Community Extracting using Intersection Graph and Content Analysis in Complex Network

Toshiya Kuramochi, Naoki Okada, Kyohei Tanikawa, Yoshinori Hijikata and Shogo Nishida
Graduate School of Engineering Science, Osaka University
Osaka, Japan
kuramochi@nishilab.sys.es.osaka-u.ac.jp

***Abstract*—Many researchers have studied complex networks such as the World Wide Web, social networks, and the protein interaction network. They have found scale-free characteristics, the small-world effect, the property of high-clustering coefficient, and so on. One hot topic in this area is community detection. For example, the community shows a set of web pages about a certain topic in the WWW. The community structure is unquestionably a key characteristic of complex networks. In this paper, we propose a new method for finding communities in complex networks. Our proposed method considers the overlaps between communities using the concept of the intersection graph. Additionally, we address the problem of edge inhomogeneity by weighting edges using the degree of overlaps and the similarity of content information between sets. Finally, we conduct clustering based on modularity. And then, we evaluate our method on a real SNS network.**

***Keywords*-Complex Network; Community Extraction; Intersection Graph; Hierarchical Clustering; Text Mining; SNS Network**

I. Introduction

Many researchers, having studied complex networks such as SNS networks, the WWW, and the protein interaction network, have reported scale-free characteristics, the small-world effect, the property of high-clustering coefficient, and so on [2, 3, 5, 15]. Recently, the community structure in complex networks is gaining increased attention from many researchers. The community structure shows the appearance of densely connected groups of nodes, with only sparse connections among groups. Many community detection methods have been proposed based on the definition presented above [4, 16]. Analyses of community structure have been conducted in various complex networks. A community in an SNS network shows a set of people with the same background or hobby. Additionally, WWW communities show sets of web pages related to a certain topic [8] and those in the protein interaction network show sets of proteins having the same function [11].

For community detection, researchers have started to show interest in whether overlaps between communities can be extracted [6, 20, 22, 23, 30]. The overlaps signify that one node belongs to several communities. For example, one person usually belongs to several communities, as do groups of college friends and groups of business members in social networks. One page is categorizable within several groups in the WWW. For example, the Apple Inc. page is categorizable among "computer" category pages and "audio" category pages. It is important that a method of community detection be able to assign a node not only to one community but also to several communities.

The weights of all edges in complex networks are assumed to be the same in many community detection methods [9, 25]. However, edges are rarely homogeneous in real networks. For example, various human connections such as those of businesses, hobbies and organizations exist in SNS networks. Similarly, various links such as internal links, advertisement links and links to other sites exist in the WWW. It is important that the weights of these edges are not be treated as identical. They should be set individually.

Many researchers use hierarchical clustering methods when they divide networks into clusters. Most hierarchical clustering methods require advance input [7, 29]: the number of clusters to be extracted. However, the number of real communities is often unknown in real networks. Therefore, it is important that the number of proper clusters be decided automatically.

We propose a new method of community detection that can solve the problems described above. Our proposed method can extract overlaps between communities using the idea of the intersection graph. The intersection graph is generated from a set by following process: each subset in the whole set are regarded as one node, and each node pairs connect if there are some common elements between them. For example, in WWW, the whole set contains many web pages and each subset is a set of strong connected pages. We also determine the weights of the edges in the intersection graph using two types information: the overlaps of the members and the similarity of content information such as text information and attribute information which appear in the network. The text information comprises sentences attached to nodes and edges. For example, in SNS, the information is self-introduction and friend introduction. The attribute information comprises values, words, or phrases to pre-determined attributes attached to nodes. For example, in the user profile in SNS, the information includes birthdays and hobbies. Moreover, we use the hierarchical clustering

method based on modularity proposed by Newman et al. [16]. This method does not necessitate manual input of the number of clusters.

The remainder of this paper is organized as follows. We introduce related works in Section II and describe our proposed method in Section III. In Section IV, we present an implementation of the method and apply it in a real SNS network. Moreover, we evaluate the extracted clusters and confirm the effectiveness of the method in Section V. Finally, we describe conclusions and future works in Section VI.

II. Related Works

The problem of community detection in complex networks has been examined in various areas such as those of computer science and medical science [3].

Some researchers have attempted to extract communities in complex networks including the overlaps between communities. The overlaps mean that one node belongs to several communities. Everett et al. found them using the idea of the intersection graph [7]. Palla et al. also found them by detecting cliques whose size was k and merging the cliques that shared $k-1$ nodes [20]. Fuzzy clustering is often used to extract the overlaps between communities [6]. This clustering method considers the notion of fuzziness and can assign one node to several communities. Researchers have proposed methods of community detection using fuzzy clustering [22, 23, 30].

Our study weights edges in complex network for dealing with edge inhomogeneity. Weighting edges in a network (usually a document network or hyperlink network) is popular in the area of the information retrieval. Some researchers improved the effectiveness of link analysis using content information. Jiang et al. measured the similarity between words using link information and content information of words [12]. Abe et al. proposed a method that combined link analysis with anchor text analysis and improved the extraction accuracy of relevant web pages [1]. Hung et al. improved the HITS algorithm by analyzing anchor text [10].

Many researchers use hierarchical clustering methods for detecting communities. The methods need input, which is the number of clusters preliminarily. Newman et al. reported modularity as an indicator of how well the clusters are formed [17]. They proposed some clustering methods based on modularity [16, 18, 19]. This method does not obviates manual input of the number of clusters.

Our proposed method considers the overlaps between communities using the idea of the intersection graph. Furthermore, we address the problem of edge's inhomogeneity by weighting edges using the degree of overlaps and the similarity of content information between sets (nodes of the intersection graph). Finally, we conduct a clustering method based on modularity, which does not necessitate manual input of the number of clusters. No study deal with all the above problems for detecting communities.

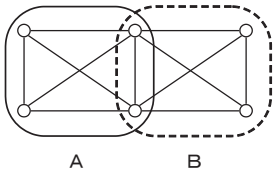

Figure 1. Two sets which are closely connected to each other

III. Proposed Method

In this section, we explains our proposed method. We provide a summary of the process and the design concept of our method in section III-A. In Sections III-B-III-E, we explain the details of the process.

A. Summary and Design Concept of Our Proposed Method

The input of our proposed method is a graph of $G = (V, E)$, where V stands for the set of nodes and E signifies the set of edges. Additionally, content information is given to the nodes and the edges. We apply the following four steps to this graph and describe the details of each step in Sections III-B-III-E.

Step 1. Enumeration of dense subgraphs: This method enumerates dense subgraphs (generally, they are called cliques) from an input graph of $G = (V, E)$.

Step 2. Conversion to the intersection graph: This method regards each subgraph enumerated in Step 1 as one special node and converts the input graph G to the intersection graph of $G' = (V', E')$.

Step 3. Calculation of the weights of special edges: This method calculates the weights of special edges using the degree of overlaps and the similarity of content information between sets (dense subgraphs).

Step 4. Clustering based on modularity: This method divides the special nodes into clusters using a clustering method based on modularity.

We applied the method of Everett et al. [7] to Step 1 and Step 2. First, their method enumerates maximal cliques as dense subgraphs in an input graph of $G = (V, E)$. A clique is a subgraph in which an edge exists between any two nodes. A maximal clique is a clique in which no other node in the graph can be included to create a larger clique. It is also called a complete subgraph. Next, the method converts the input graph G into the intersection graph $G' = (V', E')$. Finally, the method conducts hierarchical clustering based on modularity for the intersection graph.

Our method further calculates the weights of special edges between special nodes (dense subgraphs) in the intersection graph in Step 3. This step addresses the edge inhomogeneity in network, as described in Section I. For example, we obtained a social network shown in Figure 1. In this graph, two dense subgraphs (sets A and B) exist. If A is a group of people who belong to the same university and B is a group of people who do the same business, then two sets must be

separated. However, if both sets are groups of people who belong to the same university, two sets must be regarded as the same community. A structure similar to that shown in Figure 1 often exists in actual networks. When all edges are treated similary, it is difficult to distinguish the two cases described above.

Our method calculates the weights of special edges in the intersection graph using information of two types. One is the member information of elements of each set. The method calculates the degree of overlaps between sets (dense subgraphs). The other is content information (text information and attribute information) appearing in each set. The method calculates the similarity between sets using a vector space model [24]. This model is frequently used in the study of information retrieval. It subsumes each set as one vector and calculates the similarity between sets. Our method combines these two calculated values for weighting special edges.

Finally, in Step 4, the method finds communities in the weighted intersection graph in Step 3. This step conducts clustering based on modularity, which does not necessitate input of the number of clusters that we want to extract.

B. Step 1. Enumeration of Dense Subgraphs

Our method enumerates dense subgraphs in the input graph of $G = (V, E)$ in Step 1. A typical dense subgraph is a maximal clique. There exist various types of dense subgraphs such as n-clique, n-clan, k-plex, and k-core which relax a link condition is exist [25, 28]. Our method is applicable to any of clique types.

C. Step 2. Conversion to the Intersection Graph

Our method regards each dense subgraph enumerated in Step 1 as one special node and makes the intersection graph $G' = (V',\ E')$ from the input graph $G = (V,\ E)$ in Step 2. When several sets (dense subgraphs) $S_i\ (i = 1, \cdots, n)$ are generated, our method generates a special node v'_i for each set S_i. If a common element exists in two arbitrary nodes v'_i and v'_j, then a special edge is put between them. The intersection graph is a new graph composed of special nodes and special edges [14]. When the method puts a special edge between special nodes, we can set the threshold of the number of common elements between the subgraphs corresponding to these special nodes.

D. Step 3. Calculation of the Weights of Special Edges

Our method calculates the weights of edges in the intersection graph using the degree of overlaps and the similarity of the content information between special nodes (each special node can be regarded as one set) in Step 3.

Many types of the degree of overlaps betweetn arbitary two sets X and Y $(d(X, Y))$, such as co-occurrence frequency, mutual information, Dice coefficient, Simpson coefficient, and Jaccard coefficient [13, 21] are used. For example, Jaccard coefficient is defined as below:

$$d(X, Y) = \frac{|X \cap Y|}{|X \cup Y|} \tag{1}$$

Our method uses vector space model [24] to calculate the similarity of the content information between two arbitrary sets X and Y. The method regards each set as one vector and calculates the *tf·idf* value for the keyword in the texts in the set. This *tf·idf* value becomes the element of the vector. Finally, the method calculates the similarity $sim(X, Y)$ between vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ corresponding to two sets X and Y using cosine similarity.

$$sim(X, Y) = \cos\theta = \frac{\boldsymbol{x} \cdot \boldsymbol{y}}{\| \boldsymbol{x} \| \| \boldsymbol{y} \|} \tag{2}$$

The method calculates the weights $w(i,\ j)$ for the special edge between special nodes v'_i and v'_j (corresponding to set X and Y) using the degree of overlaps of sets $d(X,\ Y)$ and the similarity of content information $sim(X,\ Y)$. In the simplest way, the weights $w(i,\ j)$ are calculable by the product (eq. (3)) or weighted sum (eq. (4)) of both indicators.

$$w(i, j) = w(X, Y) = \alpha d(X, Y) sim(X, Y) \tag{3}$$

$$w(i, j) = w(X, Y) = \alpha d(X, Y) + \beta sim(X, Y) \tag{4}$$

We can also use other types of calculation function such as emphasizing the degree of overlaps (eq. (5)) or the similarity of content information (eq. (6)).

$$w(i, j) = w(X, Y) = \frac{sim(X, Y)}{1 + \epsilon - d(X, Y)} \tag{5}$$

$$w(i, j) = w(X, Y) = \frac{d(X, Y)}{1 + \epsilon - sim(X, Y)} \tag{6}$$

Here, $\epsilon\ (0 < \epsilon < 1)$ is a constant used to keep the denominator from being 0.

E. Step 4. Clustering Based on Modularity

Our method conducts clustering for community detection in the intersection graph in Step 4. When a method extracts several clusters in a network, we must evaluate the currently detected clusters. Modularity is a broadly accepted indicator for evaluation. The indicator is simple and intuitive. Therefore, we adopt a clustering method based on the modularity that is suggested by Newman et al.. This method is based on the idea that a random network shows no community structure. When k communities are given and P_k is defined as the sets of these communities, the module function $Q(P_k)$ is the following.

$$Q(P_k) \ = \ \sum_i (e_{ii} - a_i^2) = \mathrm{Tr}(e) - |e^2| \tag{7}$$

$$\begin{cases} e_{ij} & = & \frac{1}{2m} \sum_{vw} A_{vw} \delta(c_v,\, i) \delta(c_w,\, j) \\ a_i & = & \frac{1}{2m} \sum_v k_v \delta(c_v,\, i) \end{cases}$$

A_{vw} means adjacency matrix. In case of unweighted graph, A_{vw} is 1 if nodes v and w are connected and 0 otherwise. In case of weighted graph, A_{vw} means the weights between nodes v and w. k_v means the degree of node v ($k_v = \Sigma_w A_{vw.}$). The method supposes that the nodes are divided into communities such that node v belongs to community c_v. The δ-function $\delta(i, j)$ is 1 if $i = j$ and 0 otherwise and m is the number of edges in the graph. The method also defines a $k \times k$ symmetric matrix e. The element e_{ij} is set as the number of edges that links a node in community i to a node in community j divided by the total number of edges in the network. The trace of this matrix $\mathrm{Tr}(e) = \Sigma_i e_{ii}$ gives the number of edges that connects nodes in the same community divided by the total number of edges in the network; clearly a good division into communities should have a high value of this trace. The row with sums $a_i = \Sigma_j e_{ij}$ represents the number of edges that connects to the nodes in community i divided by the total number of edges in the network. $|\boldsymbol{x}|$ stands for the sum of the elements of the matrix $\boldsymbol{x}$. If the division becomes more properly, then the ratio of the edge in the community to the edges in the network becomes a higher value. Consequently, the value of the module function Q is increased. This coincides with the definition of community described in Section I.

The clustering method based on modularity is aimed at maximizing the module function Q. As the most basic method, Newman proposed a greedy approach to optimize Q [18]. This method is an agglomerative hierarchical clustering method that initially assumes each node as a community. It repeatedly searches for a pair of communities whose joining gives the greatest increase ΔQ in Q. It continues joining communities until all of them form one community. ΔQ is calculated as below equation:

$$\Delta Q = 2(e_{ij} - a_i a_j) \tag{8}$$

Finally, it chooses the division with the highest Q as a result. In fact, this method requires no input of the number of clusters. Researchers have proposed various clustering methods based on modularity in addition to the basic method above. For example, they are the methods that use edge betweenness [17] or eigenvalue decomposition [19, 28]. Our method is applicable to any of these methods.

IV. Applied to the SNS Network

We next apply our method to a real social network. Our method is applicable to networks with various relations among nodes and with the content information representing these relations. We select mixi[1] for this study which is the most popular SNS in Japan. The mixi users write self-introductions for themselves and friend introductions for their friends. Therefore, it is easy to obtain the content information. Additionally, the mixi network has various relations between users such as the connections of universities, working places, and hobbies.

[1] http://mixi.jp/

Table I
Statistical information of the dataset

Test subject	No. nodes	No. edges	Average degree	Clustering coefficient	No. true communities
1	1836	9619	5.24	0.664	9
2	2261	19425	8.59	0.56	11
3	1279	10015	8.07	0.66	8
4	1183	19423	4.28	0.665	11
5	2859	16079	5.62	0.652	12
6	2906	24774	8.53	0.609	16
7	1772	6925	3.91	0.594	11
8	2654	16484	6.21	0.654	12
9	4268	18517	4.34	0.501	13
10	1251	11824	9.45	0.619	11
11	1360	8404	6.18	0.668	11
12	1333	8007	6.01	0.683	7
13	2408	22152	9.2	0.466	7
14	1980	13070	6.6	0.565	9
15	3480	31999	9.2	0.531	9
16	5063	37302	7.37	0.527	16
17	2186	21282	9.74	0.444	9
18	1929	13754	7.13	0.501	11
19	1603	11065	6.9	0.581	10
20	2506	20126	8.03	0.526	9

In Section IV-A, we describe how we have collected the dataset and the details of it. And then, we explain our purpose of evaluation (in Section IV-B), how we have implemented each method (in Section IV-C), and the evaluation method of extracted clusters (in Section IV-D).

A. Dataset

We make a dataset for the evaluation inviting test subjects who give true relationships between them and their friends. We followed users from a test subject to two in the radius (from the test subject up to the friends of test subject's friends). This link structure was stored in the dataset. Additionally, we collected friend introductions, self-introductions and user attributes as content information. A friend introduction comprises sentences that introduce a user's friend. A self-introduction comprises sentences that introduce a user personally. Users can write friend introductions only for their connected friends. They must write self-introductions for themselves. The user attributes are values, words or phrases for attributes such as gender, present address, age, birthday, blood type, hometown, hobby, career, and affiliations. We asked the test subjects to answer true relation names for each user in the dataset. True relation names are names for relations such as those of universities, working places and hobbies between the test subject and user in the dataset. The test subjects are 20 users who are all university students. We present the statistical information of the dataset in Table I.

B. Purpose of Evaluation

We verify the four questions through the evaluation.

- **Whether our method achieves better results than the conventional method:** We compare this method with the conventional method proposed by Everett et al. The conventional method converts an input graph into the intersection graph and conducts a simple hierarchical clustering.
- **Whether the clustering method based on modularity extracts better cluster:** We use the clustering method based on modularity. We compare this method with a simple hierarchical clustering, and examine the contribution of this method.
- **Whether it is efficient to use content information for weighting edges:** Our method uses not only information about the degree of overlaps of sets but also the content information. We compare the method using both kinds of information with the method using only the degree of overlaps of sets. We confirm the effectiveness of the content information.
- **Whether the kinds of content information affect the results:** As described in Section III-A, complex networks have content information of two kinds attached to nodes and edges. The former corresponds to self-introductions and user attributes and the latter corresponds to friend introductions in SNS networks. We examine whether the results change according to the kind of content information.

C. Implementation

1) Parameter settings of the proposed method: We adopt the maximal clique as the dense subgraph in Step 1. We can use various sizes of the maximal clique (the clique threshold). If the clique threshold is 5, then the method uses only the maximal cliques that comprise more than four nodes. We set 3, 4 and 5 as the clique threshold. However we implemented the case in which the clique threshold is more than 5, the maximal clique did not exist in almost all test subjects. When the method puts a special edge between special nodes, we can set the threshold of the number of common elements (the overlap threshold) in Step 2. We set 1, 2, 3 and 4 as the overlap threshold. We set the threshold condition $(a,\ b)$ for which a is the clique threshold and b is the overlap threshold. We conduct nine threshold conditions $(a,\ b)$ = (3, 1), (3, 2), (4, 1), (4, 2), (4, 3), (5, 1), (5, 2), (5, 3) and (5, 4). When the clique threshold is 5, the network of three test subjects (test subjects 3, 4, 7) did not include a case in which five people including the test subject were mutually connected. The method extracted no clusters in these cases. We use the data for 17 test subjects in conditions (5, 1), (5, 2), (5, 3) and (5, 4).

In Step 3, we selected the Jaccard coefficient (eq. (1)) as the degree of overlaps of sets. We select friend introductions, self-introductions, and user attributes as the content information. Our method extracts nouns as keywords by conducting morphological analysis of the content information. These nouns become elements of the vector. The method calculates *tf·idf* values for all nouns within one maximal clique (corresponding to a special node). The maximal clique can represent one vector. The similarity between maximal cliques is calculated using eq. (2). Finally, the weights between maximal cliques are calculated using eq. (6). We set $\epsilon = 0.1$ in this experiment. We use a greedy approach that merges repeatedly to find the combination that maximizes the increment of the modularity [18].

2) Implementation of community extraction method: As described in section IV-C1, we use content information of several types: friend introductions, self-introductions, and user attributes. We respectively designate the cases using friend introductions, self-introductions, and user attributes as *FIA* (with friend introduction analysis), *SIA* (with self-introduction analysis), and *UAA* (with user attribute analysis). We also examine the combination of the types of content information. For example, when we use both friend introductions and self-introductions, we designate the case as *FIA+SIA*. Actually, we examine the cases *FIA*, *SIA*, *FIA+SIA*, and *FIA+SIA+UAA*. We designate these cases using the content information as *WithCA* (with content analysis). Hereinafter, we regard FIA as a representative example of WithCA. We examine the contribution of the usage of the content information. We implemented the case using only the degree of the overlaps between sets (Jaccard coefficient) as the weights of edges in Step 3. We designate the case *NonCA* (without content analysis).

We implemented Everett's method as a baseline method [7]. The method comprises three steps. The first two steps of the method are the same as the first two steps (Step 1 and Step 2) of our method. Unlike our method, it conducts a simple hierarchical clustering in the third step. We adopt the following hierarchical clustering method. The method searches for a pair of special nodes that have maximal Jaccard coefficient and merges the pair, repeatedly. We must set the number of output clusters in this method beforehand. We implemented two cases: one outputs the clusters when the number of clusters including the test subject becomes the number of true communities (Table 1). We designate this case as *Everett's method*. The other outputs the clusters when the number of clusters including the test subject becomes the number of clusters extracted by NonCA. We designate this case as *Everett's method**. Everett's method* was implemented in order to examine the contribution of clustering based on modularity.

D. Evaluation Method of Extracted Clusters

We extract clusters of two kinds: a cluster that includes the test subject and a cluster that does not contain the test subject. The test subject cannot judge the connection of members in the latter clusters. Therefore, we specifically addressed only those clusters containing the test subject.

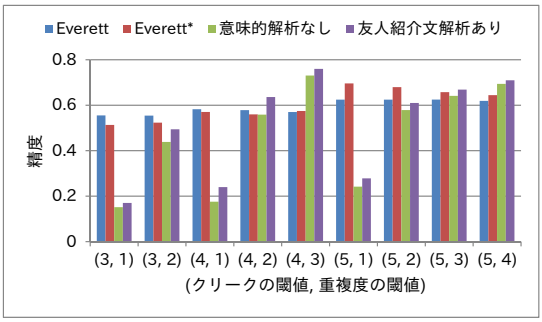

Figure 2. Comparing the precision of Everett's method and our method

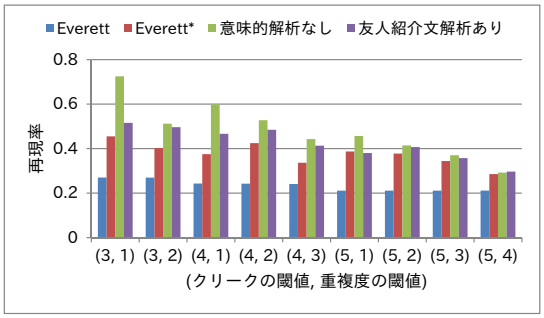

Figure 3. Comparing the recall of Everett's method and our method

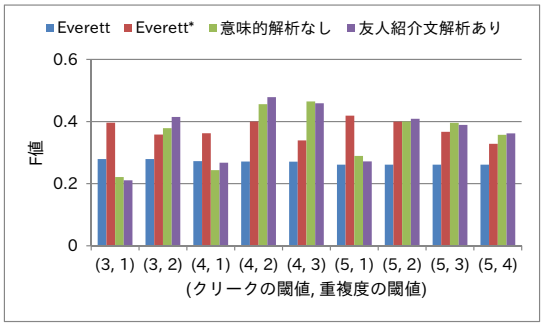

Figure 4. Comparing the F-measure of Everett's method and our method

To measure how accurate the extracted clusters are, we adopt the following evaluation process.

Step 1. A test subject enumerates all relation names for each person in the dataset (number of relation names is regarded as the number of true communities (Table 1)). The test subject can see the top page of each person in mixi.

Step 2. We assume that the relation in the extracted cluster corresponds to each relation name. Then, we calculate the precision, recall, and F-measure per relation name for the cluster (For the calculation, we consider that the relation name is the name of correct relation for the cluster). The precision, recall, and F-measure of a relation name N are calculated as follows.

- Precision(N) = No. people whose relation name is N in the extracted cluster / No. people in the extracted cluster
- Recall(N) = No. people whose relation name is N in the extracted cluster / No. people whose relation name is N in the dataset
- F-measure(N) = (2 · Precision(N) · Recall(N)) / (Precision(N) + Recall(N))

Step 3. We use the highest F-measure calculated in Step 2 among all relation names as the F-measure of the extracted cluster. We regard the relation name that marked the highest F-measure as the relation in the cluster. We also use the precision and recall calculated using the relation as the precision and recall of the clusters.

Step 4. We calculate the average values of the precision, recall, and F-measure of all clusters. These values are regarded as the evaluation value of one test subject.

V. Evaluation of Extracted Clusters

In this section, we examine that our method could extract better communities than the conventional method, and we also assess the performance of the clustering method based on modularity and that of the content information analysis. We show the average values of the precision, recall, and F-measure in all 20 test subjects in the conventional method and our method in Figure 2-4.

A. Comparison of Our Method and Conventional Method

First, we compare our method (NonCA, FIA) with the conventional method (Everett's method). In precision (Figure 2), the results of our method tend to become better when both thresholds are large. That is because the edges which represent weakly relation between users are removed due to high thresholds.

The precision of our method is very low when the overlap threshold is 1 because of the clustering method based on modularity. In this method, we repeatedly merge a pair of nodes to maximize the increment ΔQ (eq. (8)) of module function Q. The nodes connected to a large number of nodes are hard to merge under the influence of a member $a_i a_j$. Therefore, the nodes with lower degree (in other words, the nodes distant from the center of the network) are merged preferentially. Many of these nodes represent users who have weakly relationship to subjects, and the subjects could not give any relation name to such users in this experiment. In conditions whose overlap threshold is 1, our method extracts giant clusters contain both subjects and such users. We consider this is the reason why the precision of our method become low when the overlap threshold is 1. Other hand, in the conventional method, there are no major changes in precision and recall by varying the threshold condition. Since the conventional method merges node pairs in descending order of Jaccard coefficient, node pairs whose degree of overlaps is small is hard to merged. In precision,

we cannot determine which method is better.

In recall (Figure 3), despite the results of our method become worse with large overlap threshold, our method shows higher recall than the conventional method in all conditions.

At last, in F-measure (Figure 4), the results of our method overcome that of the conventional method in all conditions except whose overlap threshold is 1. Overall, we found our method brings a better result than the conventional method.

B. Contribution of Clustering Based on Modularity

Next, in order to examine the contribution of clustering based on modularity, we compare Everett's method* and NonCA. Both methods extract same number of clusters, but use another clustering method: Everett's method* uses the simple hierarchical clustering method and NonCA uses the clustering method based on modularity. In precision (Figure 2), as mentioned above, the clustering method based on modularity produces lower precision when the overlap threshold is 1. Everett's method* shows higher precision with lower overlap threshold, and NonCA shows higher precision with higher overlap threshold. In recall (Figure 3), the results of NonCA exceed that of Everett's method* in all threshold conditions. In a comprehensive evaluation in F-measure (Figure 4), NonCA is greater than Everett's method* in most conditions except the conditions whose overlap threshold is 1. We found that clustering based on modularity, although which is greatly affected by network characteristics, outputs better clusters with adjusting the threshold condition.

C. Contribution of Content Analysis

And then, to find out that content information analysis how affect precision, recall and F-measure, we compare NonCA and FIA. In all threshold conditions, the precisions of FIA are greater than that of NonCA (Figure 2) , other hand, the recall of FIA are tend to be less than that of NonCA (Figure 3). We cannot determine which method is better because the difference in the F-measure is only slightly (Figure 4). The cause of these results might be that the effect of member information of elements is stronger than that of content information in the mixi network. In mixi, a user needs to request another user and the latter user must approve the request to become a friend. These processes strengthen the edges of mixi network. Therefore, the usage of content information did not affect the F-measure so much.

However, we demonstrated the possibility that the use of the content information improves the precision. It is useful for applications that demand high precision. One example of such applications is a friend recommendation system. We can create a friend recommendation system using the extracted clusters of our method. The friend recommendation system helps a user to find friends by recommending the people unconnected to the user in the same cluster (hereinafter, 'the candidates'). Users regard precision as more important than recall when several hundred people exist as the candidates for recommendation. Even if a user wants the system to recommend many candidates without omission, we can satisfy the demand by recommending multiple times. If the user becomes friends with some of the recommended users, then the network around the user expands. When we apply our method to the new network again, the friend recommendation system can obtain new candidates for the recommendation.

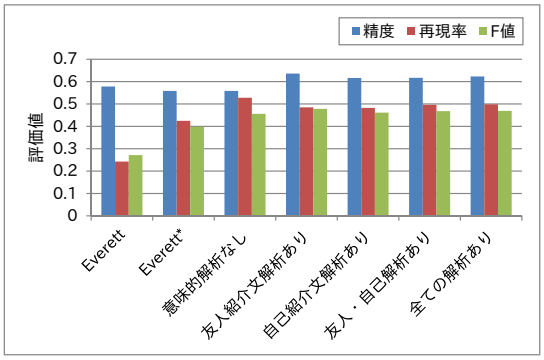

Figure 5. Evaluation in condition $(4, 2)$ using content information of various kinds

D. Contribution of the Kind of Content Information

Finally, we examine how change the results depending on the type of content information to be used. We evaluate the cases using friend introductions, self-introductions and user attributes as the content information in condition (4, 2). We present results of Everett's method, Everett's method*, NonCA, FIA, SIA, FIA+SIA, and FIA+SIA+UAA in Figure 5. The results of cases using content information are mutually similar: the highest case is FIA, whose F-measure is 0.478; the lowest case is FIA+SIA+UAA, whose F-measure is 0.458. This is considered because the impact of the type of content information is weak compared to the link information in a mixi network.

VI. Conclusion and Future Work

As described in this paper, we proposed and evaluated a new method for community detection. Our method is particularly useful for revealing overlaps between communities, for dealing with the inhomogeneity of relations between nodes, and for automatically determining the number of clusters. We sought to solve these problems using the consept of the intersection graph, the weights of edges using member information and content information and a clustering method based on modularity. We applied our method to the mixi network. By comparing our method with Everett's method, we demonstrated the superiority of our method in the evaluation. Moreover, we compared the

case using content information with the case not using the content information. We showed that the case using content information improved the precision of the extracted clusters. As future work, we will apply our method to other complex networks such as the World Wide Web and the protein interaction network.

REFERENCES

[1] Abe, T., Toyoda, M., Kitsuregawa, M., Improving Contents-based Web Information Retrieval Using Anchor Texts and Link Analysis, DEWS (2003).

[2] Albert, R., Barabasi, A.-L., Statistical mechanics of complex networks, Review of Modern Physics, Vol.74, pp.47-97 (2002).

[3] Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., Hwang, D.U., Complex Networks: Structure and Dynamics. *Phys. Rep.* Vol.424, pp.175-308 (2006).

[4] Danon, L., Duch, J., Guilera, A. D., Arenas, A., Comparing community structure identification, J. Stat. Mech, p.09008 (2005).

[5] Dorogovtsev, S. N., Mendes, J. F. F., Evolution of networks. Advances in Physics, Vol.51, No.4, pp.1079-1187 (2002).

[6] Dunn, J. C., A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters, J. Cybernet. Vol.3, pp.32-57. (1973).

[7] Everett, M. G., Borgatti, S. P., Analyzing Clique Overlap. Connections Vol.21, No.1, pp.49-61 (1998)

[8] Flake, G. W. et al., Self-Organization of the Web and Identification of Communities, IEEE Computer, Vol.35, No.3, pp.66-71 (2002).

[9] Gregory, S., An Algorithm to Find Overlapping Community Structure in Networks, Proc. of PKDD'07, pp.91-102 (2007).

[10] Hung, B. Q. et al., HITS Algorithm Improvement using content Text Portion, Web Intelligence and Agent Systems, Vol.8, No.2, pp.149-164 (2010).

[11] Huss, M., Holme, P., Currency and commodity metabolites: Their identification and relation to the modularity of metabolic networks. Preprint q-bio/0603038 (2006).

[12] Jiang, J. J., Conrath, D. W., content Similarity Based on Corpus Statistics and Lexical Taxonomy, Proceedings on International Conference on Research in Computational Linguistics, Taiwan (1997).

[13] Manning, C. D., Schütze, H., Foundations of statistical natural language processing, The MIT Press, London (2002).

[14] McKee, T. A., McMorris, F. R., Topics in Intersection Graph Theory, Proc. of SIAM International Conference on Society for Industrial & Applied Mathematics, Philadelphia (1999).

[15] Newman, M. E. J., The Structure and function of complex networks. SIAM Review, Vol.45, pp.167-256 (2003).

[16] Newman, M. E. J., Detecting community structure in networks, Eur. Phys. J. B Vol.38, pp.321-330 (2004).

[17] Newman, M. E. J., Girvan, M., Finding and evaluating community structure in networks, Phys. Rev. E Vol.69, p.026113 (2004).

[18] Newman, M. E. J., Fast algorithm for detecting community structure in networks, Phys. Rev. E Vol.69, p.066133 (2004).

[19] Newman, M. E. J., Finding community structure in networks using the eigenvectors of matrices , Phys. Rev. E Vol.74, p.036104 (2006).

[20] Palla, G., Derenyi, I., Farkas, I., Vicsek, T., Uncovering the overlapping community structure of complex networks in nature and society. Nature Vol.435, No.7043, pp.814-818 (2005).

[21] Rasmussen, E., Clustering Algorithms, Information Retrieval: Data Structures and Algorithms. Frakes, W. B., Baeza-Yates, R. (Eds.) (1992).

[22] Reichardt, J., Bornholdt, S., Detecting fuzzy community structures in complex networks with potts model, Physical Review Letters Vol.93, p.218701 (2004).

[23] Reichardt, J., Bornholdt, S., Statistical mechanics of community detection, Physical Review E Vol.74, 016110 (2006).

[24] Salton, G., Wong, A., Yang, C. S., A vector space model for automatic indexing, ACM Vol.18, pp.613-620 (1975).

[25] Scott, J., Social Network Analysis: A Handbook, 2nd ed. Sage Publications, London (2000).

[26] Scripps, J., Pang-Ning Tan, Abdol-Hossein Esfahanian, Node Roles and Community Structure in Networks (2007).

[27] Tasgin, M., Haluk Bingol,Community Detection in Complex Networks using Genetic Algorithm (2007).

[28] Wasserman, S., Faust, K., Social Network Analysis: Methods and Applications, Cambridge University Press (1994).

[29] White, S., Smyth, P., A spectral clustering approach to finding communities in graphs, Proc. of SIAM International Conference on Data Mining (2005).

[30] Zhang, S., Wang, R., Zhang, X., Identification of overlapping community structure in complex networks using fuzzy c-means clustering, pp.483-490. (2007).