FREQUENT SUB GRAPH IDENTIFICATION USING TWO-PHASE FSM ALGORITHM

P V GEETHA SRAVANI[1], J LAKSHMI[2]

[1] PG Student, Eswar College of Engineering, Narasaraopet

[2] Asst. Professor, Eswar College of Engineering, Narasaraopet

Abstract:

Mining frequent subgraphs from a collection of input graphs is an important task for exploratory data analysis on graph data. However, if the input graphs contain sensitive information, releasing discovered frequent subgraphs may pose considerable threats to individual privacy. In this paper, we study the problem of frequent subgraph mining (FSM) under the rigorous differential privacy model. We present a two-phase differentially private FSM algorithm, which is referred to as DFG. In DFG, frequent subgraphs are privately identified in the first phase, and the noisy support of each identified frequent subgraph is calculated in the second phase. In particular, to privately identity frequent subgraphs, we propose a frequent subgraph identification approach, which can improve the accuracy of discovered frequent subgraphs through candidate pruning. Moreover, to compute the noisy support of each identified frequent subgraph, we devise a lattice-based noisy support computation approach, which leverages the inclusion relations between the discovered frequent subgraphs to improve the accuracy of the noisy supports. Through formal privacy analysis, we prove that DFG satisfies ϵ-differential privacy. Extensive experimental results on real datasets show that DFG can privately find frequent subgraphs while achieving high data utility.

1. INTRODUCTION:

FREQUENT subgraph mining (FSM) is a fundamental and essential problem in data mining research. Given a collection of input graphs, FSM aims to find all subgraphs that occur in input graphs more frequently than a given threshold. FSM has practical importance in a number of applications, ranging from bioinformatics to social network analysis. For example, discovering frequent subgraphs in social networks can be vital to understand the mechanics of social interactions. Despite valuable information the discovery of frequent subgraphs can potentially provide, if the data is sensitive (e.g., mobile phone call graphs, trajectory graphs and web-click graphs), directly releasing the frequent subgraphs will pose considerable concerns on the privacy of the users participating in the data. Differential privacy has been proposed as a way to address such problem. Unlike the anonymization-based privacy models (e.g., k-anonymity [3], l-diversity [4]), differential privacy offers strong privacy guarantees and robustness against adversaries with prior knowledge. In general, by randomization mechanisms, such as adding a carefully chosen amount of perturbation noise, differential privacy assures that the output of a computation is insensitive to the change of any individual record, and thus restricting privacy breaches through the results.

To this end, we present a novel differentially private f requentsubgraph mining algorithm, which is referred to as DFG. DFG consists of two phases, where frequent subgraphs are privately identified in the first phase, and the noisy support of the identified frequent subgraphs is calculated in the second phase. In the first phase of DFG, to privately identify frequent subgraphs, we propose a level-wise frequent subgraph identification approach. In this approach, given the candidate subgraphs at a certain level, by leveraging the idea of binary search, we first put forward a binary estimation method to estimate the number of frequent subgraphs at this level.

2. EXISTING SYSTEM:

The key contributions of this paper are summarized as follows.

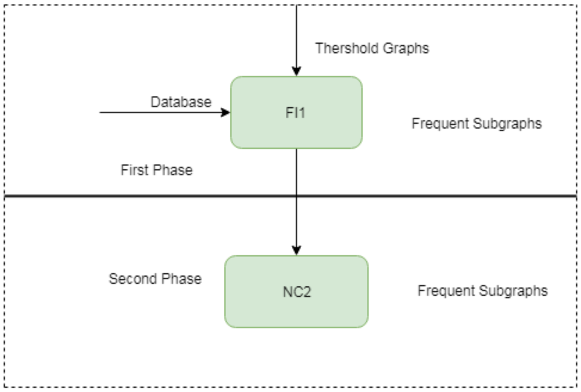
- We present a novel two-phase algorithm for mining frequent subgraphs under differential privacy called DFG, which consists of a frequent subgraph identification phase and a noisy support computation phase. To our best knowledge, it is the first algorithm which can find frequent subgraphs from a collection of input graphs with high data utility while satisfying ϵ-differential privacy. In addition, it can be easily extended for mining other frequent patterns (e.g., frequent itemsets and frequent sequences).
- To privately identify frequent subgraphs from the input graphs, we propose a frequent subgraph identification approach, which includes a binary estimation method and a conditional exponential method.
- To calculate the noisy support of each identified frequent subgraph, we propose a lattice-based noisy support computation approach, which includes a count accumulation method and an error-aware path construction method.
- Through formal privacy analysis, we prove that our DFG algorithm guarantees ϵ-differential privacy. We conduct extensive experiments to evaluate the performance of our algorithm. Experimental results show that our DFG algorithm can privately find frequent subgraphs with high data utility.
- To demonstrate the generality of our DFG algorithm, we also extend it for mining frequent itemsets and sequences, and conduct experiments to evaluate the performance of the extended algorithms. Experimental results show that the extended algorithms can also obtain good performance on differentially private itemset mining and differentially private sequence mining. The rest of our paper is organized as follows.

We provide a literature review in Section 2. Section 3 presents the necessary background on differential privacy and frequent subgraph mining. We identify the technical challenges in designing a differentially private FSM algorithm show the details of our DFG algorithm. The experimental results are reported Finally, we conclude the paper.

3. PROPOSED SYSTEM:

This paper is extended from our previous work. The significant additions in this extended manuscript can be summarized as follows. First, we propose a simple and effective method, namely error-aware path construction, for constructing the set of paths which are taken as input in the count accumulation method. Compared to our previous work, which has to employ two methods (i.e., the path construction and path extension methods) to get the final set of paths, we use only this new method to construct the set of paths. Since the error-aware path construction method can directly reduce the errors of noisy supports during computation, it results in better data utility than the two methods proposed in our previous work, especially when the threshold is relatively low. Second, to illustrate the generality of our DFG algorithm, we extend it for mining both frequent itemsets and frequent sequences. In particular, based on the unique characteristics of FIM, we also propose an optimized approach for computing the noisy support of frequent item. Experimental results on real datasets show that the extended algorithms can also obtain good performance. Third, to evaluate the efficiency of our DFG algorithm, we compare it against the non-private FSM algorithm FSG Experimental results show that our DFG algorithm can obtain comparable efficiency to the FSG algorithm

4. ARCHITECTURE:



5. ALOGRAITHAM

- ✓ **Effect of Privacy Budget**
- ✓ **Effect of NC2 Approach**
- ✓ **Extending DFG for Mining Frequent Sequences**
- ✓ **Extending DFG for Mining Frequent Itemsets**

Effect of Privacy Budget:

The performance of the three algorithms for mining top-50 frequent subgraphs under varying privacy budget ϵ on datasets Cancer and HIV. We can see DFG consistently achieves the best performance at the same level of privacy. All these algorithms perform in a similar way: the utility of the results is improved when ϵ increases. This is because, when ϵ increases, a smaller amount of noise is required and a lower degree of privacy is guaranteed.

Effect of NC2 Approach:

We also evaluate the effectiveness of our lattice-based noisy support computation approach (denoted by NC2) on datasets HIV and SPL. We compare NC2 with two algorithms. The first one is the straightforward algorithm proposed (denoted by Naive), where the noisy supports of identified frequent subgraphs are obtained by directly perturbing their true supports. The second one calculates the noisy support of identified frequent subgraphs by using the noisy support derivation approach (denoted by ND2) proposed in our conference paper [24]. The difference between NC2 and ND2 is that NC2 utilizes the error-aware path construction method to construct the set of paths, while ND2 uses the path construction and path extension methods to construct that set of paths

Extending DFG for Mining Frequent Sequences:

We also extend our DFG algorithm to mine frequent sequences under ϵ-differential privacy. We denote this extended algorithm by DFS. We compare DFS against a state-of-the-art algorithm called PFS2, which privately finds the sequences whose support exceeds a given threshold via sampling-based candidate pruning. In the experiments, we also use two real datasets.

Extending DFG for Mining Frequent Itemsets:

To demonstrate the generality of our DFG algorithm, we extend it to mine frequent itemsets under ϵ-differential privacy. In particular, we denote the extended algorithm which adopts the general approach (i.e., NC2) to compute the noisy support of frequent itemsets by DFI, and denote the extended algorithm which adopts the optimized approach proposed in Sec. 8.4 to compute the noisy support of frequent itemsets by DFI-OPT. We compare DFI and DFI-OPT with the following two algorithms:

1) the PFP-Growth algorithm proposed, which privately finds the itemsets based on FP-growth;

2) thePrivBasis algorithm proposed, which privately finds the k most frequent itemsets by constructing the basis set. In the experiments, we use two real datasets.

Algorithm:

DFG Algorithm definition:

A data-flow graph (DFG) is a graph which represents a datadependancies between a number of operations. Any algorithm consists of a number of ordered operations. Since examples are always better than words, consider the procedure for finding the root of a quadratic equation (algorithm assumes real roots).

Straight Forward Algorithm:

For each pair of triplets, one from each molecule that define 'almost' congruent triangles, compute the transformation that superimposes them. Count the number of point pairs, which are 'almost' superimposed and score the hypotheses by this number. Pick the highest ranking hypotheses and improve the transformation by replacing it with the best RMSD transformation for all matching pairs.

6. IMPLEMENTATION

Admin

In this module, the Admin has to login by using valid user name and password. After login successful he can do some operations such as adding Categories, Adding Sub-Categories, Adding Documents for that

Categories, Viewing and authorizing users, Viewing All Documents with Images and Comments, Viewing All Documents based on Frequent Subgraph Mining and Applying Frequent Subgraph Mining and Showing Scores based on Categories, Viewing All Documents based on Frequent Pattern Mining along with Quality Aware Sub-Graph Matching and Matched Probability, Viewing All Users Search Transaction and Documents Scores Results in a graph.

Adding Categories and Sub-Categories

In this module, the admin adds the category details such as category name and add sub-categories based on Categories. These details will be stored into the database.

Adding Documents

In this module, the admin adds Documents based on categories and sub-categories which include details such as, document image, document name, uses, description and document file. These details will be stored into the database.

Authorize Users

In user's module, the admin can view the list of users who all registered. In this, the admin can view the user's details such as, user name, email, address and admin authorizes the users.

View All Documents

In this module, the admin can view all the documents uploaded and their comments. Details include document name, document category, document sub-category, document image, uses, document description and contents of document.

View All Documents Based on Frequent Subgraph Mining by Categories

In this, the admin can view all documents based on category and applies Frequent Sub graph Mining, which adds all scores for particular document and divides by number of users given scores and stores those results in to table. In this, we can see all calculated and stored Scores for documents based on categories.

View All Documents Based on Frequent Pattern Mining by Categories

In this, the admin can view all the keywords used by all the users for searching documents will be listed along with the number of matched documents and matched probability(Matched Documents : Total Documents) will be shown. We can see what documents found for particular keyword just by clicking on particular keyword.

All User Search Transaction

In this module, the keywords which are used to search documents by all the users will be listed. Details such as user, keyword, found document title, category, sub-category and search date.

View All Document Scores Results

In this, the scores for documents will be calculated by adding all scores given by all users for particular document.

User

In this module, there are n numbers of users are present. User should register before doing any operations. Once user registers, their details will be stored to the database. After registration successful, he has to login by using authorized user name and password. Once Login is successful user will do some operations like viewing their profile details, search documents based on document title, document description and document contents and giving scores and comments. Finding Top K Keywords which are all used to search documents maximum number of times by users, Viewing Other User Comments on documents and user search history.

Search Documents

The user can search documents based on document title, document description and document contents and he can download document. User can view document contents, details and can give scores by selecting (1,2,3,4,5,6,7,8,9,10) any number from dropdown list and he can comment on document.

Top K Keywords

In this , the user can find Top K Keywords by entering any number and The keywords which are all used maximum numbers of times by users for searching will be listed as for number of top keywords you want to list.

View Other Users Comments on Documents

In this, the user can see all comments which are all posted by other users on documents.

View Document Search History

In this module, the user can view all the keywords that he used to search documents and found results such as document title, category, subcategory and date of search.

7. FUTURE WORK:

There are two broad settings for privacy-preserving data analysis. The non-interactive setting aims at developing privacypreserving algorithms that can publish a synthetic dataset, which can then be used to support various data analysis tasks. The interactive setting aims at developing customized privacy-preserving algorithms for various data analysis tasks. For the non-interactive setting, analyze and systematize the state-of-the-art graph data privacy and utility techniques. In particular, they propose and develop a uniform and open-source graph data sharing/publishing system called SeGraph. SeGraph is the first system that enables data owners to anonymize data by existing anonymization techniques, measure the data’s utility, and evaluate the data’s vulnerability against modern De-Anonymization attacks. Different from the work, in this paper, we focus on the interactive setting.We broadly categorize existing studies on differentially private frequent pattern mining into three groups based on the type of pattern being mined, and review each group of studies as follows. A number of studies have been proposed to address the frequent itemset mining (FIM) problem under differential privacy. Bhaskar et al. Utilize the exponential mechanism and Laplace mechanism to develop two differentially private FIM algorithms. To meet the challenge of high dimensionality in transaction databases, Li et al. Introduce an algorithm which projects the high-dimensional database onto lower dimensions. Find the utility and privacy tradeoff in differentially private FIM can be improved by limiting the length of transactions. They propose a transaction truncating method to limit the length of transactions. Propose a transaction splitting method to limit the length of transactions. Based on the FPgrowth algorithm. Present an efficient algorithm called PFP-growth for mining frequent itemsets under differential privacy.

8. CONCLUSION:

In this paper, we study the problem of designing a frequent subgraph mining (FSM) algorithm, which can satisfy ϵ-differential privacy and achieve high data utility. We present a differentially private FSM algorithm called DFG, which consists of a frequent subgraph identification phase and a noisy support computation phase. DFG can be easily extended for mining other frequent patterns, such as frequent itemsets and frequent sequences. Through privacy analysis, we prove that our DFG algorithm satisfies ϵ-differential privacy. Extensive experiments on real datasets show that the proposed DFG algorithm can privately find frequent subgraphs with good data utility.

REFERENCES

[1] R. Bhaskar, S. Laxman, A. Smith, and A. Thakurta, “Discovering frequent patterns in sensitive data,” in KDD, 2010.

[2] C. Dwork, “Differential privacy,” in ICALP, 2006.

[3] L. Sweeney, “k-anonymity: A model for protecting privacy,” Int. J.Uncertain. Fuzziness Knowl.-Base Syst, 2002.

[4] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam,“l-diversity: Privacy beyond k-anonymity,” in ICDE, 2006.

[5] E. Shen and T. Yu, “Mining frequent graph patterns with differential privacy,” in KDD, 2013.

[6] F. McSherry and K. Talwar, “Mechanism design via differential privacy,” in FOCS, 2007.

[7] N. Li, W. Qardaji, D. Su, and J. Cao, “Privbasis: frequent it emset mining with differential privacy,” in VLDB, 2012, pp. 305–316.

[8] C. Zeng, J. F. Naughton, and J.-Y. Cai, “On differentially private frequent itemset mining,” in VLDB, 2012.

[9] S. Xu, S. Su, X. Cheng, Z. Li, and L. Xiong, “Differentially private frequent sequence mining via sampling-based candidate pruning,” in ICDE, 2015.

[10] S. Ji, W. Li, P. Mittal, X. Hu, and R. A. Beyah, “Secgraph: A uniform and open-source evaluation system for graph data anonymization and de-anonymization.” in USENIX Security Symposium, 2015, pp. 303–318.

[11] C. Dwork, F. McSherry, K. Nissim, and A. Smith, “Calibrating noise to sensitivity in private data analysis,” in TCC, 2006.

[12] X. Cheng, S. Su, S. Xu, and Z. Li, “Dp-apriori: A differentially private frequent itemset mining algorithm based on

transaction splitting," Computers & Security, 2015.

AUTHORS PROFILE:

P V Geetha Sravani is a student pursuing MTech(CSE) in Eswar college Of Engineering, Narasaraopet, Guntur.

J. LAKSHMI M.Tech in Computer Science & Engineering. She is currently working as an Asst Professor in Eswar College of Engineering, Narasaraopet, Guntur, India. She is having about 11 years of teaching experience in different Engineering Colleges