Proceedings of
INTERNATIONAL CONFERENCE ON COMPUTING, COMMUNICATION AND ENERGY SYSTEMS
(ICCCES-16)
In Association with IET, UK & Sponsored by TEQIP-II
29th -30th, Jan. 2016

Paper ID: CSEIT07

DP-FIM FOR DATA LINKAGE USING ONE CLASS CLUSTERING TREE FOR MOVIE RECOMMENDATION

Ms. Bhagyashri A. Patil
Prof. Mrs. R. J. Deshmukh
Department Of Computer Science and Technology, DOT Shivaji University Kolhapur India

Abstract— Frequent itemset plays a vital role in many data mining fields such as- finance and biology. However, release of powerful patterns and trends are increasing concern on personal privacy, so the problem is –How to perform frequent itemset mining on transaction database which satisfies differential privacy? The proposed approach is called as DP-FIM (Differentially Private Frequent Itemset Mining) which can concurrently provide huge level of data utility and huge level of data privacy. It is very difficult to achieve data utility and privacy when existing system having long transaction. A system uses a transaction splitting technique to divide a long transaction into sub-transaction whose cardinality is no more than specified limit. To avert information loss generated by transaction splitting, run time estimation method is used to estimate defined support of itemset in original database. Data linkage compares records in one database with records in another database to match them. Data linkage is performed among the tables to cluster the data. But, existing data linkage methods do not handle one to many, many to many and many to one linkage field values. This system investigates a method which links different entities by using OCC-Tree (One Class Clustering Tree) in which inner node contains attribute from one database and leaf node holds close representation of matching records from another database.

Keywords—Data Mining, Differential Privacy, Data Linkage, Frequent Itemset, One class clustering tree,Recommandation System.

I. INTRODUCTION

Frequent itemset mining is an essential component in number of important data mining tasks with broad applications, ranging from web usage mining system to location based recommendation system.[1][8] The patient health care records and user behavior records are hypersensitive data, so releasing of exposed frequent itemset might act as considerable risks to personal privacy.

Frequent itemset mining algorithm creates a privacy matter how we can confident that the reporting frequent itemset in database does not alert private information about individuals whose data is being studied? This problem is compounded by the fact that we may not even know what data the individuals would like to protect nor what background information might consumed by an attacker. These combining factors are the exactly the once addressed by differential privacy, whichguarantees that the existence of individuals data in training database does not notify much about that individual. [6][9]

The main problem is to form a differentially private frequent itemset algorithm which can simultaneously provide huge level of data utility and huge level of data privacy. This task is very challenging due to the possibility of long transaction. In particular, if transaction has more items than specified number of items then items are deleted until the transaction is under the limit. However, such approach may cause too much information loss and result in poor performance, so divide the long transaction into sub-transaction and guarantee that the cardinality of sub-transaction is under the specified limit.

The most suitable frequent itemset mining algorithm is FP growth algorithm which extracts bottleneck of Apriori algorithm. An Apriori algorithm is a breath first search where candidate set is generated. In contrast, FP growth is a depth search algorithm which require no candidate generation. The problem of Apriori algorithm is dealt by compact data structure called as frequent pattern tree or FP-tree which stores the database in tree structure and every item has linked list going through all transaction that contains item.

A differentially private frequent itemset algorithm based on FP growth algorithm also called as PFP growth algorithm. The PFP growth algorithm has two phases-1] Preprocessing Phase 2] Mining Phase. In preprocessing phase, the privacy and utility trade off can be achieved by assigning the length of transaction. The Preprocessing phase generates the renewed database from existing database. In mining phase, the input is renewed database and given threshold can privately determine frequent itemset. [15]

Data linkage is also called as record linkage which is a process of analyzing different data items that refers to same entity among different data sources. The main aim of data linkage is to join database that do not share foreign key or common identifier. To implement a data linkage purpose a method based on "One Class Clustering Tree (OCC-Tree)" is used which indicates that entities should be linked together. [13][14] An OCC-tree is a tree in which the inner nodes consist only of features describing the first set of entities, while the leaves of the tree represent the features of their matching entities from second database. The OCC-tree can be used for
recommendation system where main goal is to match new user of the system with the items that they are expected to like based on their demographic attribute [14]. Here, we implement movie recommendation system which can suggest a set of movies to user based on their interest, or the popularities of the movies. Although, a number of movie recommendation systems have been proposed, most of these cannot recommend movies to the existing users or to the new user. In this paper, we propose systems which suggest movie to new user and others also.

II. LITERATURE REVIEW

Most of the formerly proposed algorithms for mining frequent itemset can be grouped into: the Apriori method and

the FP-growth method. The Apriori method [1] (**Agrawal & Srikant, 1994**) finds frequent itemsets which generate candidate itemset. To avoid generating too much candidates, the FP-growth method uses FP-tree **(Grahne Zhu, 2005; Liu, Lu, Lou, Xu, & Yu, 2004)** to store databases and applies a divide and conquer method to mine frequent itemsets directly, which make it much more efficient than Apriori method.[2][3]

Differential privacy is first proposed by **Dwork et al. (2006)** and has developed as the de facto standard notation for research in private data analysis [4]. **Bhaskar et al. (2010) [7]** presented two differentially private algorithms for top-k frequent pattern mining which adopt exponential mechanism (**McSherry and Talwar, 2007**) [6] and Laplace mechanism (**Dwork et al., 2006**) respectively [5].

Zeng C, Naughton JF, Cai J-Y (2012) proposed a transaction truncating approach where items in a long transactions are deleted until the transactions cardinality is under a stated limit. Based on a transaction truncating approach, they presented a differentially private frequent itemset mining algorithm. [8]

To meet the challenge of high dimensionality of transactional database **Liet al. (2012)** proposed the PrivBasis algorithm which plans the input database onto sevral sets of dimensions for differentially private top-k frequent itemset mining [9] . **Bonomi and Xiong (2013)** presented a two-phase algorithm for differentially privately mining both prefixes and substring patterns. In the first phase, it brings about frequent prefixes and a candidate set of substring patterns. In the second phase, it clarifies the count of the probable frequent substrings patterns [10].

Based on Markov Chain Monte Carlo (MCMC) sampling, **Shen and Yu (2013)** proposed differentially private frequent graph pattern mining algorithm which does not build on the output of a non-private mining algorithm [11]. **Chen et al. (2011)** proposed a probabilistic top-down algorithm to efficiently generate release of set valued data in a differentially private manner with secured utility for mining frequent itemset [12].

Ma'ayan Dror, Asaf Shabtai, Lior Rokach, and Yuval Elovici (2014) proposed an OCCT (One Class Clustering Tree) approach in which the task of matching entities from two different data source that do not share a common identifier is a data linkage [13]. **A.Gershman et al. (2010)** proposed a new method for decision-tree-based recommender systems [14].

III. SYSTEM ARCHITECTURE

Our work is related to mine frequent itemsets to achieve data linkage via-transaction splitting and OCC-Tree with differential privacy. We propose an OCC-Tree based on differentially private frequent itemsets via transaction splitting for data linkage to give the recommendation system. The proposed system consist two main phases,

Phase-1- Frequent item set mining phase and
Phase-2- OCC-tree generation phase

Phase 1: Frequent item set mining phase

Input: Training Database
Output: Pattern Sets

The phase-I consists of following stages: A] Pre-processing Phase-

In this step of phase 1, if size of transaction is greater than described threshold then split the transaction into sub transactions and it gives the guarantee that each sub-transaction is under the threshold (i.e.-limit). The pre-processing phase is carried out only once for given database. By using smart splitting method, we limit the length of transactions without much information loss. For example, consider itemset {a,b,c} and (d,e,f} are frequent and specified threshold is 4. Given transaction t= {a,b,c,d,e,f} if we truncate the transaction t becomes as {a,b,c,d} so the itemsets which are frequent in original database may turn into infrequent itemset .

Algorithm 1- Smart Splitting Of Transaction-
Input - Transaction t of length p, maximum length constraint Lm;
Output-q= [p/Lm]
R ← Φ
Consider initial node list N_L;
for i from 1 to q do
  $t_i \leftarrow \Phi$
  select a node n_l with highest number of items from N_L;
  Add the items in n_l into ti and remove n_i from N_L;
  Sort the remaining node in N_L;
  For each node n_l' in N_L do
    if $|t_i| + |n_l'| <= L_m$
      Add the items in n_l' into t_i and remove from n_l'
    end if
  end for
  add t_i into R
  end for
Return R;

Fig-2- Algorithm for Smart Splitting Of Transaction

Pre-processing phase creates Transformed database or renewed database.

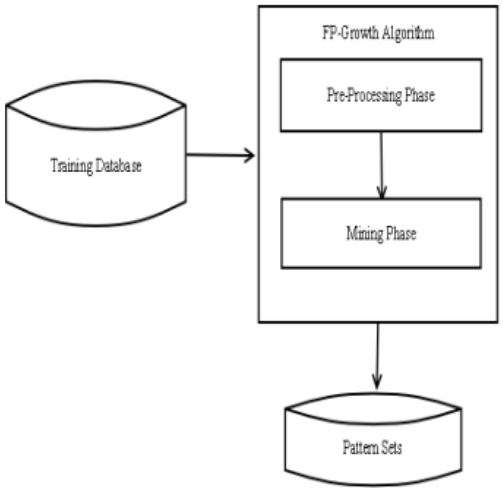

Fig-1: Flow diagram of Phase-1 B] Mining Phase

In the mining phase, given the transformed database and a specified threshold, we secretly identify frequent itemsets. The Transaction splitting might bring frequency

information loss, so we define a run-time estimation method to estimate the original support of itemsets in the original database and to balance the information loss caused by transaction splitting. At the time of mining put dynamic reduction method to compute the number of support computation, so it break the amount of noise essential for differential privacy. Mining phase gives pattern sets as output.

In run time estimation method, suppose a transaction t= {a,b,c,d} is branched into t1={a,b} and t2= {c,d} . Suppose, the support of itemset may decrease from 1 to 0 due to splitting, the information loss may be occurs. So, we propose run time estimation method which consists of noisy support of itemset in transformed database to estimate its actual support in transformed database and to compute its actual support in the original database. Here, we estimate "Average" support to determine whether itemset is frequent and its "maximum" support to decide whether to use itemset to generate candidate frequent itemset.

Phase 2: OCC-tree generation phase

Input: Pattern sets generated by Phase-1 and records for data linkage

Output: recommended output

The phase-II consists of following stages:

A] LPI Algorithm

Least probable intersections (LPI) algorithm proposes a distinct combination of attributes as a unique identifier of an entity. The goal is to find a splitting attribute for which there is the least amount of identifiers that are shared, in comparison to a random split of the same size.

Least Probable intersection method (LPI) performed for each possible split .If all possible splitting attributes is smaller than predefined threshold then random splitting is formed.

B] OCC-Tree

OCC-Tree consists of attributes which can be derived from the LPI algorithm in tree like structure. The goal is to achieve a tree which consists of a small amount of nodes. Smaller tree would better to conclude and to avoid the over fitting and forms straightforward representation for human eye which is easy to understand. Therefore, it is essential to use an effective splitting criterion in order to build the tree. OCC-Tree is similar to clustering tree where each node represents a cluster and whole tree describes the hierarchy. In particular, we create the cluster by analyzing the attributes representation in first table (T_A) while data is clustered from the table (T_B).

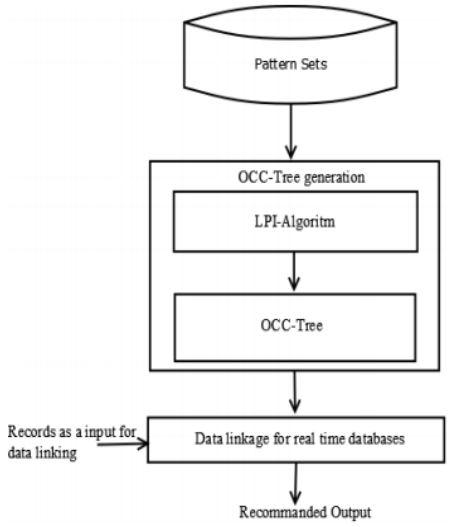

Fig-3- Flow Diagram for Phase-2

C] Data linkage for Real Time Databases

During the linkage phase, each possible pair of test records is tested against the linkage model in order to determine if the pair is a match. This process produces a score representing the probability of the record pair being a true match for the movie.

IV. CONCLUSION

In this paper, a novel technique for the differentially private mining of frequent patterns is studied. FP Growth algorithm will be developed which will consists of preprocessing phase and mining phase to achieve huge level of data privacy and huge level of utility as well as efficiency. A one class clustering tree method for data linkage to generate a decision tree in which the inner nodes consist only of features describing the first set of entities, while the leaves of the tree represent the features of their matching entities from second database The decision tree developed by one class clustering tree approach have been extensively used in movie recommendation system in which linkage may be one to one or many to many or may be many to one . The recommendation system will provide high quality recommendation.

V. REFERENCES

[1] Agrawal, R., & Srikant, R. (1994). Fast algorithm for mining Association rules. In:VLDB, pp. 487–499.

[2] Grahne, G., & Zhu, J. (2005). Fast algorithms for frequent itemset mining using FPtrees.IEEE TKDE Journal, 17(10), 1347–1362.

[3] Liu, G., Lu, H., Lou, W., Xu, Y., & Yu, J. X. (2004). Efficient mining of frequent itemsets using ascending frequency ordered prefix-tree. DMKD Journal, 9(3), 249–274.

[4] Dwork C, McSherry F, Nissim K, Smith A. Calibrating noise to sensitivity in private data analysis. In: TCC; 2006. p. 265-84.

[5] Dwork C. Differential privacy: a survey of results. In: TAMC; 2008. p. 1-19.

[6] McSherry F, Talwar K. Mechanism design via differential privacy.In: FOCS; 2007. p. 94-103.

[7] Bhaskar R, Laxman S, Smith A, Thakurta A. Discovering frequent patterns in sensitive data. In: KDD; 2010. p. 503-12.

[8] Zeng C, Naughton JF, Cai J-Y. "On differentially private frequent itemset mining." PVLDB 2012; 6(1):25-36

[9] Li N, Qardaji WH, Su D, Cao J. Privbasis: frequent itemset mining with differential privacy. PVLDB 2012;5(11):1340-51.

[10] Bonomi L, Xiong L. A two-phase algorithm for mining sequential patterns with differential privacy. In: CIKM; 2013. p. 269-78.

[11] Shen E, Yu T. Mining frequent graph patterns with differential privacy. In: KDD; 2013. p. 545-53

[12] Chen R, Mohammed N, Fung BCM, Desai BC, Xiong L. Publishing set-valued data via differential privacy. PVLDB 2011;4(11):1087-98

[13] Ma'ayan Dror, Asaf Shabtai, Lior Rokach, and Yuval Elovici, "OCCT: A One-Class Clustering Tree for implementing One-to-Many Data Linkage", IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 26, NO. 3, MARCH 2014.

[14] A. Gershman et al., "A Decision Tree Based Recommender System," Proc. 10th Int'l Conf. Innovative Internet Community Services, pp. 170-179, 2010.

[15] Sen Su, Shengzhi Xu, Xiang Cheng, Zhengyi Li, and Fangchun Yang, "Differentially private frequent itremset mining via transaction spliitting"

.