A Pooled Structure for Data Confidentiality using AES and K-NN Classifier

Nischala Thakur, PANTHANGI PAVITHRA,PRAVEENA BETHA

Assistant professor,
Department of CSE Engineering,
Visakha Institute of Engineering & Technology,
Division,GVMC,Narava, Visakhapatham, Andhra Pradesh.

Abstract— Knowledge may be extracted from enormous data sets using data mining techniques. Predicting a certain result from a set of inputs is what classification is all about. Customer data may be stored in enormous amounts on a server in the cloud. We shall know the full potential of categorization when we analyze such vast datasets. However, one of the drawbacks of cloud computing is that data is stored on remote servers, making it accessible to anybody. As a result, the vast majority of businesses have shunned cloud computing. Customers' personal information must be protected by these firms. Encryption is one method of ensuring the safety of data. But encrypted data cannot be classified. To solve the DMED (Data Mining over Encrypted Data) conundrum, we have written this article. We offer a unified architecture for data confidentiality using AES and the k-NN classifier.

INTRODUCTION

The cloud handles all of the organization's data management needs, so the company does not have to. The cloud paradigm may be used by a wide range of enterprises, including healthcare, finance and insurance, and scientific research. Yet these firms have client data that cannot be outsourced in order to maintain confidentiality. Encrypting the data is one approach to ensure that the information is secure.

Example: An insurance business could desire to employ cloud services for archiving its customers' personal information. On the cloud, the data may be accessed by the owner at any time. Class labels are in column (m+1) of the cloud, therefore this is not an issue. We believe that the encryption method used is semantically safe. encrypted relational data are included in X's encoded data.

When the agent wishes to categorize consumers depending on their level of risk. Based on the categorization algorithm, this will be implemented. The agent must create a class label A using all of the customer's information in order to categorise their data. The problem is that this A includes sensitive information that should be encrypted before being sent to the cloud.

In the scenario above, firms who employ cloud computing for data mining confront a difficult decision. Most people would say decode the information and do data mining, however this threatens the data's security. Because of this, we will need to do data mining on encrypted data. We recommend using homomorphic encryption in this study as the optimal method. The consumer data may be outsourced.

A. ProblemStatement

There are n entries in Tarun's (the owner's) relational table (X) and an additional m+1 attributes. Let ap and q be the elements of the pth row and qth column, respectively. To keep the encrypted database safe, the query is referred to as r.

B. OurContributions

The AES and K-NN classifiers are used in this study to provide a cloud computing architecture for encrypted relational data. Make sure Tarun can only encrypt or outsource relational data from customers. He should not be involved in any additional cloud computing calculations. No data from consumers should be accessible or viewable to Danush, who should only be allowed to provide certain characteristics for the query r. Tarun should not know the topic of Danush's query in order to protect Danush's privacy. As a result, we are safeguarding the information of both the owner and the agent in the same manner. It is also important that the cloud is not able to access either of the people's customer data or the query they submitted. Due of its semi-honest (honest yet inquisitive) nature, the cloud is used in this way. Data access patterns may be deduced from the owner's involvement in the cloud, and the cloud may be able to anticipate the customer's data based on this. It is possible to infer that a person is a heart patient just by listening to them talk about anything having to do with cardiology.

Section 2 of the article offers a review of the literature relevant to the topic at hand. The k-NN algorithm, which was utilized to create the framework described above, is discussed in Section 3. Detailed information on conflicts in the article may be found in Section 4. This is detailed in detail in Section 5 of the framework. Section 6 of the article covers the technical details. Section 7 concludes the article and discusses the study's future directions.

LITERATURESURVEY

The use of homomorphic encryption may be appropriate for relational data, as we have previously discussed [1]. Data mining over encrypted data becomes simpler with the help of this technology. However, this is not the case due to the high implementation costs of these methods. While this study is ongoing, it is possible to use these algorithms at the most cost-effective rate. At least 30 seconds is required to do a single query using this technique on today's high-end computers [2].Data mining over encrypted data is performed here instead utilizing a mix of AES and K-NN methods from AES and RSA. As explained in Shamir's technique [3], here we require two parties to execute data mining as indicated. The only difference is that, in contrast to our job, they need three parties there.

A. ClassificationofConfidentialData

It was Lindell and Pinkas [5] and Agrawal and Srikant [4] that came up with the idea of data categorization. Both data perturbation and data distribution are under the umbrella of the categorization term "confidential data." There was a first attempt at a data perturbation approach, however it proved unsuccessful for sensitive material (e.g., [6]). As an alternative, we may create a basic classifier and utilize it instead. As mentioned by Shamir in his approach, the data is dispersed among numerous parties (e.g., [7]). As a result, we came to the conclusion that even this was not functioning with semantically safe data.

B. ProcessingEncryptedData

We want to guarantee that the information that has been facilitated by the proprietor, or in this model, Tarun, ought to be made protected to really handle it. The cloud should never have access to the data. These are customer's personal and sensitive information. That is why it was suggested in prior articles that the owner's role be minimized. Consequently, the client was limited to hosting the data of other customers but was not allowed to modify it. Even if you succeed in implementing this strategy, your relational database will be littered with duplicated rows as a consequence. In addition to this, we know that totally homomorphic algorithms are not a feasible alternative. So we use the AES algorithm, which is a homomorphic method, to encrypt data. In this way, the overhead of fully homomorphic algorithms is eliminated from the data mining process.

To prevent the host from learning Danush's query and attempting data manipulation while hosting, the owner's query should be secured in the same way that it was before. As a result, the owner will only be responsible for hosting the data and will have no extra responsibilities. Due to the fact that mining is not done directly on cloud data, the query is not exposed to the cloud. It is better if we take data from the cloud and then apply our algorithms to it without the owner of that data or even the cloud itself knowing about it. As a result, if anybody looks at the data, it will seem to be meaningless since it does not include any private client data.

C. AESCryptosystem

Symmetric encryption is used in AES, which is a cryptosystem. An iterative process is at work here. The four stages are: replacing bytes, moving rows, mixing columns, and adding a round key. An encrypted 128-bit block of input data is decrypted using a key that is also 128-bit long. S-Box bytes may be used in place of the first step. Two rows of four bits each are used to denote a single data point in a single S-Box. Moving rows is the next stage. The first row has not been moved. The second row has been relocated by one byte to the left. Shift left by two bytes to make room for the third row of bytes. The fourth row has been moved 3 bytes to the left. The following step will use this as the result. After that, the 4X4 matrix is multiplied by a second constant matrix, which is used to mix the columns. This step's secret key is then used to X-OR the matrix. In Fig. 1, you can easily see all of the information I have discussed.

Depending on how many rounds an AES cryptosystem has, the key length changes. To make it homomorphic, though, one may use the same encryption key for all of its operations. Every time we encrypt with the same key and information, we will obtain the same ciphertext. Semantically secure ciphers, like AES, can not be decrypted even if someone has the cipher text. AES encryption and decryption are identical except for the sequence in which they are carried out.

ESSENTIALSFORSEMANTICSECURITY

Along with AES, the K-NN algorithm also plays a role in cryptographic key exchanges. To accomplish the classification, the K-NN algorithm uses the encrypted data and the query as input. This study devotes a significant amount of time on classifying data, since it is one of the foundational functions of data mining.

The encrypted database X'has a query q that must be performed by Danush. The properties of Danush's question may be mathematically stated as r. The encrypted database and the query are used as inputs in the classification procedure. C1 and C2 are the two parties we will be focusing on in this section. Encrypted data is held by C1 and the AES algorithm key is held by C2. When Danush inputs his inquiry, we encrypt it using the AES technique in order to ensure that it is secure. The encrypted query has been sent to C1 for processing.

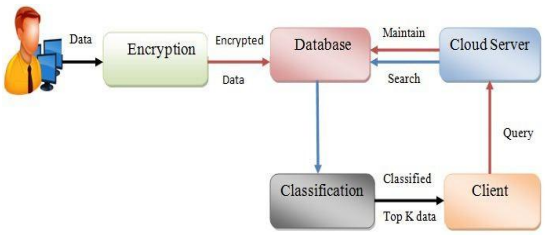

Fig1.Systemarchitecture

The SCMC's second characteristic is its ability to safely navigate data. Class labels are multiplied by actual numbers so that the decision tree may be transferred from C1 to Danush in this instance Q is sent to C2 and Danush receives the actual number. Decrypted data will be sent to Danush via C2. When all of this is done, Danush will get the following from C1: rq. He will do a mod n on the result of subtracting the two integers. The output is the answer to the question.

MUTUALPROTECTIONOFDATA

Two parties' C1 and C2 are assumed in this protocol. The fundamental problem is that any participant may depart from the procedure and try to change data at any time. If data is encrypted, it is possible to safeguard it from cyberattacks. There is a chance they will try to mess with the data while it is being processed. To continue the k-NN with updated inputs, C1 might, for example, exit halfway through the protocol after receiving the required information. It is not practicable in this scenario since neither party has adequate knowledge to carry out the whole procedure on its own. To prevent such nefarious manipulations, we have shared the data across many parties.

However, if both parties are attempting to modify the data, then securing it is pointless. If just one side is malevolent, then the foregoing is true. You may avoid this sort of conduct by allowing the honest person, who has no knowledge of information, to properly compute the facts. Furthermore, attempting to verify the parties or data at each stage would drastically raise the project's cost and time.

Changing responsibilities at random intervals is another option that may be adopted in the future in addition to just distributing the info. C1 is responsible for the data processing while C2 is responsible for authenticating it; in this case, C1 and C2 are both C1. C1 should verify the data after a period of time, followed by C2 completing the computation.

PERFORMANCE

According to K-NN, the computing cost of the algorithm grows linearly with the number k. We were able to see the time grow as we increased the value of k from 3 to 30. The k-NN method accounts for the majority of the processing time. The SCMC algorithm takes just a little quantity of data. There are 524 entries and four characteristics in the data set that we are using. Attribute-wise, we use the AES technique to encrypt this dataset. It is also worth noting that the AES method consumes a significant amount of time when conducting the encryption.

To be clear, this approach is not efficient in practice. All transactions, even on the most powerful machines, require a long time to process. That being said, there is always room for improvement, even in the most little ways. Parallelism will be required. Parallelization is not used in any of the above computations.

CONCLUSION AND FUTURE WORK

Some of the most popular algorithms for classifying data have been developed in the last decade. Unified k Nearest Neighbors is the best classifier for classifying datasets that have been provided by third parties. This will categorize encrypted data by verifying that the data is secured with the necessary semantic security. The algorithm's efficiency may be increased by giving the bare minimum of security. In this way, enterprises may outsource their data without worrying about security issues in the cloud and yet take use of the cloud's many benefits.

Malicious parties may be an issue, as we have already warned you. As a result, it is necessary to assign specified times for the parties to complete their respective tasks. It is still an issue of efficiency, since the planned work would increase current efficiency to the point where the effort is rendered ineffective. So, study is needed to ensure that the aforementioned task can be implemented without affecting productivity.

REFERENCES

1. A. Shamir, “How to share a secret,” Commun. ACM, vol. 22,pp.612–613,1979.
2. R. Agrawal and R. Srikant, “Privacy-preserving data mining,”ACMSigmodRec., vol.29,pp. 439–450,2000.
3. Y. Lindell and B. Pinkas, “Privacy preserving data mining,” inProc. 20th Annu. Int. Cryptol. Conf. Adv. Cryptol., 2000, pp.36–54.
4. L.Xiong,S.Chitti,andL.Liu,“Knearestneighborclassificationacrossmultipleprivatedatabases,”inProc.15th ACM Int. Conf. Inform. Knowl. Manage., 2006, pp.840–841.
5. X.Xiao,F.Li,andB.Yao,“Securenearestneighborrevisited,” in Proc. IEEE Int. Conf. Data Eng., 2013, pp.733–744.
6. ShivlalMewada,SharmaPradeep,GautamS.S.,“ClassificationofEfficientSymmetricKeyCryptographyAlgorithms”, International Journal of Computer

Science andInformation Security (IJCSIS) USA, Vol. 14, No. 2, pp (105-110), Feb2016.ISSN:1947-5500

7. B. K. Samanthula, Y. Elmehdwi, and W. Jiang, "k-nearestneighbor classification over semantically secure encryptedrelationaldata,"eprintarXiv:1403.5001,2014.
8. A. Shamir, "How to share a secret," Commun. ACM, vol. 22, pp. 612–613, 1979.
9. R. Agrawal and R. Srikant, "Privacy-preserving data mining," ACM Sigmod Rec., vol. 29, pp. 439–450, 2000.

