ISSN:2348-2079

Volume-6 Issue-2

International Journal of Intellectual Advancements and Research in Engineering Computations

Trust-But-Verify: Verifying Result Correctness of Outsourced Frequent Item set Mining

[1]Mrs.K.E.Eswari, M.C.A., M.Phil.,M.E., Associate Professor/MCA
[2]Mr.G.Vignesh, III MCA
Department of MCA, Nandha Engineering College, Erode-52.
E-Mail ID: eswari.eswaramoorthy@nandhaengg.org, vigneshgunasekar.bsc@gmail.com

***Abstract*- Data Mining is popularizing the computing paradigm in which data is outsourced to a third-party service provider (server) for data mining. Outsourcing, however, raises a serious security issue: how can the client of weak computational power verify that the server returned correct mining result. This project focuses on the specific task of frequent itemset mining. It also consider the server that is potentially untrusted and tries to escape from verification by using its prior knowledge of the outsourced data. It propose efficient probabilistic and deterministic verification approaches to check whether the server has returned correct and complete frequent itemsets. Our probabilistic approach can catch incorrect results with high probability, while our deterministic approach measures the result correctness with 100 percent certainty. It also design efficient verification methods for both cases that the data and the mining setup are updated. This project demonstrates the effectiveness and efficiency of our methods using an extensive set of empirical results on real datasets.**

***Key terms* :Big Data, Frequent itemset ,Mining,Homomorphic**

I.INTRODUCTION

Deterministic approach is based on an efficient authenticated data structure, which is built upon standard Merkle trees and bilinear-map accumulators. It enables the proof based verification. The existing systems optimize the verification algorithm by reducing the number of proofs for both correctness and completeness verification. The results show that a small number of proofs is sufficient to verify the correctness and completeness of a large set of frequent itemsets.To enforce this principle, they proposed an algorithm that employs generalization and suppression to group semantically close diagnosis codes together in a way that enhances data utility. Additionally, considered protecting data in which a certain diagnosis code may occur multiple times in a patient record. They designed an algorithm through suppression a subset of the replications of a diagnosis code.In this project, Homomorphic protocols solving this problem onsuppression-based and generalization-based k-anonymous and confidential databases are proposed. The protocols rely on well-known cryptographic assumptions..Homomorphic protocol is aimed at suppression-based anonymous databases, and it allows the owner of DB to properly anonymize the tuple t, without gaining any useful knowledge on its contents and without having to send to t's owner newly generated data. To achieve such goal, the parties secure their messages by encrypting them.In order to perform the privacy-preserving verification of the database anonymity upon the insertion, the parties use a commutative and homomorphic encryption scheme. The second protocol is aimed at generalization-based anonymous databases to support privacy-preserving updates on a generalization-based k-anonymous DB.

II. RELATED WORKS

Rosario Gennaro, Craig Gentry, Bryan ParnoVerifiable Computation enables a computationally weak client to "outsource" the computation of a function F on various inputs x1... xkto one or more workers. The workers return the result of the function evaluation, e.g., yi= F (xi), as well as a proof that the computation of F was carried out correctly on the given value xi. The verification of the proof should require substantially less computational effort than computing F (xi) from scratch. They present a protocol that allows the worker to return a computationally-sound, non-interactive proof that can be verified in O (m) time, where m is the bit-length of the output of F. The protocol requires a one-time pre-processing stage by the client which takes O (|C|) time, where C is the smallest Boolean circuit computing F. Their scheme also provides input and output privacy for the client, meaning that the workers do not learn any information about the xi or yivalues.

Spurred by developments such as cloud computing, there has been considerable recent interest in the data-mining-as-a-service paradigm. Users lacking in expertise or computational resources can outsource their data and mining needs to a third-party service provider (server). Outsourcing, however, raises issues about result integrity: how can the data owner verify that the mining results returned by the server are correct? They present AUDIO, an integrity auditing framework for the specific task of distance-based outlier mining outsourcing. It provides efficient and practical verification approaches to check both completeness and correctness of the mining results. The key idea of their approach is to insert a small amount of artificial tuples into the outsourced data; the artificial tuples will produce artificial outliers and non-outliers that do not exist in the original dataset. The server's answer is verified by analyzing the presence of artificial outliers/non-outliers, obtaining a probabilistic guarantee of correctness and completeness of the mining result. Their empirical results show the effectiveness and efficiency of their method.

Ran Canetti, Ben Riva, Guy N. RothblumThe current move to Cloud Computing raises the need for verifiable delegation of computations, where a weak client delegates his computation to a powerful server, while maintaining the ability to verify that the result is correct. Although there are prior solutions to this problem, none of them is yet both general and practical for real-world use. They demonstrate a relatively efficient and general solution where the client delegates the computation to several servers, and is guaranteed to determine the correct answer as long as even a single server is honest. A protocol for any efficiently computable function, with logarithmically many rounds, based on any collision-resistant hash family. The protocol is set in terms of Turing Machines but can be adapted to other computation models.An adaptation of the protocol for the X86 computations model and a prototype implementation, called Quin, for Windows executables.They described the architecture of Quin and experiment with several parameters on live clouds. They show that the protocol is practical, can work with nowadays clouds, and is efficient both for the servers and for the client. They executed several experiments of the full protocol. For each experiment they ran the protocol several times with one cheating cloud that cheats on one out of three randomly chosen states. Those states were chosen to be close to the end of the computation (around 80%−85% of the total number of steps).

Dario Fiore and Rosario GennaroOutsourced computations (where a client requests a server to perform some computation on its behalf) are becoming increasingly important due to the rise of Cloud Computing and the proliferation of mobile devices. Since cloud providers may not be trusted, a crucial problem is the verification of the integrity and correctness of such computation, possibly in a public way, i.e., the result of a computation can be verified by any third party, and requires no secret key – akin to a digital signature on a message. They present new protocols for publicly verifiable secure outsourcing of Evaluation of High Degree Polynomials and Matrix Multiplication. Compared to previously proposed solutions, ours improve in efficiency and offer security in a stronger model. The paper also discusses several practical applications of their protocols.The rise of Cloud Computing raises several new security problems that must be addressed by the research community.
In particular, a fundamental component of any secure cloud computing approach is a mechanism that enforces the integrityand correctness of the computations done by the provider on behalf of a client.

Charalampos Papamanthou1, Roberto TamassiaThey study the design of protocols for set-operation verification, namely the problem of cryptographically checking the correctness of outsourced set operations performed by an untrusted server over a dynamic collection of sets that are owned (and updated) by a trusted source. They presented new authenticated data structures that allow any entity to publicly verify a proof attesting the correctness of primitive set operations such as intersection, union, subset and set difference. Based on a novel extension of the security properties of bilinear-map accumulators as well as on a primitive called accumulation tree, their protocols achieve optimal verification and proof complexity (i.e., only proportional to the size of the query parameters and the answer), as well as optimal update complexity (i.e., constant), while incurring no extra asymptotic space overhead. The proof construction is also efficient, adding a logarithmic overhead to the computation of the answer of a set-operation query. In contrast, existing schemes entail high communication and verification costs or high storage costs. Applications of interest include efficient verification of keyword search and database queries. The security of their protocols is based on the bilinear q-strong Diffie-Hellman assumption.

Michael T. Goodrich, Duy Nguyen, Olga OhrimenkoThey consider the problem of verifying the correctness and completeness of the result of a keyword search. They introduce the concept of an authenticated web crawler and present its design and prototype implementation. An authenticated web crawler is a trusted program that computes a specially crafted signature over the web contents it visits. This signature enables

(i) The verification of common Internet queries on web pages, such as conjunctive keyword searches—this guarantees that the output of a conjunctive keyword search is correct and complete;

(ii) The verification of the content returned by such Internet queries—this guarantees that web data is authentic and has not been maliciously altered since the computation of the signature by the crawler. In solution, the search engine returns a cryptographic proof of the query result.

Both the proof size and the verification time are proportional only to the sizes of the query description and the query result, but do not depend on the number or sizes of the

web pages over which the search is performed. As experimentally demonstrated that the prototype implementation of their system provides a low communication overhead between the search engine and the user, and fast verification of the returned results by the user.

Bryan Parno, Mariana Raykova, Vinod VaikuntanathanThe wide variety of small, computationally weak devices and the growing number of computationally intensive tasks makes the delegation of computation to large data centers a desirable solution. However, computation outsourcing is useful only when the returned result can be trusted, which makes verifiable computation (VC) a must for such scenarios. In this work, they extend the definition of verifiable computation in two important directions: public delegation and public verifiability, which have important applications in many practical delegation scenarios. Yet, existing VC constructions based on standard cryptographic assumptions fail to achieve these properties.

As the primary contribution of their work, they establish an important (and somewhat surprising) connection between verifiable computation and attribute-based encryption (ABE), a primitive that has been widely studied. Namely, they showed that how to construct a VC scheme with public delegation and public verifiability from any ABE scheme. The VC scheme verifies any function in the class of functions covered by the permissible ABE policies. This scheme enjoys a very efficient verification algorithm that depends only on the output size. Strengthening this connection, they showed the construction of a multi-function verifiable computation scheme from an ABE with outsourced decryption. A multi-function VC scheme allows the verifiable evaluation of multiple functions on the same preprocessed input.Traces deviating from common trace population rules are removed.

- The resultant filtered traces are then separated into multiple clusters.
- By clustering common traces together, it is expected that the learner is able to learn better and over-generalization of a subset of traces is not propagated to other clusters. These clusters of filtered traces are then inputted to a specification miner.
- The algorithm has been shown gain to significant performance improvement over TraceMiner and FP-TraceMiner.
- To provide an efficient method to mine the specifications from program execution traces.

III. METHODOLOGY

Mining specifications can be done by using Association rule mining. Association rules mining is a very popular data mining techniques and it finds relationships among the different entities of records (for example specifications records). It has received a great deal of attention in the field of knowledge discovery and data mining. The problem of association rules mining was introduced was improved to obtain the Apriori algorithm. The Apriori algorithm employs the downward closure property- if an itemset is not frequent, any superset of it cannot be frequent either. The Apriori algorithm performs a breadth-first search in the search space

- The algorithm has been shown gain to significant performance improvement over Trace Miner and FP-Trace Miner.
- This proposed system also gives an efficient method to mine the specifications from program execution traces.

Traces deviating from common trace population rules are removed. The resultant filtered traces are then separated into multiple clusters. By clustering common traces together, it is expected that the learner is able to learn better and over-generalization of a subset of traces is not propagated to other clusters. These clusters of filtered traces are then inputted to a specification miner. This algorithm confirms the usefulness of the proposed method in discovering software specifications in iterative pattern form. Besides mining software behavioral pattern, it is believed that the proposed mining technique can potentially be applied to other knowledge discovery domains

A. TRANSACTION ENTRY

In this module contains a frequent item details which is involved in each transaction. In the grid view control, all the records are displayed from which the records can be modified and new values can be updated. In addition if an item support count is higher than the minimum support count then it will be highlighted.

B. FREQUENT ITEM TRACES

This modules considered only a node with maximum support count otherwise nodes are removed from each transaction. In addition transaction entries are ordered based on the support count. These details are stored in 'Ordered' table and viewed by using grid view control.

C. PROBABILISTIC APPROACH

In this module apply the probabilistic approach to catch mining result and the predefined correctness/completeness requirement with high probability. The key idea is to construct a set of (in)frequentitemsets from real items, and use these (in)frequent itemsets as evidence to check the integrity of the server's mining results.

D. DETERMINISTIC APPROACH

The deterministic approach to catch any incorrect/incomplete frequent itemset mining answer with 100 percent probability. The key idea of our deterministic solution is to require the server to construct cryptographic proofs of the mining results. Both correctness and completeness of the mining results are measured against the proofs with 100 percent certainty.

E. ANONYMITY

a.Attribute Creation

In this module, attribute id, name, data type and suppress type (No Suppress, Semi Suppress, Full Suppress) are added to the database table. During the Value

Generalization Hierarchy creation, the attribute id is selected using the combo box control.

b.Value Generalization Hierarchy

In this module, during the form load, the attribute ids are fetched from 'attributes' table. An attribute id is selected, its name is displayed; the original value in the data set and the 'After VGH' value is keyed in. Data grid view control is also provided to list all the records. Any record value can be modified and updated to the database using 'Update' button.

F. DATA SETS

Data Set Creation

Based on the attributes given and their data type, a data set with 'n' (Attributes count) number of columns is dynamically created. The record values are keyed in using the data grid view control.

Show Original Data Set

Based on the values given, the record values from the table 'DataSetValues' are displayed using the data grid view control.

Show Suppressed Data

The record values from the table 'DataSetValues' are replaced with '*' marks for suppressed columns table and are displayed using the data grid view control.

Show Generalized Data

The record values from the table 'DataSetValues' are replaced with 'ValueGeneralizationHierarchy 'table and are displayed using the data grid view control. During the replace any single value or range of values are substituted and displayed.

G. VIEW

Show Attributes

In this module, attribute id, name, data type and suppress type (No Suppress, Semi Suppress, Full Suppress) are viewed from the database table 'Attributes'.

Show Data Set Values

Based on the values given, the record values from the table 'DataSetValues' are displayed using the data grid view control.

Show Value Generalization Hierarchy

In this module, the attribute id, name, Original value and VGH values are displayed. Data grid view control is provided to view all the records.

Show Suppressed Data

The record values from the table 'DataSetValues' are replaced with '*' marks for suppressed columns table and are displayed using the data grid view control.

Show Witness Set

After the VGH values applied, the records may contain duplicate values in all columns. Those values are eliminated and witness set is prepared. These records are displayed here.

H. HOMOMORPHIC PROTOCOL

Suppression Based Method

In this module, the suppression based method is used. In Step 1 DB Owner Send coded non suppressed column values (EA(cDeltaI)) to Data Provider. Then Data Provider gives his tuple values of non-suppress columns. In Step 2 Data Provider codes own tuple values and also DB Owners' tuple value. In Step 3 DB Owner Decrypts EB(EA(cDeltaI)). Both db owner and data provider process the data without disturbing the privacy here.

IV. ANONYMITY DEFINITIONS

Consider T {t1; ... ;tn} over the attribute set A. The idea is to form subsets of indistinguishable tuples by masking the values of some well-chosen attributes. In particular, when using a suppression-based anonymization method, we mask with the special value *, the value deployed by (database owner) for the anonymization. When using a generalization-based anonymization method, original values are replaced by more general ones, according to a priori established value generalization hierarchies (VGHs). The following notation is used. Quasi-Identifier (QI): A set of attributes that can be used with certain external information to identify a specific individual.With respect to suppression-based anonymization, QI can be classified into two subsets: suppressed attributes QIs and nonsuppressed attributes QIs . When T is k-anonymous, then for every tuple T, there exists a subset of tuples {t1;...;tz} T(zk – 1) such that for every attribute in QIs, the corresponding value is replaced by * (indicating suppressions of the original values).For generalization-based anonymization, it is assumed that each attribute value can be mapped to a more general value.

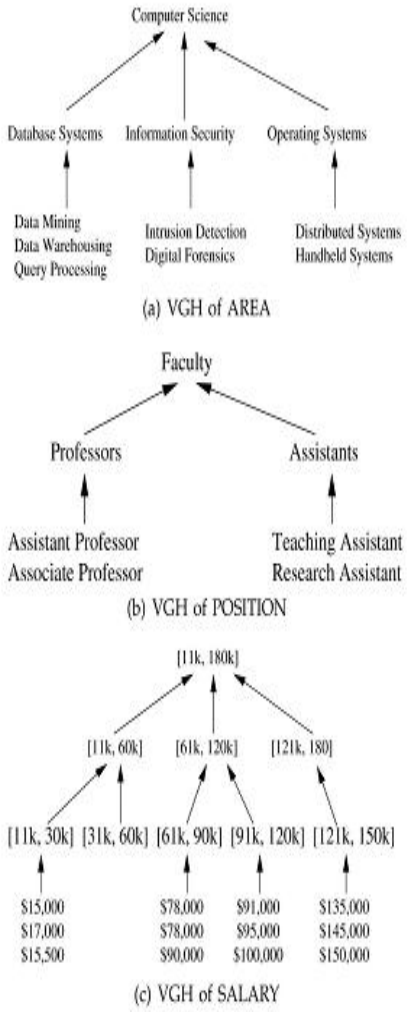

Fig 1: A typical Value Generalization Hierarchy

The main step in most generalization based k-anonymity protocols is to replace a specific value with a more general value.

A. PROTOCOL OPERATION

To provide secure communication between database owner and data provider (the records given from database owner and data provider should not be identified by data provider and vice-versa), a commutative, product-homomorphic encryption scheme E is introduced.A commutative, product-homomorphic encryption scheme ensures that the order in which encryptions are performed is irrelevant (commutativity) and it allows to consistently perform arithmetic operations over encrypted data (homomorphic property).

Further, for the security proofs we require that the encryption scheme E satisfies the indistinguishability property. The scheme should produce an encryption method which should be product-homomorphic.

Given a finite set K of keys and a finite domain D, a commutative, product-homomorphic encryption scheme E is a polynomial time computable function E : K * D -> D satisfying the following properties.

1. Commutativity. For all key pairs K1,K2 value D, the following equality holds:
$E_{K1} (E_{K2}(d)) = E_{K2} (E_{K1}(d))$
2. Product-homomorphism. For every K and every value pairs d1;d2 the following equality holds:
$E_K (d) . E_K (d2) = E_K (d1.d2)$
3. Indistinguishability. It is infeasible to distinguish an encryption from a randomly chosen value in the same domain and having the same length. In other words, it is infeasible for an adversary, with finite computational capability, to extract information about a plain text from the cipher text.

Finally, a simple tuple coding scheme is introduced which is used in the next protocol operation. For example Alice and Bob agree on a set {g1;g2 ... ;gu} of generators of D. Let d be a tuple {d1;d2; ... ;du} with elements taken from q, the encoding of a tuple d is defined as

$$C(<d1,d2,\ldots,du>) = gi^{di}{}_{modq.} i=1$$

B. ALGORITHMS USED FOR DATABASE UPDATE

The protocol works as follows: At Step 1, Alice sends Bob an encrypted version of i, containing only the s non-suppressed QI attributes. At Step 2, Bob encrypts the information received from Alice and sends it to her, along with encrypted version of each value in his tuple t. At Steps 3-4, Alice examines if the nonsuppressed QI attributes of i is equal to those of t.

STEPS
1. Alice codes her tuple i into c(<v1'; ... ;vs'>), is denoted as c(i). Then, she encrypts c(i) with her private key and sends E_Ac(i) to Bob.
2. Bob individually codes each attribute value in t to get the tuple of coded values < C (V_1), …… C (V_U) >, , encrypts each coding and E_A (c(i)) with his key B and sends (i) < E_B (C (V_1), …… C (V_U)) >; and (ii) E_B (E_A (c (i))) to Alice.
3. Since E is a commutative encryption scheme,Alice decrypts

$$E_B(E_A(C (i)))$$

4. Since the encrypted values sent by Bob are ordered according to the ordering of the attributes in T (assume this is a public information known to both Alice and Bob), Alice knows which are, among the encrypted values sent by Bob, the one corresponding to the suppressed and nonsuppressed QI attributes. Thus, Alice computes
E_B (C (V_1) X ……X E_B (C (V_S)(A))
where v1; ... ;vs are the values of nonsuppressed attributes contained in tuple t. As already mentioned, E is a product-homomorphic encryption scheme. Based also on the definition of function C (.), this implies that Expression (A) is equal to

$$E_B (C (< V_1 \ldots\ldots V_S >))$$

5. Alice checks whether
E_B (C (< V_1 …… V_S >)) = E_B (C (< V'_1 …… V'_S >))
If true, t (properly anonymized) can be inserted to table T. Otherwise, when inserted to T, t breaks k-anonymity.

V. CONCLUSION

In this project, a new algorithm TM-Trace Miner is presented using the vertical database representation. Trace ids of each trace set are transformed and compressed to continuous transaction interval lists in a different space using transaction tree and frequent trace sets are found by transaction intervals intersection along a lexicographic tree in depth-first order. Through this project the TM-Trace Miner algorithm has been gain to significant performance improvement over Trace Miner and FP-Trace Miner.

This project also gives an efficient method to mine the specifications from program execution traces. Traces deviating from common trace population rules are removed. The resultant filtered traces are then separated into multiple clusters. By clustering common traces together, it is expected that the learner is able to learn better and over-generalization of a subset of traces is not propagated to other clusters.

These clusters of filtered traces are then inputted to a specification miner. This algorithm confirms the usefulness of the proposed method in discovering software specifications in iterative pattern form. Besides mining software behavioral pattern, it is believed that the proposed mining technique can potentially be applied to other knowledge discovery domains

REFERENCES

[1] J. Vaidya and C. Clifton, "Privacy Preserving Association Rule Mining in Vertically Partitioned Data," Proc Eighth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD '02), pp. 23-26, July 2002.

[2] L. Sweeney, "k-Anonymity: A Model for Protecting Privacy," Int'l J. Uncertainty, Fuzziness and Knowledge-based Systems, vol. 10, pp. 557-570, May 2002.

[3] S. Bu, L. Lakshmanan, R. Ng, and G. Ramesh, "Preservation of Patterns and Input-Output Privacy," Proc. IEEE 23rd Int'l Conf. Data Eng., pp. 696-705, Apr. 2007.

[4] D. Denning and T. Lunt. A multilevel relational data model. In Proc. of the IEEE Symposium on Research in Security and Privacy, pages 220-234, Oakland, 1987.

[5] R. Agrawal and R. Srikant. Privacy preserving data mining. Proc. 2000 SIGMOD, pp. 439–450.

[6] P. Chan. On the accuracy of meta-learning for scalable data mining. Journal of Intelligent Information Systems, 8:5{28, 1997.

[7] Ford Motor Corporation. Corporate citizenship report http://www.ford.com/en/ ourCompany/communityAndCulture/buildingRelationships /strategicIssues/firestoneTireRecall.htm, May 2001.

[8] P. Chan. An Extensible Meta-Learning Approach for Scalable and Accurate Inductive Learning. PhD thesis, Department of Computer Science, Columbia University, New York, NY, 1996. (Technical Report CUCS-044-96).

[9] D. W.-L. Cheung, V. Ng, A. W.-C. Fu, and Y. Fu. Efficient mining of association rules in distributed databases. Transactions on Knowledge and Data Engineering, 8(6):911{922, Dec. 1996.

[10] M. Kantarcioglu and C. Clifton. Privacy-preserving distributed mining of association rules on horizontally partitioned data. In The ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery (DMKD'02), June 2 2002.