considered malicious. In real world applications, the former case behaviors are more often and more realistic; all participants have mutual interest to follow a protocol. Semi-honest adversaries are faced in our protocol as the only information revealed and send by the Miner are the final outcomes. The Miner could be also considered an internal adversary. The homomorphic primitive, both for nominal and numeric attribute values, ensures that the original data will not be revealed to any attacker or the Miner.

Active attackers trying to modify the transmitted messages and alter the final results or disclose sensitive data, are confronted using integrity mechanisms (SHA-1). Our protocol is designed so participants are able to send only once their data (blank or missing inputs are excluded) and can run the protocol only once per computer system, confronting denial of service attacks.

Paillier cryptosystem at its initial mode is vulnerable to chosen plaintext attacks. The usage of a random variable (variable M in the current protocol) is important to confront such attacks.

VII. CONCLUSION

Various data mining techniques have been introduced with regards to the detection and prediction of important information that is hidden within statistical databases. Due to the fact that such databases often contain sensitive data, their disclosure throughout mining processes can compromise privacy. The current work aims to solve this problem by presenting a properly designed privacy preserving data mining technique, developed for an environment in which distributed databases can either be horizontally or vertically partitioned. A trusted third party (Miner) conducts all operations, by collecting data from at least three parties. Through Paillier cryptosystem, data exchanged during the execution of the protocol is encrypted. The Miner decrypts all received data using the homomorphic primitive, assuring at the same time the privacy of the individuals. The cryptographic approach is considered the most appropriate in terms of accuracy. The evaluation of conducted experiments results to an effective but also efficient protocol for both database partition types. In the future, ensemble methods such as Random Forests or Gradient Boosting Machines could be exploited and compared with the current proposal, in order to find the most efficient and accurate algorithm. A research on other cryptosystems, like El-Gamal's elliptic curve cryptography, could also be a future research in order to strike a balance between security and efficiency.

REFERENCES

[1] General Data Protection Regulation (GDPR). (2018). Available at:https://gdpr-info.eu/ [Accessed 31 Oct. 2018].

[2] Clifton C (2001) Privacy Preserving Distributed Data Mining. In: 13th European Conference on Machine Learning, pp 19-23.

[3] Clifton C, Marks D (1996) Security and Privacy Implications of Data Mining. In Proceedings of the 1996 ACM SIGMOD Workshop on Data Mining and Knowledge Discovery, Montreal, Canada, pp 15-19.

[4] Kantarcioglu M, Clifton C (2004) Privacy preserving distributed mining of association rules on horizontally partitioned data. IEEE Transactions on Knowledge and Data Engineering 16(9):1026-1037.

[5] Kantarcioglu M, Vaidya J (2003) Privacy Preserving Naive Bayes Classifier for Horizontally Partitioned Data. IEEE ICDM Workshop on Privacy Preserving Data Mining, pp 3-9.

[6] Wright R, Yang Z (2004) Privacy-Preserving Bayesian Network Structure Computation on Distributed Heterogeneous Data. In Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), Seattle, WA, USA, pp 713-718.

[7] Zhan J, Matwin S, Chang L (2008) Privacy-Preserving Naive Bayesian Classification over Horizontally Partitioned Data. Data Mining: Foundation and Practice, 118:529-538.

[8] Sweeney L (2002) k-Anonymity: a model for protecting privacy. International Journal on Uncertainty, Fuzziness and Knowledge-based systems 10(5):557-570.

[9] Verykios V, Bertino E, Fovino I, Parasiliti Provenza L, Saygin Y, Theodoridis Y (2004) State-of-the-art in privacy preserving data mining. ACM SIGMOD Record 33(1):50-57.

[10] Aggarwal CC, Yu PS (2008) A General Survey of Privacy-Preserving Data Mining Models and Algorithms. In C. C. Aggarwal, & P. S. Yu, Privacy-Preserving Data Mining, pp 11-52.

[11] Agrawal D, Aggarwal C (2001) On the Design and Quantification of Privacy Preserving Data Mining Algorithms. In: 12th ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, pp 247-255.

[12] Yao ACC (1986) How to generate and exchange secrets. In: 27th Annual Symposium on Foundations of Computer Science, pp 162-167.

[13] Goldreich O (1998) Secure multi-party computation. Working Draft.

[14] Lindell Y, Pinkas B (2009) Secure multiparty computation for privacy-preserving data mining. J. Privacy Confidentiality, 1(1): 59-98.

[15] Skarkala ME, Maragoudakis M, Gritzalis S, Mitrou L (2011) Privacy preserving tree augmented Naive Bayesian multi-party implementation on horizontally partitioned databases. In Proc. 8th Int. Conf. Trust, Privacy, Security Digit. Bus, pp. 62-73.

[16] Baudron O, Fouque PA, Pointcheval D, Stern J, Poupard G (2001) Practical multi-candidate election system. In: PODC'01: Proceedings of the twentieth annual ACM symposium on Principles of distributed computing, ACM, New York, USA, pp 274-283.

[17] Yang Z, Zhong S, Wright R (2005) Privacy-preserving classification of customer data without loss of accuracy. In: SDM'2005 SIAM International Conference on Data Mining. DOI: https://doi.org/10.1137/1.9781611972757.9.

[18] Paillier P (1999) Public-key cryptosystems based on composite degree residue classes. In: Advances in Cryptography - EUROCRYPT 99, pp 223-238.

[19] Gkoulalas-Divanis A, Verykios VS (2009) An overview of privacy preserving data mining. XRDS 15, 4, Article 6 (June 2009), 4 pages. DOI: https://doi.org/10.1145/1558897.1558903.

[20] Qi X, Zong M (2012) An Overview of Privacy Preserving Data Mining, Procedia Environmental Sciences, Volume 12, Part B, 2012, Pages 1341-1347.

[21] Malik MB, Ghazi MA, Ali R (2012) Privacy preserving data mining techniques: Current scenario and future prospects. In Proc. 3rd Int. Conf. Comput. Commun. Technol. (ICCCT), Nov. 2012, pp. 26-32.

[22] Bertino E, Fovino IN, Provenza LP (2005) A framework for evaluating privacy preserving data mining algorithms. Data Mining and Knowledge Discovery 11(2), 121-154.

[23] Bertino E, Lin D, Jiang W (2008) A Survey of Quantification of Privacy Preserving Data Mining Algorithms. In: Aggarwal C.C., Yu P.S. (eds) Privacy-Preserving Data Mining. Advances in Database Systems, vol 34. Springer, Boston, MA.

[24] Kantarcioglu M, Jin J, Clifton C (2004). When do data mining results violate privacy?. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '04). ACM, New York, NY, USA, 599-604. DOI:http://dx.doi.org/10.1145/1014052.1014126.

[25] Scardapane S, Altilio R, Ciccarelli V, Uncini A, Panella M (2018) Privacy-Preserving Data Mining for Distributed Medical Scenarios. In: Esposito A., Faudez-Zanuy M., Morabito F., Pasero E. (eds) Multidisciplinary Approaches to Neural Computing. Smart Innovation, Systems and Technologies, vol 69. Springer, Cham.

[26] Huai M, Huang L, Yang W, Li L., Qi M (2015) Privacy-Preserving Naive Bayes Classification. In Proceedings of the 8th International Conference on Knowledge Science, Engineering and Management - Volume 9403 (KSEM 2015), Songmao Zhang, Martin Wirsing, and Zili Zhang (Eds.), Vol. 9403. Springer-Verlag, Berlin, Heidelberg, 627-638.

[27] Liu k, Kargupta H, Ryan J (2006) Random projection-based multiplicative data perturbation for privacy preserving distributed data mining. In IEEE Transactions on Knowledge and Data Engineering, vol. 18, no. 1, pp. 92-106, DOI: 10.1109/TKDE.2006.14.

[28] Vaidya J, Shafiq B, Basu A, Hong Y (2013) Differentially private naive bayes classification, IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), vol. 1, pp. 571-576, 2013.

[29] Vaidya J, Clifton C (2002) Privacy preserving association rule mining in vertically partitioned data. In: 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp 639-644.

[30] Agrawal R, Srikant R (2000) Privacy-preserving data mining. In: 2000 ACM SIGMOD Conference on Management of Data 29(2):439-450.

[31] Kargupta H, Datta S, Wang Q, Sivakumar K (2003) On the privacy preserving properties of random data perturbation techniques. In: Proceedings of the Third IEEE International Conference on Data Mining (ICDM03). Melbourne, Florida.

[32] Zhang P, Tong Y, Tang S, Yang D (2005) Privacy Preserving Naive Bayes Classification. In: Li X., Wang S., Dong Z.Y. (eds) Advanced Data Mining and Applications. ADMA 2005. Lecture Notes in Computer Science, vol 3584. Springer, Berlin, Heidelberg.

[33] Clifton C, Kantarcioglu M, Vaidya J, Lin X, Zhu MY (2002) Tools for Privacy Preserving Distributed Data Mining. ACM SIGKDD Explorations 4(2):28-34.

[34] Lindell Y, Pinkas B (2002) Privacy Preserving Data mining. Journal of cryptology 15(3): 177-206.

[35] Pinkas B (2002) Cryptographic techniques for privacy-preserving data mining. ACM SIGKDD Explorations Newsletter 4(2):12-19.

[36] Vaidya J, Kantarcioglu M, Clifton C (2008) Privacy-preserving Naive Bayes classification. The VLDB Journal 17(4):879-898.

[37] Yi X, Zhang Y (2009) Privacy-preserving naive Bayes classification on distributed data via semi-trusted mixers. Information Systems 34(3):371-380.

[38] Vaidya J, Clifton C (2004) Privacy preserving naive bayes classifier for vertically partitioned data. In: SDM. SIAM, pp 522-526.

[39] Tassa T (2014) Secure mining of association rules in horizontally distributed databases. IEEE Trans. Knowl. Data Eng., 26(4): 970-983.

[40] Goethals B, Laur S, Lipmaa H, Mielikinen T (2004) On private scalar product computation for privacy-preserving data mining. In Information Security and Cryptology, Berlin, Germany:Springer-Verlag, vol. 3506, pp. 104-120.

[41] Keshavamurthy BN, Sharma M, Toshniwal D (2010) Privacy Preservation Naive Bayes Classification for a Vertically Distribution Scenario Using Trusted Third Party. October 2010 ARTCOM '10: Proceedings of the 2010 International Conference on Advances in Recent Technologies in Communication and Computing.

[42] Gao C, Cheng Q, He P, Susilo W, Li J (2018) Privacy-preserving Naive Bayes classifiers secure against the substitution-then-comparison attack. Information Sciences, Volume 444, pp 72-88.

[43] Zhang N, Wang S, Zhao W (2005) On a new scheme on privacy-preserving data classification. In: Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining, ACM New York, USA, pp 374-383.

[44] Mitchell T (1997) Machine Learning. McGrawHill.

[45] Friedman N, Geiger D, Goldszmidt M (1997) Bayesian network classifiers. Machine Learning 29(2-3):131-163.

[46] Chow CK, Liu CN (1968) Approximating discrete probability distributions with dependence trees. IEEE Transactions on Information Theory 14:462-467.

[47] Magkos E, Maragoudakis M, Chrissikopoulos V, Gritzalis S (2009) Accurate and Large-Scale Privacy-Preserving Data Mining using the Election Paradigm. Data and Knowledge Engineering 68(11):1224-1236.

[48] Dua, D. and Graff, C. (2019). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.