ISSN(O): 2456–026X

Semantic Inference Model Implementation for Database Security using Frequent Pattern Tree

Sonal Jaiswal
Computer Science & Engineering Department,
Rungta College of Engineering and technology Bhilai.
Email: Sonal_2504@yahoo.co.in

Toran Verma
Computer Science & Engineering Department, Rungta College of Engineering and technology Bhilai.
Email: Vermatoran24@gmail.com

Abstract: Semantic inference model (SIM) consists of data dependency, relational database schema, and domain specific semantic knowledge & learning. Along these lines, a Semantic Inference Model (SIM) speaking to them as probabilistic surmising channels to get to any information from the framework. In any association there are diverse sorts of information and some information is most essential for association. The association meets expectations with distinctive office and diverse items. In any association each individual not generally knows all advancement work of the association. On the off chance that association is adding to some new item then everyone is not mindful of the points of interest at essential stage. In this paper we have proposed semantic inference model for data base security using frequent pattern tree, the method we have implemented detects and avoids unauthenticated users.

Keywords: data mining, semantic inference model, frequent pattern tree

I. Introduction

Data mining is used to deal with very large amount of data which are stored in the data ware houses and databases, to find out desired interesting knowledge and information. Many data mining techniques have been proposed such as, association rules, decision trees, neural networks, etc. It has become the point of attention from many years. One of the most well-known techniques is semantic inference model. In this paper we proposed semantic inference model to database security using frequent pattern tree.

Inference Framework

The proposed inference detection system consists of three modules: knowledge acquisition, semantic inference model (SIM), and security violation detection including user social relation analysis.

- The knowledge acquisition module separates information reliance learning, information plan learning and area semantic information. In view of the database outline and the information sources, we can extricate information reliance between properties inside of the same element and among elements. Area semantic learning can be inferred by semantic connections with particular requirements and standard
- A semantic inference model can be developed taking into account the procured information. The Semantic Inference Model (SIM) is an information show that consolidates information outline, reliance and semantic learning. The model connections related characteristics and elements and also semantic learning required for information induction. Consequently SIM speaks to all the conceivable connections among the properties of the information sources. A Semantic Inference Graph (SIG) can be built by instantiating the elements and qualities in the SIM. For a given inquiry, the SIG gives derivation channels to gathering touchy data. In light of the induction channels got from the SIG, the Violation Detection module combines the new query request with the request log, and it checks to see if the current request exceeds the prespecified Threshold of information leakage. If there is collaboration according to social relation analysis.
- The violation detection module will choose whether to answer the present question in view of the gained learning among the vindictive gathering individuals and their social connection to the present user.

Frequent Pattern Tree

In the field of information mining, the most well known calculation utilized for example revelation is FP Growth calculation. To manage the two fundamental downsides of Apriority calculation in a novel, compacted information structure named as FP tree is built, which is prefix-tree structure putting away quantifiable data about successive examples. In view of FP tree a regular example development calculation was created. It's a two-stage approach. In first step an incessant example tree is built checking database twice. In first go of database, information is checked and bolster mean every thing is ascertained, rare examples are erased from

the rundown and remaining examples are sorted in plummeting request. In 2nd go of database, FP Tree is fabricate. In 2nd stage utilizing FP development calculation regular examples are removed from FP Tree. Contingent FP tree base and Conditional FP tree are in light of hub connection property and prefix way property.

Working of FP tree:-

Step -1	Create an empty F-List array F[]
Step -2	For every item in every transaction of the database, set F[item] +=1
Step -3	Sort the array F
Step -4	Create an empty tree T with null as the root node
Step -5	For every transaction in the Database, sort the transaction according to F-List, and add the items one by one to the tree T
Step -6	Maintain a reference of the items in the tree, and if present in more than one location, keep all the references.
Step -7	Starting with the leaf node, construct the conditional FP-Tree for each element.
Step -8	Generate the frequent Item Sets.

II. Problem Description

- One of the main issues faced by database security professionals is avoiding inference capabilities. Basically, inference occurs when users are able to piece together information at one security level to determine a fact that should be protected at a higher security level.
- Access control mechanisms are commonly used to protect users from the divulgence of sensitive information in data sources. However, such techniques are insufficient because malicious users may access a series of innocuous data, and from the received answers, the malicious users may employ inference techniques to derive sensitive information. The user can pass series of queries that are related to each other and may lead them to a specific data set. By the result of these queries the user may detect the relationship between the dates and ultimately predict the sensitive data. This may cause a serious database security issue.

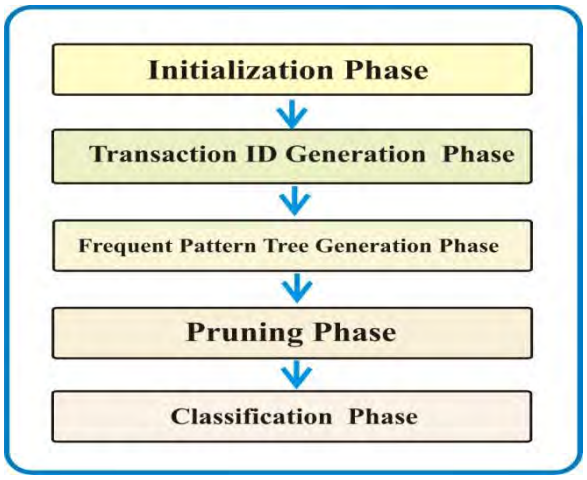
If certain measures are taken to keep a track of the user queries than also there is a problem, most users usually work as a team, and each member can access the information independently. Afterwards, the members may merge their knowledge together and jointly infer the sensitive information.

Basically the techniques used in semantic inference model (SIM) for violation detection are very complex in nature. In most of the approach database semantics and domain knowledge has to be assessed which itself is a very complex task.

- Lack of data mining techniques has also been observed in the related works.
- Tree/graph construction in semantic inference graph is complex. It has been observed that there is a lack of mathematical approach in construction of semantic inference graph (SIG).
- Semantic inference model (SIM) emphasizes on intruder detection. It stops certain users to access sensitive data relations but then also there is a possibility of violation in the database. This drawback builds a necessity of developing an approach to avoid inference detection. So a detection as well as avoidance approach should be developed.

III. Proposed Methodology

This proposed method is about the Semantic inference model implementation for database security using frequent pattern tree performing the following steps work. Fig no 1 is followed proposed method.



Phase 1: Initialization Phase

The first step of this work is to input database log file, and then assign a unique prime number for each attributes in the database. The log file consist of the data that shows the frequency of the data the users have assessed.

Step –1	Input database log file
Step –2	For each attributes in the database. Assign unique prime number. End

Phase 2 Transaction ID generation Phase

The table contains one section for every dynamic exchange. This incorporates Transaction ID and last LSN, where last LSN portrays the LSN of the latest log record, for the exchange of this procedure is called a transaction Id generation. This algorithm is the first step of input key attributes that generates a unique id then in the second step in which each unique id in the transactions are multiplied to create a unique transaction id.

Step –1	Input key Attributes and their unique id.
Step –2	For each row in the log file. Multiply the unique id of each entry and End

Phase 3 Frequent pattern tree generation Phase FP tree is constructed, in which the prefix-tree structure storing quantifiable the information about frequent patterns. Based on FP tree, a frequent pattern growth algorithm is developed. It has two-step approach. In the first pass, the algorithm counts occurrence of items (attribute-value pairs) in the dataset, and stores them to 'header table'. In the second pass, it builds the FP-tree structure by inserting instances. Items in each instance have to be sorted by descending order of their frequency in the dataset, so that Phase 4 Pruning Phase

To solve the problem of huge memory usage of FP-tree construction and traversal in FP-growth, Dynamic-prune, which is a concurrent frequent item sets mining algorithm i.e based on FP-tree proposed. On the one hand, by recording the change of support counts of frequent items during the process of FP-tree construction, dynamic FP-tree pruning is implemented. This phase are step one is input the frequent pattern tree then second step is input the minimum support of the FP tree the transaction minimum support then else condition is discard.

Step –1	Input the frequent pattern tree.
Step –2	Input Minimum support
Step -	If transition follows minimum support Keep it Else Discard it End

Phase 5 Classification Phase

FP tree classification is classified by support and query. This phase are first steps input the minimum support and satisfying the rules the next step is input the columns are sensitive then next step is frequency of columns of user is greater than to support.

Step - 1	Input the minimum Support. Satisfying rules
Step -2	Input the columns which are sensitive

Step 3	For each sensitive columns
	If frequency of columns of user is greater than to support Then
	assign intruder
	Else
	Genuine

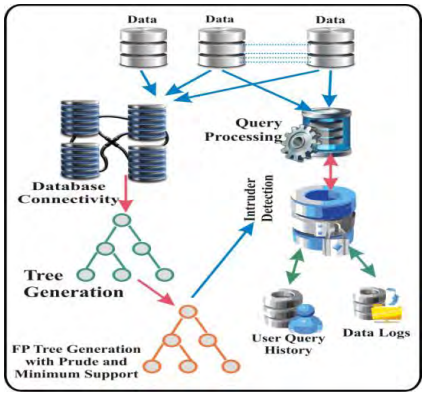


Fig. 2: Proposed Method

Example

Step 1 database connectivity

Step 2 transaction id generation based on prime number assignment

S.No.	Transaction	Transformation	Transaction ID
1	A,B,C	2,3,5	30
2	A,D,F	2,7,13	182
3	B,D,E,F	3,7,11,13	3003
4	A,C,E	2,5,11	110
5	A,B,C,D,E	2,3,5,7,11	2310
6	A,C,D,E,F	2,5,7,11,13	10010
7	A,C,D,F	2,5,7,13	910

Fig. 4: Transaction id generation

Step 3 adjancy matrix generation A [nxm] matrix is generated with elements 0&1. n= number of users. m= number of transaction. 0 = non accessed data

1 = accessed data

Step 4 transaction matrix generation A [TxTid] matrix is generated with elements 0&1.

T= transaction

Tid = transaction id

= non accessed data

= accessed data

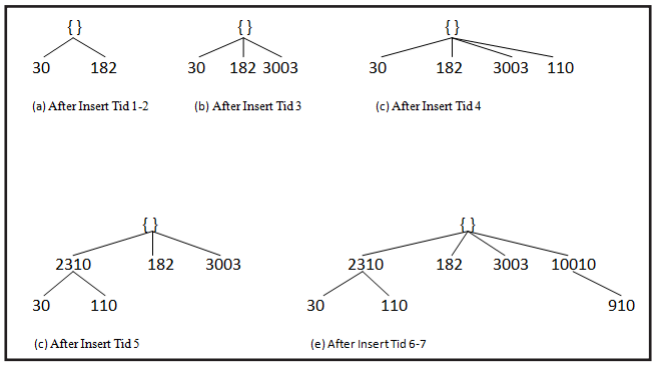


Fig. 5: Frequent Pattern Tree Generation

Step 5 calculation of frequency of data Frequency of data (Fd1) = summation of all the columns of adjancy matrix.

Frequency of data (Fd2) = summation of all the columns of transaction matrix.

Step 6 applying minimum support i.e., threshold (Th)

Th = given by database admin If {Fd2 > Th} Then user is intruder.

Else The user is genuine.

IV. Result

This graph shows the frequency of the data assessed by the users in each column. That means that how frequently the column

data has been assessed by the user. In the following graph we use number of users in Y axis and number of columns in X axis. When number of users is 100 and number of columns is 26. X axis represented column and Y axis represented by a users.

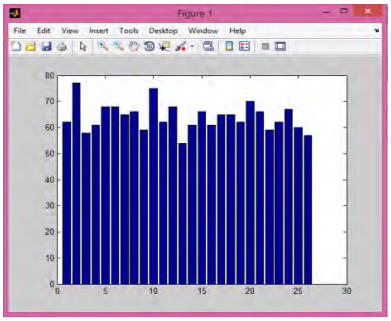


Fig. 6: Cloumn vs users graph

Next graph is showing the number of user is 12 and number of columns is 30 they represented is a bar graph form

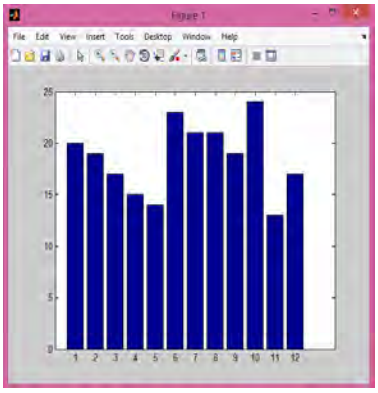


Fig. 7: Cloumn vs users graph

Fig no 5 is represented by the evolution time on increasing the number of user's dependent of the time. Execution time is calculated by the on the basis of mat lab tic and tock" and the result in showing a box In healthy network the performance of PDF is always fine and also possible to reaches at 100% for certain time duration. X axis represented execution time number of users and Y axis represented a no number of users.

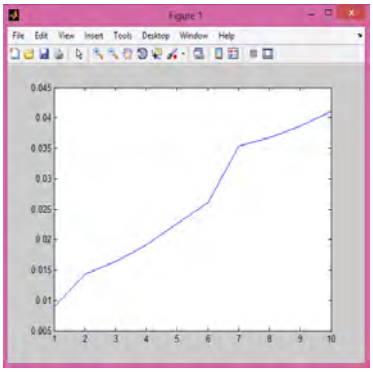


Fig. 8: Evaluation time on increasing the no. of users

Fig no 6 is represented by Execution time on increasing the number of data set. Execution time is calculated by number of data set. X axis represented data set and Execution time on increasing the data set Y axis represented a no number of users.

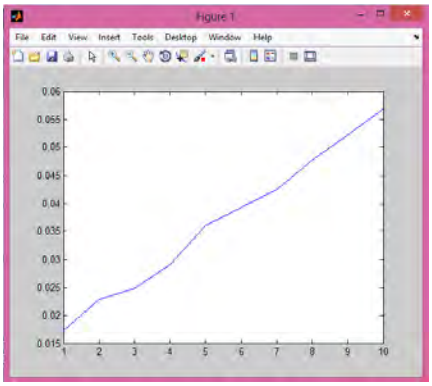


Fig. 9. Represents a executation time

V. Conclusion and Future Scope

Conclusion

Web mining is the Data Mining type that automatically discovers or extracts the information from web documents. In this paper, An enhanced novel approach on semantic inference model for database security in FP tree. Our goal of research is secure communication of the data base. We proposed a technique to prevent users to infer sensitive information from a series of seemingly innocuous queries. Based on the data dependency, the database schema and the semantic knowledge, we constructed a semantic inference model (SIM) that links all the related attributes and thus, represent all possible inference channels from any attributes to the set of pre-assigned sensitive attributes.

Future Scope

In Future, we have plan following things:

(i) In this project we have used long mathematical calculations, we are trying to simplify mathematical calculation or to use other methods for better performance and efficiency.

(ii) More Security at upper level and decrease the number of users of the database.

References

[1] Y. Chen, and W. W. Chu, " Database security protection via inference detection," *IEEE International Conference on Intelligence and Security Informatics*, May 2006.

[2] R. Chopade, A. Savyanavar and B. N. Jagdale, "Semantic inference model for database inference de-

tection violation," *International Journal on Computer Science and Engineering (IJCSE)*, vol. 3, no. 2, Feb. 2011.

[3] F. Gullo, "From patterns in data to knowledge discovery: What data mining can do 3rd international conference frontiers in diagnostic technologies," *Physics Procedia*, vol. 62, pp. 18-22, 2015.

[4] S. Nasreen, M. A. Azam, and S. Khurram, "Frequent pattern mining algorithms for finding associated frequent patterns for data streams: a survey," *The 5th International Conference on Emerging Ubiquitous Systems and Pervasive Networks*, vol. 37, pp. 109-116, 2014.

[5] K. Gadia, and K. Bhowmick, "Parallel text mining in multicore systems using FP-Tree algorithm," *International Conference on Advanced Computing Technologies and Applications*, vol. 45, pp. 101-110, 2015.

[6] R. Mythilya, A. Banu, and S. Raghunathan "Clustering models for data stream mining," *International Conference on Information and Communication Technologies*, vol. 46, pp. 619-626, 2015.

[7] S. Mishra, D. Mishra, and S. K. Satapathy. "Fuzzy frequent pattern mining from gene expression data using dynamic multi-swarm particle swarm optimization," *2nd International Conference on Computer, Communication, Control and Information Technology*, vol. 4, pp. 797-801, 2012.

[8] H. A. Park, T. Kim, M. Li, H. S. Shon, J. S. Park, and K. H. Ryu, "Application of gap constraints given sequential frequent pattern mining for protein function prediction," *Osong Public Health Res Perspect*, vol. 6, no. 2, pp. 112-120, 2015.

[9] K. Okoye, and A. R. Tawil, "A semantic rule based approach supported by process mining for personalised adaptive learning," *The 5th International Conference on Emerging Ubiquitous Systems and Pervasive Networks*, vol. 37, pp. 203-210, 2014.

[10] P. Brauna, A. Cuzzocrea, C. Leung, R. K. Mackinnon, and S. K. Tanbeer, "A tree-based algorithm for mining diverse social entities," *18th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems*, vol. 35, pp. 223-232.

[11] A. Bhandari, A. Gupta, and D. Das, "Improvised a priori algorithm using frequent pattern tree for real time applications in data mining," *Procedia Computer Science*, vol. 46, pp. 644-651, 2015.

[12] M. Alwadi and G. Chettyb, "Energy efficient data mining scheme for high dimensional data," International Conference on Information and Communication Technologies, *Procedia Computer Science*, vol. 46, pp. 483-490, 2015.

[13] M. H. N. Shahraki, N. Mustapha, and M. N. B. Somalian, "Efficient candidacy reduction for frequent pattern mining, *International Journal of Computer Science and Information Security (IJCSIS)*, vol. 6, no. 3, 2009.

[14] U. M. Fayyad, "Data mining and knowledge discovery in databases applications in astronomy and planetary science," AAAI-96 Proceedings,

[15] S. G. Shanthi, and A. S. Thanamani, "Web page categorization using web mining," *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, vol. 1, no. 7, 012.