International Journal of Application or Innovation in Engineering & Management (IJAIEM)
Web Site: www.ijaiem.org Email: editor@ijaiem.org, editorijaiem@gmail.com
Volume 2, Issue 1, January 2013 ISSN 2319 - 4847

A Survey of Graph Pattern Mining Algorithm and Techniques

Harsh J. Patel[1], Rakesh Prajapati[2], Prof. Mahesh Panchal[3], Dr. Monal J. Patel[4]

[1,2]M.E.C.S.E. ,KITRC, KALOL

[3]HOD, M.E.C.S.E., KITRC, KALOL

[4]Prof., Manish Institute of Computer Studies,Visnagar

ABSTRACT

Mining graph data is the extraction of novel and useful knowledge from a graph representation of data. The most natural form of knowledge that can be extracted from graphs is also a graph, we referred it as patterns. Many graph mining algorithms have been proposed in recent past researchers; all this algorithms rely on a very different approach so it's really hard to say that which one is the most efficient and optimal in the sense of performance. This paper investigates on comparison of graph mining algorithms and techniques for finding the frequent patterns.

Keywords: graph mining, frequent pattern, apriori based approach, pattern growth approach, NP-complete

1. INTRODUCTION

The graph representation has gained popularity in pattern recognition and machine learning. Frequent pattern mining (FPM) is an important part of graph mining that helps to discover *patterns* that conceptually represent relations among discrete entities. Developing algorithms that discover all frequently occurring sub graph in a large graph dataset is particularly challenging and computationally intensive, as graph and sub graph isomorphism play a key role throughout the computations. According to style of finding the frequent pattern from the given graph, it is mainly to types,

1.1 Apriori-Based Approach

It is uses a generate-and-test approach – generates candidate item sets and tests if they are frequent: One is Generation of candidate item sets is expensive (in both space and time) second Support counting is expensive i.e., Subset checking, Multiple Database scans (I/O).

1.2 Pattern-Growth Approach

It allows frequent item set discovery without candidate generation. Two steps: 1.Build a compact data structure called the FP-tree 2.extracts frequent item sets directly from the FP-tree.

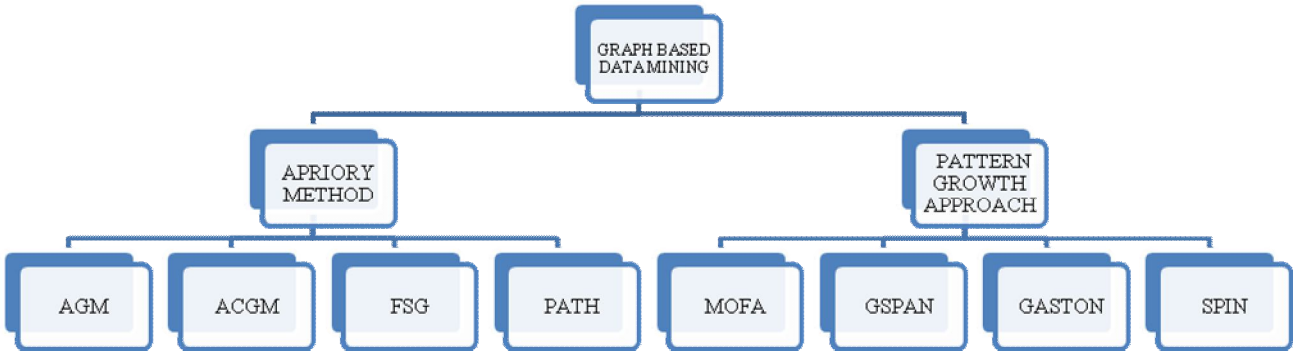

Figure 1 Graph Mining Approches [34]

2. ALGORITHM AND TECHNIQUES

Many of the already existing mining methods also apply in case of graphs, but management of that methods are far more challenging to implement because of the additional constraints which arise from the structural nature of the graph. In spite of these challenges, a number of techniques have been developed for traditional mining problems such

as frequent pattern mining, clustering, and classification. In this section, we will provide a survey of many of the structural algorithms for graph mining.

2.1 Mining Patterns from Graphs

The challenges of frequent pattern mining have been understood in the case of mining transactional data [1]-[2].After studied these challenges researcher decide that the methods of data mining can also be apply on graph mining. The main difference in the case of graph is that the process of determining the support is quite different. The problem can be defined in different ways depending upon application domain. In the first case, we have a bunch of graphs, and we wish to determine all patterns which support a fraction of the corresponding graphs [3]-[4]-[5]. In the second case, we have a single large graph, and we wish to determine all patterns which are supported at least a certain number of times in this large graph [6]-[7]-[4]. For the first case, where we have a data set containing multiple graphs, most of the well known techniques for frequent pattern mining with transactional data can be easily extended. For example, *Apriori*-style algorithms can be extended to the case of graph data, by using a similar level wise strategy of generating (k + 1)-candidates from k-patterns. The difference in the process of join operation. Two graphs of size k can be joined, if they have a structure of size (k − 1) in common. The *size of this structure* could be defined in terms of either nodes or edges. In the case of the AGM algorithm [3], this common structure is defined in terms of the number of common vertices. Thus, two graphs with k vertices are joined, only if they have a common sub graph with at least (k − 1) vertices. A second way of performing the mining is to join two graphs which have a sub graph containing at least (k − 1) edges in common. The FSG algorithm proposed in [4] can be used in order to perform edge based joins. It is also possible to define the joins in terms of arbitrary structures. For example, it is possible to express the graphs in terms of edge-disjoint paths. In such cases, sub graphs with (k +1)-edge disjoint paths can be generated from two graphs which have k edge disjoint paths, of which (k − 1) must be common. Another strategy which is often used is that of *pattern growth techniques*, in which frequent graph patterns are extended with the use of additional edges [8]-[9]-[10]. For the second case in which we have a single large graph, a number of different techniques may be used in order to define the support in presence of the overlaps. A common strategy is to use the size of the maximum independent set of the overlap graph to define the support. This is also referred to as the *maximum independent set support*. In [11], two algorithms HSIGRAM in which a breadth-first search approach is used in order to determine the frequent sub graphs and VSIGRAM are proposed in which a depth-first approach is used for determining the frequent sub graphs within a single large graph. As in the case of standard frequent pattern mining, a number of variations are also possible in the case of finding graph patterns, such as determining maximal patterns [10], closed patterns [12], or significant patterns [13]-[14]-[15]. Frequent pattern mining has been found to be particularly useful in the chemical and biological domain [6]-[16]-[17]-[18]. Frequent pattern mining techniques have been used to perform important functions in this domain such as classification or determination of metabolic pathways. Frequent graph pattern mining is also useful for the purpose of creating graph indexes. In [19], the frequent structures in a graph collection are mined, so that they can be used as features for an indexing process. The similarity of frequent pattern membership behavior across graphs is used to define a rough similarity function for the purpose of filtering. In general graph pattern mining techniques have the same range of applicability as they do for the case of vanilla frequent pattern mining.

Inokuchi, Washio and Motoda [20] in 1998 proposed a novel approach name AGM to efficiently mine the association rule among the frequently appearing substructure in a given graph dataset. A graph is represented by adjacency matrices and the frequent patterns appearing in the matrices are mined through the extended algorithm of the basket analysis. Agarwal and Srikant [21] in 1994 considered the problem of discovering association rules between items in a large database of sales transaction. They presented two new algorithms for solving this problem that are fundamentally different from the known algorithm**.**Blockeel and Raedt [22] in 1998 introduced a first order framework for top-down induction of logical decision tree. Top-down induction of decision trees is the best known and most successful machine learning technique. It has been used solve numerous practical problems. It employs a divide-and conquers strategy, and in this it differs from its rule-based competitors which are based on covering strategies. Kuramochi and Karypis [24] in 2001 presented a computationally efficient algorithm for finding all frequent sub graphs in large graph databases. Yan and Han [25] in 2002 investigated new approaches for frequent graph-based pattern mining in graph datasets and proposed a novel algorithm called gSpan. gSpan is a graph-based substructure pattern mining. This discovered frequent substructures without candidate generation. Huan, wang and Prince [26] in 2003 proposed a novel sub graph mining algorithm: FFSM, which employs a vertical search scheme within an algebraic graph framework. They have developed to reduce the number of redundant candidates proposed. Their studied on synthetic and real datasets demonstrates that FFSM achieves a substantial performance gain over the current start-of-the art sub graph mining algorithm gSpan. Yan and Han [27] in 2003 proposed to mine close frequent graph patterns. A graph G is closed in a database if there exists no proper sub graph of G that has the same support as G. A closed graph pattern mining algorithm, Close Graph, is developed by exploring several interesting looping methods. Their performance studied shown that Close Graph not only dramatically reduces unnecessary subgroups to be generated but also substantially increases the efficiency of mining, especially in the presence of large graph patterns. Huan, Wang, Prins and Yang [28] in 2004 developed a new algorithm that mines only maximal frequent sub graphs, that is sub graph that are not a part of any other frequent sub

graphs. Their algorithm can achieve a five-fold speed up over the current state-of-the-art sub graph mining algorithms. Meinl, Borgelt and Berthold [23] in 2004 shown that is possible to mine meaningful, discriminative molecular fragments from large databases. Using an existing algorithm that employs a depth-first strategy and a sophisticated ordering scheme allows avoiding costly embeddings throughout the candidate growth process, which in turn enables us to find also larger fragments. Krasky, Rohwer, Schroeder and Selzen [28] in 2006 discussed on a combined bioinformatics and chemo informatics approach for the development of new ant parasitic drugs. In ParMol package they have implemented four of the most popular frequent sub graph miners using a common infrastructure: MoFa, gspan, FFSM and Gaston. They also added additional functionality to some of the algorithms like parallel search, mining directed graphs and mining in one big graph instead of a graph database. Meinl, Worlein, Fischer, and Philippsen [28] in 2006 presented the thread-based parallel versions of MoFa and gSpan that achieve speedup up to 11 on a shared memory SMP system using 12 processors. Yang, Parthasarthy and Sadayappan in 2010 presented a novel approach to data representation for computing this kernel, particularly targeting sparse matrices representing power-law graphs. They shown their representation scheme, coupled with a novel tiling algorithm, can yield significant benefits over the current state of the art GPU and CPU efforts on a number of core data mining algorithms such as Page Rank, HITS and Random Walk with Restart. A graph transaction is represented by adjacency matrices and the frequent patterns appearing in matrices are mined through the extended algorithm. These are modeled as attribute graph in which each vertex represents an atom and each edge a bond between atoms. Each vertex carries attribute that indicates the atom type. Frequent structures are graphs that are isomorphic to a large number of sub graphs in the data graph. Frequent structures form building blocks for visual exploration and data mining of semi structured data. To overcome this Shayan Ghazizadeh and Sudarshan S. Chawa the [29] presented a novel approach of summary data structure to prune the search space and to provide interactive feedback. This is called SEus is the three-phase process: first phase (*summarization*), they preprocess the given dataset to produce a concise summary. This summary is an abstraction of the underlying graph data. This summary is similar to data guides and other (approximate) typing mechanisms for semi structured data [30]-[31]-[32]-[33]. In the second phase (*candidate generation*), method interacts with a human expert to iteratively search for frequent structures and refine the support threshold parameter. Since the search uses only the summary, which typically fits in main memory, it can be performed very rapidly (interactive response times) without any additional disk accesses. Although the results in this phase are approximate (a supper set of final results), they are accurate enough to permit uninteresting structures to be filtered out. When the expert has filtered potential structures using the approximate results of the search phase, an accurate count of the number of occurrences of each potential structure is produced by the third phase (*counting*).

3. CONCLUSION AND FUTURE WORK

The main challenge in the development of the algorithm for gaining high performance to enhance graph mining process is graph isomorphism which is the most costly step since it is an NP-complete problem. Hence, reducing the number of graph isomorphism is a promising direction which would save computational time. Due to increasing size and computational complexity of pattern in computer sciences the need for efficient graph mining algorithm is increasing. Still there is a scope of improvement in graph mining algorithm; the improvement can be in speed or sensitivity. The future graph mining algorithm can be works like query language for finding the desired pattern by querying graph data for high performance.

REFERENCES

[1] R. Agrawal, R. Srikant: Fast algorithms for mining association Rules in large databases, *VLDB Conference*, 1994.
[2] J. Han, J. Pei, Y. Yin: Mining Frequent Patterns without Candi Date Generation. *SIGMOD Conference*, 2000.
[3] A. Inokuchi, T. Washio, H. Motoda: An Apriori-based Algorithm For Mining Frequent Substructures from Graph Data. *PKDD Conference*, pages 13–23, 2000.
[4] M. Kur amochi, G. Karypis: Frequent sub graph discovery. *ICDM Conference,* pp. 313–320, Nov. 2001.
[5] N. Vanetik, E. Gudes, S. E. Shimony: Computing Frequent Graph Patterns from Semi-structured Data. *IEEE ICDM Conference*, 2002.
[6] B. Bringmann, S. Nijssen: What is frequent in a single graph? *PAKDD Conference*, 2008.
[7] M. Fiedler, C. Borgelt: Support computation for mining frequent sub graphs in a single graph. *Workshop on Mining and Learning with Graphs (MLG'07)*, 2007.
[8] C. Borgelt, M. R. Berthold. Mining molecular fragments:Finding Relevant Substructures of Molecules *ICDM Conference*, 2002.
[9] X. Yan, J. Han. Gspan: Graph-based Substructure Pattern Mining. *ICDM Conference*, 2002.
[10] J. Huan, W. Wang, J. Prins, J. Yang. Spin: Mining Maximal Frequent Subgraphs from Graph Databases. *KDD Conference*,2004.

[11] M. Kuramochi, G. Karypis. Finding frequent patterns in a large sparse graph. *Data Mining and Knowledge Discovery*, 11(3): pp. 243–271, 2005.
[12] X. Yan, J. Han. CloseGraph: Mining Closed Frequent Graph Patterns, *ACM KDD Conference*, 2003.
[13] H. He, A. K. Singh: Efficient Algorithms for Mining Significant Substructures from Graphs with Quality Guarantees. *ICDM Conference*, 2007.
[14] S. Ranu, A. K. Singh. GraphSig: A scalable approach to Mining significant sub graphs in large graph databases. *ICDE Conference*, 2009.
[15] X. Yan, J. Han. CloseGraph: Mining Closed Frequent GraphPatterns, *ACM KDD Conference*, 2003.
[16] M. Deshpande, M. Kuramochi, N. Wale, G. Karypis. Frequent Substructure-based Approaches for Classifying Chemical Compounds. *IEEE Transactions on Knowledge and Data Engineering*, 17: pp. 1036– 1050, 2005.
[17] J. Huan, W. Wang, D. Bandyopadhyay, J. Snoeyink, J. Prins,A. Trop-sha: Mining Spatial Motifs from Protein Structure Graphs. *Research in Computational Molecular Biology (RECOMB)*, pp. 308–315, 2004.
[18] M. Koyuturk, A. Grama, W. Szpankowski: An Efficient Algorithm for Detecting Frequent Subgraphs in Biological Networks. *Bioinformatics*, 20:I200–207, 2004.
[19] X. Yan, P. S. Yu, J. Han. Graph indexing: A frequent structure based approach. *SIGMOD Conference*, 2004.
[20] A. Inokuchi, T. Washio, H. Motoda, "An Apriori-based Algorithm for Mining Frequent substructures from Graph Data. In proc. 2000 European Symp. Principle of Data mining and knowledge Discovery (PKDD'00), 1998, pp. 13-23.
[21] R. Agrawal, R. Srikant, "Fast Algorithms for mining association rules. In the proc. Of the 20th Int. conf. on very large databases (VLDB), 1994.
[22] H. Blockeel, L.D. Raedt, "Top-down induction of first-order logic decision trees", Artificial Intelligence, 101, 1998, pp. 285-297.
[23] S. Chakrabarti, B. Dom, P. Indyk, "Enhanced hypertext categorization using hyperlinks" ACM, (SIGMOD'98), 1998, pp.307-318.
[24] M. Kuramochi, G. Karypis, "Frequent Sub graph Discovery" In Proc 2001 Int. conf. Data mining (ICDM'01).
[25] X. Yan, and J. Han, "gSpan: Graph-Based Substructure Pattern Mining." In Proc. 2002 Int. conf. Data mining, 2002, pp.721-724.
[26] J. Huan, W. Wang and J. Prins, "Efficient Mining of frequent Subgraph in the Presence of Isomorphism." In Proc. 2003 int. conf. Data mining (ICDM'03), 2003, pp. 549-552.
[27] X. Yan and J. Han CloseGraphs: Mining Closed Frequent Graph Patterns. In proc. 2003 ACM SIGKDD Int. conf. knowledge Discovery and Data Mining(KDD'03), 2003, pp.286-295.
[28] J. Huan, W. Wang, J. Prins and J. Yang,"Spin: mining maximal frequent subgraph from graph Databases", KDD04 Seattle, Washington, USA, 2004.
[29] Shayan Ghazizadeh and Sudarshan S. Chawathe. Seus: *Structure extraction using summaries*. In Discovery Science, pages 71–85, 2002.
[30] P. Buneman, S. B. Davidson, M. F. Fernandez, and D. Suciu. Adding structure to unstructured data. In *Proc. of the 6th International Conference on Database Theory*, 1997.
[31] R. Goldman and J. Widom. Dataguides: Enabling query formulation and optimization in semistructured databases.In *Proc. of the Twenty-Third International Conference on Very Large Data Bases*, pages 436–445, 1997.
[32] S. Nestorov, S. Abiteboul, and R. Motwani. Inferring structure in semistructured data. In *Proc. of the Workshop on Management of Semistructured Data*, 1997.
[33] S. Nestorov, J. Ullman, J. Wiener, and S. Chawathe. Representative objects: Concise representations of semistructured, hierarchial data. In *roc of the International Conference on Data Engineering*, pages 79–90, 1997.
[34] D. Kavitha, B.V.Manikyala Rao, V. Kishore Babu. A Survey on Assorted Approaches to Graph Data Mining. International journal of computer application, volume-14,January 2011.

AUTHOR

Mr. Harsh Patel received the B.E. degree in Computer Engineering from S.K.Patel College of Engineering in 2011.His Master degree in Computer Science and Engineering is pursuing from KITRC, Kalol, India. His research area is in the field of Graph mining.

Mr. Rakesh Prajapati received the B.E. degree in Computer Engineering from Government Engineering College, Chankheda in 2011.His Master degree in Computer Science and Engineering is pursuing from KITRC, Kalol, India. His research area is in the field of Data mining.

Prof. Mahesh Panchal received his B.E. and M.E. degree in Computer Engineering. And his PhD pursuing form G.T.U. He has 9 years of teaching experience and his area of interest is in Data mining.currentlly he is working at KITRC, kalol as a head of department in computer department.

Dr. Monal Patel is working as an assistant professor at Manish Institute of Computer Studies, Visnagar. She has 7 years of teaching experience in the field of computer science and application. She also completed her Ph.D. from Hemchandracharya North Gujarat University, Patan.Her current research focus includes on-line learning and education in virtual and immersive environment and also in data mining. She has published 10 research papers in several national and international journals.