www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 3 Issue 5, May 2014, Page No. 5833-5838

Demographical Implementation of Graph Classification in Educational Network Using Graph Mining Technique

S.A Amala Nirmal Doss[1] , *Dr.S.P.Victor*[2]
[1] **Research Scholar, MS University, Tirunelveli.**
amalnirmaldoss@rediffmail.com.

[2]**HOD/MCA-**
St. Xavier's College Tirunelveli,
drvictorsp@rediffmail.com.

ABSTRACT

Graphs become increasingly important in modeling complicated structures, such as circuits, images, chemical compounds, protein structures, biological networks, social networks, the Web, workflows, and XML documents. The Graph mining offers a convenient way to study structured datum with different level of implications. Our conventional setup initially focuses with dataset and its entity. This paper perform a detailed study of classified datum of graph classification towards variant clusters in the field of graph mining which can be carried out with identification and analysis strategies. We will implement our Integrated graph mining techniques with real time implementation of Educational network Domains. We will also perform survey analysis strategies for the successful implementation of our proposed research technique in several sampling domains with a maximum level of improvements. In near future we will implement the cluster mining techniques for analyzing the Graph sub structure behaviors.

Keywords: Graph mining, Education, cluster, substructure, classification

I. Introduction

The graph classification is based on a graph's important substructures. This work can create a binary feature vector based on the presence or absence of a certain substructure (subgraph) and apply an off-the-shelf classifier.

Since the entire set of subgraphs is often very large, this work must focus on a small subset of features that are relevant. The most straightforward approach for finding interesting features is through frequent pattern mining. However, frequent patterns are not necessarily relevant patterns. For instance, in chemical graphs, ubiquitous patterns such as C-C or C-C-C are frequent, but have almost no significance in predicting important characteristics of chemical com- pounds such as activity, toxicity, etc. Boosting is used to automatically select a relevant set of subgraphs as features for classification. LPBoost (Linear Program Boost) learns a linear discriminate function for feature selection. To obtain an interpretable rule, this work need to obtain a sparse weight vector, where only a few weights are nonzero. Graph boosting can achieve better accuracy than graph kernels, and it has the advantage of discovering key substructures explicitly at the same time.

The problem of graph classification is closely related to that of XML classification. This is because XML data can be considered an instance of *rich graphs*, in which nodes and edges have features associated with them. Consequently, many of the methods for XML classification can also be used for structural graph classification. In a rule-based classifier (called *XRules*) was proposed in which this work associate structural features on the left-hand side with class labels on the right-hand side. The structural features on the left- hand side are determined by computing the structural features in the graph which are both *frequent* and *discriminative* for classification purposes. These structural features are used in order to construct a prioritized list of rules which are used for classification purposes. The top-k rules are determined based on the discriminative behavior and the majority class label on the right hand side of these k rules is reported as the final result.

Classification Algorithms for Graph Data Classification is a central task in data mining and machine learning. As graphs are used to represent entities and their relationships in an increasing variety of applications, the topic of graph classification has attracted much attention in both academia and industry. For example, in pharmaceutics and drug design, we are interested to know the relationship between the activity of a chemical compound and the structure of the compound, which is represented by a graph. In social network analysis, we study the relationship between the health of a community (e.g., whether it is expanding or shrinking) and its structure, which again is represented by graphs. Graph classification is concerned with two different but related learning tasks.

Label Propagation. A subset of nodes in a graph is labeled. The task is to learn a model from the labeled nodes and use the model to classify the unlabeled nodes.

Graph classification. A subset of graphs in a graph dataset is labeled. The task is to learn a model from the labeled graphs and use the model to classify the unlabeled graphs.

The concept of label or belief propagation is a fundamental technique which is used in order to leverage graph structure in the context of classification in a number of relational domains.

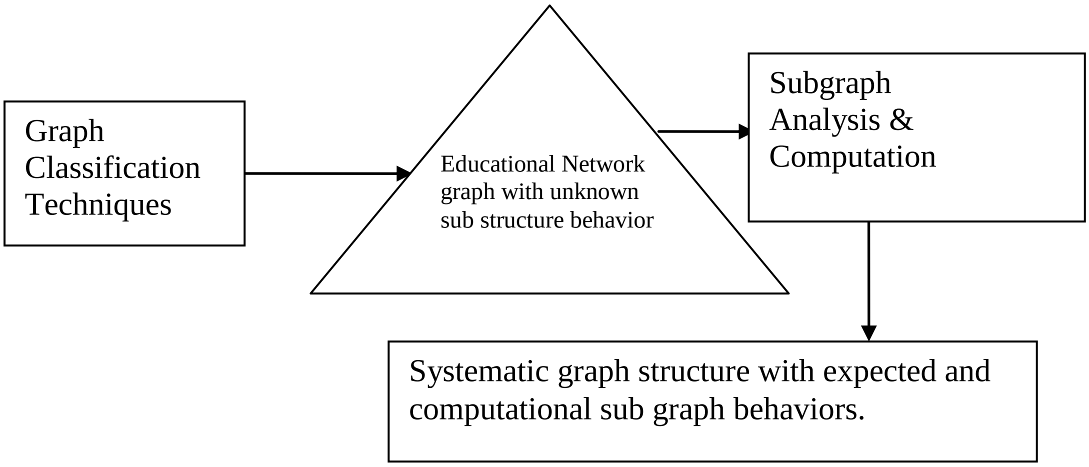
The scenario of label propagation occurs in many applications. As an example, social network analysis is being used as a mean for targeted marketing. Retailers track customers who have received promotions from them. Those customers who respond to the promotion (by making a purchase) are labeled as positive nodes in the graph representing the social network, and those who do not respond are labeled as negative. The goal of target marketing is to send promotions to customers who are most likely to respond to promotions. It boils down to learning a model from customers who have received promotions and predicting the responses of other potential customers in the

Social network. Intuitively, we want to find out how existing positive and negative labels propagate in the graph to unlabeled nodes. Based on the assumption that "similar" nodes should have similar labels, the core challenge for label propagation lies in devising a distance function that measures the similarity between two nodes in the graph.

One common approach of defining the distance between two nodes is to count the average number of steps it takes to reach one node from the other using a random walk. However, it has a significant drawback: it takes O(n3) time to derive the distances and O(n2) space to store the distances between all pairs. However, many graphs in real life applications are sparse, which reduces the complexity of computing the distance. For example, Zhou et al introduces a method whose complexity is nearly linear to the number of non-zero entries of the sparse coefficient matrix.

II.PROPOSED METHODOLOGY

This proposed methodology focuses on the implementation of a Graph classification algorithmic strategy to identify and analyze the unknown sub graph behaviors by implementing the cluster computations.



Graph Classification Techniques

Educational Network graph with unknown sub structure behavior

Subgraph Analysis & Computation

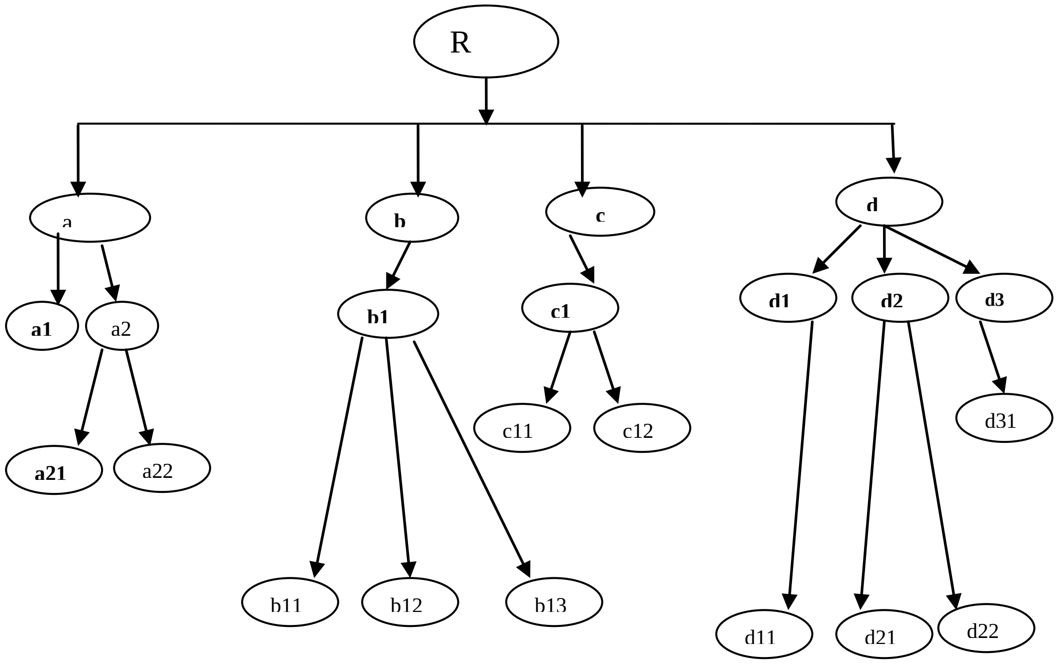
Systematic graph structure with expected and computational sub graph behaviors.

Figure 1.1: Proposed Graph mining structure

Implementation of Algorithmic strategies.

Consider the Class room network in a ABC Engineering college with known node behaviors as follows, The network contains 23 nodes with Root node R act as the faculty and there exists 4 groups of students labeled as a,b,c,d where "a" provides academic performance "All Pass" group of Gifted children, the group "b" acted as Good learners group, the group "c" represents the "Average learners "group and d represent the "Slow learners" group. The level represents the node performance and attachment as friendship as a group study pattern. Each node works well and earns their course based on their performance in internal examination marks. The group "a" trained with "ADVANCED PROBLEM SOLVING" material,"b" with "ACADEMIC PROBLEM SOLVING" material,"c" with "COACHING AND DRILLING" material and finally "d" with "LINGUISTIC,COUNSELL AND EXAMINATION POINT OF VIEW" material. But one or two nodes with exceptions. In the second year the lateral entry students 6 to 12 students based on total intake, arrived to append the class strength. The problem of splitting them into 4 groups can be resolved by the proposed graph classification of sub structures strategies.

Figure 1.2: Social Corporate network graph



The following table illustrates the analysis of our graph in figure 1.2 where the Type represents the performance group level .the average marks are represented by the assessment of 3 different question papers of 100 marks towards each node. The Time taken can be computed as Response time in Minutes. The level represents the node level in the graph. Sibling represents the node status adjacency with a sibling(maximum 3 only allowed),child represents number of nodes to be controlled(Information sharing purpose).The Allocation status for the default nodes are computed through their performance responses but for the lateral entry nodes are unknown which will be computed by the proposed algorithmic strategy.

Table 1.1: Promotional Credit Table with unknown values

Node number	Type	Average Marks	Time Taken(180 minutes)	Sibling	Child	Allocation Status Level
a	Gifted	99.9	150	Yes	Yes	0
a1	Gifted	97.8	147	Yes	No	1
a2	Gifted	98.9	165	Yes	Yes	1
a21	Gifted	95.8	130	Yes	No	2
a22	Gifted	96.9	125	Yes	No	2
b	Good	89.9	175	Yes	Yes	0
b1	Good	85.9	176	No	Yes	1
b11	Good	80.1	172	Yes	No	2
b12	Good	79.5	173	Yes	No	2
b13	Good	75.4	174	Yes	No	2
c	Average	69.9	180	Yes	Yes	0
c1	Average	64.5	179	No	Yes	1
c11	Average	63.2	180	Yes	No	2
c12	Average	61.4	180	Yes	No	2
d	Slow	59.9	164	Yes	Yes	0
d1	Slow	57.8	160	Yes	Yes	1
d2	Slow	55.7	159	Yes	Yes	1
d3	Slow	52.6	158	Yes	Yes	1
d11	Slow	52.3	100	No	No	2
d21	Slow	51.8	98	Yes	No	2
d22	Slow	51.3	97	Yes	No	2
d31	Slow	50.6	95	No	No	2
X1	Unknown	---	---	---	---	---
X2	Unknown	---	---	---	---	---
X3	Unknown	---	---	---	---	---
X4	Unknown	---	---	---	---	---
X5	Unknown	---	---	---	---	---
X6	Unknown	---	---	---	---	---

Find the values for X1, X2, X3, X4, X5, and X6 based on the following algorithmic strategies,

```
Start
Type (X (i)) =Nil, Sib(X (i)) =Nil, Child(X (i)) =Nil, Level(X (i)) =Nil
1.For each node i, Conduct 3 Tests with mixture level of pre executed tests and compute the average marks Mark(X(i)) and average Time taken as Time(X(i)).   ,
  If Average Marks (X (i) >=90.0 then Type="Gifted"
  Else if Average Marks(X (i) >=75.0) and Average Marks (X (i) <90.0 then Type="Good"
  Else if Average Marks(X (i) >=60.0) and Average Marks (X (i) <75.0 then Type="Average"
  Else Type="Slow"

2. For each node I, traverse to Group of sub roots a, b, c, d according to their performance type, If equal use the average Time(X(i)) as minimal
  If Average (Mark(X (i))) > sub root then add X (i) as new sub root (new group under faculty)
          No change
  Else
```

IfAverageMarks (Left (sub root)) < X (i) and Child (Sub root) <3 then add X (i) as a Left child

IfAverageMarks (Left (sub root)) > X (i) and Average Marks (Right (sub root)) < X (i) and Child (Sub root) <3 then add X (i) as a middle child

IfAverageMarks(Left(sub root)) > X(i) and Average Marks(Right(sub root)) > X(i) and Child(Sub root) <3 then add X(i) as a right child.

3. If Left child (sub root)>3 Continue step 2 on right Child (sub root) else add it as a new child to the root (Faculty new group)

4. Goto Step1 and continue the process still all the nodes are allocated .The allocation strategy checks for all the levels 1, 2, 3 in each correspondence subgraphs.

Update Allocation level X (i)

Stop

III. COMPUTATION

The evaluation table for the lateral entry students is as follows,

Table 1.2 Computation table

Node	Average Marks	Response time
X1	86.2	175
X2	75.6	180
X3	98.6	160
X4	62.1	180
X5	98.5	175
X6	50.1	104

For the node X1

According to step 1
X1 belongs to Type "b"
According to step 2 x1 act as b2 since better than sub root b1
X1 lies as a new child to b

For the node X2

According to step 1
X2 belongs to Type "b"
According to step 2 b1 is full continue to step 3
According to step 3 X2 act as a new child to X1 i.e. b2
X1 lies as a new child to b2 named as b21

For the node X3

According to step 1
X3 belongs to Type "a"
According to step 2 X3 acts as a new child to a1
X1 lies as a new child to a1 named as a11

For the node X4

According to step 1
X4 belongs to Type "c"
According to step 2 X4 acts as a new child to c1
X4 lies as a new child to c1 named as c13

For the node X5

According to step 1
X5 belongs to Type "a"
According to step 2 X5 acts as a new child to a1
X lies as a new child to a1 named as a12

For the node X6

According to step 1
X6 belongs to Type "d"
According to step 2 X6 acts as a new child to d1
X6 lies as a new child to a1 named as d12

Table 1.3: Student allocation table with computed values

Node number	Type	Average Marks	Time Taken(180 minutes)	Sibling	Child	Allocation Status Level
a	Gifted	99.9	150	Yes	Yes	0
a1	Gifted	97.8	147	Yes	No	1
a2	Gifted	98.9	165	Yes	Yes	1
a21	Gifted	95.8	130	Yes	No	2
a22	Gifted	96.9	125	Yes	No	2
b	Good	89.9	175	Yes	Yes	0
b1	Good	85.9	176	No	Yes	1
b11	Good	80.1	172	Yes	No	2
b12	Good	79.5	173	Yes	No	2
b13	Good	75.4	174	Yes	No	2
c	Average	69.9	180	Yes	Yes	0
c1	Average	64.5	179	No	Yes	1
c11	Average	63.2	180	Yes	No	2
c12	Average	61.4	180	Yes	No	2
d	Slow	59.9	164	Yes	Yes	0
d1	Slow	57.8	160	Yes	Yes	1
d2	Slow	55.7	159	Yes	Yes	1
d3	Slow	52.6	158	Yes	Yes	1
d11	Slow	52.3	100	No	No	2
d21	Slow	51.8	98	Yes	No	2
d22	Slow	51.3	97	Yes	No	2
d31	Slow	50.6	95	No	No	2
X1=b2	Good	86.2	175	Yes	Yes	1
X2=b21	Good	75.6	180	No	No	2
X3=a11	Gifted	98.6	160	Yes	No	2
X4=c13	Average	62.1	180	Yes	No	2
X5=a12	Gifted	98.5	175	Yes	No	2
X6=d12	Slow	50.1	104	Yes	No	2

IV. RESULTS AND DISCUSSION:

The implementation of our proposed methodology computes the expectation of node behaviors in syntactical way. The final net work may obtain the following structures if implemented in an optimistic approach as follows,

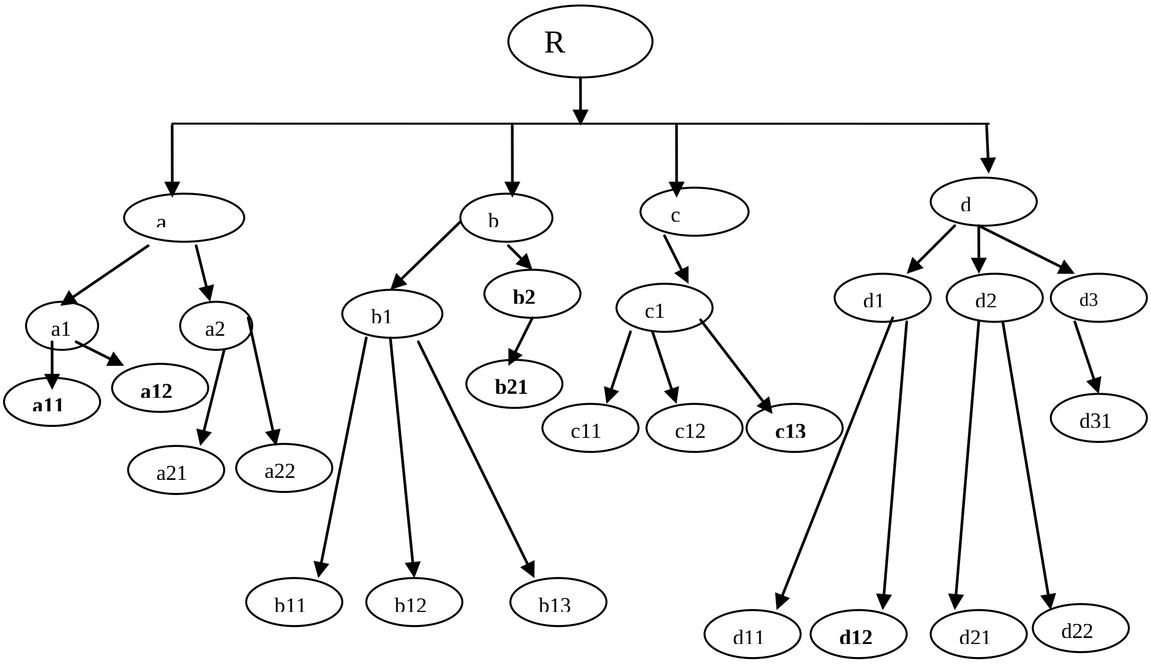


Figure 1.3: Social Corporate network graph with predicted node responses.

V.CONCLUSION:

In this paper, we implemented the graph mining technique of graph classification with our proposed algorithmic strategy. By using this strategy we adopt the group study structure and also the selection of course ware towards each clustered group in order to attain the examination results and incorporated knowledge with the maximum level of efficiency. This graph mining techniques is based on the classification, clustering, decision tree approaches, which are the graph mining fundamentals. In addition, the strategies are supporting the optimistic way of stimulus response feature. We also have highlighted the research contributions and found out some limitations in different research works. Consequently, this work also depicts the critical evaluation in which prediction has been taken out to show the similarities and differences among different node responsibilities equitant to educational network clients. The spatiality of this work is that it reveals the literature review of different graph mining techniques and provides a vast amount of information under a single paper. In our future work, we have planned to propose a cluster mining method based on graph mining technique, provide its implementation and compare its results with the different existing classification based graph mining algorithms.

REFERENCES

[1] G. Di Fatta, S. Leue, E. Stegantova. “Discriminative Pattern Mining in Software Fault Detection.”*Workshop on Software Quality Assurance*, 2006.

[2] E. W. Dijkstra. “A note on two problems in connection with graphs. *Numerische Mathematik”* ,269-271.

[3] T. Falkowski, J. Bartelheimer, M. Spilopoulou. “Mining and Visualizing the Evolution of Subgroups in Social Networks”, *ACM International Conferenceon Web Intelligence*, 2006.

[4] 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.

[5] J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney. “Statistical properties of community structure in large social and information networks.” In WWW, pages 695-704, 2008.

[6] J. Feigenbaum, S. Kannan, A. McGregor, S. Suri, J. Zhang. “Graph Distances in the Data-Stream Model”. *SIAM Journal on Computing*, 38(5): pp.1709–1727, 2008.

[7] J. Ferlez, C. Faloutsos, J. Leskovec, D. Mladenic, M. Grobelnik. “Monitoring Network Evolution using MDL”. *IEEE ICDE Conference*, 2008.

[8] F. Eichinger, K. B-ohm, M. Huber. “Improved Software Fault Detection with Graph Mining”. *Workshop on Mining and Learning with Graphs*,2008.

[9] F. Eichinger, K. B-ohm, M. Huber. “Mining Edge-Weighted Call Graphs to Localize Software Bugs“. *PKDD Conference*, 2008.

[10] W. Fan, K. Zhang, H. Cheng, J. Gao. X. Yan, J. Han, P. S. Yu O. Verscheure.Direct “Mining of Discriminative and Essential Frequent Patterns via Model-based Search Tree“. *ACM KDD Conference*, 2008.

[11] C. Liu, F. Guo, and C. Faloutsos. Bbm: “Bayesian browsing model from petabyte-scale data. In KDD“, pages 537-546, 2009.

[12] Y. Low, J. Gonzalez, A. Kyrola, D. Bick son, C. Guestrin, and J. M. Heller stein. “Graph lab: A framework for parallel machine learning“. In UAI, pages 340-349, 2010.

[13]R. Gemulla, E. Nijkamp, P. Haas, and Y. Sisma-nis. “Large-scale matrix factorization with distributed stochastic gradient descent“. In Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 69-77. ACM, 2011.

[14] A. Ghoting, R. Krishnamurthy, E. P. D. Pednault,B. Reinwald, V. Sindhwani, S. Tatikonda, Y. Tian,and S. Vaithyanathan. “System: Declarative machine learning on map reduce“. In ICDE, pages 231-242, 2011

[15] U. Kang, H. Tong, J. Sun, C.-Y. Lin and C. Faloutsos. “Gbase: an ancient analysis platform for large graphs“.VLDB J., 21(5):637-650, 2012.