CHAPTER-4

MODULAR IDENTIFICATION OF FREQUENT SUBGRAPH STRUCTURES IN GRAPH DOMAIN

This Chapter describes the frequent sub graph structure identification using Graph Mining Technique©.

The Frequent Subgraph search plays a vital role to study structured graph with different level of implications. Our conventional setup initially focuses with subgraph and its vertex-edge connectivity. This paper perform a detailed study of classified subgraph of different sub structure towards variant vertex clusters in the field of graph mining which can be carried out with the comparison search prediction strategies[14]. We will implement our frequent sub graph identification techniques with the implementation of Mathematical Computation based edge cluster domains related to its frequency of occurrence. 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 Identification of Frequent Sub structure graph mining techniques for predicting the Graph sub structure behaviors [10].

4.1 Extraction Schema Methodology

This proposed methodology focuses on the implementation of a Graph Conversion algorithmic strategy to search the requested graph details by implementing the conversional computations. 

© This work covers our paper S.Siva Kumar and Dr.S.P.Victor “Modular Identification of Frequent Sub Graph Structures in Structured Graph Domain Using Graph Mining Technique”, International Journal of Innovative Technology and Research (IJITR) ISSN: 2320-5547, Volume-2, Issue-4-July-2014 pages 1098 – 1100.
Frequent Subgraph Mining (FSM) is the essence of graph mining. The objective of FSM is to extract all the frequent subgraphs, in a given data set, whose occurrence counts are above a specified threshold [13].

**4.2 Extraction Schema Implementation**

Consider the possible sample graph structures ordered by number of vertices.

Given a graph dataset $D$, find sub graph $g$, s.t

$$\text{Freq}(g) \geq 0$$

Where $\text{freq}(g)$ is the maximum percentage of graphs in $D$ that contains $g$.

![Sample Frequent Subgraph with 4 vertices](image)

**Figure 4.1:** Proposed Frequent Sub graph mining structure

**Figure 4.2:** Sample Frequent sub Graph mining structures
The graph representation are computed as follows,

START

Step 1 : Graphs Edges are sorted as per the vertices in ascending order with a given. 0

   Of frequency

Step 2 : Edge representation can be done through the following computation.

   For a 3 vertices graph (I, j) exists then value=1 else value=0

   (a,b) if edge exists then “1” else “0”

   (a,c) if edge exists then “1” else “0”

   (b,c) if edge exists then “1” else “0”

Step 3 : Identify all the edge values as a string

Step 4 : SubGraph frequency matching can be done by collecting the common value

   tableset for each graph

   Compute the total value T for all the edges

   Compare T>= 0

   Collect the edges and map it with the graphset

   Formulate the common subgraph.

STOP
4.3 IMPLEMENTATION RESULT

Now the proposed algorithmic computational values for figure 4.1 we obtain the following tabulation,

<table>
<thead>
<tr>
<th>Graph</th>
<th>a-b</th>
<th>a-c</th>
<th>a-d</th>
<th>a-e</th>
<th>a-f</th>
<th>b-c</th>
<th>b-d</th>
<th>b-e</th>
<th>c-d</th>
<th>c-e</th>
<th>c-f</th>
<th>d-e</th>
<th>d-f</th>
<th>e-f</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>G2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>G3</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>G4</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Total-T</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>

Find the Total Value T>=3 edges

Given \( T = 3 \)

The feasible solution edges are a-b, b-c and b-d

In G1, G2, G3 the edges a-b, b-c, b-d form a frequent subgraph. Hence the solution.

![Sample Frequent Subgraph with 4 vertices](image)

**Figure 4.3**: Solution for proposed Frequent Sub Graph mining structures
4.4 CONCLUSION

This chapter describes the implemented the frequent sub graph mining technique of graph identification approach with our proposed algorithmic strategy, the advantage of such simplifications is that the complexity of graph matching is reduced when tackling the original problem and this allows the use of techniques that require evaluating a lot of individuals through the search for the best solution. This approach may include many relationships that can be decisive when searching for a satisfactory matching. The overall method proves to be highly efficient compared to mining significant and open trees, dramatically reducing running time and number of features mined. Moreover, the experimental results revealed that the expressiveness of frequent sub graph matching impact influence optimization representatives is significantly higher than that of open trees, because a lower number of features are associated with better accuracy, mainly due to higher specificity, reducing false alarms in matching tasks. In our future work, we have planned to propose a new classification method based on graph mining technique, provide its implementation and compare its results with the different existing classification based graph mining algorithms.

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.