Research Community Mining with Topic Identification

Ryutaro Ichise, Hideaki Takeda
Principles of Informatics Research Division,
National Institute of Informatics
2-1-2 Hitotsubashi Chiyoda-ku
Tokyo, 101-8430, Japan
{ichise,takeda}@nii.ac.jp

Taichi Muraki
TriAx Corporation
4-29-3 Yoyogi Shibuya-ku
Tokyo 151-0053, Japan
muraki@triax.jp

Abstract

Since research trends can change dynamically, researchers have to keep up with these new trends and undertake new research topics. Therefore, research communities for new research domains are important. In this paper, we propose a method to discover research communities. The key features of our method are a network model of papers and a word assignment technique for the communities obtained. We show our system based on the proposed method and discuss our system through case studies and experiments.

1 Introduction

As information technologies progress, we can obtain research information faster than before. However, the technologies covering a wide area can change just as rapidly. Therefore, all researchers must not only continuously follow new trends of research but also must investigate new research topics. When we undertake new research topics, we need to know the research communities of researchers with the same research topic or same interest. As a result, we need an effective community mining method for finding them.

In order to find research communities, we usually use bibliography information. The methods include co-citation analysis [13, 3] and bibliographic coupling [8]. Although these methods are very useful for analyzing research topics from the global viewpoint of all bibliography data, we cannot always understand what the discovered communities represent. CiteSeer [12] and Google Scholar [4] are able to handle research communities from a micro viewpoint because they handle the co-author and citation information from the bibliographies and use the information for individual researchers. Although these systems are good for finding local communities involving an author, they are not suitable for finding research communities close to the author. Börner et al. [2] propose the use of co-author networks to find research communities by weighted graphs. Their system uses heuristics to separate communities without interaction. Ichise et al. [5] propose a community mining method based on the interaction of users. Although the proposed system of Ichise et al. supports community mining from both a global view and a local view with several mining indexes, it does not identify the research topics of the communities obtained. In this paper, we propose a method to discover research communities with identified topics.

This paper is organized as follows. In Section 2, we discuss the proposed method for research community mining. In Section 3, we explain the system design of our system and show some case studies of our system. In Section 4, we discuss our system and finally, in Section 5, we present our conclusions.

2 Research Community Mining

2.1 Network Model of a Research Community

Several network models using bibliographies to represent research communities have been proposed [5]. In this paper, we focus on the co-author relationships of a research paper to find the research communities, because co-authorship relations are basic elements in research groups.

Let's start with a simple paper model. We assume the paper model consists of keywords and authors' names. In this case, we can establish an author's research interest or specialty by the keywords in the author's works. Therefore, authors who write a paper collaboratively share the same interest or specialty, as represented by the keywords. If we consider the authors as nodes and the shared interest or specialty as edges labeled by keywords, we can represent the bibliography information as researcher networks.

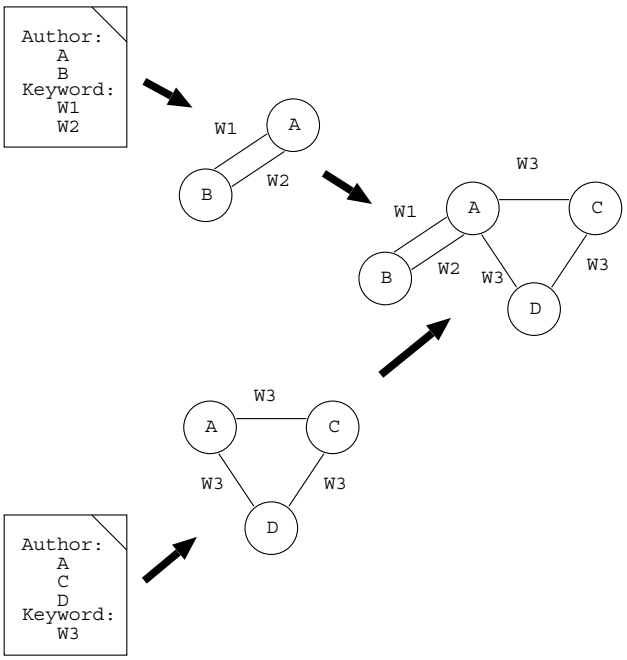

Figure 1. Network model of researchers.

Let us explain our model using an example. Assume that we have two papers, as shown on the left side of Figure 1. One was written by two authors, A and B, and has two keywords, W_1 and W_2. Another was written by three authors, A, C and D, and has the keyword W_3. We can compose graphs of the authors and edges from the two papers, as shown in the middle of Figure 1. Then, the entire graph, which is the joint representation generated from the two papers, is shown on the right of Figure 1.

As one can see, we can obtain a labeled graph from the bibliography data with our modeling. Then, the next question is how to discover research communities from this graph. We define a research community as a cluster which is densely connected by the same research interest or specialty. Therefore, the research communities we want to obtain are clusters which have their edges labeled by the same keywords. Since our network model provides the research specialty on the edges as labels, we can obtain the research communities by eliminating the edges of no interest to the system user. In other words, when the user specifies a research specialty, most of the edges which are not related to the specified research area can be deleted and the communities the user wants to obtain will stand out. This process produces the research communities. For example, when the user specifies W_3 for the networks in Figure 1, the edges labeled W_1 and W_2 will be eliminated. As a result, researcher B is isolated from the graph and we can find the research community consisting of researchers A, C and D.

2.2 Property Assignments for Communities

Since the clusters obtained by our method are only connected by user-specified relationships, we can consider each cluster as a research community. However, each cluster does not have its own property or identification. In other words, if the user does not have enough knowledge about the researchers, the user may not understand the meaning of the communities because there is no information about them. In order to solve this problem, we propose a method of assigning keywords for each obtained community.

In our paper model, the papers written by the authors in each community have keywords. If some words appear often in such papers, we can consider these words as a property for the community. However, if we simply counted the occurrences of the keywords in these papers, the relationships between keywords would be lost. For example, if half of the papers written in the community has keyword X and the other half has keyword Y, the keywords X and Y are not good for the property of the community because it is hard to understand the relationship between X and Y. To avoid this problem, we consider frequent keywords as units of the papers. The algorithm to identify the property of the community is as follows:

1. Select papers which are written by the authors in the community from a paper database.
2. The selected papers are analyzed by the Apriori algorithm [1]. In this process, the keywords in a paper are treated as an item, and the papers are treated as transactions.

As a result, we can obtain keyword pairs for each community. We assign these keyword pairs as the property of the community.

3 Community Mining System

We implemented the proposed method in an actual system using Java. In this section we describe our system design with some examples.

3.1 System Design

Our system consist of three components for community mining. The components are as follows:

- Keyword map view
- Chart view
- Community view

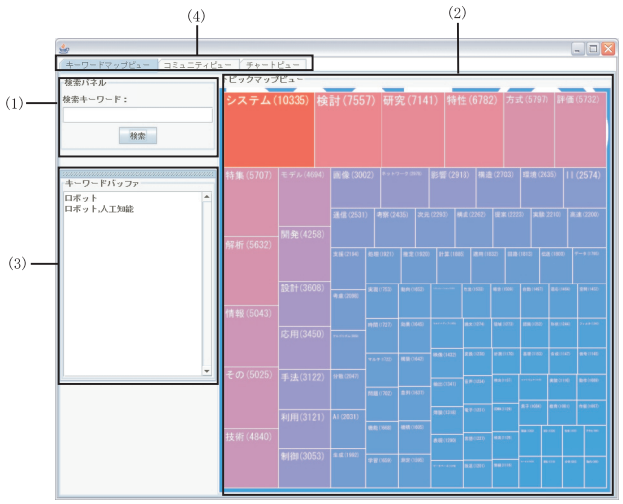

Figure 2. Screen shot of the keyword map view.

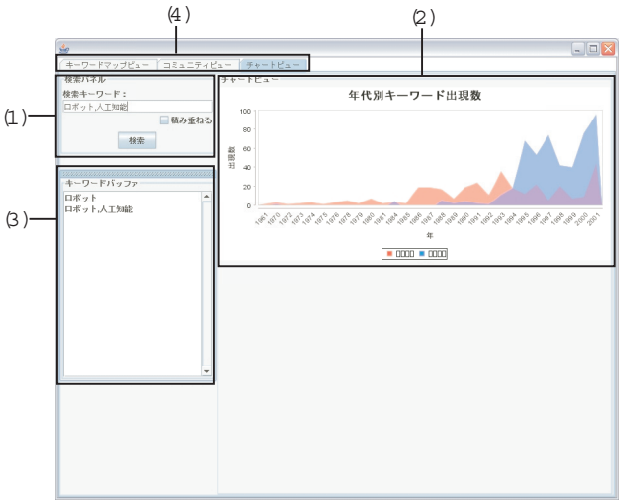

Figure 3. Screen shot of the chart view.

The first component, *keyword map view*, is a support system for users to select an appropriate keyword for starting the system. Figure 2 is a screen shot of the component. The upper left window (1) is used to input a keyword by the user. Then, the related words are shown in the right space (2). The view is inspired from *newsmap* [10]. Each block in the map represents a keyword which is frequently used in papers with the user-specified keyword. The size of the block represents the number of occurrences of the keyword. In other words, the keywords represented with a larger box are more used with the user-specified keywords. When the user clicks on a block, the keyword for the block is set at the search window (1), and the map is repainted for the keywords which are user-specified keyword and a clicked keyword. The bottom left area (3) on the screen is a keyword buffer area, which keeps a history of the searched keywords. The component is constructed using prefuse [11].

The second component, *chart view*, is also a support system for users to select an appropriate keyword. The only difference between the previous component and this is the area on the right. In the keyword map view, the right area represents a current snapshot of the status of the specified keywords. On the other hand, in the chart view, we can see the trend of the keywords. Figure 3 is an example of a screen shot. When we search keywords, we can obtain the keyword trend in right window (2). The vertical axis represents the number of papers which include the keyword, and the horizontal axis represents the publishing year. As a result, we can easily understand what keyword is currently more used, what keyword has disappeared recently, and so on. This component is constructed using JFreeChart [6]. The search window (1) and keyword buffer (3) are shared between the keyword map view and the chart view. The two views can be changed by clicking the upper tab (4). Therefore, we can easily find the keywords to find a community by using these views.

The third component is the *community view*. The screen shot of this component is shown in Figure 4. This is the main component of our community mining system, which is based on the method we described in the previous section. As in the previous components, the right area, including the search window (1) and the keyword buffer (3), is shared in all three views. These views can be easily used by clicking the upper tab (4) with the same keywords. The upper right area (2) is used for showing the communities produced by the proposed method. The component is constructed using JUNG [7]. The bottom right window (5) is used for representing the properties for each community. When a community in (2) is selected, the properties are calculated by the method presented in Section 2.2 and the results are shown in (5). The window (6) is used for showing statistics of the communities, such as the number of nodes in a community. The bottom left window (7) is used for filtering the edges in the community or searching in the community. In addition to this function, we are able to access the bibliography list of the specified researcher and personal co-author networks from this component.

3.2 Bibliography Data

In the present study, we used a part of the CiNii database [9] to obtain bibliography information. The database is composed of SGML (Standard Generalized Markup Language) data. The CiNii database contains bib-

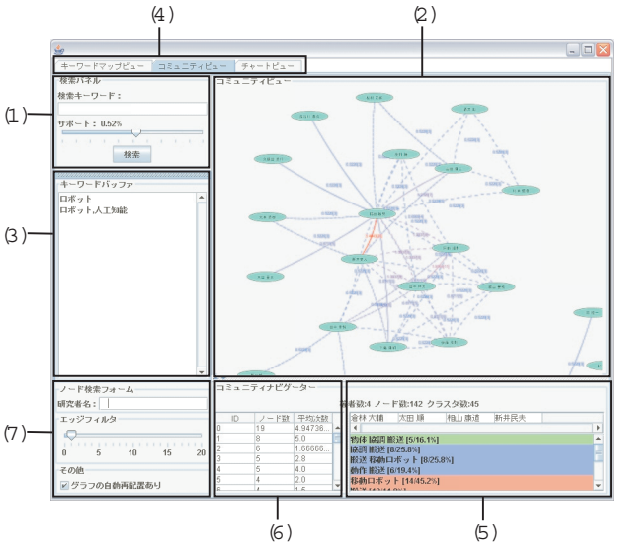

Figure 4. Screen shot of the community view.

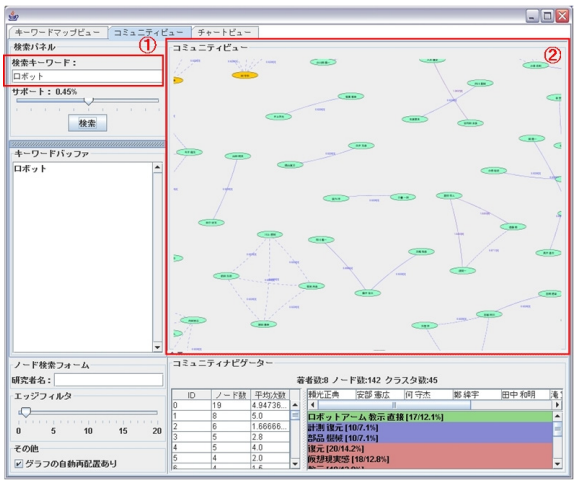

Figure 5. A search example for “robotics.”

liography entries such as title, author and publication year. We used 131,000 records, 93,000 records, 358,000 records, 519,000 records, 40,000 records and 1,087,000 records for the paper, researcher, author, co-author, keyword and label, respectively. The paper and researcher entries denote the number of records for the papers and the number of records for the researchers, respectively. The author entries denote the number of authors for each paper. For example, the record is three when three researchers write a paper collaboratively. The co-author entries denote the number of combinations of authors for a paper. For example, when a paper is written by four authors, it is counted as ${}_4C_2 = 6$ for the paper. The keyword entries denote the number of kinds of keywords. In our present study, we used the words in a title as keywords. The label entries denote the number of keyword labels for each paper. For example, the record is three when the paper title has three keywords.

3.3 Case Study of the Community Mining System

In order to demonstrate how the proposed system works, we next show the system behavior using actual examples. We will show how to find communities for “robotics” by using the community mining system. First, the user inputs in the search window the research field that the user want to know. In this case, “robotics” is specified by the user. When we select the tab of the keyword map view, we can find related keywords such as “model” and “design.” Then, the user is able to find related words and refine the word to what the user wants to find out about the community. If the user is interested in the trends of “robotics” and “artificial intelligence,” we can obtain the keyword occurrences of both words for each year by selecting the chart view. Then, the user can find out that the word “robotics” has been used from the early days; however, the words “artificial intelligence” have been used only recently. Through using the keyword map view and chart view, we are able to learn about the research situation of the field “robotics.” We assume the user does not discontinue the use of the keyword “robotics” for community mining, even if he use the two components[1]. Then, the user inputs “robotics” in the search window (1) for the community view. This example is shown in Figure 5. The research communities are shown as graphs (2) in Figure 5. As you can see from this example, we can obtain many communities for “robotics,” such as “robot arm” or “virtual reality.” If the user prefers another size for the research communities shown on the system, the user can adjust the graph size by using the slide bar in the bottom left window (7) in Figure 4. When the user uses the slide bar, the system changes the threshold of the edges for display. You can find this situation in Figure 6. The screen on the far right has more edges than the screen on the far left. When a particular community is specified from the graphs, the user can browse the community property, shown by (3) and (4) in Figure 7. When the user finds a person of interest, the user can browse the publication list and personal graph. When a particular researcher is selected with a right click, a pop-up menu appears, as in Figure 8. There are two items. One is used for showing the bibliography list (5) and the other is used for showing the egocentric personal co-authorship graph (6).

[1]Of course, he can change the keyword if he finds a more appropriate keyword for community mining by using the two components.

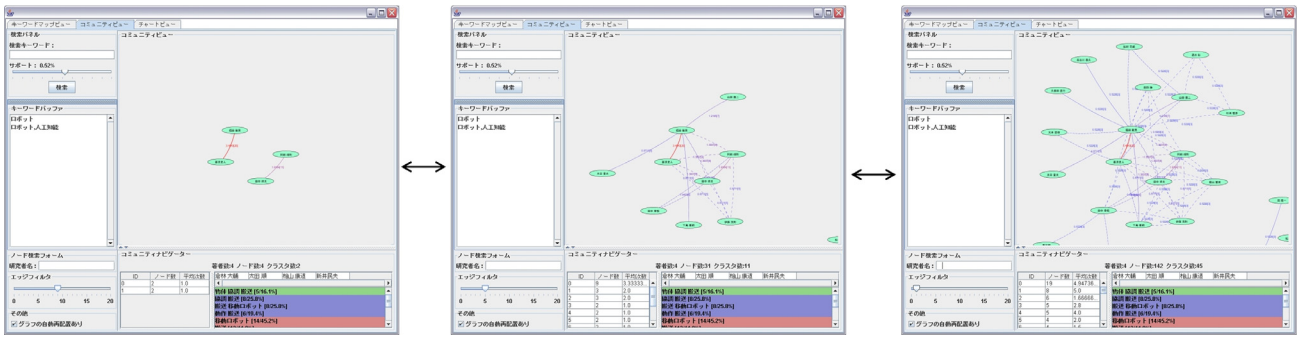

Figure 6. Adjusting the community size by using the slide bar.

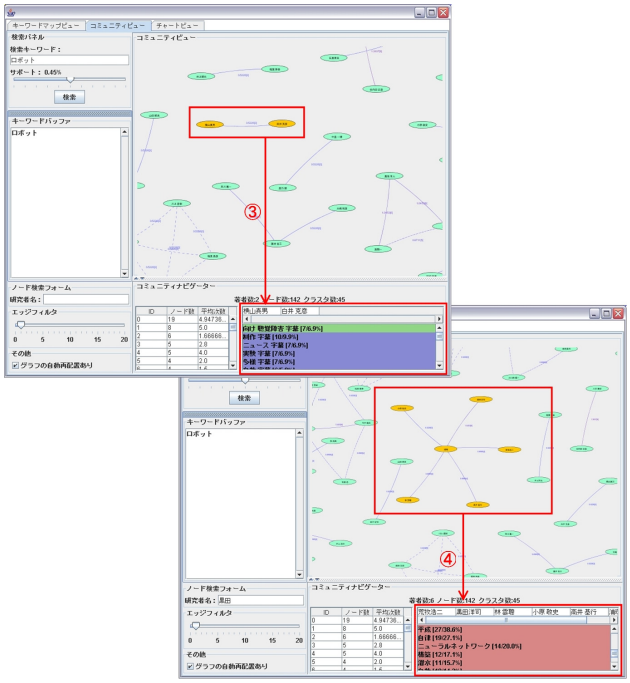

Figure 7. Keyword listing for the specified communities.

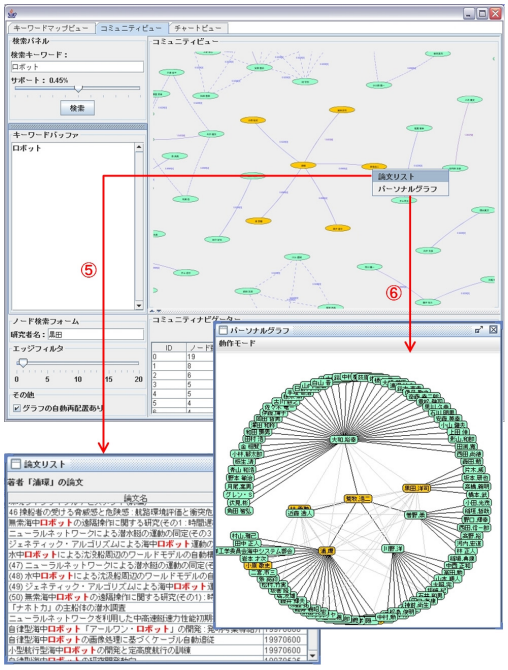

Figure 8. Publication listing and personal graph.

3.4 Experiment

In order to evaluate our method from the qualitative aspect, we analyzed the communities obtained by our method. The communities were constructed using the word "discovery." Although our method discovered many communities, we selected five communities. The assigned keywords and the number of nodes for each community are shown in Table 1. The longest pairs of keywords which appeared as the top three in frequency were selected for the table.

Communities No. 2, No. 3, No. 4 and No. 5 represent research groups in universities. Most of the assigned words for the communities are valid. However, our method assigns meaningless words such as "workshop" and "report." In future work, we plan to develop a method to suppress the assignment of meaningless words. We believe such words can be identified by a simple method of retrieving stop words.

Community No. 1 is the largest community. Since the community includes a very famous professor in the "discovery" domain, he bridged several communities. As a result, this community is a research group in an academic society. Our system assigns general words for this commu-

Table 1. Keywords obtained for the topic "discovery."

Community ID	Number of nodes	Keywords
No. 1	17	{discovery} {algorithm} {special issue}
No. 2	5	{Japanese poem, similarity} {poem, similarity, extraction} {English sentence, technology}
No. 3	3	{heuristics, method} {database, heuristics} {knowledge, exception, discovery}
No. 4	2	{definition, occurrence, lambda calculus} {logic, program} {unification, extension}
No. 5	2	{*th, workshop} {scientific, discovery} {*th, report}

nity, such as "discovery" and "algorithm." In addition, the system cannot generate a long pair of words for this community because it's a large community. In future work, we will develop a method to identify a person who is a bridge between different groups. Incidentally, the word "special issue" was assigned in Community No. 1 because the community members edit a special issue for a journal.

4 Discussion

Obviously, our research community mining system supports exploration of knowledge domains. One of the main differences between the conventional community mining system Ver. 1 [5] and the proposed system is keyword use. Ver. 1 does not use keywords for mining communities. This function enables us to find a community without complete knowledge of the research domains where the user wants to mine. In addition, the keyword map view and chart view are good support systems for choosing a keyword with understanding of its usage. One of the issues of using keywords is the use of different keywords for the same research field, such as "pervasive" and "ubiquitous." The keyword map view would support the finding of similar keywords and the chart view would support the deciding of which word is popular or recently used.

The community mining view also improves upon Ver.1 by using keywords. Keywords enable the proposed system to divide communities into a readable size. In addition, since the property of a community can be shown by our algorithm, the user can easily understand the community's specialty. The shared window between three views, including the search window and keyword buffer, is easy to use. When a user starts to use the community mining system, he or she does not have definite images of the mined community. Therefore, the user has to iterate to refine the word which is input to the system. The shared window supports such actions.

Unfortunately, the current system has the limitation of trust in the keywords, because we utilize title words as keywords. Since most of the records in the CiNii database do not include keywords or abstracts, we made an explicit decision for this treatment. However, this approach sometimes does not assign the appropriate keywords for papers. In future work, we have to investigate a keyword assignment method for such records.

5 Conclusion

In this paper, we proposed a research community mining method. The key feature of our research is the modeling of papers and researchers. This modeling enables us to eliminate the edges in large clusters. In addition, the modeling can also help to retrieve communities for particular topics. We also proposed a method to assign a word to each cluster. We implemented our method as an interactive system and showed how to mine communities with our system. The case studies and experimental results show that the proposed method looks promising.

References

[1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In J. B. Bocca, M. Jarke, and C. Zaniolo, editors, *Proc. 20th Int. Conf. Very Large Data Bases, VLDB*, pages 487–499. Morgan Kaufmann, 12–15 Sept. 1994.

[2] K. Börner, L. Dall'Asta, W. Ke, and A. Vespignani. Studying the emerging global brain: Analyzing and visualizing the impact of co-authorship teams. *Complexity*, 10(4):58–67, 2005.

[3] C. Chen and R. J. Paul. Visualizing a knowledge domain's intellectual structure. *Computer*, 34(3):65–71, 2001.

[4] Google scholar, 2006. http://scholar.google.com/.

[5] R. Ichise, H. Takeda, and K. Ueyama. Community mining tool using bibliography data. In *Proceedings of the 9th International Conference on Information Visualization*, 2005.

[6] JFreeChart, 2006. http://www.jfree.org/jfreechart/.

[7] JUNG, 2006. http://jung.sourceforge.net/.

[8] M. M. Kessler. Bibliographic coupling between scientific papers. *American Documentation*, 14(1):10–25, 1963.

[9] CiNii (Citation Information by NII). National Institute of Informatics, 2006. http://ci.nii.ac.jp/.

[10] newsmap, 2004. http://www.marumushi.com/apps/newsmap/.

[11] prefuse, 2006. http://prefuse.sourceforge.net/.

[12] Scientific literature digital library, 2006. http://citeseer.ist.psu.edu/.

[13] H. Small. Co-citation in the scientific literature: A new measure of the relationship between two documents. *Journal of the American Society of Information Science*, 24:265–269, 1973.