ISSN: 0975-766X
CODEN: IJPTFI
Available Online through
Research Article
www.ijptonline.com

A BRIEF REVIEW ON APPLICATION OF GRAPH THEORY IN DATA MINING

Abhinav Chanana*, Tanya Rastogi, M.Yamuna
VIT University, Vellore, Tamil Nadu, India.
Email: abhinav.chanana@gmail.com

Received on 22-05-2016 *Accepted on 25-06-2016*

Abstract

Graph theory is becoming progressively important as it is applied to other fields of mathematics, science and technology. It is being actively used in areas as varied as biochemistry, electrical engineering, computer science and operations research. The main application of graph theory in data mining is graph mining. The need for mining structured data has increased in the past few years. Graphs are one of the best studied data structures in computer science and discrete mathematics. The relational aspect of data is explained by graph mining. The main aim of graph mining is to provide new principles and effective algorithms to mine topological substructures embedded in graph data. This article provides a brief review on four theoretical based approaches of graph based data mining. Brief description of application of graph mining is also provided.

Keywords: Graph, Data mining, Graph mining, Approaches, Applications

Introduction

In Mathematics and computer science, the study of graphs is graph theory where graphs are data structures used to model entities (called vertices) pairwise. Data mining is the process of extracting useful information and interesting patterns from large amounts of data sets by applying methods from artificial intelligence, database systems, statistics and machine learning. During the past few years, the field of data mining has arose as a unusual field of research, exploring interesting research matters and developing challenging real-life applications. In data science most of the data we deal with can modeled into graphs. These graphs can be mined using algorithms and approaches in graph theory. Application of graph theory in data mining is called graph mining. Graph mining has become a very significant topic of research recently because of the development of many applications to data mining problems in various fields. Graph mining is the process

of extracting (mining) frequent (sub) graph patterns. A (sub) graph is frequent if its occurrence frequency which is called support in a given set of data is more than a minimum support threshold. Graph mining has become a very significant topic of research recently because of the development of many applications to data mining problems in various fields. The various approaches of graph mining is explained in the following section. Applications of graph mining is also discussed. Many of the search algorithms in graph mining are from artificial intelligence but some from mathematics are also used. Almost every underlying data can be modeled into graphs. So graph theory is used to mine useful information from such data sets

Preliminary Note:

In this section we provide the basic details required for this survey.

A **point** is an exact position or location on a surface. A point is also referred as a node [20].

A **graph** is an ordered pair G = (V, E) comprising a set V of vertices or nodes or points together with a set E of edges or arcs or lines, which are 2-element subsets of V (i.e. an edge is related with two vertices, and the relation is represented as an unordered pair of the vertices with respect to the particular edge). To avoid ambiguity, this type of graph may be described precisely as undirected and simple. [21].

A subgraph of a graph G is another graph formed from a subset of the vertices and edges of G. The vertex subset must include all endpoints of the edge subset, but may also include additional vertices. A spanning subgraph is one that includes all vertices of the graph; an induced subgraph is one that includes all the edges whose endpoints belong to the vertex subset [22].

Graph mining is something like structure mining or structured data mining is the process of finding and extracting useful information from semi structured data sets [16].

An **Adjacency matrix** is a square matrix used to represent a finite graph. The elements of the matrix indicate whether pairs of vertices are adjacent or not in the graph[17].

Graph theory has emerged and grown as one of the best areas of mathematics that can support to study multiple domains. In this brief review we concentrate on the contribution of graph theory to data mining. For this purpose we have split the article into five parts [2]

1. Greedy Based Approach

2. Inductive Database Based Approach.
3. ILP Based Approach.
4. Mathematical Approach.
5. Kernel Function Based Approach.

Since the review is restricted to graph mining, various results related to data mining are omitted. Also since it is a very brief review some interesting results on graph mining are also omitted. We apologize to the authors for omitting many results due to the crisp note.

Approaches of Graph Mining

Greedy Search Based Approach

As Graph Mining is in the initial stages of development the first break through that came in this field was Greedy based algorithms in the early 1994. Two techniques came into being the SUBSUE and GST both used framework of Greedy Search Based. Both the approaches were graph representations of some structures that were used in everyday life.eg. Graph having resemblance to semantic networks or a physical network like electric circuits.

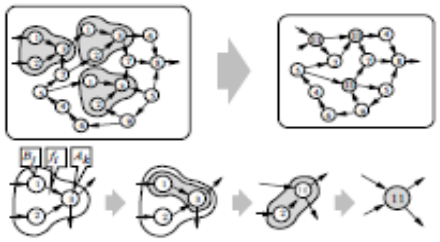
In [1] J. Cook et al describe the method 'SUBDUE' that deals with conceptual graphs belonging to class of connected graphs. This method is premier and played an important role in the field of greedy search-based graph mining. In [2] T. Washio et al tells that the vertex set V (G) is R∪C where R and C are the sets of labeled vertices representing relations and concepts respectively. The edge set E(G) is U which is a set of labeled edges.

The input for SUBDUE is a structured data represented as labeled graph. This algorithm provides sub-structures, again as labelled graphs as output. The sub-structure is a sub-graph to the input graph. The derived sub graph is considered a **concept. SUBDUE** basically looks for sub graph which can best compress the input Graph G based on the Minimum Description Length (MDL) principle. This algorithm that SUBDUE uses is based on computationally-constrained beam search. It uses the greedy approach as it starts with one vertex in the input graph G and grows by adding nodes to it. At each step of addition of nodes the total description(DL) $I(G_s)+I(G|G_s)$, which is defined as the sum of the DL of the sub graph that is $I(G_s)$ and the DL of the input graph $I(G|G_s)$ where all instances of sub graphs are replaced by nodes [2]. This algorithm stops on achieving the minimum total description. Along with MDL, SUBDUE has evaluation methods like **SIZE** and **SETCOVER** also. The **size** measure is faster to process than the MDL measure yet is less steady.

In [3] M. Mukherjee et al gives value(S, G) = size(G)/(size(S) + size(G|S)), where size(G)=(#vertices(G)+#edge(G)), and (G|S) is G compressed with substructure [3]. The set-cover is still under research and slowly evolving.

SUBDUE is completely Greedy in nature once it has added a node it never backtracks. This algorithm takes decision what is best at the given instance. The advantages of using this approach is that it performs approximate matching to allow slight variations of sub graphs. It can also use background knowledge in terms of predetermined sub graphs. After the best structure is found it is extracted and over written on the input graph.

In [4] K. Yoshida describe the method GBI (Graph Based Induction) in which frequent appearing patterns are identified and analyzed. Similar to SUBDUE, GBI also finds a subgraph which has the minimum size by replacement of each subgraph with one vertex which would compress the graph on every iteration. This algorithm would prevent the graph to continue compressing that is it would not be reduced to a single vertex. This algorithm is known to handle both types of graph: Directed and Undirected Graphs. Genetic Algorithm like opportunistic beam search is used to derive a local minimum solution. This algorithm worked pairwise at each step in the search was to find a strongly linked paired of vertices by an edge to chunk. Snapshot – 1 explains chunking of vertices in directed graphs [2].



Snapshot - 1 Graph compression by pairwise chunking.

This algorithm breaks the vertices i.e chunks the triplets (A; F; B) which minimizes the graph size. A and B are the vertices and F is link between A and B. This process of breaking or removing the vertices is continued till the time local minimum is achieved. Chunking of vertices can be nested and one or both vertices of paired nodes can be chunked out. GBI remember the relation between the vertices that is the links and the original graph can be constructed at time of search. Later in [5] W. Geamsakul et al give improvements focused on using other measures than frequency.

Inductive Database Based Approach

In [7] L. De Raedt explains a work in the structure of inductive database having viable computational proficiency is MolFea framework in light of the level-wise form space calculation. This technique performs the complete inquiry of the

ways inserted in a diagram information set where the ways fulfill monotonic and hostile to monotonic measures in the adaptation space. In [8] A. Srinivasan et al tells that MolFea system has been applied to the Predictive Toxicology Evaluation challenge data set. The variant space is a pursuit subspace in a cross section structure. This uses the minimum and maximum support like in association algorithms. If a substructure has more than minimum support as sub-structures then that substructure would have influence on the whole structure or graph properties as a whole. It is used in the medical branch.

ILP Based Approach

Intersection of Machine Learning and Logic Programming is known as Inductive Logic Programming (ILP). The use of logic for the illustration of multi-relational data characterises ILP. The use of logic for the depiction of multi-relational data and the exploration for syntactically legal hypotheses constructed from predicates provided by the related knowledge are the two main principles of ILP.

In [6] Nikhil S. Ketkar defines ILP process as mainly an exploration in which the states are hypotheses and the aim is the hypotheses that is recurrent or that differentiates positive and negative examples. The way of structuring hypothesis space and the search strategy used to explore it can characterize an ILP system. Classification of an ILP system can be done on the basis of four keys. ILP may study a single concept or multiple concepts. Depending on whether ILP systems use human assistance in the process of learning they may be interactive or non-interactive. Depending on how ILP systems receive examples they may be batch or incremental. ILP systems may study concepts from start or review a theory which is called theory revision systems. ILP systems may be empirical ILP systems or incremental ILP systems. Empirical ILP systems are non-interactive, single concept batch learners that construct concepts from start. Incremental IPL systems are incremental, interactive theory revisers who study multiple concepts. FOIL, CProgol, Golem, SMART, G-Net, CHILLIN, TILDE and WARMR are some ILP systems.

Mathematical Approach

In Mathematical graph theory based approach we mine a complete set of subgraphs under primarily support measure. The primary approach is Apriori-based Graph mining (AGM) system. In [2] T. Washio et al tells that the basic principle of AGM and Apriori algorithm are analogous to each other. Under this approach a lattice of graph nodes is built. A node at the k^{th} level of the lattice has k vertices and number of supporting cases surpasses a user-specified minimum support. In

[15] Akihiro Inokuchia at al explain that the method can devise a rule "IF subgraph G_a is in transaction G, the union of subgraphs $G_a \cdot G_b$ is also contained in G with a ceratin confidence level".

When a transaction comprising of a chemical graph and virtual vertices stating molecular properties is given, we can acquire rules representing structure activity relationships. This method was used to study mutagenicity data for 230 aromatic nitro compounds. The frequent graphs are searched in a bottom up manner by creating candidates having an extra vertex where each graph is a single vertex. While searching for connected graphs an edge should be added between the extra vertex and the vertices of smaller frequent graphs. One graphs establishes one transaction.

The transformation of graph structured data into an adjacency matrix with less computational effort. Let the size of the graph be the number of vertices in it, an adjacency matrix of a graph whose size is s be X_s, the ij-element of X_s, x_{ij} and its graph, $G(X_s)$. AGM can manage graphs comprising of labelled vertices and labelled edges.

The Apriori-based Graph Mining system can mine general subgraph, induced subgraph, connected subgraph, ordered subtree, unordered subtree, subpath and other various types of subgraphs. This approach constructs 'association rules whose support and confidence exceed thresholds specified by the user.

In [9] M. Kuramochi tell that a work Frequent SubGraph discovery system also takes alike definition of canonical labelling of graphs and is based on the adjacency matrix. This approach increases the efficiency of deriving the canonical labels by using graph vertex invariants such as the degree of each vertex in the graph.

The transaction ID (TID) method is also introduced to increase the efficiency of the candidate generation of recurrent graphs. FSG introduces an efficient search algorithm using "core" which is a shared part of the size m-1 in the two frequent subgraphs of the size k under the limitation of the frequent subgraphs to connected graphs. By limiting the common part of the two recurrent graphs to the core this approach increases the joining efficiency.

After we obtain the candidate set, their frequency counting is showed by checking the cardinality of the juncture of both TID lists. The advantage of FSG process is that it runs fast but it also consumes much memory space.

In [10] X. Yan describe Graph-based substructure pattern mining as a DFS based canonical labelling approach. Instead of using adjacency matrix it uses tree representation of each graph to define the code the graph. Among the codes, the quasi-tree expression has the smallest code in terms of the lexicographical order and is the canonical form, and the corresponding code is the canonical label.

This code is called DFS code as the code is derived in the DFS algorithm. Then all codes are sorted according to ascending lexicographical order, and the matching of the code starting from the first elements among the codes are conducted by using DFS in the sorted order. gSpan is an efficient approach in both computational time and memory consumption. Snapshot – 2 describes the gSpan algorithm [11].

```
Algorithm:

Input:   s, a DFS code;
D, a graph data set;
min_sup,the minimum support threshold;
Output: The frequent graph set, S.
Method:S← ϕ ;
Call gspan(s,D,min_sup,S);
Procedure: PatternGrowth Graph(s,D,min_sup,S)
1 ) if s≠ dfs(s) then
2 )        return;
3)  insert s into S;
4 )  set C to ϕ;
5 )  scan D once ,find all the edges e such that s can be
right-most extended to s◊ re;
6 )   insert  s ◊ re into C and count its frequency;
7 )  sort C in DFS lexicacographic order;
8 )   for each frequence to s ◊ re  in C do
9 )   sSpan( s ◊ re,D,min_sup,S);
10) return;
```

Snapshot – 2: Describing the gSpan algorithm [11].

Kernel Function Based Approach

A Kernel function K defines a similarity between two graphs G_x and G_y. In [2] T. Washio explains that for the application to graph-based data mining, the key issue is to find the good combinations of the feature vector XG and the mapping ϕ: $X_G \rightarrow$ H to define appropriate similarity under abstracted inner product $<\phi(X_{Gx});\phi(X_{Gy})>$. In [12] H. Kashima et al proposed a composition of a kernel function characterizing the similarity between two graphs G_x and G_y based on the feature vectors consisting of graph invariants of vertex labels and edge labels in the certain neighbor area of each vertex. In [14] V. Vapnik tells that using this the classification of graphs can be done by Support Vector Machine (SVM). The given training data consists of graphs which have binary classes and SVM is trained for the classification of each group. Similarity defined by the Kernel function is used to classify the graphs although the similarity is not complete in terms of graph isomorphism.

In [13] R. Kondor describe another framework of kernel function related with graph structures which is called “diffusion kernel". It has a theoretical but not a dedicated relation with graph based data mining. In this each vertex of the graph

structure contains an instance. The diffusion process along the edges of the graph evaluates the similarity among instances. Some experiments report that the similarity evaluation in the structure characterizing the relations among the instances provides better performance in classification and clustering tasks than the distance based similarity evaluation [2].

Applications of Graph Mining

Graph in recent days has taken over in providing easy realization of normal day to day problems. From 3D Printing, Function Mapping to Virtual Reality Headset. As the use of graph increased so did the need for extracting and analyzing its substructures which is known as Graph Mining. Algorithms mentioned in the research paper can be to extract sub-graph.

A. Chemical and Biological Applications

Discovery of new drugs is a time consuming and expensive job. The molecules can be visualized as graph where atoms being the nodes and the bonds being the edges between nodes. The properties each drug has can be predicted by analyzing the substructure's available e.g. analyzing properties of aspirin on basis of the substructures. Prediction of properties of drug by graph mining.

B. Search Engines

The ranking algorithms use graph mappings of websites to understand the relevance and popularity.

C. Social Networking

The social network is growing at phenomenal speed and in social network sites every person is considered as a node while friendship between them can be seen as the edges between the nodes. The class of friends like 'best friend', 'normal friend' etc. is done using graph mining. Training the system to identify such structures and comparing it with existing ones. This helps in customer satisfaction and improved connectivity.

D. Detection of financial crimes

The activities that are illegal are represented as a graph, and that graph is searched in a large set of financial transactions.

E. Consumer Behavior Analysis

Representation of purchases of a consumer as a graph and analyzing the purchases to achieve customer satisfaction.

Finally we would like to note that there are many other researches in graph mining. An approach to derive induced subgraphs of graph data was proposed by Geibel and Wysotzki [18]. Their approach can be used to search recurrent induced subgraphs in the set of graph data. Liquiere and Sallantin proposed an approach [19] to fully search homomorphically equivalent subgraphs which are very less general over a given set of graphs and do not consist of any identical triplet of the labels of two vertices and the edge direction between the vertices within each subgraph. They explain that the computational complexity to search for the above mentioned class of subgraph is polynomial. As various graphs in real-world problems like chemical compound analysis are comparatively more general so the polynomial characteristics of this method is not applicable in real-time cases.

Conclusion

The journey of writing this review article has been an interesting experience, as it revealed the fact how graph theory can be linked to other fields. The amazing results that a tiny graph can contribute will be of interest for anyone interested in implementing graph theory. In this review various graph based data mining approaches were explained in the former half of the article. The latter half of the review describes the various applications of graph mining. This review will be of interest to anyone interested in applying graph theory to data mining.

References

1. J. Cook and L. Holder. Substructure discovery using minimum description length and background knowledge. J. Artificial Intel. Research, 1:231-255, 1994.
2. H. Motoda, and T. Washio State of the Art of Graph-based Data Mining. ACM SIGKDD Explorations Newsletter Homepage archive Volume 5 Issue 1, July 2003 Pages 59-68.
3. M. Mukherjee and Lawrence B. Holder. Graph Based Data Mining on Social Networks. In Proceedings of the ACM KDD Workshop on Link Analysis and Group Detection, 2004.
4. K. Yoshida, H. Motoda, and N. Indurkhya. Graph-based induction as a unified learning framework. J. of Applied Intel. 4:297-328, 1994.
5. W. Geamsakul, T. Matsuda, T. Yoshida, H. Motoda, and T. Washio. Classifier construction by graph-based induction for graph-structured data. In PAKDD'03:Proc. of 7th Pacific-Asia Conference on Knowledge Discovery and Data Mining, LNAI2637, pages 52-62, 2003.

6. Nikhil S. Ketkar, Lawrence B. Holder and Diane J. Cook. Comparison of Graph-based and Logic-based Multi-relational Data Mining in SIGKDD Explorations Volume 7, Issue 2.
7. L. De Raedt and S. Kramer. The level wise version space algorithm and its application to molecular fragment finding. In IJCAI'01: Seventeenth International Joint Conference on Artificial Intelligence, volume 2, pages 853-859, 2001.
8. A. Srinivasan, R. King, and D. Bristol. An assessment of submissions made to the predictive toxicology evaluation challenge. In IJCAI'99: Proc. of 16th International Joint Conference on Artificial Intelligence, pages 270-275, 1999.
9. M. Kuramochi and G. Karypis. Frequent subgraph discovery. In ICDM'01: 1st IEEE Conf. Data Mining, pages 313-320, 2001.
10. X. Yan and J. Han. gspan: Graph-based substructure pattern mining. In ICDM'02: 2nd IEEE Conf. Data Mining, pages 721-724, 2002.
11. Sadhana Priyadarshini and Debahuti Mishra. An Approach to Graph Mining Using Gspan Algorithm. Int'l Conf. on Computer & Communication Technology(ICCCT'10).
12. H. Kashima and A. Inokuchi. Kernels for graph classification. In AM2002: Proc. of Int. Workshop on Active Mining, pages 31-35, 2002.
13. R. Kondor and J. Lafferty. Diffusion kernels on graphs and other discrete input space. In ICML'02: Nineteenth International Joint Conference on Machine Learning, pages 315-322, 2002.
14. Akihiro Inokuchia, Takashi Washioa, Takashi Okadab and Hiroshi Motoda. Applying the Apriori-based Graph Mining Method to Mutagenesis Data Analysis. Journal of Computer Aided Chemistry, Vol.2, 87-92(2001).
15. P. Geibel and F. Wysotski. Learning relational concepts with decision trees. In ICML'96: 13th Int. Conf. Machine Learning, pages 166-174, 1996.
16. M. Liquiere and J. Sallantin. Structural machine learning with galois lattice and graphs. In ICML'98: 15thInt. Conf. Machine Learning, pages 305-313, 1998.
17. http://tex.stackexchange.com/questions/295673/what-is-the-difference-between-points-coordinates-and-anchors-in-tikz-pgf-and

Corresponding Author

Abhinav Chanana*,

Email: *abhinav.chanana@gmail.com*