FP-Tree based Association Rule Mining in Academic Social Networks to Refine RDF Framework and FOAF

K. Sobha Rani
MVGR College of Engineering
Vizianagaram
Andhra Pradesh

KVSVN Raju
ANITS
Bheemunipatnam
Visakhapatnam

V. Valli Kumari
AU College of Engineering
Andhra University
Visakhapatnam

ABSTRACT

In order to analyze large scale social networks, different strategies are being implemented. The traditional methods of data mining are getting transformed to be suitable to the requirements of the web based information available in different structured and unstructured formats. The process of web mining, a versatile methodology of data mining, involves modified mining techniques applied on documents spread across the World Wide Web. In this proposed work, the process of association rule mining is applied to Academic Social Networks, an offshoot of Social networks. This technique involves processing of profile pages of members stored in FOAF format and retrieving the association rules from the data to identify the strength of relationship amongst researchers. The RDF vocabulary, a standard format of web based identity representation is the building block of this entire framework. Traditional applications of association rule mining is referred in the context of Market Basket analysis, and its application is projected to analyze dense network and derive association patterns amongst members of the academic social network.

General Terms

Data Mining, Social Networks

Keywords

Academic Social Networks, RDF, FOAF, Association Rule Mining, FP-Tree

1. INTRODUCTION

1.1 Academic Social Networks

Social networks are built upon social relations among people who share common interests, activities etc. Social networking sites represent members through profile pages which contain their personal information like homepage, fields of interest, hobbies, contact details etc. However, the information is hidden in heterogeneous and distributed web pages. Many social networking sites offer search and mining services, by allowing a person to register and identify ties based on profile. Such networks can be briefly classified as informational, professional, educational, academic, news groups, sports based and so on. These networks can be visualized using graphs where users represent nodes and their relation represents edges. Social network analysis will be made simpler with the graphical representation and graph metrics would be useful to measure the dynamics of the network and individual properties of nodes.

Academic social networks represent collaboration of researchers and their publications. Different academic social networks include academia.edu, arnetminer.org, academic.research.microsoft.com, researchgate.net etc. It is observed that collaborative effort by people across the globe makes research strong from different perspectives. This proposed approach is based upon services offered by http://arnetminer.org[23] which concentrates on accurately extracting researcher profile information from the web by integrating data from different sources. Coauthor path and coauthor graph presents visually the relation of co-authorship between researchers.

1.2 RDF Specification

Resource Description Framework (RDF) [16], [20] is a W3C standard for describing resources. The necessity of Machine-understandable data from Machine-readable data has lead to the specification of RDF framework. It is a foundation for processing metadata, provides interoperability between applications that exchange machine understandable information on the web. RDF is a model for representing named properties and their values. In object oriented terminology, resources correspond to objects and properties correspond to instance variables. In this context people i.e. members of social network are referred as the resource. In RDF, the *resource* is identified by an URI and is represented as ellipse. Resources have *properties* which in turn have *values*. RDF graph has nodes and arcs that connect nodes and the RDF triple comprises subject, predicate and object nodes. Concrete syntax needed to create and exchange metadata is developed by using Extensible Markup Language, XML. RDF also requires namespace facility to precisely associate property with the schema that defines the property.

Consider a simple example of the sentence: *The individual whose name is Ora Lassila, email <lassila@w3.org>, is the creator of http://www.w3.org/Home/Lassila.* This can be represented in general as "*<subject> HAS <predicate> <object>*" and can be depicted as an RDF graph as shown in Figure 1.

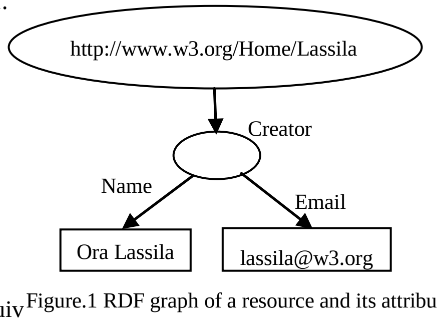

Figure.1 RDF graph of a resource and its attributes

The equiv graph would be as specified below using the namespace, properties and values. This representation is both human and machine understandable and is a standard for efficient data exchange.

```
<rdf:RDF>

<rdf:Description
about="http://www.w3.org/Home/Lassila">

<s:Creator
rdf:resource="http://www.w3.org/staffId/8
574"/>

</rdf:Description>

<rdf:Description
about="http://www.w3.org/staffId/8574">

<v:Name>Ora Lassila</v:Name>

<v:Email>lassila@w3.org</v:Email>

</rdf:Description>

</rdf:RDF>
```

1.3 FOAF Representation of a Person

"Friend of A Friend"(FOAF) [18] project is based around the use of machine readable web home pages for people, groups, companies and others. FOAF is a linked information system that is built using decentralized semantic web technology designed to allow integration of data across variety of applications, web services and software systems. FOAF vocabulary is defined as a dictionary of named properties and classes using W3C's RDF technology. FOAF integrates three kinds of networks; a) *social networks* b) *representational networks* and c) *information networks*. FOAF provides authoritative documentation of contents, status and purpose of RDF/XML vocabulary and are published as linked documents in the web. Main FOAF terms are grouped as Core, Social Web and Linked Data utilities. The Core terms describe characteristics of people and social groups that are independent of time and technology. FOAF defines classes of foaf:Person, foaf:Document, foaf:Image etc. and properties such as foaf:name, foaf:mbox , foaf:homepage etc. A member of a social network is represented with basic RDF template as given below.

```
<foaf:Person rdf:about ="#danbri"
xmlns:foaf="http://xmlns.com/foaf/0.1/">

<foaf:name>Dan Brickley</foaf:name>

<foaf:homepage
rdf:resource="http://danbri.org/"/>

<foaf:openid
rdf:resource="http://danbri.org/" />

<foaf:img rdf:resource="/images/me.jpg"/>

</foaf:Person>
```

1.4 Association Rule Mining – Generation of FP-tree

Association rule mining is an important data mining task that is used to discover frequent patterns that appear together in a large database. The major projection of this process is described in the context of market-basket analysis, but is also applicable to social network analysis to identify people who collaborate and document clustering based on similarity of terms and so on. Semi-structured and multidimensional data of social network makes the data mining process complex. Social interactions can be depicted effectively by the predictive and descriptive patterns that are derived by the task of data mining. The basic search of social network i.e. the thread behind solving research issues like Link Prediction, Community Discovery, Expert Finding is implemented through Association Rule Mining.

Most of resent research works like [1], [2], [13] preferred FP-Tree based approach than traditional apriori methods. Apriori traverses the graph in Breadth First Search (BFS) and generates all possible combinations of association, where as FP growth algorithm traverses the graph in Depth First Search(DFS) and only grows patterns that are frequent. FP tree is advantageous as it is highly condensed and need less database scans. Hence, this analysis is based on FP-Tree generation to identify frequent patterns of peoples' association.

As per the first definition given by Agrawal and Srikant [13] in 1993, *"Given a set of distinct items I and a set of transactions D, where any transaction T from D contains only items from I, an association rule R is an implication "X to Y," where X and Y are unrelated items from I. The rule R has the support s in D if s% of transactions in D contains both items X and Y, and it has the confidence c if c% of transactions in D that contain X also contain Y."*

In the context of academic social network analysis, association rules can be redefined by considering publications and authors as follows:

Definition 1: *"Given a set of distinct authors A and a set of publications P, where any publication p from P contains only authors from A, an association rule R is an implication "X to Y", where X and Y are unrelated authors from A. The rule R has support s in P if s% of publications in P contains both authors X and Y, and it has the confidence c if c% of publications in P that contain X also contain Y."*

Let P= set of publications created by set of authors A and is represented as a combination of co-authors, title of the publication and the conference/journal in which it was published.

P= $\{A_p \subseteq A$, "Title", "Conf/Journal"$\}$

Here A is the set of authors whose profile pages are published in the social network.

Each publication P_i contains a subset of items from the set A. Set of items in the transaction is termed as an Itemset. Rules are generated based on the frequently occurring itemsets. The quality of rule is measured by *support* and *confidence*. The items that satisfy the minimum support and confidence threshold (set by us and is domain specific) are termed as frequent itemsets. If A_1 and A_2 are two authors,

Support of $A_1(Sup_{A1})$ =

$$\frac{\#publications\ by\ A1}{Total\ \#publications}$$

Confidence of $A_1 \rightarrow A_2(Conf_{A1\text{-}A2})$ =

$$\frac{\#publications\ by\ both\ A1\ and\ A2}{\#publications\ by\ A1}$$

FP-Tree construction process involves two stages of activities. First stage performs scanning the database and identifying the frequent itemsets. Then infrequent items are pruned and frequent items are sorted in descending order of their frequency. Then in the second stage, FP-tree is constructed by traversing the transactions and adding branches accordingly. The frequently collaborated co-authors are retrieved by this FP-Tree generation. This method also projects most influencing authors in the academic social network.

1.5 API to Process RDF

Jena is a Java API [21] which can be used to create and manipulate RDF graphs. Jena framework includes API to process RDF data, ontology for handling OWL [17], the Web Ontology Language, inference engine for reasoning data sources, and a query engine compliant with SPARQL the protocol for RDF specification. Jena has object classes to represent graphs, resources, properties and literals. The interfaces representing resources, properties and literals are called Resource, Property and Literal respectively. In Jena, a graph is called a model and is represented by the Model interface.

The syntax for creating a model, resource and properties is specified below

```
// create an empty Model

Model model =
ModelFactory.createDefaultModel();

// create the resource

Resource johnSmith =
model.createResource("http://somewhere/Jo
hnSmith");

// add the property

 johnSmith.addProperty(VCARD.FN, "John
Smith");
```

The John Smith resource is then created and a property added to it. The property is provided by a "constant" class VCARD which holds objects representing all the definitions in the VCARD schema. Jena provides constant classes for other well known schemas, such as RDF and RDF schema themselves, Dublin Core and OWL.

As described in Section 1.2, nodes and arcs of RDF model are represented as triples containing *subject, predicate* and *object*. Each arc in RDF model is called a statement that asserts a fact about a resource. Jena API also provides interfaces to read and write RDF documents and also its XML equivalents.

The key RDF package that can be used by is `com.hp.hpl.jena.rdf.model`. The API has been defined in terms of interfaces so that application code can work with different implementations without change. This package contains interfaces for representing models, resources, properties, literals, statements and all the other key concepts of RDF, and a ModelFactory for creating models. This framework is used to process the FOAF documents of persons i.e. members of academic social network retrieved through the profile pages of the domain http://arnetminer.org[9] and to present their association.

2. BACKGROUND AND RELATED WORK

Visualization of an academic research community by Jie Tang et al.[9] through Arnetminer offers expert, topic and conference search whose data is collected from different data sources. The dataset used in this proposed analysis is downloaded from [3] which provide FOAF documents representing an author profile. Unified approach of social search and discovery offered by Einat et al.[10] suits the requirements of web 2.0 and large enterprise applications. Social ranking proposed by them deal with ranking of all entities retrieved by social search engine. Basic classification as discussed by Ting [15] has classified the task as web content mining, structure mining and Web usage mining. Web people search proposed by Dmitri et al.[11] uses connection analysis for searching web pages related to a person. The retrieved results for a query are clustered based on the relevance to the search query.

Han et al.[1] have proposed FP-Tree that mines frequent patterns without candidate generation, where complex task of candidate generation is eliminated and large datasets are compressed. They also have proposed that this technique outperforms the Apriori technique in terms of time and memory constraints. Fast algorithms for AR Mining proposed by Rakesh Agarwal et al.[13] and Gosta et al.[14] have shown effective solutions for FP tree construction and AR mining.

Takama et al.[2] have proposed Association Rule Mining based adaptive search engine that uses RDF documents to observe users’ information retrieval behavior and retrieve rules from the patterns. Moradi et al.[4] have presented analysis on XML-adopted and XML-specific association rule mining. They further classified the mining techniques as semantic based, apriori based etc. The issue of XML documents having irregular and complex structure can be resolved by using FOAF documents which follow RDF framework.

Sleeman et al.[5] have developed a system that identifies co-referent FOAF documents by using logical constraints and other heuristics. SVM based classification and clustering techniques were used to identify co-referent documents. Yutaka et al [6] have devised methods to compute trust amongst members of an academic community based on FOAF documents. They identified edge labels of the social network graph with relations like co-authors etc.

Andrea et al.[7] have presented an XML-based middleware language and system that supports the KDD process known as KDDML. They presented the solution for requirements like data preprocessing and querying using KDDML. Yale and Fleximine are some other similar frameworks. Semantic relations from textual web content mining proposed by Tao et al.[8], GP-Close helps to identify underlying association patterns of RDF metadata. Optimized search offered by Sonia et al.[12] offer graph based search using identity tags of social relations. Ontology extraction from web pages proposed by Peter Mika [22] emphasizes semantic based analysis on social networks. Mining static and dynamic XML documents and deriving association rules and clustering have been presented in the article of IBM developer works [26].

Rest of the paper is organized as follows; Section 3 discusses the problem and Section 4 provides the system architecture proposed to solve the issue. Experimentation and results are provided in Section 5 and the paper is concluded by providing future scope in Section 6.

3. PROBLEM DESCRIPTION

Academic social networks like arnetminer [23], Microsoft research [24] etc. project network of authors and publications based on the research domains. Graphical representation of such networks like social graph, co-author graph and egocentric graph will present authors along with their co-authors, research interests and other profile information. The information retrieved from different sources is integrated and represented using an FOAF document as a link to the profile page. Sample FOAF document elements for an author from [23] is shown here.

```
<rdf:RDF

xmlns:rdf="http://www.w3.org/1999/02/22-
rdf-syntax-ns#"

xmlns:foaf="http://xmlns.com/foaf/0.1/">

<foaf:Person rdf:ID="me">

<foaf:name>Zhengping Jin</foaf:name>

</foaf:Person>

<foaf:knows>  <foaf:Person>

       <foaf:name>Wenmin Li</foaf:name>

       <foaf:homepage rdf:resource =
"http://arnetminer.org/person/wenmin-li-
644700.html"/>

</foaf:Person>  </foaf:knows>

<foaf:publications rdf:resource="Haiyan
Sun,Qiaoyan Wen,Hua Zhang,Zhengping

 Jin,Wenmin Li: Cryptanalysis and
improvement of two certificateless three-
party authenticated key agreement
protocols:CoRR: -1--1"/>

</rdf:RDF>
```

The FOAF document shown above contain elements like <foaf:Person>, <foaf:knows>, <foaf:publications> and <foaf:homepage> etc. which are defined in FOAF standard vocabulary[18] that describe the properties of every member of the network. <foaf:knows> elements gives details of the person who is known to the current referring author and <foaf:publications> gives the details of publications of the author.

Due to the vast number of members and their publications, it is difficult and time consuming to retrieve the details of collaboration made by an author in a specific domain. Hence, there is a need of analysis which highlights the collaboration amongst members of the network. The association derived will also identify the active members of the network and also the trends of research in different domains. This process could even lead to trust computation which is another important research issue. This study is confined to identify association amongst members of academic social networks. The identification of frequently collaborating co-authors will help one to achieve concrete opinions about collaborative effort made by the members. Present domain offers only quantitative analysis of publication count but not exactly studying the collaborative behavior of the members which is the missing component. The proposed perspective of solving this issue is an attempt to identify the underlying hidden patterns that will benefit large scale social network analysis.

4. ARCHITECTURE OF ARMinASN

4.1 The Pre-Process

This proposed approach of Association Rule Mining in Academic Social Networks is completely dependent on frequent patterns paradigm, as apriori based methods need to generate huge number of candidate sets and need to perform multiple database scans. In case of social network data, it is huge and contains very different patterns when compared to traditional market basket data. The basic strategy of divide-and-conquer makes this more suitable to this analysis.

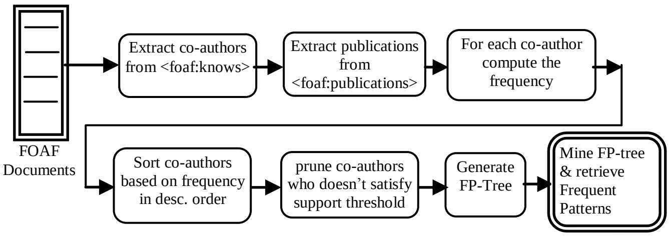
FOAF
Documents
Extract co-authors from <foaf:knows>
Extract publications from <foaf:publications>
For each co-author compute the frequency
Sort co-authors based on frequency in desc. order
prune co-authors who doesn't satisfy support threshold
Generate FP-Tree
Mine FP-tree & retrieve Frequent Patterns

Figure. 2 Architecture for Generation of Association Rules

The entire architecture can be sub divided into two phases

Phase 1: Constructing FP Tree → involves scanning the FOAF documents and identifying the frequency of each author and to build a tree with authors who satisfy the minimum support threshold.

Phase 2: Mining the FP Tree → involves generation of frequent patterns.

Phase 1 needs more emphasis, as it deals with scanning lengthy documents and parsing the required elements. Two different approaches of horizontal and vertical format of data representation are followed here to simplify this task of data representation. In order to parse an RDF document, XML parsers are used that extract required elements from the FOAF documents.

In the first two stages of this architecture shown in Figure 2, XQuery[25] is used to retrieve the elements and form XML documents containing authors' names and publications. Data modeling and expression handling as specified in [19] gives a clear process model for XML documents. Data types, operators and functions provided by the W3 standard of XQuery give the flexibility for efficient handling of XML documents. In order to retrieve co-authors' names, value from <foaf:knows> → <foaf:Person> → <foaf:name> tags are extracted. To retrieve publications, rdf:resource attribute from <foaf:publications> is extracted and tokenized to extract author's names. The sample syntax that retrieved 585 publications and 470 authors from single FOAF page is given below:

```
/rdf:RDF/foaf:Person/foaf:knows/foaf:Person/foaf:name

/rdf:RDF/foaf:Person/foaf:publications/@rdf:resource
```

4.2 Data Representation

The process can further be simplified by using bit vectors to represent the author-publication details as shown below. Bit vector length is equal to the #publications and the bit corresponding to the author is set to 1 in the vector. Simple "AND" is enough to mine the frequent item sets.

Horizontal Data Format	**Vertical Data Format**
{ P_{id} : A_i,A_j..A_n }	{A_{id}: P_i,P_j,..P_m}

Where P_{id} specifies the ID of publications and A_{id} specifies the Author ID along with his publications. In the horizontal data format authors' ID are mapped against publication ID and in vertical data format publications' ID are mapped against author's ID.

Example Bit Vector Representation:

	P1	P2	P3	P4	P5	P	P7	P8	P9	P1	Σ
A1	1	0	0	1	1	0	1	0	0	1	5
A2	1	0	1	1	0	1	0	1	1	1	7
A3	0	1	0	0	0	0	0	1	0	0	2

The term frequency of each author from authors' XML document is updated with the frequency count that is verified from the XML document of publications. The authors' data is then sorted in descending order of frequency and list is pruned based on the predefined minimum support threshold. Now FP-Tree is generated by parsing the resultant XML document as specified in Algorithm in section 4.1.

Some publications extracted from a FOAF document of one author are listed below. The publication details are rewritten for simplicity and anonymity using ID instead of names.

Pub. ID	Authors	Authors list sorted by freq.
P1 →	N1, N10, N2, N3, N4	→N3, N10, N2, N4, N1
P2→	N5, N10, N3, N2, N6	→N3, N10, N2, N5, N6
P3→	N4, N10, N7, N3	→N3, N10, N4, N7
P4→	N13, N10, N9, N3	→N3, N10, N9, N19
P5→	N8, N9, N3, N2, N10	→N3, N10, N2, N8, N9
P6→	N4, N10, N7, N3, N11	→N3, N10, N4, N7, N11
P7→	N3, N10	→N3, N10
P8→	N8, N9, N3, N10, N2	→N3, N10, N2, N8, N9
P9→	N3, N10, N12	→N3, N10, N12
P10→	N3, N10, N2	→ N3, N10, N2
P11→	N3, N14, N12, N10	→ N3, N10, N12, N14

Table 1a. Header Table

Id	Node	Freq.
N1	Haiyan Sun	1
N2	**Hua Zhang**	**5**
N3	**Zhengping Jin**	**11**
N4	**Wenmin Li**	**3**
N5	Lin Cheng	1
N6	Liming Zhou	1
N7	**Qi Su**	**2**
N8	**Min Zhang**	**2**
N9	**Jie Zhang**	**3**
N10	**Qiao Yan Wen**	**11**
N11	Yanjiong Wang	1
N12	**Hongzhen Du**	**2**
N13	Lu Zhao	1
N14	Huijuan Zuo	1

Table 1b. Frequent Items sorted descending

Id	**Node**	**Freq.**
N3	**Zhengping Jin**	**11**
N10	**Qiao Yan Wen**	**11**
N2	**Hua Zhang**	**5**
N4	**Wenmin Li**	**3**
N9	**Jie Zhang**	**3**
N7	**Qi Su**	**2**
N8	**Min Zhang**	**2**
N12	**Hongzhen Du**	**2**

From the publication details, the frequency of authors is counted. If minimum support threshold is set as 2, then the nodes whose frequency is less than the value are pruned and tree is generated for the remaining nodes. The frequency count is shown in Table 1, for which the FP-Tree is shown in Figure 3. Once, the FP-tree is generated, the next phase is to mine the FP-Tree in order to derive Association rules i.e. identify frequent patterns. For simplicity, the FP-Tree of a single FOAF document is projected here. Pruning is done by considering 2% number of authors as the minimum support**.**

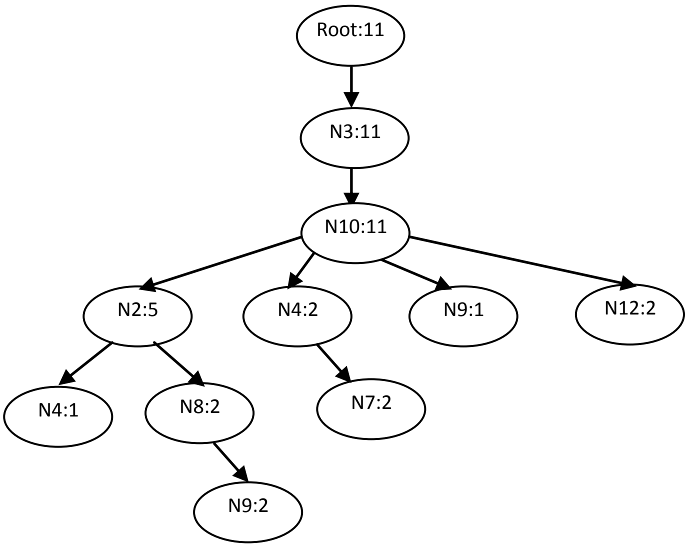

Figure 3. FP-Tree corresponding to data in Table 1.b

From the above process, it is observed that level-1 item set is as shown in table and level-2 item set is {N3,N10} because their support is 100% in this example. These statistics are based on the local data and hence, the support and confidence measures are applicable to purely local dataset. When the experimentation is expanded to larger level of documents and to the large scale social networks, the statistics differ from the current values.

4.3 The Algorithm: FP-Growth

Input: The Collection of FOAF Documents

Output: The Set of Frequent Co-authors

Phase -1, FP-Tree construction

Scan the FOAF document, D, parse and retrieve A- the set of authors and PA- the author's names from publications.

a. For each A, compute frequency count by scanning the set of PA

b. Sort A in descending order of support count and refer it as FA, set of frequent authors.

c. Create root of FP-Tree, label it as null. For each publication in PA, sort the author's names as per the order of FA. Let '*f*' be the first name and F be the remaining list. Call insert_tree([*f*|F], T) as specified here. If T has child N such that N.author-name=*f*.author-name, then increment N's count by 1. Else create new node N and let it's count be 1, its parent linked to T and its node-link to the nodes with the same item-name via the node-link structure. If F is non-empty, call insert_tree(F,A) recursively.

Phase -2, FP-Tree mining

```
Procedure FP_growth(Tree,$\alpha$)
a.  If Tree contains a single path P then
b.  For each combination $\beta$ of the nodes in the path P
c.  Generate pattern $\beta \cup \alpha$ with support_count = minimum support count of nodes in $\beta$
d.  else for each $a_i$ in the header of Tree {
e.  Generate pattern $\beta = a_i \cup \alpha$with support_count = $a_i$.support_count;
f.  Construct $\beta$'s conditional pattern base and then $\beta$'s conditional FP_Tree$\beta$;
g.  If Tree$_\beta \neq \phi$ then call FP_growth(Tree$_\beta$, $\beta$); }
```

5. THE EXPERIMENTATION TO REFINE FOAF DOCUMENT

Depending on the frequent co-authors identified in previous phases, the original FOAF document is modified with the Jena API as described here. New properties of support and confidence are added to the resource <foaf:knows> and their list is re-written in sorted order of support for efficient traversal of the documents. The sample source code to add property and the resultant FOAF document part is shown below. The arguments are given as static for simplicity, but are retrieved dynamically from different file streams.

Sample Source Code

```
String inputFileName = “id14539.rdf”;

String xyzURI =
“http://aminer.org/person/xyz14539”;

Model model =
ModelFactory.createDefaultModel();

InputStream in =
FileManager.get().open(inputFileName);

model.read(new InputStreamReader(in),
                                    "");

Resource vcard =
model.getResource(xyzURI);

vcard.addProperty(support, 0.6);

vcard.addProperty(confidence, 0.8);
```

Refined FOAF Document

```
<foaf:knows>

<foaf:Person>

        <foaf:name>xyz14539</foaf:name>

        <foaf:homepage rdf:resource =
"http://aminer.org/person/xyz14539.htm"/>

<foaf:support>0.6</foaf:support>

<foaf:confidence>0.8</foaf:confidence>

</foaf:Person>

</foaf:knows>
```

As an additional task, the derived association rules are stored as new metadata tags in an RDF document that can be used for statistical analysis. The search engine can use this meta-data to filter out inactive authors and identify strongly connected members of the academic social network.

The experiments were held on the dataset of FOAF documents released by http://www.arnetminer.org[3][23] whose size after extraction is 4GB. The basic analysis started with FOAF documents of each individual taken from the profile pages of each member of the network. Once processing and retrieval of meaningful results is done, those are applied to be tested on the entire dataset and then final rules are retrieved. The frequent patterns derived are again randomly checked with the result of Weka for cross validation.

If Data Mining is chosen as the domain of interest, then the authors worked in the domain are 8805 who contributed to a total publication count of 16304 in 1407 journals/conferences. In the domain of Social Network, the statistics are 9768, 8719 and 1362 respectively. This algorithm is applied by selecting one random author and all his co-authors' profiles in each domain. The selected FOAF documents varied in size from 2KB to 174KB. When the author's names are merged into a single document, they come to 5,132KB and their distinct research interest constituted 5,189KB of memory.

When FP-tree algorithm is applied on a single document of 62KB, 14 frequent patterns of 2-itemset and 5 frequent patterns of 3-itemset are retrieved when the minimum support threshold is set as 30% and minimum confidence threshold is set as 80%. These statistics widely varied to each document because of their own differences of collaborative effort. Hence a global FP-tree mining is performed on the combined document and results of network statistics are computed.

6. CONCLUSION

The process of deriving association rules amongst members of an academic social network demonstrated here can be applicable to any social network where the members have their profile stored in XML format or any other similar standard structured format. This method can still be extended to perform multi-level association rule mining to identify association of authors in a specific research domain. In this proposal, one technique of data mining i.e. association rule mining is implemented, however other techniques of classification and clustering can also be performed on the pre-processed data for analysis on the social networks which is one of the interesting domains of research. The use of RDF, a standard and structured framework made this process as a benchmark that can be used for other data mining tasks in future. The extension proposed to FOAF vocabulary will reduce lot of processing time during information retrieval and project social network at different required levels of abstraction based on research domains.

6. REFERENCES

[1] Han, Jiawei et al., "Mining frequent patterns without candidate generation: A frequent-pattern tree approach", Data mining and knowledge discovery 8.1 (2004): 53-87. ACM 2000.

[2] Takama, Yasufumi and Shunichi Hattori, "Mining association rules for adaptive search engine based on RDF technology.", Industrial Electronics, IEEE Trans. on 54.2 (2007): 790-796.

[3] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang and Zhong Su, "ArnetMiner: Extraction and Mining of Academic Social Networks.", in Proc. of the 14th ACM SIGKDD Intnl Conf. on Knowledge Discovery and Data Mining (SIGKDD'2008). pp.990-998.

[4] Moradi, Mohammad, and Mohammad Reza Keyvanpour. "An analytical review of XML association rules mining." Artificial Intelligence Review (2013): 1-24. DOI 10.1007/s10462-012-9376-5, Springer.

[5] Sleeman, Jennifer, and Tim Finin. "Computing foaf co-reference relations with rules and machine learning." Proc. 3rd Intnl workshop on social data on the web. 2010.

[6] Y. Matsuo, H. Tomobe, K. Hasida, M. Ishizuka, "Finding social network for trust calculation", pp. 510–514 Proc. of 16th European Conference on Artificial Intelligence (ECAI2004)

[7] Romei, Andrea, Salvatore Ruggieri, and Franco Turini. "KDDML: a middleware language and system for knowledge discovery in databases." Data & Knowledge Engineering 57.2 (2006): 179-220, 2005 Elsevier

[8] Jiang, Tao, Ah-Hwee Tan, and Ke Wang. "Mining generalized associations of semantic relations from textual web content." Knowledge and Data Engineering, IEEE Trans. on 19.2 (2007): 164-179.

[9] Tang, Jie, et al. "Arnetminer: An expertise oriented search system for web community." Semantic Web Challenge (2007).

[10] Amitay, Einat, et al. "Social search and discovery using a unified approach."Proc. of the 20th ACM conf. on Hypertext and hypermedia. ACM, 2009.

[11] Kalashnikov, Dmitri V., et al. "Web people search via connection analysis."Knowledge and Data Engineering, IEEE Transactions on 20.11 (2008): 1550-1565.

[12]Lajmi, Sonia, et al. "Extended Social Tags: Identity Tags Meet Social Networks." Computational Science and Engineering, 2009. CSE'09. Intnl Conf. Vol. 4. IEEE, 2009.

[13]Agrawal, Rakesh, and Ramakrishnan Srikant. "Fast algorithms for mining association rules." Proc. 20th Int. Conf. Very Large Data Bases, VLDB. Vol. 1215. 1994.

[14] Grahne, Gosta, and Jianfei Zhu. "Fast algorithms for frequent itemset mining using fp-trees." Knowledge and Data Engineering, IEEE Trans. on 17.10 (2005): 1347-1362.

[15] Ting, I-Hsien. "Web mining techniques for on-line social networks analysis."Service Systems and Service Management, 2008 Intnl Conf on. IEEE, 2008.

[16] http://www.w3.org/TR/REC-rdf-syntax/

[17] http://www.w3.org/TR/owl-semantics/

[18] http://xmlns.com/foaf/spec/

[19] http://www.w3.org/TR/xquery/

[20] http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/

[21] http://jena.apache.org/index.html

[22] Mika, Peter. "Ontologies are us: A unified model of social networks and semantics.", Web Semantics: Science, Services and Agents on the World Wide Web 5.1 (2007): 5-15.

[23] http://arnetminer.org

[24] http://www.academic.research.microsoft.com

[25] Jacky W.W.Wan, Gillian Dobbie, "Extracting Association Rules from XML Documents using XQuery", WIDM'2003, Nov 2003, New Orleans, Louisiana, USA.

[26] http://www.ibm.com/developerworks/library/x-datamine2/index.html