

SECURED DATA MINING WITH (ALA) ANTI LEECHING ALGORITHM

U. Manju, J. Jayapradha

Abstract—Top-k pairs and top-k objects requests have to get significant notice by the research group. The framework handles several top-k pairs and top-k object queries. It allowed to uses a different gaining window. We use different types of k and a different size of sliding windows. Also the framework to allow the users to allows the users explain arbitrarily complex gaining functions and supports out of the data streams. To all the queries that use the same scoring functions. We need to maintain only one k skyband. In existing, we present the first approach to answer a broad class of top-k pairs and top-k objects queries, but this only concentrates on retrieval speed but there is possibility for leaching attack which our proposed system can overcome.

Index Terms—Top-k pairs, Top-k object query, Leaching attack, Data mining.

I. INTRODUCTION

A scoring function that computes the score of an object, a top-k objects query returns k objects with the smallest scores. Given a scoring function; that computes the score of a pair of object; a top-k pair query returns k pairs with the smallest scores among all possible pairs of objects. k closest pairs queries, k furthest pairs queries and their variants are some well studied examples of top-k pairs queries that rank the pairs on distance functions.

Due to the importance of the top-k queries, numerous algorithms have been proposed to answer several variants of the top-k objects and top-k pair queries [1], [2], [3], [4], [5], [6]. Our focus in this paper is on developing efficient techniques for top-k queries over sliding windows. Top-k objects queries over sliding windows have many applications and have received significant research attention in the past few years [6], [7], [8]. However, Top-k pair query over sliding windows has not been studied well. Therefore, our main focus in this paper is on presenting the techniques for top-k pair queries. Then, we show that the framework can be used to answer top-k objects queries.

Top-k pair queries have many interesting applications in different areas such as wireless sensor network, stock market, traffic monitoring and internet applications, etc. For instance, top-k pair queries can be used for pair-trading [9]. Pair-trading is a market neutral strategy according to which two correlated stocks that follow same day-to-day price movement (e.g., Coca-Cola and Pepsi) may be used to earn profit when the correlation between them weakens, i.e., one stock goes up and the other goes down. The profit can be earned by buying the underperforming stock and

selling it when the divergence between the two stocks returns to normal. A top-k pairs query can be issued to obtain the pairs of stocks that are correlated (e.g., they belong to the same business sector and have similar fundamentals such as market caps, dividends, etc.) and display different trends. Pair-trading can be profitable only if the trader is the first one to capitalize on the opportunity [9]. Hence, the trader may want to continuously monitor the top-k pairs from the most recent data (e.g., a sliding window containing most recent n items). In this proposed system we using leaching algorithm.

We present a unified approach to answer a broad class of top-k pairs query including the k closest pairs queries, the k furthest pairs queries and their variants. The expected performance of the proposed algorithms is optimal when the queries involve two or less attributes. Extensive experiments demonstrate the efficiency of our proposed algorithms.

We present efficient techniques to answer a broad class of top-k pairs query over sliding windows. We provide a detailed complexity analysis and show that the storage requirement and the performance of our algorithms are reasonably close to the lower bound. We verify this by an extensive experimental evaluation and demonstrate the efficiency of our approach.

II. RELATED WORKS

A. Top-k Pairs and Top-k Queries

Top-k pairs and top-k objects queries have received significant attention by the research community. In existing, we present the first approach to answer a broad class of top-k pairs and top-k objects queries.

Consider another example of an online auction website. A user may be interested in finding the pairs of products that have similar specifications but are sold at very different prices (i.e., different final bids). Such pairs may be used to understand the user behavior and market trends, e.g., suitable bidding time for buyers and suitable bidding closing time for sellers etc. An analyst or a user may issue the following query to obtain top-k pairs of such products sold during some days.

The existing systems have no security for database. Leeching attack can be done through SQL injection. Coding analyzation for weak code detection is not present.

B. Leaching Process

To overcome the leeching and SQL injection attack in the areas like existing system, we propose Anti-leeching algorithm. Using the existing system, query processing speed will be high during transaction by the user. Verification is done in all the fields to prevent from leeching attack. If any query hits the database then privilege verification will be done. If any attacking query enters into the system then the alarm notification will be send to all the

U. Manju, Krishnasamy College of Engineering and Technology, Cuddalore, Tamil Nadu, India,

J. Jayapradha, Krishnasamy College of Engineering and Technology, Cuddalore, Tamil Nadu, India.

database admin. If the privilege is from admin then it will send the query to all other admin. They will verify whether that query is an attacking query or not. If yes that query is blocked

Encryption is done between admin during transfer of response. This process will prevent from hacking attack. If the privilege is from client, it will verify the privilege, the privilege is other than client's privilege then it will send the query to all the admin, They will verify whether that query is an attacking query or not. If yes that query is blocked. If the privilege is from user, it will verify the privilege, the privilege is other than user's privilege then it will send the query to all the admin, They will verify whether that query is an attacking query or not. If yes it will block that query and also the user. Admin can verify the path of the attacking query. Develop a static code analyzer tool. Apply the weak coding to the tool. Our static code analyzer tool can analyze the applied coding and check the path of the attack and it will report the weak places by which the attacker may attack. External user also can verify their coding by using our static code analyzer tool.

Both data mining performance as well as security for database is provided. The most threatening leeching attack cannot be done here. Anti leeching attack algorithm not only detects attack it also prevents the attackers from sensitive data and provides high security. The security leaks can be found at coding level so the developers can modify them to make strong.

III. METHODS

The proposed system to implement the partition parallelism we follow below methods:

A. SQL Injection Attack Detection

In this module we are going to create an user application by which the User is allowed to access the data from the Server of the Service Provider. Here first the User wants to create an account and then only they are allowed to access the Network. Once the User creates an account, they are to login into their account and request the Job from the Service Provider. Based on the User's request, the Service Provider will process the User requested Job and respond to them. All the User details will be stored in the Database of the Service Provider. In this Project, we will design the User Interface Frame to Communicate with the Server through Network Coding using the programming Languages like Java/.Net. By sending the request to Server Provider, the User can access the requested data if they authenticated by the Cloud Service Provider.

Bank server will be created for maintaining all bank transaction. Clients will be registered under banks for further transactions. Server will maintain black list. This black list will contain the existing attackers list.

Data base admin will be created to maintain the database privileges. It will allocate all the privileges according to the designation. There will be n number of db admin to maintain this database depending upon the organization.

The clients to request to bank data base system for online transaction. The request privileges sent to database system. So the bank databases to send privilege to client. Also the database send authentication to client.

When the queries from different privilege tries to execute in the database a notification will be sent to all admin. They will verify and double check the query for harm full query. All db admin should check and accept the query then only the query will execute success fully. If a single db admin denies the query then the query will not be executed and the user will be blocked (Fig. 1).

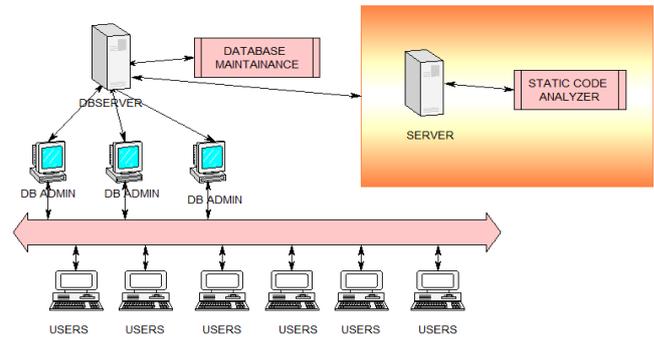


Fig. 1 Leeching processes.

The client sent to the query implementation to server. The query verified by database to if the query is any vulnerable it will be denied database. If the query accepted by database it is executed to client. Then automatically services provided by database to client.

The banks sever to provide the services for query verification, query acceptance, denying query. Client's to implements the query to bank server. The query verified by server, if it is valid, it executed, otherwise denying this query. Now the clients to access the bank database through online. The client request to bank sever, it provide bank services to clients.

D. Static Code Analyzer Creation

After blocking the attacker the software will be applied to the static code analyzer which will check for the weak code design. This weak code will be highlighted for modification to make it strong and to prevent future attacks. This makes the software mores stronger.

The static code analyzer used to analyze the query it valid or not. The databases to provide code for analyzer to verify the client query. If the code is weak the analyzers, that detected weak code is sent to database.

In many real-world applications, the objects do not arrive in correct order due to various reasons such as network delay and data sent from different sources [10], [11]. Such streams are called out-of-order data streams. In out-of-order streams, the age of an object does not denote the time since it has been in the sliding window (i.e., the time since it was received) but it denotes the time since it was sent to the server. Hence, the age of a newly received object may be larger than the age of objects received earlier.

We vary the number of colors (each object is randomly assigned one color) and study the performance of our algorithms for heterochromatic and homochromatic queries. Note that the homochromatic query is the same as a non-chromatic query when only one color is used. The cost of both homochromatic and heterochromatic queries is lower than the cost of non-chromatic queries. The cost of homochromatic queries decreases with the increase in number of colors because the number of valid pairs decreases. In contrast, the cost of heterochromatic queries increases because the number of valid pairs increases when the number of colors is larger [12-25].

IV. CONCLUSION

In this paper we have presented techniques for answering ad-hoc top-k queries over streaming data. We have introduced techniques based on the notion of geometric arrangements and presented their practical realization in a data streaming scenario. We have presented analytical and experimental results quantifying the tradeoffs around the choice of various parameters inherent in our techniques. Our results demonstrate the practical utility of our methods.

We present efficient techniques to answer a broad class of top-k pairs and top-k objects queries over sliding windows. The efficiency of the proposed techniques is evaluated by a detailed complexity analysis and an extensive experimental study. The proposed framework can handle arbitrary scoring functions, supports queries with any window size and works for out-of-order data streams. We using that leaching algorithm, both data mining performance as well as security for database is provided. The most threatening leeching attack cannot be done here. Anti leeching attack algorithm not only detects attack it also prevents the attackers from sensitive data and provides high security. The security leaks can be found at coding level so the developers can modify them to make strong [26-28].

REFERENCES

- [1] M.A. Cheema, X. Lin, H. Wang, J. Wang, and W. Zhang, "A Unified Approach for Computing Top-k Pairs in Multidimensional Space," Proc. IEEE 27th Int'l Conf. Data Eng. (ICDE), 2011.
- [2] G.R. Hjaltason and H. Samet, "Incremental Distance Join Algorithms for Spatial Databases," Proc. ACM SIGMOD Int'l Conf. Management of Data, 1998.
- [3] A. Corral, Y. Manolopoulos, Y. Theodoridis, and M. Vassilakopoulos, "Closest Pair Queries in Spatial Databases," Proc. SIGMOD Int'l Conf. Management of Data, 2000.
- [4] M. Smid, "Closest-Point Problems in Computational Geometry," Handbook on Computational Geometry, North-Holland, 1997.
- [5] C. Yang and K.-I. Lin, "An Index Structure for Improving Nearest Closest Pairs and Related Join Queries in Spatial Databases," Proc. Database Eng. and Applications Symp. (IDEAS), 2002.
- [6] K. Mouratidis, S. Bakiras, and D. Papadias, "Continuous Monitoring of Top-k Queries over Sliding Windows," Proc. ACM SIGMOD Int'l Conf. Management of Data, 2006.
- [7] K. Mouratidis and D. Papadias, "Continuous Nearest Neighbor Queries over Sliding Windows," IEEE Trans. Knowledge and Data Eng., vol. 19, no. 6, pp. 789-803, June 2007.
- [8] C. Böhm, B.C. Ooi, C. Plant, and Y. Yan, "Efficiently Processing Continuous k-NN Queries on Data Streams," Proc. IEEE 23rd Int'l Conf. Data Eng. (ICDE), 2007.
- [9] G. Vidyamurthy, Pairs Trading: Quantitative Methods and Analysis. John Wiley & Sons, 2004.
- [10] Z. Shen, M.A. Cheema, X. Lin, W. Zhang, and H. Wang, "Efficiently Monitoring Top-k Pairs over Sliding Windows," Proc. IEEE 28th Int'l Conf. Data Eng. (ICDE), 2012.
- [11] R. Fagin, A. Lotem, and M. Naor, "Optimal Aggregation Algorithms for Middleware," J. Computer and System Sciences, vol. 66, 2003.
- [12] S. Nepal and M.V. Ramakrishna, "Query Processing Issues in Image (Multimedia) Databases," Proc. 15th Int'l Conf. Data Eng. (ICDE), 1999.
- [13] U. Gěuntzer, W.-T. Balke, and W. Kießling, "Optimizing Multi-Feature Queries for Image Databases," Proc. 26th Int'l Conf. Very Large Data Bases (VLDB), 2000.
- [14] L.H. U, N. Mamoulis, and M.L. Yiu, "Continuous Monitoring of Exclusive Closest Pairs," Proc. 10th Int'l Conf. Advances in Spatial and Temporal Databases (SSTD), 2007.
- [15] F. Angiulli and C. Pizzuti, "An Approximate Algorithm for Top-k Closest Pairs Join Query in Large High Dimensional Data," Proc. Int'l Database Eng. and Applications Symp. (DKE), 2005.
- [16] P. Zhou, D. Zhang, B. Salzberg, G. Cooperman, and G. Kollios, "Close Pair Queries in Moving Object Databases," Proc. 13th Ann. ACM Int'l Workshop Geographic Information Systems (GIS), 2005.
- [17] G. Das, D. Gunopulos, N. Koudas, and N. Sarkas, "Ad-Hoc Top-k Query Answering for Data Streams," Proc. 33rd Int'l Conf. Very Large Data Bases (VLDB), 2007.
- [18] Z. Shen, M.A. Cheema, and X. Lin, "Loyalty-Based Selection: Retrieving Objects that Persistently Satisfy Criteria," Proc. 21st ACM Int'l Conf. Information and Knowledge Management (CIKM), 2012.
- [19] D. Papadias, Y. Tao, G. Fu, and B. Seeger, "Progressive Skyline Computation in Database Systems," ACM Trans. Database Systems, vol. 30, pp. 41-82, 2005.
- [20] W. Zhang, X. Lin, Y. Zhang, W. Wang, and J.X. Yu, "Probabilistic Skyline Operator over Sliding Windows," Proc. IEEE 25th Int'l Conf. Data Eng. (ICDE), 2009.
- [21] D.E. Knuth, The Art of Computer Programming, Volume I: Fundamental Algorithms, second ed., Addison-Wesley, 1973.
- [22] E.M. McCreight, "Priority Search Trees," SIAM J. Computing, vol. 14, no. 2, pp. 257-276, May 1985.
- [23] M. Blum, R.W. Floyd, V.R. Pratt, R.L. Rivest, and R.E. Tarjan, "Time Bounds for Selection," J. Computer and System Sciences, vol. 7, pp. 448-461, 1973.
- [24] P. Tsaparas, T. Palpanas, Y. Kotidis, N. Koudas, and D. Srivastava, "Ranked Join Indices," Proc. 19th Int'l Conf. Data Eng. (ICDE), 2003.
- [25] G. Cormode, F. Korn, and S. Tirthapura, "Time-Decaying Aggregates in Out-of-Order Streams," Proc. 27th ACM SIGMODSIGACT-SIGART Symp. Principles of Database Systems (PODS), pp. 89-98, 2008.
- [26] M. Liu, M. Li, D. Golovnya, E.A. Rundensteiner, and K.T. Claypool, "Sequence Pattern Query Processing over Out-of-Order Event Streams," Proc. IEEE 25th Int'l Conf. Data Eng. (ICDE), pp. 784-795, 2009.
- [27] <http://db.csail.mit.edu/labdata/labdata.html>, 2014.
- [28] S. Börczsonyi, D. Kossmann, and K. Stocker, "The Skyline Operator," Proc. 17th Int'l Conf. Data Eng. (ICDE), pp. 421-430, 2001.



U. Manju completed her B.Tech.(CSE) degree in the year 2013. At present she is pursuing M.E. (CSE) in Krishnasamy College of Engineering and Technology, Cuddalore, Tamil Nadu, India. Her research interests lies in the areas of Big Data, Knowledge and Data Engineering in Data Mining, Cryptographic Security in Cloud Computing and Secure Computing.



J. Jayapradha received the M.E degree in Computer Science and Engineering from Anna University, Chennai in 2008. She is currently working as Assistant Professor in Krishnasamy College of Engineering and Technology. She has published few papers in National and International conferences and journals. Her area of interests includes Operating system, Compiler design and Software Engineering. She is a member of ISTE.