Extracting Community Cores in Heterogeneous Citation Network

Zin Mar Yin, Soe Soe Khaing
University of Technology, Yatanarpon Cyber City
zinmaryinn@gmail.com, khaingss@gmail.com

Abstract

The identification of closely connected cores in a complex network has become an important feature in the area of community mining in recent years. According to the network topology, it is true that a large and complex graph have a set of densely connected sub-graphs as the cores. Extraction these cores can reveal that some unexpected connection patterns between nodes in this type of network. In this paper, a simple and efficient intersection-based algorithm is proposed for finding community cores in multi-relational citation network by using the strength of modularity-based Louvain method. The proposed system is applied on a part of Citeseer^X digital library [5] as the real-world data set of heterogeneous citation graph. Graph nodes represent individual papers of Citeseer^X data set which are linked by three types of above relationships. The experimental results show that the proposed algorithm is highly effective in core extraction with comparable time complexity.

Keywords: community mining, community core, citation network, Citeseer^X data set

1. Introduction

The most effective representation of the interconnections of real world objects is the graph-based network. For example, in the World Wide Web, node represents a single page and edge as a hyperlink. Others are social networks, food webs, metabolic structures, etc. The mathematical formulation and topological investigation of the complex network have a significant benefit to reveal some unexpected relationships of a particular object or a group of them. As crime detection, the chain of offences can be easily detected from a closed cluster of connected persons.

Graph mining methods naturally divides the network of interest into sub-graphs or sub-clusters and the analysis point of view is to investigate those groups for intended purposes. Most of the early works on graph partition methods were intended to clusters the corresponding graph into disjoint or non-overlapping communities [4, 16]. However, to identify well-defined communities in graph clustering, it is realized that an individual object may belong to multiple communities at the same time, and is likely to have more relationships to other objects outside of its community than inside. For example, in a student social network, a node becomes an individual student and the relationships between the students become a set of edges, such as same interest of study, interest and hobby, native city or country, religions and culture, etc.

For the point of view of discovering overlapping communities, many methods have also been proposed in the literature. Unfortunately, they have the limitations in the knowledge of global network topology, strict parameters to be processed and the time consuming for large scale networks [15]. These limitations point out that the requirement of efficient algorithm for finding community cores in large and complex network with reasonable time complexity.

In this paper, the extraction of tight-knit nodes in multi-relational set of connections is proposed. Each relationship can be configured as a single relational network. From this homogeneous network structure, we apply the modularity-based graph partition method, the Louvain method [3], in order to cluster a series of sub-graphs. Then, the combination of these resulted sub-clusters to identify the overlapping core structures in the experimental data set is presented.

The paper is organised as follows. In Section 2, a review of the relevant literature in community studying as well as the identification of core structures in different types of networks are presented. As the Section 3, the architecture of the proposed system is discussed in accordance with the theoretical foundation. Proposed dense node clustering algorithm is presented in Section 4. Section 5 presents the implementation in the case-study of Citeseer^X data set and its evaluations. Finally, Section 6 discusses the concluding remarks and the future research avenues.

2. Related Work

A social network is modelled by a graph, where the nodes represent individuals, and an edge between the individuals. Most of the existing methods on community mining assume that there is only one kind of relation in the networks, and the mining results are independent of the users' needs or preferences. In reality, there exist multiple, heterogeneous social networks, each representing a particular kind of relationship is participate a distinct role in a particular task. Multi-relational networks or multiple edge types or heterogeneous networks have been discussed in [6-9].

In contrast, a multi-relational network is a network with a heterogeneous set of edge labels which can represent relationships of various types in a single data structure [8]. Edges may either represent "friendship", "kinship", or "collaboration" but not all of them together [9]. Similarities between two papers can be based on common authors, where they are published, keyword similarity, citations, etc [10].

Finding core structures in various types of networks are also discussed in [10-13]. The idea of identifying a champion of a community by using (α,) Community algorithm is proposed in [10]. For Autonomous Systems in Internet, the discovery of central dense-core in end-to-end routes is implemented by introducing a randomized sublinear algorithm in [11]. The analysis of the virtual communities in Douban.com, a Chinese web site is used and the optimization of the user interests' concentration ratio in user groups is also described in [12].

3. System Architecture

3.1. Decomposing Heterogeneous Network

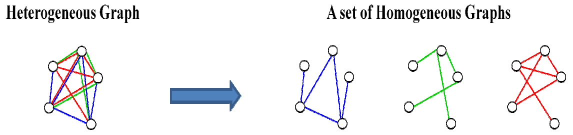

Figure 1.The decomposition of heterogeneous network into a set of homogeneous types

Community mining becomes one of the major trends in graph clustering areas. However, almost all existing community detection algorithms are based on homogeneous type of graph which means edges represent a definition of only one relationship. In reality, an object may have the existence of more than one meaning of connection and each kind of relationship may play a distinct role in a particular task. Therefore, the proposed system has the notion of heterogeneous complex network as a set of homogeneous type of graph, as shown in above.

3.2. Mining Communities in Homogeneous Networks

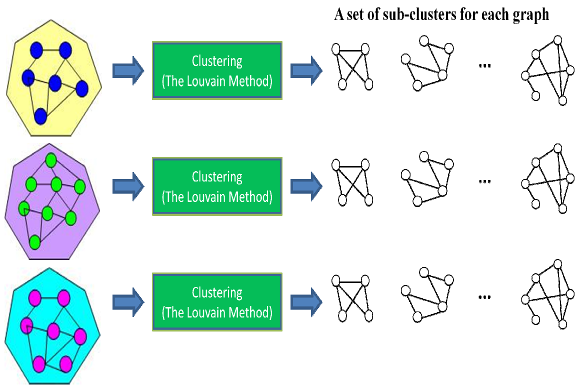

Figure 2. The graph clustering in each single relational network by using the Louvain Method

In the study of complex networks, nodes represent individual objects and edges represent relationships between each pair of objects. A common property of these networks is their community structure which reveals the existence of densely connected groups of vertices, with only sparser connections between groups. This type of graph clustering focuses on the detection and characterization of such network structure.

One of the most widely used methods for community finding is modularity maximization. Modularity is a benefit function that measures the quality of a particular division of a network into communities. The modularity maximization method detects communities by searching over possible divisions of a network for one or more that have particularly high modularity. The modularity Q is defined as follows:

Q = (number of edges within communities) □ (expected number of such edges)

In other words, the modularity Q measures the fraction of the edges in the network that connect vertices of the same type, i.e., within-community edges, minus the expected value of the same quantity in a network with the same community division but with random connections between the vertices [4]. This modularity is defined as

$$Q = \frac{1}{2L}\sum_{i,j}\left[A_{i,j} - \frac{k_i k_j}{2L}\right]\delta(i,j).$$

Here $A_{i,j}$ is the ijth element of the adjacency matrix ($A_{i,j}$ = 1 if a link exists between i and j and (A_{ij} = 0 otherwise), L is the total number of network links, k_i is the degree of node i, and δ(i,j) equals 1 if i and j belong to the same group, and equals 0 otherwise.

For large networks it is computationally impractical to maximize the modularity over all possible partitions of the network and is required to apply the approximation method. The approximation algorithm for optimizing modularity on large networks was proposed by Blondel et al [3], as follows:

$$\Delta Q = \left[\frac{\sum_{in} + k_{i,in}}{2m} - \left(\frac{\sum_{tot} + k_i}{2m}\right)^2\right] - \left[\frac{\sum_{in}}{2m} - \left(\frac{\sum_{tot}}{2m}\right)^2 - \left(\frac{k_i}{2m}\right)^2\right]$$

And this method can also be known as *the Louvain Method* in the community mining environment.

The quality of the partitions resulting from these methods is often measured by the so-called modularity of the partition, which is a scalar value between -1 and 1 that measures the density of links inside communities as compared to links between communities. As the preservation of the degree of each vertex, this method is based on the randomization of the edges in the network. A modularity Q = 0 corresponds to a random network, in which two nodes are connected with probability that is proportional to their respective degrees. The value of Q that is close to 1, which is the maximum, indicates strong community structure.

3.3. Extracting Dense-cores in Heterogeneous Network

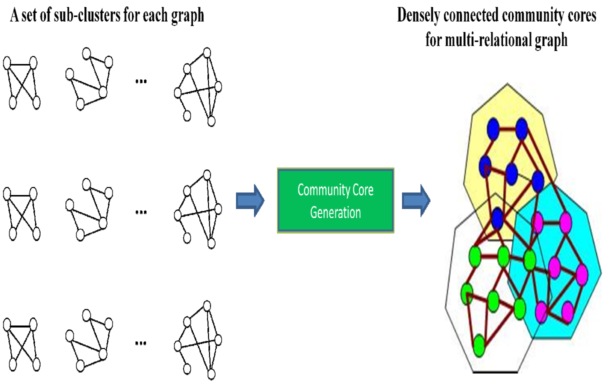
A set of sub-clusters for each graph
Densely connected community cores for multi-relational graph
Community Core Generation

Figure 3. The generation of community cores from a set of sub-clusters in each homogeneous graph

For the consideration of heterogeneous network, Cai et al [9] introduced that there are different types of relationships in real-world network and each relation can be treated as a single relational network. Such kind of complex network can be called multi-relational network or heterogeneous network.

In order to find the closely related community cores in heterogeneous type of network, this paper proposed an efficient dense node clustering algorithm in the portion of community core generation. the time complexity of our algorithm has two unfolds: O (N log N) for the Louvain method, where N is the number of nodes in a network to be clustered by modularization, and O (n^2) for the intersection-based core generation, where n is the number of sub-clusters in each homogeneous graphs as the worse-case. In comparison with other community core extracting algorithms [10-13], the advantages of our algorithm is that it does not depend on the increasing number of nodes in the intended graph. Moreover, our algorithm can be effectively applied in heterogeneous types of graph with less time-consuming with the help of the Louvain method, which is a homogeneous graph clustering technique.

4. Dense Node Clustering Algorithm

An efficient dense node clustering algorithm for extracting the community cores is proposed in this paper and described in figure 4. Firstly, citation data D1, co-citation data D2 and activity bibliography data D3 are analyzed from Citeseer^x websites. D1, D2 and D3 are clustered by using Gephi Toolkit which is applied the Louvain method based on modularity maximization theory. Secondly, Core_gen function is called two times. Intersetdata_1 and Intersetdata_2 are obtained. Thirdly, Core_gen function with two parameters Intersetdata_1 and Intersetdata_2 and community core Resultdata is returned. Core_gen algorithm is described in figure 5. In this method, before finding core data, the parameter is sorted.

```
BEGIN
    GET D1 for Attribute 1;
    GET D2 for Attribute 2;
    GET D3 for Attribute 3;
    Cd1=Cluster D1;
    Cd2=Cluster D2;
    Cd3=Cluster D3;
    ISdata_1= Core_gen (Cd1,Cd2);
    ISdata_2= Core_gen (Cd1,Cd3);
    Resultdata= Core_gen (ISdata_1, ISdata_2);
END
```

Figure 4. Dense Node Clustering Algorithm

```
Core_gen(String[] first,String[] second)
BEGIN
              result = new ArrayList<String>();
              Sort(first); Sort(second);
              int i = 0;
              int j = 0;
             WHILE(i < first.length &&
                                     j < second.length)
                    IF(first[i]<second[j])
                           ++i;
                    ELSEIF (first[i]>second[j])
                           ++j;
                    ELSE
                       IF (!result.contains(first[i]))
                          result.add(first[i]);
                           ++i;
                           ++j;
                       ENDIF
                   ENDIF
              ENDWHILE
              RETURN result;
             }
END
```

Figure 5. Community Core Generation Algorithm

5. Implementation

5.1. Citeseer^X Data Set

Citeseer^x is a scientific literature digital library and search engine that focuses primarily on the literature in computer and information science. Citeseer^x aims to improve the dissemination of scientific literature and to provide improvements in functionality, usability, availability, cost, efficiency, comprehensiveness, and timeliness in the access of scientific and scholarly knowledge. [5] Rather than creating just another digital library, Citeseer^x attempts to provide resources such as algorithms, data, metadata, services, techniques, and software that can be used to promote other digital libraries. Citeseer^x has developed new methods and algorithms to index PostScript and PDF research articles on the Web.

For a network of Citeseer^x, same vertices with different meanings of edges are used in this implementation. Citation graph is constructed with a reference to a published or unpublished source according to the original Citeseer^x citation links, as shown in figure 6.

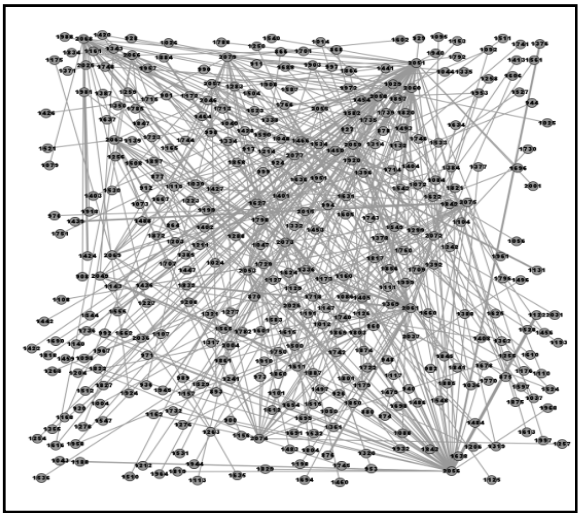

Figure 6. Citation Graph

Co-citation graph is also constructed that two documents are co-cited if they are both independently cited by one or more document.

Active bibliography graph is constructed that tow documents are bibliographically coupled if they both cite one or more documents in common.

Similar to the Citation graph of Citeseer^x, Co-citation graph and Active Bibliography graphs are built to perform the homogeneous graph partition algorithm: the Louvain method which is done the vertices clustering. The sub-clusters of Citation graphs are mentioned by the colors in figure 7. To get the community cores, our proposed efficient dense nodes clustering algorithm, described in section 4, is applied to the series of each homo-graph clusters. The result of community cores is expressed in table 1.

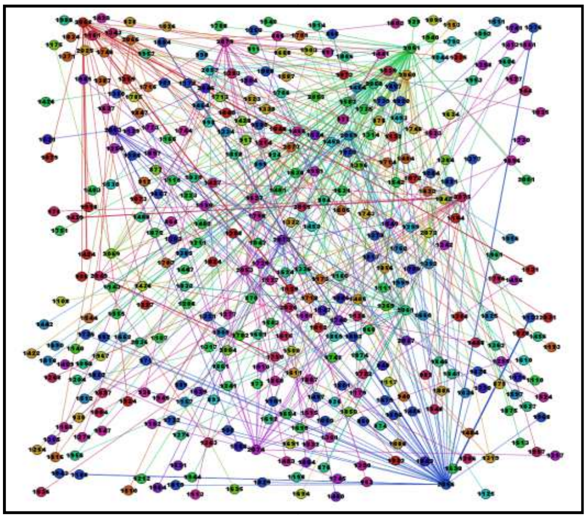

Figure 7. Citation Sub-clusters Graph

5.2. Empirical Evaluation

This section evaluates the proposed technique which is experimented in Citeseer^X data set. As mentioned in Table 2, the experimental results show that the sub-clusters of community cores in the case study data set. They form the well-defined area of paper nodes in the citation graph in the heterogeneous type of network. Therefore, the proposed algorithm is able to identify the community cores in attractive time complexity.

6. Conclusion and Future Works

The proposed system studies the problem of mining community cores in heterogeneous complex network. By exploiting the modularity-based Louvain method, this system finds all the sub-clusters in a set of homogeneous graphs. Then, the efficient dense nodes clustering algorithm is proposed to extracting the community cores taking the advantage of simple intersection-based method. The proposed algorithm is experimented in a portion of real-world Citeseer^X data set. The experimental results show that this algorithm is able to discover the interesting community cores with comparable time complexity.

Communit	**Paper ID**
Core-1	id-1636, id-2051
Core-2	id-1532,id-1533,id-2053, id-2079
Core-3	id-2056, id-900
Core-4	id-1037,id-1204,id-2058,id-2061,id-874, id-924
Core-5	id-1453,id-2057
Core-6	id-1268,id-1997,id-2075

Table 1. Community Core Results

Community	PaperID	Paper Title	**Remark**
Core_1	1268	Discovering user access pattern based on probabilistic latent factor model	Latent Semantic Analysis
	1997	Using Probabilistic Semantic Latent Analysis for Web Page Grouping	
	2075	A Latent Usage Approach for Clustering Web Transaction and Building User	
Core_2	1636	On the Markov equivalence of chain graphs, undirected graphs, and acyclic	Markov Chain
	2051	A Characterization of Markov Equivalence Classes for Acyclic Digraphs	
Core_3	1532	Learning to classify text from labeled and unlabeled documents	Document Classification
	1533	Learning to extract symbolic knowledge from the World Wide Web	
	2053	A Bayesian Approach to Filtering Junk E-Mail	
	2079	A Comparison of Event Models for Naive Bayes Text Classification	
Core_4	2056	A Formal Basis for Architectural Connection	SW architecture
	900	A Formal Approach to Software Architecture	
Core_5	1453	Survey on Independent Component Analysis	ICA
	2057	FastISA: A Fast fixed-point algorithm for Independent Subspace Analysis	
Core_6	1037	An evaluation of statistical approaches to text categorization	Text categorization
	1204	Context-sensitive learning methods for text categorization	
	2058	A comparison of event models for Naive Bayes text classification	
	2061	A Re-Examination of Text Categorization Methods	
	924	A Neural Network Approach to Topic Spotting	

Table 2. The Clusters of Community Cores Result

References

1. http://www.gephi.org
2. Cai, D., Shao, Z., He, X., Yan, X., and Han, J., “Mining Hidden Community in Heterogeneous Social Networks”, Technical report, Computer Science Department, UIU (UIUCDCS-R-2005-2538, May, 2005)
3. Blondel, V. D., Guillaume, J. -L., Lambiotte, R. and Lefebvre, E., “Fast Unfolding of Community Hierarchies in Large Network”, 2008, J. Stat. Mech. P1008.
4. Newman, M. E. J. and Girvan, M. “Finding and Evaluating Community Structure in Networks”, Physical Review E 69, 2004.
5. http://citeseerx.ist.psu.edu/
6. Stroele, V., Zimbrao, G., Souza, J. M., “Modeling, Mining and Analysis of Multi-Relational Scientific Social Network”, Journal of Universal Computer Science, Vol.28, No.8, 2012.
7. Rodriguez, M. A., Shinavier, J.,“Exposing Multi-Relational Networks to Single-Relational Network Analysis Algorithms”, December 9, 2010.
8. Rocklin, M., Pinar, A., “On Clustering on Graphs with Multiple Edge Types”, September 8, 2011.
9. Cai, D., Shao, Z., He, X., Yan, X., and Han, J., “Community Mining from Multi-relational Networks”, PKDD'05, Proc. of 2005 European Conf. on Principles and Practice of Knowledge Discovery in Databases, 2005.
10. Wang, L., Hopcroft, J., He, J., Liang, H. and Suwajanakorn, S., “Extracting the Core Structure of Social Networks using(α , β) Community”, Internet Mathematics Volume 9, Issue 1, 2013.
11. Gonen, M., Ron, D., Weinsberg, U., Wool, A.,”Finding a Dense-Core in Jellyfish graphs”, Computer Networks 52(15):2831–2841, 2008.
12. Chen, H., Cheng, X., Liu, Y., “Finding Core Members in Virtual Communities”, Poster Paper, April 21-25, 2008.
13. Qiao, S., Tang, C., Peng, J., Fan H. and Xiang, Y.,”VCCM Mining: Mining Virtual Community Core Members Based on Gene Expression Programming”, In Intelligence and Security Informatics, Vol. 3917 of Lecture Notes in Computer Science, pages 133-138, Springer, Germany, 2006.
14. Chen, J., Zaiane, O. R. and Goebel, R., “Local Community Identification in Social Networks”, International Conference on Advances in Social Networks Analysis and Mining (ASONAM), Athens, Greece, July 20-22, 2009.
15. Nguyen, N. P., Dinh, T. N., Nguyen, D. T. and Thai, M. T., “Overlapping Community Structures and their Detection on Social Networks”, In Proceedings of SocialCom/PASSAT. 2011, 35-40.
16. Andersen, R., Chung, F. and Lang, K., “Local graph partitioning using Pagerank vectors”, In Proc. 47th IEEE Symp. Found. Comp. Sci. (FOCS), 2006.