A-RAFF: A Ranked Frequent Pattern-growth Subgraph Pattern Discovery Approach

Saif Ur Rehman[1], Sohail Asghar[2]

[1] Department of computer science, Abasyn University, Islamabad, Pakistan
[2] Department of computer science, COMSATS Institute of Information Technology, Islamabad, Pakistan
Saifi.ur.rehman@gmail.com, sohail.asg@gmail.com

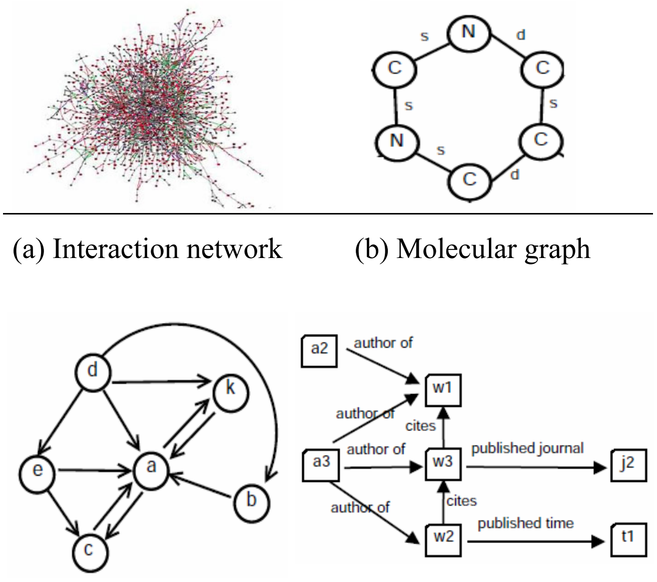
Abstract

Graph mining is one of the arms of Data Mining in which voluminous complex data are represented in the form of graphs and mining is done to infer useful knowledge from them. Frequent subgraph mining (FSM) is an active research field and is considered as the essence of graph mining. FSM is defined as finding all the subgraph patterns that occur frequently over the entire set of graphs. FSM is extensively used in graph clustering, classification and building indices in the databases. In literature, different FSM algorithms have been proposed such as AGM, FSG, SPIN, SUBDUE, gSpan, FFSM, CloseGraph, FSG, GREW. Most of these FSM techniques perform very well for small to medium size graph datasets, but the computational cost of FSM becomes very critical when the graph size is increased. In accession to this, the number of frequent subgraphs patterns grows exponentially with the increasing size of graph datasets. Consequently, in this research work, a novel FSM approach A RAnked Frequent pattern-growth Framework (A-RAFF) is proposed. This work is a preliminary work to study on how to make A-RAFF both computational effective and avoid the generation of the huge number of useless frequent subgraph patterns. A-RAFF has achieved efficiency by embedding the ranking of discovering FSGs during the mining process. The experiments on the three different real benchmark graph datasets demonstrated that the mining results of A-RAFF are very promising as compared to the existing FSM techniques.

Keywords: Graph mining, Frequent subgraphs, Apriori based FSGs, Pattern growth based FSGs

1 Introduction

The primary goal of data mining is to discover the statistically significant and hidden knowledge from the data [1-2]. The data used in data mining can be represented in various formats of structured data such as tables, graphs, etc. Modeling structured data as graphs generate an expressive and general purpose structure. Recently, traditional data mining techniques such as clustering, classification, frequent pattern mining and indexing have now been extended to the graph [4, 78]. Different examples of graph represented data are shown in Figure 1(a) to Figure 1(d).



(a) Interaction network (b) Molecular graph

(c) Behavior graph (d) Bibliography graph

Figure 1. Different graphs represented Data

Graph mining is a well-explored area of research in which voluminous complex data are represented in the form of graphs and mining is done to infer knowledge from them [79]. Graph mining is widely used for several applications, for example, 3D motifs discovery in protein structures [5-6], extracting significant subgraphs from protein-protein interaction networks [7-8], link spam detection in web data [9-10], mining attributed patterns over semantic data [11-12], drug discovery [14-15], discovering relationships in social networking web sites [15-16, 28], discovery of dense subgraph [18-20], among others. In graph mining algorithms, typically a labelled and immutable graph is used as an algorithm input and during mining process those patterns are mined satisfying some algorithm-

[*] Corresponding Author: Saif ur Rehman; E-mail: Saifi.ur.rehman@gmail.com

DOI: 10.3966/160792642019012001024

specific property (such as frequency above some user provided threshold). Some popular graph mining subdomains are included: graph searching [21-22], Approximate graph pattern mining [23-25], Graph classification [26-27], Frequent subgraph mining [29], Web structure mining [30-31], Graph indexing [3, 32].

Frequent Subgraph Mining (FSM) is the core research area of graph mining. FSM aims to discover all the subgraph patterns, whose occurrences within a graph dataset are above a user-defined threshold. These subgraph patterns are called Frequent Subgraph Patterns (FSPs, hereafter). Theoretically, FSM can be formulated as a search in a search space, modelled by a lattice, consisting of all possible subgraph patterns [14]. FSM plays an essential role in many graph mining applications such as chemical compound analysis [12, 21], document image clustering [33], software bug [34], web content mining [35-38], social network mining [39-42], email mining [43-45] and anomaly detection [46-47]. Over the period, 1994 to present, large number of FSM algorithms are proposed. These FSM algorithms have been highlighted in Section 3, Literature review on FSM, in this article.

In this paper, we aim to propose an effective and efficient novel FSM approach called **A RA**nked **F**requent pattern-growth Framework (A-RAFF). In the proposed A-RAFF, labelled undirected graph datasets are used. A-RAFF is based on pattern-growth. This is due to the fact that pattern-growth discovers the entire set of frequent subgraphs patterns without involving costly operation of candidate generation [28]. A-RAFF involves a novel ranking mechanism to rank the discovered FSGs and thus results in most interesting and significant subgraph patterns. A detailed discussion on A-RAFF is provided in Section 4 of this paper. The A-RAFF preliminary experimental results obtained from different benchmark graph datasets are promising and demonstrated that the A-RAFF approach can mine all FSPs in a more efficient manner as compared to the existing FSM approaches such as gSpan, Close-Graph, SPIN, FFSM, FSP, Gaston. The contribution of this paper may thus be summarized as: (i) summarization of different well-known FSM techniques based on different identified common characteristics; (ii) a novel frequent subgraph pattern discovery architectural framework, called A-RAFF with embedded ranking mechanism of frequent subgraph; (iii) an efficient and effective frequent subgraph discovery algorithm; (iv) experimental evaluation of the proposed A-RAFF; (v) performance evaluation of the proposed A-RAFF with the state-of-the art FSM approaches on different graph datasets.

The rest of this paper is organized as follows. Some background knowledge about graph theoretic is described in Section 2. Section 3 discusses about the related work. Problem formulation and the proposed A-RAFF technique are discussed in Sections 4. Section 5 and Section 6 discuss about the Experimental settings and Performance evaluations respectively. This paper is closed with Conclusion and future work in Section 7, followed by the Acknowledgement and References.

2 Graph Preliminaries

Most of the concepts in graph mining are directly taken from graph theory. For better understanding, there are some mathematical terms that need to be discussed before proceeding on to the research work done in this study. Here defined terminologies are frequently used in this work. Different graph notations used throughout the paper are given in Table 1.

Table 1. Different notations used in this paper

Notation	Description
G_D	Graph Database
E(G)	An edge of a graph *G*
G	A graph in graph database
DFS	Depth First Search
BFS	Breadth first search
V(G)	Vertex of a graph *G*
GI	Graph isomorphism
FSGs	Frequent Subgraphs
FSM	Frequent Subgraph Mining
FSPs	Frequent Subgraph Patterns

Definition 1: A *graph* $G=\{V,E\}$ consists of a set of objects $V=\{v_1,v_2,\ldots,v_n\}$ called vertices or nodes and another set of objects $E=\{e_1,e_2,\ldots,e_n\}$ called edges. The order of a graph is denoted by $|V|$ and size by $|E|$.

Definition 2: A graph $G_2=\{V_2,E_2\}$ is a subgraph of another graph $G_1=\{V_1,E_1\}$ iff $V_2\subseteq V_1$ and $E_2\subseteq V_1$ $\wedge(V_1,V_2)\in E_2\rightarrow V_1\in V_2$ and $v_2\in V_2$. The G_1 is called a supergraph of G_2.

Definition 3: Let $G_1=(V_1,E_1,\alpha_1,\beta_1)$ and $G_2=(V_2,E_2,\alpha_2,\beta_2)$ be two graphs. G_2 is an induced subgraph of G_1,if $V_2\subseteq V_1$, $\alpha_1(v)=\alpha_2(v)$ for all $v\in V_2$, $E_2=E_1\cap(V_2\times V_2)$, and $\beta_1(e)=\beta_2(e)$ for all $e\in E_2$. Given a graph $G_1=(V_1,E_1,\alpha_1,\beta_1)$, if any subset $V_2\in V_1$of its vertices uniquely defines a subgraph, this subgraph is called the subgraph induced by V_2.

Definition 4: Two graphs $G_1=(V_1,E_1)$ and $G_2=(V_2,E_2)$ are isomorphic if they are topologically identical to each other, that is, there is a mapping from $G_1$to G_2 such that each edge in E_1 is mapped to a single edge in E_2 and vice versa. In the case of labelled graphs, this mapping must also preserve the labels on the vertices and edges.

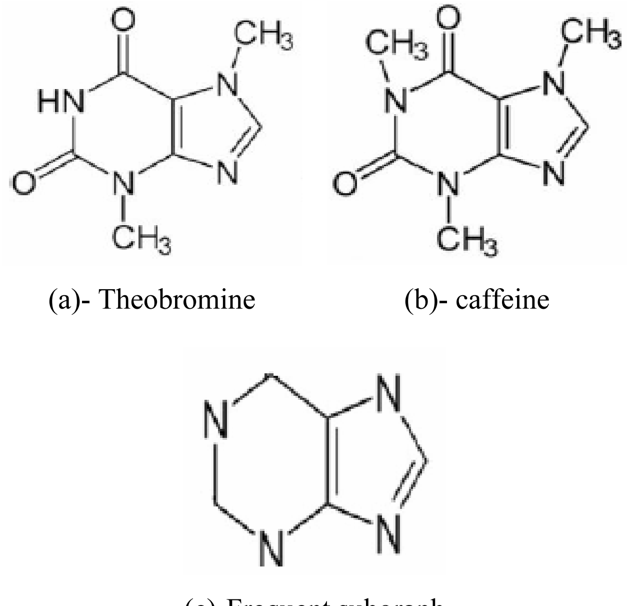
Definition 5: Given two graphs $G_1=(V_1,E_1)$ and $G_2=(V_2,E_2)$, the problem of (sub) graph isomorphism (GI) is to find an isomorphism between G_2 and a subgraph of G_1, that is, to determine whether or not G_2 is included in G_1.

Definition 6: Support threshold is an interestingness measure (normally a numeric value) supplied by the user. In the mining process, this measure is used to check whether a given object (graph, cluster, itemset etc.) passes the support threshold value or not. Usually, during the mining process, those objects are kept in the result set which meet this support threshold where as others are ignored or removed in future steps of the mining process. In graph mining, the support threshold is given by,

$$Support(G) = G_1 / |N| \tag{1}$$

In the above Eq. (1) G_n represents the graph database transactions which contain a specific graph G and N denotes total number of graphs in the given graph database [48].

Definition 7: A frequent subgraph (FSG) is a graph whose support is no less than a minimum user specified support threshold (σ). Given a labelled graph dataset $GD = \{G_1, G_2, \ldots, G_n\}$, support or frequency of a subgraph g is the percentage (or number) of graphs in GD where g is a subgraph [49-50]. In Figure 2, an example of the FSG is given. The graphs in Figure 2 (a) and Figure2(b) represents two chemical compounds Theobromine and Caffeine respectively. If the support threshold is assumed to be 2, i.e.σ > =2, then the subgraph structure given in Figure 2(c) gives one of the possible FSGs which is found in both graphs of the chemical compounds (as its support value σ = 2).



(a)- Theobromine (b)- caffeine

(c)-Frequent subgraph

Figure 2. Two chemical graphs and their frequent subgraph

3 Literature Review on FSM

Development of frequent subgraph algorithms is particularly challenging and computationally, as graph and subgraph isomorphism play very significant role throughout the computation.

It is widely accepted in the literature that FSM techniques are classified into two categories: (1) Apriori-based approaches; and (2) pattern growth-based approaches [48, 54-57]. These two categories are similar in spirit to counterparts found in association rule mining, namely the Apriori algorithm and pattern-growth algorithm [4] respectively. Both of these approaches aim to identify the frequently occurring subgraph patterns from a given collection of small graph sets or within one large graph. These two approaches are different from each other in the way they mine the FSPs.

In the last few decades, numerous FSM algorithms developed in both approaches such as FSG [53], FS3 [63], AGM [51], gSpan [52], CloseGraph [58], Subdigger [64], SPIN [59], Gaston [50], Mofa [33], Margin [61] and LC-. mine [65], FSP [77]. In this work, different FSM techniques are summarized in Table 3 and Table 4 at the end of this paper based on the different identified common parameters found in these approaches. These include (1) nature of the graph inputs; (2) FSM techniques output nature; (3) search strategy adopted by the each FSM technique; (4) graph type addressed; (5) graph representation; (6) isomorphic test used; and (7) candidate generation methodology. Interested readers are further referred to the latest FSM survey articles [13, 48, 54-57].

Although, different FSM algorithms have been used effectively to discover FSGs in domains involving subgraphs which are relatively small in volume. However, when such FSM algorithms are applied to more substantial domains, including image mining, text mining and social network mining, the computational complexity becomes critically very high due to the combinatorial explosion, encountered with respect to the number of possible FSPs [48, 80]. Therefore, many existing approaches to FSM cannot cope with large graph datasets [49-52]. Moreover, mostly FSM techniques mine a prohibitively large number of FSGs during the mining process. This is due to the fact that support threshold (σ) is kept low as the FSM algorithms attempt not to miss any significant FSPs [48]. Therefore, affecting the performance of the FSM algorithm, as the analysis of a large number of frequent subgraph patterns is both difficult as well as resource intensive [20, 42, 57, 64]. Therefore, there is a dire need to devise such an algorithm which can handle datasets of massive size and reduce the number of FSPs with no compromise on missing of any significant FSPs. To address these issues, in this paper a new FSM technique is proposed, called A-RAFF. A-RAFF can mine FSGs more efficiently as compared to other FSM techniques. This is discussed in the next section in details.

Table 3. Apriori approach FSM Techniques Comparison

Authors & Techniques	Search Strategy	Isomorphic Test	Graph Type	Input Nature
		SubgraphGeneration	Graph Representation	Output Nature
SPIN [59]	Greedy Search	Exact	General Graphs	Set of graphs
		Join	Adjacency Matrix	Maximal Frequent subgraphs
FSG [53]	BFS	Exact	Undirected	Set of graphs
		Level-wise join	Adjacency List	Frequent Connected Subgraphs
AGM [51]	BFS	Exact	Not limited	Set of graphs
		Level-wise join	Canonical Adjacency Matrix	Incomplete
HSIGRAM [76]	BFS	Adjustable	Labelled, Undirected	Single Large Graph
		Level-wise join	Canonical Adjacency Matrix	Frequent Subgraphs
FFSM [60]	DFS	Exact	Labelled, Undirected	Set of graphs
		Join extension	Adjacency Matrix	Frequent Subgraphs
Dynamic GREW [62]	DFS	Exact	Dynamic graphs	Uncertain set of Graphs
		Level-wise join	Adjacency Matrix	Dynamic patterns in Frequent Subgraphs

Table 4. Pattern growth based approach FSM Techniques Comparison

Authors & Techniques	Search Strategy	Isomorphic Test	Graph Type	Input Nature
		Subgraph Generation	Graph Representation	Output Nature
gSpan [52]	DFS	Exact	Labelled, Undirected	Set of graphs
		Rightmost Extension	Adjacency Matrix	Complete
CloseGraph [58]	DFS	Exact	Static	Set of graphs
		Rightmost Extension	Adjacency Matrix	Incomplete
SUBDUE [29]	Greedy	Approximate	Labelled	Single Large Graph
		Level-wise Search	Adjacency Matrix	Complete
GASTON [50]	DFS	Exact	Labelled, Undirected	Set of graphs
		Extension	Hash Table	Complete
MOFA [33]	DFS	Exact	Labelled, Undirected	Set of graphs
		Rightmost Extension	Adjacency Matrix	Complete
TSP [49]	DFS	Exact	Dynamic	Set of graphs
		Extension	Adjacency Matrix	Incomplete

4 A Proposed Frequent Subgraph Mining Approach: A-RAFF

In this section, the proposed FSM approach called A-RAnked Frequent pattern-growth Framework (A-RAFF) for FSGs discovery is described, highlighting the major characteristics of the A-RAFF.

In the proposed FSM approach, labelled undirected graph datasets are used. The goal of A-RAFF framework is to discover a small collection of ranked frequent subgraphs patterns from a database of labelled input graphs. A-RAFF used the basic characteristics of the pattern-growth scheme to discover the FSPs, which is solely works on the divide and conquer strategy. A-RAFF is based on pattern-growth as pattern-growth discover the entire set of frequent subgraphs patterns without involving costly operation of candidate generation during the mining process. An abstract level architectural framework for A-RAFF is shown in Figure 3 and the complete algorithms involved in A-RAFF are given in Figures 4, Figure 5 and Figure 6.

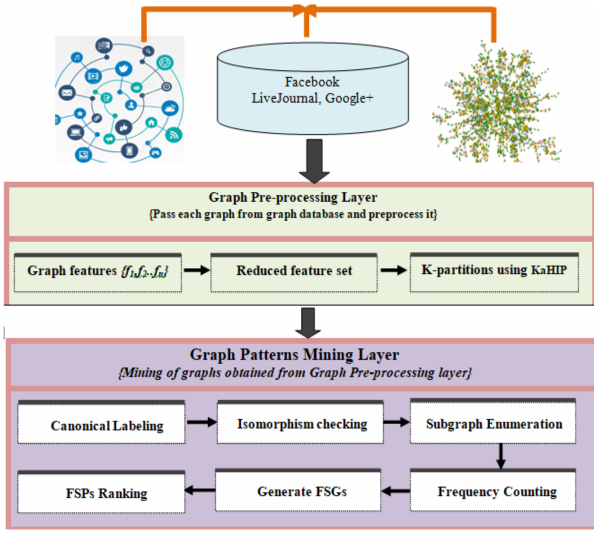

Facebook
LiveJournal, Google+
Graph Pre-processing Layer
{Pass each graph from graph database and preprocess it}
Graph features {f1,f2..fn}
Reduced feature set
K-partitions using KaHIP
Graph Patterns Mining Layer
{Mining of graphs obtained from Graph Pre-processing layer}
Canonical Labeling
Isomorphism checking
Subgraph Enumeration
Frequency Counting
Generate FSGs
FSPs Ranking

Figure 3. Architectural framework of the proposed A-RAFF

```
Proposed Algorithm 1. A-RAFF ($\mathbb{GD}, \sigma, RF$)
Input: $\mathbb{GD}$ = a graphs dataset of the labelled undirected graphs
       $\sigma$ =minimum threshold
Output: $RF$ , a set of ranked frequent subgraphs
    count all the features and put in the feature set "F"
    for each feature f in the F-set do
    Identify most informative graph features
    repeat until all features are checked
    partition the graph using KaHIP Tool based on an F-set [71]
$F \leftarrow \phi F$ denotes the frequent subgraph patterns set
$F \leftarrow$ discovered 1-frequent subgraphs patterns
    for each graph $g_i\_partition, \in G_i$ do
  FSPs ($\mathbb{GD}$, $g_i\_partition, \sigma, F$ )
end
FSP-Rank(F)
return RF
```

Figure 4. Proposed Algorithm for A-RAFF

```
Proposed Algorithm 2. FSPs ($\mathbb{GD}$ $g_i\_partition, \sigma, RF$ )
Output: F mined frequent subgraph patterns
1. if $g\_i\_partition, \in F$ then
2.     return
3. else
4.  $F \leftarrow F \cup g_i\_partition$
5. end
6. extend $g_i\_partition$ by adding all edges "e" $\in \mathbb{GD}$ such that
7.  $extended\_g_i \leftarrow g_i\_partition \cup e$
8. for each $extended\_g_i$ from Line. (7) do
9. if support ( $extended\_g_i$ ) $\geq \sigma\,|\mathbb{GD}|$ then
10.         FSPs($\mathbb{GD}$, $extended\_g_i, \sigma, F$ )
11. else
12.         return
13. end
14. return F
```

Figure 5. Proposed Algorithm for FSPs discovery

```
Proposed Algorithm 3. FSP-Rank (FSPs, RF)
Input: FSPs = frequent subgraph patterns
Output: RF, set of ranked frequent subgraph patterns
compute λ using (3)
for each FSP in FSPs do
score $(D_i) \leftarrow \sum_{i=0}^{n}$ (in−degree+out−degree)
compute $f(R_k)$ using (2)
 $RF \leftarrow RF \cup FSP$
next
return RF
```

Figure 6. Proposed Algorithm for FSP-Rank measure

The A-RAFF has two layers: graph pre-processing layer and graph patterns mining layer. In A-RAFF, different labelled undirected graphs are stored in graph databases. There are two functions of the graph pre-processing layer: first function is to extract the relevant graph dataset features, therefore ignoring/removing those features which are less interesting as compared to other features. The second function of the pre-processing layer is graph partitioning. Different graph partitioning algorithms exist in the literature [66-70, 72-73]. In A-RAFF, the graphs are partitioned using well-know graph partitioning tool KaHIP [71].

In the KaHIP tool, Sander and Schulz implemented a multilevel graph partitioning scheme called KaFFPa (Karlsruhe Fast Flow Partitioning). KaFFPa algorithm exploits a novel local improvement algorithm which is based on max-flow and min-cut computations. Furthermore, KaFFPa used more localized FM searches in addition to involving of a sophisticated global search strategies transferred from multi-grid linear solvers problem [72].

The graph pattern mining layer is the core layer of A-RAFF framework. This layer is responsible to discover the ranked FSPs. At the beginning of A-RAFF, a list of frequent subgraphs is derived such that the subgraphs are organized in descending order of the frequency computed value. Furthermore, using this frequency descending list, the graph collection is compressed into required frequent pattern tree (FP-tree). This FP-tree structure also maintains the subgraphs association minutiae. The creation of this FP-tree structure is the fundamental requirement of the pattern-growth approaches to FSM. Moreover, in this layer, a ranking of the discovered FSPs is incorporated. For FSPs ranking, FSP-Rank measure is proposed.

The ranking of the FSPs is performed to avoid the repetitive generation of the FSPs, therefore, reducing the resultant frequent subgraph patterns. In A-RAFF, FSP-Rank measure is proposed to calculate the Rank of the discovered FSPs. FSP-Rank is computed using the following equation:

$$f(R_k) = (1-\lambda) * \sum_{i=1}^{n} W_i + \lambda\left(\frac{|D_i|}{n_i}\right) \qquad \textbf{(2)}$$

In Eq. (2), λ is a normalized factor which is computed using equation (3). The value of λ can be between [0, 1]. W shows the sum of the weight associated with the vertices in the i^{th} frequent subgraph. D denotes the degree of the i^{th} frequent subgraphs and n denotes the total number of vertices in the i^{th} frequent subgraph.

$$\lambda = \left(\frac{\sum_{i=1}^{n}(FSG_i)}{\sum_{i=1}^{n}T(V_i)}\right) \qquad \textbf{(3)}$$

In Eq. (3), FSG_i corresponds to the number of discovered frequent subgraph and $T(V_i)$ represents the total number of the vertices found in all of the n frequent subgraphs discovered.

In any of the FSM technique graph isomorphism (GI) detection is fundamental. A significant number of efficient schemes have been proposed with the objective to reduce the computational overhead allied with GI problem. To perform GI checks, canonical labeling has been successfully used. In canonical labeling, each graph is assigned to a unique code (i.e. a sequence of bits, a string, or a sequence of numbers). This code is invariant on the ordering of the vertices and edges in the graph [74-75]. After assigning a unique code to each graph using canonical labels, GI is performed by comparing whether they have identical canonical labels. In the proposed FSM approach, A-RAFF, graph isomorphism is performed using the canonical labeling strategy adopted by [62].

Complexity Analysis: The proposed algorithm, A-RAFF, is based on the pattern-growth category. This is due to the fact that the FSM algorithms based on the pattern-growth approaches are more efficient in computational complexity than the algorithms using an Apriori approach [13]. Graph isomorphism is an unavoidable issue faced by all subgraph mining algorithm, and is a NP-hard problem which cannot be solved in polynomial time.

Both canonical labelling and determining graph isomorphism are not known to be either in P or in NP-complete [74]. In the proposed scheme, the canonical labelling strategy is used from [62], as it fully makes use of edge and vertex-labels for fast processing and various vertex invariants to reduce the complexity of determining the canonical label of a graph.

In addition to the canonical labelling and graph isomorphism, the computational complexity of the proposed A-RAFF algorithm mainly depends on computation involved in the discovery of FSPs and ranking of the frequent subgraphs. Thus, computational complexity for the loop used for the extraction of the graph features is $O(n)$, where n is total of graph features. The same is also defined for FSP-Rank. In the FSPs algorithm, there is recursion involved inside the loop. Therefore, it can easily be observed that it will compute all the frequent subgraph patterns in $O(2^{N^2})$ time.

5 Experiment Settings

This section presents details about the experimentsettings.

5.1 Experimentation Environment

A set of different experiments is performed to evaluate the performance of A-RAFF. All of these experiments are performed on a 32-bit machine running the Linux operating system with 6 GB memory and 3.0 GHz Intel processor. A-RAFF is implemented using Java. For the purpose of A-RAFF evaluation, FFSM, FSP, CloseGraph are re-implemented in Java. The executable of gSpan and Gaston are obtained from their respective authors.

5.2 Dataset Introduction

Three different graph datasets are used in the experiment. These datasets are included Chemical Compound, AIDS antiviral screen compound and DTP human tumor cell line screen (CANSO3SD).

Table 2. Dataset Statistics

Dataset	Dataset Description
DTP human tumor cell line screen (CANSO3SD)	This dataset consists of 42,247 molecules. Each molecule corresponds to a graph, atoms are represented using nodes and the bonds between them are represented by edges.
Chemical Compound	340 chemical compounds, 24 different atoms, 66 atom types, and 4 types of bonds. On average 27 vertices per graph and 28 edges per graph. The largest one contains 214 edges and 214 vertices.
AIDS antiviral screen compound	The dataset contains 43,095 chemical compounds The compounds are classified into three classes. 41179 belong to CI (Confirmed Inactive), 1081 belong to CM (Confirmed Moderately active) and 422 belong to CA (Confirmed Active)

For the preliminary analysis, these three graph datasets are used becausemost of the FSM techniques performed the performance comparison using these benchmark graph datasets. The statistics of each of the graph datasets are given in Table 2.The graph datasets considered are divided into two subsets. 80% of each graph dataset is used for training and 20% of the each graph dataset is reserved for testing of the A-RAFF.

5.3 Evaluation Parameters

Support Threshold (σ) is the most important parameter in any of the FSM schemes. The value of Support Threshold (σ) is set to 20, 30, 40 and 50. The performance of the A-RAFF with its counterparts FSM approaches is evaluated using this different range of values of σ. A detailed discussion on the performance evaluation is given in the forthcoming section of this paper.

6 Performance Evaluations

A series of different experimentation is performed on the real benchmark graph datasets. The following experiments are conducted to demonstrate the performance of A-RAFF as compared to the other well-know FSM approaches. Another factor considered during the performance comparison is the computation time. A detailed analysis is highlighted in this section.

Experiment 1. *DTP human tumor cell line screen (CANSO3SD)*: First experiment is performed using

DTP human tumor cell line screen (CANSO3SD) graph dataset. The computation time required to discover and rank the FSPs are taken on different threshold value ranging from 20 to 50 is given in Figure 7.

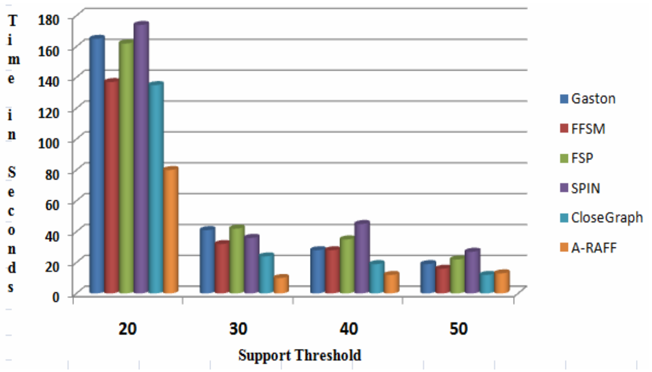


Figure 7. Performance comparison of various FSM with A-RAFF on DTP human tumor cell line screen (CANSO3SD)

These results depict that in most cases, the proposed FSM approach A-RAFF discovered and ranked all the frequent subgraphs in less time as compare to the computational time of other FSM techniques considered for comparison.

Experiment 2. *Chemical Compound*: In the second experiment for A-RAFF comparisons, the Chemical Compound dataset was used. The experimental results of the discovery of FSPs using A-RAFF and other competing FSM techniques are shown in Figure 8.

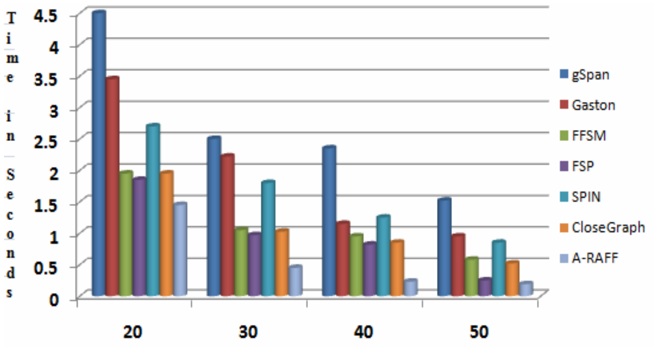


Figure 8. Performance comparison of various FSM with A-RAFF on Chemical Compound

In the experiment, it is observed that as the (σ) value is increased, the time required to discover the FSGs is also decreased. In chemical compound dataset the performance of A-RAFF is very promising and A-RAFF beat the other FSM techniques.

Experiment 3. *AIDS antiviral screen compound:* In the last experiment, AIDS antiviral screen compound graph dataset was used. In this dataset, A-RAFF performance on different threshold values is given in Figure 9. The proposed A-RAFF performed well in this experiment as well, except at 50support threshold values, performance of proposed A-RAFF was same as that of FFSM & FSP and CloseGrpah has performed well as compare to A-RAFF. In other cases, A-RAFF has beaten the other FSM techniques.

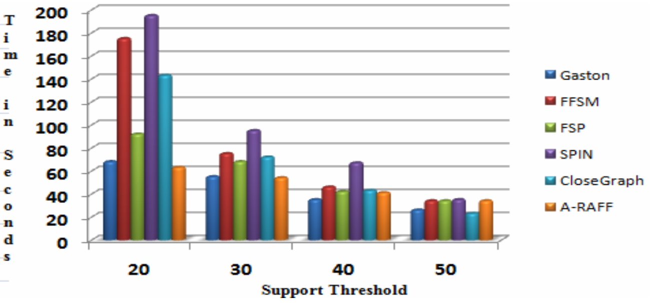


Figure 9. Performance comparison of various FSM with A-RAFF on AIDS antiviral screen compound

By looking at the different experimental results given in Figure 7, Figure 8 and Figure 9, the proposed FSM scheme A-RAFF was much better than the other well-known FSM techniques for the extraction of frequent subgraph patterns. Moreover, in some exceptional cases CloseGraph perform well this is due to the fact that CloseGraph only focus on those graphs which are closed, but the proposed A-RAFF has discovered all the possible FSGs. The results of Gaston tool were better on the AIDS antiviral screen compound graph dataset. Although, A-RAFF beat the Gaston on two out of three graph datasets, we are investigating that why A-RAFF cannot outperform Gaston tool.

7 Conclusions and Future Work

In this paper, the problem of rank frequent subgraph pattern discovery is investigated. An algorithm and a framework are presented called A-RAFF. A-RAFF falls in the pattern-growth category of FSPs. A-RAFF framework is decomposed into two distinctive layers. First layer, graph pre-processing layer, is responsible to select the most useful features for the graph and then the graphs are partitioned. In the proposed A-RAFF, KaHIP tool is used to partition the graph dataset. Graph pattern mining layer is the core layer of the A-RAFF. The outcome of the second layer is the ranked FSPs. In A-RAFF framework a novel ranking method is proposed. The efficiency of the A-RAFF is also confirmed by different benchmark real graph datasets, which are used in most of the FSM approaches. Furthermore, the performance of the A-RAFF is also examined, with gSpan, FFSM, CloseGraph, Gaston, SPIN and FSP, through extensive experiments. There is ample room for the future work in the proposed framework, A-RAFF. In future A-RAFF can be extended to big graph datasets such as obtained from social networking sites. Dynamic graphs can also be incorporated in the A-ARFF. One of the possible future scope of the present study can be to consider the different other graph features such as betweenness centrality, closeness centrality, average path length etc., while performing the ranking of the FSGs.

Acknowledgments

The authors would like to thank University of Kansas for sharing FFSM source code Mr. Xifeng Yan and Professor Jiawei Han in University of Illinois at Urbana Champaign for making available the gSpan executable. We, the authors of this article, really appreciate the valuable suggestions and feedbacks of all the anonymous reviewers of the article. Their valuable suggestions helped us to improve our work to give it the current shape.

References

[1] M. S. Chen, J. Han, P. S. Yu, Data Mining: An Overview from Database Perspective, *IEEE Transaction on Knowledge and Data Engineering*, Vol. 8, No. 6, pp. 866-883, December, 1996.

[2] J. Han, P. Jian, K. Micheline, *Data Mining: Concepts and Techniques*. Elsevier, 2011.

[3] X. Yan, P. S. Yu, J. Han, Graph Indexing: A Frequent Structure-based Approach, *Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data*, ACM, Paris, France, 2004, pp. 35-346.

[4] C. Aggarwal, H. Wang, *Managing and Mining Graph Data*, Springer, 2010.

[5] W. Dhifli, E. M. Nguifo, Motif Discovery in Protein 3D-Structures Using Graph Mining Techniques, in: M. Elloumi, A. Y. Zomaya (Eds.), *Pattern Recognition in Computational Molecular Biology: Techniques and Approaches*, Wiley, 2015, pp. 17-26.

[6] W. Dhifli, A. B. Diallo, PGR: A Graph Repository of Protein 3D-Structures, arXiv preprint arXiv: 1604.00045, January, 2016.

[7] P. Bertolazzi, M. Bock, C. Guerra, On the Functional and Structural Characterization of Hubs in Protein-protein Interaction Networks, *Biotechnology Advances*, Vol. 31, No. 2, pp. 274-286, April, 2016.

[8] J. Ji, A. Zhang, C. Liu, X. Quan, Z. Liu, Survey: Functional Module Detection from Protein-protein Interaction Networks, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 26, No. 2, pp. 261-27, February, 2014.

[9] S. Kumar, X. Gao, I. Welch, M. Mansoori, A Machine Learning Based Web Spam Filtering Approach, *2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*, Crans-Montana, Switzerland, 2016, pp. 973-980.

[10] N. Spirin, J. Han, Survey on Web Spam Detection: Principles and Algorithms, *ACM SIGKDD Explorations Newsletter*, Vol. 13, No. 2, pp. 50-64, May, 2012.

[11] A. Silva, W. Meira, M. J. Zaki, Mining Attribute-structure Correlated Patterns in Large Attributed Graph, *Proceedings of the VLDB Endowment*, Vol. 5, No. 5, pp. 466-477, January, 2012.

[12] A. Prado, M. Plantevit, C. Robard, J. F. Boulicaut, Mining Graph Topological Patterns: Finding Covariations among Vertex Descriptors, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 25, No. 9, pp. 2090-2104, September, 2013.

[13] S. Velampalli, V. M. Jonnalagedda, Frequent SubGraph Mining Algorithms: Framework, Classification, Analysis, Comparisons, *Data Engineering and Intelligent Computing*, Springer, Singapore, 2018, pp. 327-336.

[14] I. Takigawa, H. Mamitsuka, Graph Mining: Procedure, Application to Drug Discovery and Recent Advances, *Drug Discovery Today*, Vol. 18, No. 1, pp. 50-57, January, 2013.

[15] P. Csermely, T. Korcsmáros, H. J. Kiss, G. London, R. Nussinov, Structure and Dynamics of Molecular Networks: A Novel Paradigm of Drug Discovery: A Comprehensive Review, *Pharmacology & Therapeutics*, Vol. 138, No. 3, pp. 333-408, June, 2013.

[16] M. Sachan, D. Contractor, T. A. Faruquie, L. V. Subramaniam, Using Content and Interactions for Discovering Communities in social Networks, *Proceedings of the 21st International Conference on World Wide Web,* Lyon, France, 2012, pp. 331-340.

[17] I. Guy, Social Recommender Systems, in: *Recommender Systems Handbook*, Springer US, 2015, pp. 511-543.

[18] A. Gionis, C. Sourakakis, Dense Subgraph Discovery: Kdd 2015 Tutorial, *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Sydney, NSW, Australia, 2015, pp. 2313-2314.

[19] J. Chen, Y. Saad, Dense Subgraph Extraction with Application to Community Detection, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 24, No. 7, pp. 1216-1230, July, 2012.

[20] S. Koujaku, I. Takigawa, M. Kudo, H. Imai, Dense Core Model for Cohesive Subgraph Discovery, *Social Networks*, Vol. 44, pp. 143-152, January, 2016.

[21] X. Yan, F. Zhu, J. Han, P. S. Yu, Searching Substructures with Superimposed Distance, *Proceedings of the 22nd International Conference on Data Engineering*, Atlanta, GA, 2006, pp. 88-97.

[22] C. Chen, X. Yan, P. S. Yu, J. Han, D.Q. Zhang, X. Gu. Towards Graph Containment Search and Indexing, *Proceedings of the 33rd International Conference on Very Large Data Bases*, Vienna, Austria, 2007, pp. 926-937.

[23] Kelley, R. Sharan, R. M. Karp, T. Sittler, D. E. Root, B. R. Stockwell, T. Ideker, Conserved Pathways within Bacteria and Yeast as Revealed by Global Protein Network alignment, *Proceedings of the National Academy of Science of the United States of America,* Vol. 100, No. 20, pp. 11394-11399, September, 2003.

[24] R. Sharan, S. Suthram, R. M. Kelley, T. Kuhn, S. McCuine, P. Uetz, T. Sittler, R. M. Karp, T. Ideker, Conserved Patterns of Protein Interaction in Multiple Species, *Proceedings of the National Academy of Science of the United States of America* Vol. 102 No. 6, pp. 1974-1979, February, 2005.

[25] C. Chen, X. Yan, F. Zhu, J. Han, Gapprox: Mining Frequent Approximate Patterns from a Massive Network, *Proceedings of the 7th IEEE International Conference on Data Mining,* Omaha, NE, 2007, pp. 445-450.

[26] T. Kudo, E. Maeda, Y. Matsumoto, An Application to Boosting to Graph Classification, *Proceedings of the 8th Annual Conference on Neural Information Processing Systems*, Vancouver, British Columbia, Canada, 2004, pp. 729-736.

[27] J. Huan, W. Wang, D. Bandyopadhyay, J. Snoeyink, J. Prins, A. Tropsha, Mining Protein Family Specific Residue Packing Mining Protein Family Specific Residue Packing Patterns from Protein Structure Graphs, *8th Annual International Conference on Research in Computational Molecular Biology*, San Diego, CA, 2004, pp. 308-315.

[28] R. Irfan, G. Bickler, S. U. Khan, J. Kolodziej, H. Li, D. Chen, L. Wang, K. Hayat, S. A. Madani, B. Nazir, I. A. Khan, Survey on Social Networking Services, *IET Networks*, Vol. 2, No. 4, pp. 224-234, December, 2013.

[29] B. Holder, D. Cook, S. Djoko, Substructure Discovery in the SUBDUE System, *AAAIWS'94 Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining*, Seattle, WA, 1994, pp. 169-180.

[30] S. Brin, L. Page, The Anatomy of a Large Scale Hypertextual Web Search Engine, *Computer Networks and ISDN System*, Vol. 30, No. 8, pp. 107-17, September, 1998.

[31] J. M. Kleinberg, Authoritative Sources in a Hyperlinked Environment, *Journal of the ACM (JACM)*, Vol. 46, No. 5, pp. 668-677, September 1998.

[32] D. Shasha, J. Wang, R. Giugno, Algorithms and Applications of Tree and Graph Searching*, Proceedings of the 21st ACM SIGMOD-SIGACT-SIGART Symposium on Principles on Database Systems*, Madison, WI, 2002, pp. 39-52.

[33] C. Borgelt, M. R. Berthold, Mining Molecular Fragments: Finding Relevant Substructures of Molecules, *IEEE International Conference on Data Mining ICDM 2003,* Maebashi, Japan, 2002, pp. 51-58.

[34] F. Eichinger, K. Böhm, M. Huber, Mining Edge Weighted Call Graphs to Localise Software Bugs, *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, Antwerp, Belgium, 2008, pp. 333-348.

[35] A. Schenker, H. Bunke, M. Last, A. A. Kandel, A Graph Based Framework for Web Document Mining, in: S. Marinai, A. R. Dengel (Eds.), *Document Analysis Systems VI*, Springer Berlin Heidelberg, 2004, pp. 401-412.

[36] R. Baeza, P. Boldi, Web Structure Mining, in: J. D. Velásquez, L. C. Jain (Eds.), *Advanced Techniques in Web Intelligence-I*, Springer Berlin Heidelberg, 2010, pp. 113-142.

[37] B. Panda, S. N. Tripathy, N. Sethi, O. P. Samantray, A Comparative Study on Serial and Parallel Web Content Mining*, International Journal of Advanced Networking and Applications*, Vol. 7 No. 5, pp. 2882, March, 2016.

[38] S. Algur, P. Bhat, Web Video Object Mining: Expectation Maximization and Density Based Clustering of Web Video Metadata Objects*, International Journal of Information Engineering and Electronic Business*, Vol. 8 No. 1, p. 69, January, 2016.

[39] S. Chakradeo, R. Abraham, B. Rani, B. Manjula, Data Mining: Building Social Network, *Indian Journal of Science and Technology*, Vol. 8 No. S2, pp. 212-216, January, 2015.

[40] F. Jiang, K. Kawagoe, C. Leung, Big Social Network Mining for Following Patterns, *Proceedings of the Eighth International Conference on Computer Science & Software Engineering,* ACM, Yokohama, Japan*,* 2015, pp. 28-37.

[41] J. Tang, Y. Chang, C. Aggarwal, H. Liu, *A Survey of Signed Network Mining in Social Media*, arXiv preprint arXiv: 1511.07569, 2015.

[42] H. H. Shuai, C. Y. SheN, D. N. Yang, Y. F. Lan, W. C. Lee, P. s. Yu, M. S. Chen, Mining Online Social Data for Detecting Social Network Mental Disorders, *Proceedings of the 25th International Conference on World Wide Web,* Montréal, Québec, Canada*, 2016,* pp. 275-285.

[43] M. Aery, S. Chakravarthy, InfoSift: Adapting Graph Mining Techniques for Text Classification, *FLAIRS Conference*, Clearwater Beach, FL, 2005, pp. 277-282.

[44] G. Tang, J. Pei, W. S. Luk, Email Mining: Tasks, *Common Techniques, and Tools, Knowledge and Information Systems*, Vo. 41, No.1, pp. 1-31, October, 2014.

[45] L. Alsmadi, I. Alhami, Clustering and Classification of email Contents, *Journal of King Saud University-Computer and Information Sciences*, Vol. 27, No. 1, pp. 46-57, January, 2015.

[46] C. Noble, D. Cook, Graph-based Anomaly Detection, *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Washington, DC, 2003, pp. 631-636.

[47] W. Eberle, L. Holder, Discovering Structural Anomalies in Graph-based Data*, Proceeding of the 7th IEEE International Conference on Data Mining Workshops,* Washington, DC, 2007, pp. 393-398.

[48] S. U. Rehman, S. Asghar, Y. Zhuang, S. Fong, Performance Evaluation of Frequent Subgraph Discovery Techniques, In *Mathematical Problems in Engineering*, Vol. 2014, No. 2, pp. 1-6, August, 2014.

[49] M. Lahiri, T. Y. Berger-Wolf, Structure Prediction in Temporal Networks Using Frequent Subgraphs, *IEEE Symposium on Computational Intelligence and Data Mining,* Honolulu, HI, 2007, pp. 35-42.

[50] S. Nijssen, J. N. Kok, A Quick Start in Frequent Structure Mining Can Make a Difference, *Proceedings of the 10th ACM SIGKDD international conference on Knowledge Discovery and Data Mining*, Seattle, WA, 2004, pp. 647-652.

[51] Inokuchi, T. Washio, H. Motoda, An Apriori-based Algorithm for Mining Frequent Substructures from Graph Data, *Proceeding of 2000 Practice of Knowledge Discovery in Databases Conference (PKDD)*, London, UK, 2004, pp. 13-23.

[52] X. Yan, J. Han, gSpan: Graph-Based Substructure Pattern Mining, *Proceeding of IEEE International Conference on Data Mining*, Maebashi, Japan, 2002, pp. 721-723.

[53] M. Kuramochi, G. Karypis, Frequent Subgraph Discovery, *Proceeding of the Conference on Data Mining, Piscataway*,

San Jose, CA, 2001, pp. 313-320.

[54] K. Lakshmi, T. Meyyappan, A Comparative Study of Frequent Subgraph Mining Algorithms, *International Journal of Information Technology Convergence and Services (IJITCS)*, Vol. 2, No. 2, pp. 23-39, April, 2012.

[55] T. Ramraj, R. Prabhakar, Frequent Subgraph Mining Algorithms–A Survey, *Procedia Computer Science*, Vol. 47, pp. 197-204, January, 2015.

[56] V. Krishna, N. R. Suri, G. Athithan, A Comparative Survey of Algorithms for Frequent Subgraph Discovery, *Current Science (Bangalore)*, Vol. 100, No. 2, pp. 190-198, January, 2011.

[57] C. Jiang, F. Coenen, M. Zito, A Survey of Frequent Subgraph Mining Algorithms, *The Knowledge Engineering Review*, Vol. 28, No. 1, pp. 75-105, March, 2013.

[58] X. Yan, J. Han, Closegraph: Mining Closed Frequent Graph Patterns, *Proceeding of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* Washington, DC, 2003, pp. 286-295.

[59] J. Huan, W. Wang, J. Prins, J. Yang, SPIN: Mining Maximal Frequent Subgraphs from Graph Databases, *Proceedings of the 10th ACM SIGKDD international conference on Knowledge discovery and data mining*, Seattle, WA, 2004, pp. 581-586.

[60] J. Huan, W. Wang, J. Prins, Efficient Mining of Frequent Subgraphs in the Presence of Isomorphism, *the 3rd International. Conference on Data Mining, Piscataway,* Melbourne, FL, 2003, pp.549-552.

[61] L. T. Thomas, S. R.Valluri, K. Karlapalem Margin: Maximal Frequent Subgraph Mining, *ACM Transactions on Knowledge Discovery from Data (TKDD),* Vol. 4, No. 3, pp. 10, October, 2010.

[62] M. Kuramochi, G. Karypis, Grew- A Scalable Frequent Subgraph Discovery Algorithm, *Proceeding of the 4th IEEE International Conference on Data Mining,* Brighton, UK, 2004, pp. 439-442.

[63] T. Saha, M. A. Hasan, FS3: A Sampling Based Method for Top-k Frequent Subgraph Mining, Statistical Analysis and Data Mining, *the ASA Data Science Journal,* Vol. 8 No. 4, pp. 245-261, August, 2015.

[64] S. Shahrivari, S. Jalili, High-performance Parallel Frequent Subgraph Discovery, *The Journal of Supercomputing*, Vol. 71, No. 7, pp. 2412-2432, July, 2015.

[65] B. Douar, M. Liquiere, C. Latiri, Y. Slimani, LC-mine: A Framework for Frequent Subgraph Mining with Local Consistency Technique, *Knowledge and Information Systems,* Vol. 44, No. 1, pp. 1-25, July, 2015.

[66] A. S. Muttipati, P. Padmaja, Analysis of Large Graph Partitioning and Frequent Subgraph Mining on Graph Data, *International Journal of Advanced Research in Computer Science*, Vol. 6, No. 7, September, 2015.

[67] R. Preis, R. Diekmann, PARTY- A Software Library for Graph Partitioning, *Advances in Computational Mechanics with Parallel and Distributed Processing*, Civil-Comp Press, Kippen, Scotland, 1997, pp. 63-71.

[68] H. Meyerhenke, B. Monien, T. Sauerwald, A New Diffusion-Based Multilevel Algorithm for Computing Graph Partitions, *Journal of Parallel and Distributed Computing*, Vol. 69, No. 9, pp. 750-761, September, 2009

[69] B. Hendrickson, R. W. Leland, A Multilevel Algorithm for Partitioning Graphs, *Proceeding of ACM/IEEE Conference on Supercomputing*, San Diego, CA, 1995, pp. 28-28.

[70] B. W. Kernighan, S. Lin, An Efficient Heuristic Procedure for Partitioning Graphs, *The Bell System Technical Journal*, Vol. 49, pp. 291-307, February, 1970.

[71] P. Sanders, C. Schulz, Engineering Multilevel Graph Partitioning Algorithms, *Proceeding of European Symposium on Algorithms*, Bordeaux, France, 2011, pp. 469-480.

[72] P. Sanders, C. Schulz, Think Locally, Act Globally: Highly Balanced Graph Partitioning, *Experimental Algorithms Lecture Notes in Computer Science*, Vol. 7933, pp. 164-175, 2013.

[73] C. E. Bichot, P. Siarry, *Graph Partitioning*, John Wiley & Sons*,* 2013.

[74] S. Fortin, The Graph Isomorphism Problem, *Technical Report, TR96-20,* September, 1996.

[75] R. Read, D. Corneil, The Graph Isomorphism disease, *Journal of Graph Theory*, Vol. 1, No. 4, pp. 339-363, December, 1977.

[76] M. Kuramochi, G. Karypis, *Finding Frequent Patterns in a Large Sparse Graph*, SDM, 2004, pp. 345-356.

[77] S. Han, W. Keong, N. Yu, FSP: Frequent Substructure Pattern Mining, 2007 6th International Conference on Information, Communications & Signal Processing, Singapore, 2007, pp. 12-15.

[78] A. Dhiman, S. K. Jain, Optimizing Frequent Subgraph Mining for Single Large Graph, *Procedia Computer Science*, Vol. 89, pp. 378-385, December, 2016.

[79] C. H. Tai, T.H. Lee, S.H. Chiang, J. Y. Tsai, D. N. Yang, Y. H. Wu, Y. H. Chan, On Recommendation of Graph Mining Algorithms for Different Data, *2016 International Conference on Big Data and Smart Computing*, Hong Kong, China, 2016, pp. 357-360.

[80] C. Jiang, Frequent Subgraph Mining Algorithms on Weighted Graphs, Ph.D. Thesis, University of Liverpool, Liverpool, UK, 2011.

Biographies

Saif Ur Rehman is currently an assistant professor in UIIT, PMAS Arid Agriculture University, Rawalpindi, Pakistan. He received his MCS degree with distinction from Institute of Computing and Information Technology, Gomal University, DIKhan, Pakistan in 2005 and MS degree from SZABIST, Islamabad, Pakistan. Currently, he is working towards his PhD (CS) degree in Abasyn University, Islamabad, Pakistan. His research interests include data mining, graph mining, social graph analysis and big data analytics.

Sohail Asghar is working as a *Professor of Computer Science* at COMSATS Institute of Information Technology Islamabad. In 1994, he graduated with honors in Computer Science from the University of Wales, United Kingdom. He received his PhD from Faculty of Information Technology at Monash University, Melbourne Australia in 2006. Dr. Sohail has taught and researched in Data Mining and is a member of ACS, and IEEE. http://ww3.comsats.edu.pk/faculty/Faculty Details.aspx?Uid=4564