Communications for Statistical Applications and Methods
2020, Vol. 27, No. 5, 569–578

https://doi.org/10.29220/CSAM.2020.27.5.569
Print ISSN 2287-7843 / Online ISSN 2383-4757

Comparison of graph clustering methods for analyzing the mathematical subject classification codes

Kwangju Choi[a], June-Yub Lee[b], Younjin Kim[1,a], Donghwan Lee[2,c]

[a]Institute of Mathematical Sciences, Ewha Womans University, Korea;
[b]Department of Mathematics, Ewha Womans University, Korea;
[c]Department of Statistics, Ewha Womans University, Korea

Abstract

Various graph clustering methods have been introduced to identify communities in social or biological networks. This paper studies the entropy-based and the Markov chain-based methods in clustering the undirected graph. We examine the performance of two clustering methods with conventional methods based on quality measures of clustering. For the real applications, we collect the mathematical subject classification (MSC) codes of research papers from published mathematical databases and construct the weighted code-to-document matrix for applying graph clustering methods. We pursue to group MSC codes into the same cluster if the corresponding MSC codes appear in many papers simultaneously. We compare the MSC clustering results based on the several assessment measures and conclude that the Markov chain-based method is suitable for clustering the MSC codes.

Keywords: mathematical subject classification, Markov chain clustering, entropy graph clustering

1. Introduction

The mathematical subject classification (MSC) is the five-digit alphanumerical code for the organization of research papers, which are based on two mathematical databases, mathematical reviews (MR) and Zentralblatt MATH (zbMath) (Lange *et al.*, 2012). The primary purpose of the use of MSC is to help users find the literature of interest by subject area (https://msc2020.org/). Various mathematical and other scientific journals often request authors to enter the MSC codes in their articles, and the author-supplied MSC codes are used to classify submissions. Browsing the database of MR and zbMath using MSC search is an effective method of the following research in specific areas. Each paper can be involved in multiple study areas with corresponding MSC codes; therefore, it is represented by a combination of various mathematical fields. MSC codes only provide a hierarchical classification scheme; however, it is desirable to apply statistical clustering methods to identify and cluster the specific topics in the academic field by investigating multiple MSC codes presented in the same papers. It is possible to gather detailed information about an intimate relationship between all areas of pure and applied mathematics through a database of mathematical papers.

Due to the emergence of a complex structure of data, various kinds of clustering methods have been studied for partitioning a finite set of data into different clusters according to similarities. Graph clustering algorithms can easily be applied to identify communities when the data is a form of the

[1] Corresponding author 1: Institute of Mathematical Sciences, Ewha Womans University, 52, Ewhayeodae-gil, Seodaemun-gu, Seoul 03760, Korea. E-mail: younjinkim@ewha.ac.kr
[2] Corresponding author 2: Department of Statistics, Ewha Womans University, 52, Ewhayeodae-gil, Seodaemun-gu, Seoul 03760, Korea. E-mail: donghwan.lee@ewha.ac.kr

Published 30 September 2020 / journal homepage: http://csam.or.kr

Table 1: Top 10 MSC codes most frequently referred

MSC	Major category	Minor category	Num. of docs
01A70	History and biography	Biographies, obituaries, ...	23,637
81T30	Quantum theory	String and superstring theory ...	19,333
65N30	Numerical analysis	Finite elements, Rayleigh-Ritz ...	17,567
68Q25	Computer science	Analysis of algorithms and ...	17,187
47H10	Operator theory (1959–)	Fixed-point theorems ...	15,437
62M10	Statistics	Time series, auto-correlation	14,469
35Q53	Partial differential equation	KdV-like equations ...	13,526
62G05	Statistics	Estimation	13,389
05C85	Combinatorics	Graph algorithms ...	13,135
35B40	Partial differential equation	Asymptotic behavior of solutions	13,014

MSC = mathematical subject classification.

network such as social networks or modules in biological systems (Kenley and Cho, 2011). In this paper, we compare graph clustering methods for grouping MSC codes into similar clusters.

This paper is organized as follows. In Section 2, we describe how to obtain the MSC data and construct graph data for applying the graph clustering methods. In Section 3, we review two well-known graph clustering algorithms, entropy-based clustering, and Markov chain clustering (MCL). A comparative study of graph clustering methods is given in Section 4. Concluding remarks are provided in Section 5.

2. Data and preprocessing

In the MR database at the MathSciNet website (http://www.ams.org/mathscinet), each of 3,193,603 entries (papers, books, and reports) provides bibliographic information for a document published from in 1899 to in 2015 along with possibly multiple choices of MSC codes. We download P = 2,975,635 entries published in 1940–2015 with a least one MSC codes and 1,101,422 documents have a single MSC code, but 62.96% items have multiple codes (995,257 documents with two codes, 531,327 documents with three codes, 228,847 with four codes, and 116,782 documents with five to ten codes).

An MSC code consists of a two digits primary classification code (79 codes are used so far), a letter, and two digits code for minor classification; for example, 00A15 for bibliographies in the 00-General subject. There are 9,395 known MSC codes (as of Jan 1, 2016) but only N = 8,822 codes are used at least once in the P items, and we list the top ten MSC codes most frequently referred in Table 1.

We now define the binary incidence matrix $X = (x_{ip})$ using discrete MSC code information as:

$$x_{ip} = \begin{cases} 1, & \text{if } i^{th} \text{ MSC code is used in an } p^{th} \text{ document,} \\ 0, & \text{otherwise,} \end{cases}$$

where $i = 1, \ldots, N$ and $p = 1, \ldots, P$. Now, we construct a weighted graph $G = G(V, E)$ where the vertices $V = (v_1, \ldots, v_N)$ and edges E are the MSC codes and their similarity, respectively. To transform X into a form of graph, we compute an $N \times N$ adjacency matrix $M = XX^T$. Then, the $(i, j)^{th}$ element of M, M_{ij} represents the strength of edge between vertex v_i and v_j, i.e., the number of documents containing the i^{th} and j^{th} MSC code simultaneously. Intuitively, if M_{ij} is large, many documents which have both MSC code i and j, so these MSC codes will be in the same cluster. Table 2 shows 11 edges with edge strength bigger than 2,500; in addition, Figure 1 provides part of the graph that focuses on the field of Statistics (62XDD).

The entries in the adjacency matrix M associated with the code to document matrix X represent the

Table 2: Top 11 pairs of MSC codes with M_{ij} larger than 2,500

MSC_i	Major (Minor) category	MSC_j	Major (Minor) category	M_{ij}
03B35	Math ⋯ logic (Mechanization of proofs ⋯)	68T15	Computer science (Theorem proving ⋯)	2,628
05C85	Combinatorics (Graph algorithms)	68R10	Computer science (Graph theory)	4,790
05C85	Combinatorics (Graph algorithms)	68Q25	Computer science (Analysis of algorithm)	2,801
17B37	Nonassociative ring (Quantum groups ⋯)	81R50	Quantum theory (Quantum groups ⋯)	2,972
35Q30	Partial differ ⋯ (Navier-Stokes eq ⋯)	76D05	Fluid mechanics (Navier-Stokes eq ⋯)	3,721
47H09	Operator theory (Contraction-type ⋯)	47H10	Operator theory (Fixed-point theorem)	3,097
47H10	Operator theory (Fixed-point theorem)	54H25	General topology (Fixed-point and ⋯)	4,688
60K25	Probability theory (Queueing theory ⋯)	90B22	Operations research (Queues and service)	5,528
68Q25	Computer science (Analysis of algorithm)	68R10	Computer science (Graph theory)	2,949
81T30	Quantum theory (String and super ⋯)	81T60	Quantum theory (Supersymmetric field)	2,723
81T30	Quantum theory(String and super ⋯)	83E30	Relativity and gra(String and super ⋯)	6,412

MSC = mathematical subject classification.

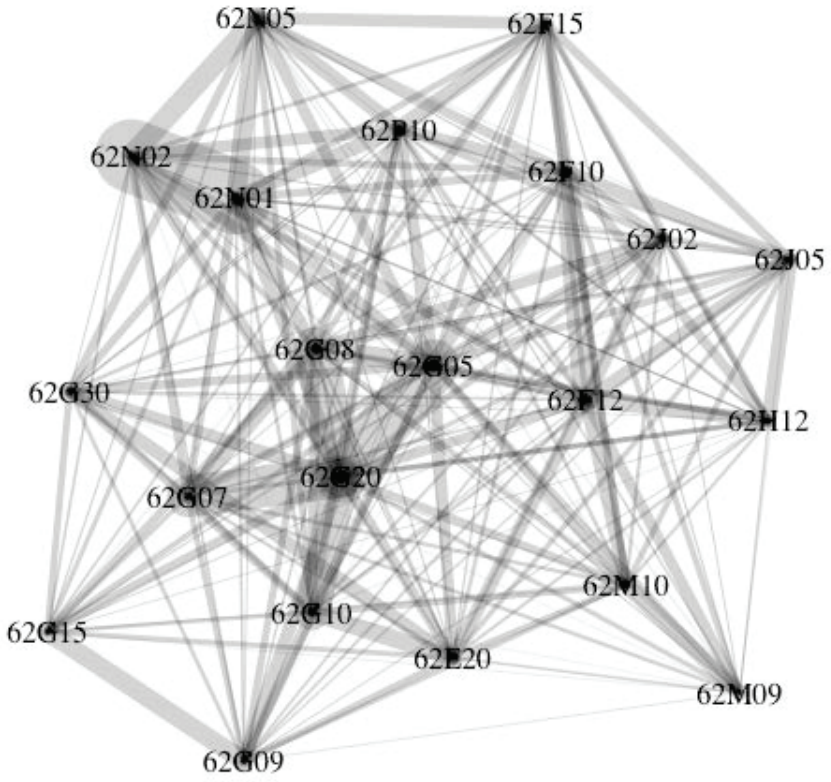

Figure 1: *A part of graph with the adjacency matrix M.*

number of links between two MSC codes so the contribution to the entries by p^{th} document depends on the number of codes m_p in the document. If a document contains two codes MSC_i and MSC_j then M_{ij} and M_{ji} will be added by one each, so total contribution on the entries of M will be 2. A document with $m_p > 1$ codes make total $P(m_p, 2)$ (m_p permutation 2) contributions to M. In order to normalize the contribution to M from each document to be one, we introduce a weighted code-to-document

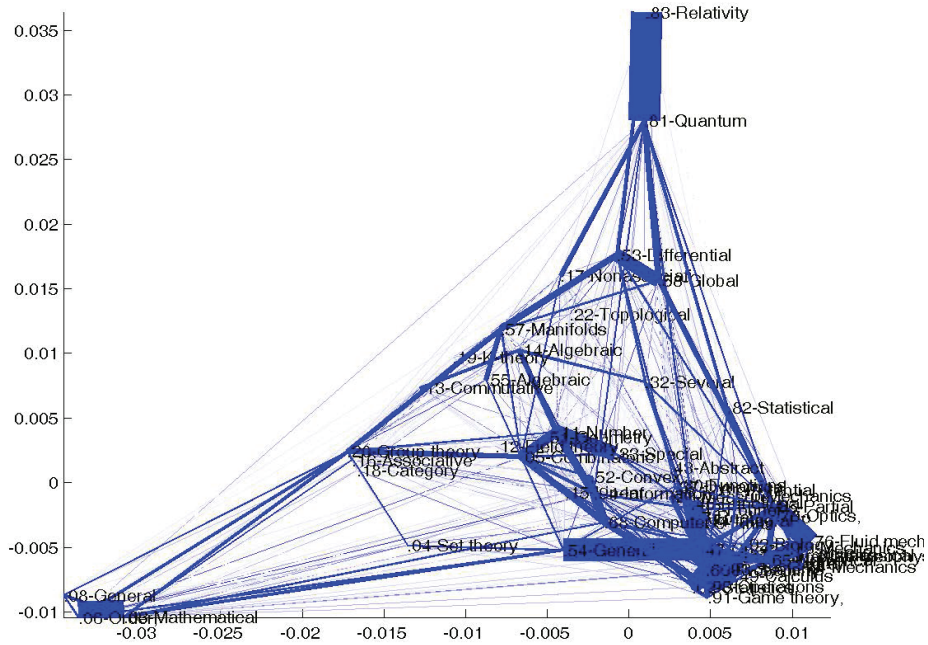

Figure 2: *Relative strength of the edges of* M^w_{ij}.

matrix X^w as:

$$x^w_{ip} = \begin{cases} \dfrac{1}{P(m_p, 2)}, & \text{if } i^{th} \text{ code is one of } m_p \text{ MSC codes in } p^{th} \text{ document,} \\ 0, & \text{otherwise,} \end{cases}$$

for $i = 1, \ldots, N$ and $p = 1, \ldots, P$ with $P(1, 2) := 1$, $P(m, 2) := m(m-1)$, $m > 1$. Then the sum of all entries of the weighted adjacency $M^w := X^w (X^w)^T$ associated with X^w will be the number of the documents, $\sum_{i=1}^{N} \sum_{j=1}^{N} M^w_{ij} = P$.

Table 2 shows the strength of the weighted adjacency matrix. Here for just simple visualization, we only use the graph with 79 nodes associated with the principal classification codes (instead of actual N = 8,822 codes for real computations). The width of each edge represents the relative strength of the edges (excluding self-interactions) of M^w_{ij}. We can easily see that 81-Quantum/83-Relativity, 06-Order.Latice/03-Math.Logic, 54-Gen.Topology/47-Oper.Theory, and 63-Fluid/35-Part.Diff.Eq pairs are closely related.

3. Methods

In this section, we review two graph clustering methods. Consider the clustering problem for partitioning N vectors of P-dimensional binary data (X) or vertices V of G into $K(\leq N)$ disjoint clusters $C_1, C_2, \ldots, C_K$. Let us define a clustering set C as a collection of K disjoint clusters $C_1, C_2, \ldots, C_K$. Note that a clustering set C of K disjoint clusters also can be regarded as an K-partition of $\mathcal{N}$={1, 2, . . . , N}. Then, $i \in C_k$ implies that an i^{th} MSC code belongs to k^{th} cluster C_k.

3.1. Entropy-based clustering algorithm

Entropy type measures for similarity among the finite set of data have been frequently used. Since it is hard to define a distance when the given set of data is discrete, we use the entropy-type measures for similarity among distinct objects. Here, we review an entropy-based clustering method Li *et al.* (2004), Chen and Liu (2005) minimizing the expected entropy of the partition. An entropy-based clustering algorithm can be formally derived in the framework of probabilistic clustering models.

For given cluster C_k, an *entropy* of the partition is defined as

$$H(C_k) = -\sum_{i \in C_k} \sum_{p=1}^{P} \sum_{s=0}^{1} \Pr(x_{ip} = s) \log \Pr(x_{ip} = s),$$

for $k = 1, \ldots, K$. For each cluster C_k, we let n_k be the number of MSC codes in each cluster C_k and $N_{p,k,1} = \sum_{i \in C_k} x_{ip}$, $N_{p,k,0} = n_k - N_{p,k,1}$. Then, with the estimated $\Pr(x_{ip} = s)$, $\widehat{\Pr}(x_{ip} = s) = N_{p,k,s}/n_k$, the estimate of $H(C_k)$ is

$$\hat{H}(C_k) = -\sum_{p=1}^{P} \sum_{s=0}^{1} \frac{N_{p,k,s}}{n_k} \log \frac{N_{p,k,s}}{n_k}.$$

An entropy-based clustering method finds $C_1, \ldots, C_K$ to minimize the following entropy measure of the partition

$$\hat{H}(C) = \frac{1}{N} \sum_{k=1}^{K} n_k \hat{H}(C_k)$$

which is the weighted sum of an estimated entropy $\hat{H}(C_k)$.

Following Li *et al.* (2004), $\hat{H}(C)$ can be approximated as

$$\begin{aligned} \hat{H}(C) &\approx \frac{1}{N} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i,j \in C_k} \sum_{p=1}^{P} \left| x_{ip} - x_{jp} \right| \\ &= \frac{1}{N} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i,j \in C_k} \left(x_{i+} + x_{j+} - M_{ij} \right), \end{aligned} \tag{3.1}$$

where $x_{i+} = \sum_{p=1}^{P} x_{ip}$ and $x_{j+} = \sum_{p=1}^{P} x_{jp}$. Note that to minimize the estimated entropy, the pairs of MSC codes with large M_{ij} tend to be in the same cluster.

For an actual algorithm implementation, we use the Monte-Carlo method suggested by Li *et al.* (2004). First, we make all MSC codes placed randomly in the K clusters with almost the same size. Then we choose one MSC code v randomly. Let C_i denote a current cluster containing a randomly chosen code v. According to the value of an estimated expected entropy, we decide if the randomly chosen code v moves from a current cluster C_i to another cluster C_j ($\neq C_i$) or not. We move the code v from a current cluster C_i to another cluster C_j if the cluster C_j (after getting a current code v) has less estimated expected entropy than a current cluster C_i. Otherwise, the code v stays in the current cluster C_i. After continuing these steps, we assign MSC codes into K disjoint clusters minimizing the estimated expected entropy. The repetitive relocation is required in the actual algorithm implementation. Note that the convergence property of the Monte-Carlo optimization is shown in Li *et al.* (2004).

3.2. Markov chain clustering algorithm

Recall that the $(i, j)^{th}$ element of an $N \times N$ adjacency matrix M, M_{ij} represents the number of papers containing the i^{th} and j^{th} MSC code. Define a *transition matrix* T for the graph G as a matrix having an $(i, j)^{th}$ element T_{ij} which is obtained as

$$T_{ij} = \frac{M_{ij}}{\sum_{j \neq i} M_{ij}},$$

where $1 \leq i \leq N$ and $1 \leq j \leq N$.

A *random walk* of a directed graph consists of a series of vertices generated at a starting vertex, reaching the next vertex by selecting one of the outgoing edges, and repeating that process. A finite Markov chain defined on the set of states, which is denoted by $V = \{v_1, v_2, \ldots, v_N\}$, is specified by the transition matrix T of the weighted graph G on the vertex set V. Therefore, there is no significant difference between a random walk on graphs and a finite Markov chain. In the weighted directed graph, all Markov chains can be thought in random walks on directed edges. A MCL algorithm generates random walks on weighted graphs for dividing a finite set of data into different clusters according to their similarity. When a random walk reaches the vertex w from a starting vertex v with high probability, vertices v and w need to be collected together into the same cluster in the MCL algorithm.

Let that an $(i, j)^{th}$ element of the matrix M^r, which is made by the r^{th} power of M, is a number of length r paths from v_i to v_j. Therefore a number of length r paths between two vertices, v_i and v_j, is obtained by raising the adjacency matrix of G to the exponent r. To subdivide MSC codes into K disjoint clusters according to their similarity, the MCL algorithm is explained in the following four steps (Van Dongen, 2008).

- Step 1. We define a matrix M^* having an $(i, j)^{th}$ element which is obtained as

$$M^*_{ij} = \begin{cases} M_{ij}, & \text{when } i \neq j, \\ 1, & \text{when } i = j. \end{cases}$$

  Then we execute a normalization of the matrix M^*. Let us call the normalized matrix M.

- Step 2 (Expansion). In this step, we execute the power of the matrix M, which means $M \times M$. Then we again call the powered matrix M. Note that the effectiveness of the power of a matrix diminishes as the flow proceeds, where a flow on the graph G is defined as a transition probability from a vertex to another vertex. The higher powers of the matrix obtained after repeating this step also reinforce more connections between MSC codes.

- Step 3 (Inflation). Let r be a given inflation parameter. Depending on the given inflation parameter r, an inflation operator $\Gamma_r : R^{N \times N} \to R^{N \times N}$ is defined as

$$\Gamma_r M_{ij} = \frac{(M_{ij})^r}{\sum_{q=1}^{N} (M_{qj})^r},$$

  where $\Gamma_r M_{ij}$ is an $(i, j)^{th}$ element of the matrix $\Gamma_r M$.

  Note that the inequality of each column becomes bigger after taking an inflation operator Γ_r. If an $(i, j)^{th}$ element of the matrix M has a big (small) value, then an $(i, j)^{th}$ element of the matrix

$\Gamma_r M$ becomes bigger (smaller), respectively. After executing a normalization of the matrix $\Gamma_r M$, we again call the normalized matrix M.

- Step 4 (Iteration). We repeat the second and third steps until the matrix M converges.

The expansion step strengthens more connections among MSC codes and subdivides MSC codes into some vast clusters. However, the inflation step both strengthens already strong currents and weakens already weak currents. To avoid dominantly massive clusters, we need the inflation step that distributes the MSC codes across the different clusters. In the MCL algorithm, the given inflation parameter r controls the extent of strengthening and weakening and influences the granularity of disjoint clusters. For the given set of data, we subdivide MSC codes into 136 disjoint clusters by setting a value 1.5 as an inflation parameter.

4. Numerical study

In this section, we apply the entropy-based clustering (Entropy) in Section 3.1 and MCL in Section 3.2 to the MSC code dataset. Furthermore, we compare these with conventional methods and evaluate the performance. As the traditional methods, we use the hierarchical clustering algorithm. First, we start by assigning each MSC code into a single cluster. Then we have N clusters containing just one MSC code. Second, we find the closest pair of clusters, which is most similar among them. Then we merge them into a single cluster. Now we have $N - 1$ clusters. Third, we compute all distances between any two clusters among $N - 1$ clusters, where a distance between two clusters, C_i and C_j, is defined as an average of all distances between any two codes $v \in C_i$ and $w \in C_j$. Then we repeat the second step to find the closest pair of clusters that are most similar and merge them into a single cluster. We repeat the second and third steps until we get K disjoint clusters.

For dissimilarity measure $\delta(x_i, x_j)$ between two MSC codes i and j, we use the following Jaccard coefficient and Simple matching coefficient:

(1) Jaccard coefficient: $\displaystyle\sum_{p=1}^{P} \frac{|x_{ip} - x_{jp}|}{\|\{p : x_{ip} = 1 \text{ or } x_{jp} = 1\}\|}$.

(2) Simple matching coefficient: $\displaystyle\sum_{p=1}^{P} \frac{|x_{ip} - x_{jp}|}{P}$.

In this paper, the hierarchical clustering algorithms based on Jaccard coefficient and Simple matching coefficient are called *hclust1* and *hclust2*. The *hclust1* method considers the effectiveness of the number of the coordinates of two codes, $x_i = (x_{i1}, x_{i2}, \ldots, x_{iP})$ and $x_j = (x_{j1}, x_{j2}, \ldots, x_{jP})$, having the value 1 in the same position.

We consider within-cluster criterion for the clustering set C that evaluates the clustering results of these four algorithms (Entropy, hclust1, hclust2, and MCL)

$$D(C) = \frac{1}{N} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i,j \in C_k} \delta(x_i, x_j),$$

where a dissimilarity $\delta(x_i, x_j)$ is a Jaccard coefficient or a Simple matching coefficient between two MSC codes, x_i and x_j. The within-cluster criteria for the clustering set C of Entropy, hclust1, hclust2, and MCL are given in Table 3. All clustering methods are similar in terms of the Jaccard coefficient.

Table 3: Dissimilarity measures of clustering methods

Within-cluster criterion	MCL	Entropy	hclust 1	hclust2
Jaccard coefficient	0.982	0.983	0.984	0.984
Simple matching coefficient	4.59 ×1E-04	4.39 ×1E-04	4.68 ×1E-04	3.81 ×1E-04

MCL = Markov chain clustering; Entropy= entropy-based clustering; hclust1 = hierarchical clustering algorithms based on Jaccard coefficient; hclust2 = hierarchical clustering algorithms based on Simple matching coefficient.

Table 4: Size of biggest four clusters

Size	Entropy	hclust1	hclust2	MCL
Biggest	926	6278	8666	601
Second	634	1605	4	469
Third	526	792	3	423
Fourth	398	8	2	395
Standard deviation	115.3391	556.0134	740.2677	96.46783

Entropy= entropy-based clustering; hclust1 = hierarchical clustering algorithms based on Jaccard coefficient; hclust2 = hierarchical clustering algorithms based on Simple matching coefficient; MCL = Markov chain clustering.

Table 5: Estimated entropy and W-Ratio value for each clustering method

Algorithm	Estimated entropy	W-Ratio
MCL	3832.066	0.71434
Entropy	2604.859	0.45628
hclust1	6094.500	0.94477
hclust2	5196.884	0.77836

MCL = Markov chain clustering; Entropy= entropy-based clustering; hclust1 = hierarchical clustering algorithms based on Jaccard coefficient; hclust2 = hierarchical clustering algorithms based on Simple matching coefficient.

In this data, 97.8 percent of MSC code pairs has a value 1 as a Jaccard coefficient. This means that the number of MSC code pairs used in different papers is extremely small. hclust2 shows the smallest simple matching coefficient; however, *hclust1* and *hclust2* insert most MSC codes into one or two big clusters (Table 4).

Next, we calculate the estimated entropy value $\hat{H}(C)$ (3.1) with the disjoint clusters $C_1, \ldots, C_K$ of each clustering algorithm and the W-Ratio as another evaluation measure. Let $T = \{T_1, T_2, \ldots, T_P\}$ be the set of documents (papers) where an p^{th} paper T_p contains m_p MSC codes, where $1 \le p \le P$. For given clustering set C, W-Ratio is defined as

$$W(C) = \frac{\sum_{k=1}^{K} \sum_{i,j \in C_k} \sum_{p=1}^{P} W_p(i,j)}{\sum_{i,j \in C} \sum_{p=1}^{P} W_p(i,j)},$$

where

$$W_p(i,j) = \begin{cases} \dfrac{1}{m_p(m_p-1)}, & \text{if an } p^{th} \text{ document } T_p \text{ contains distinct codes } i \text{ and } j, \\ 0, & \text{if } i = j \text{ or an } p^{th} \text{ document } T_p \text{ does not contain codes } i \text{ or } j. \end{cases}$$

Note that to get relatively high W-Ratio, the pairs of MSC code with the large number of papers containing them tend to be in the same cluster.

Table 5 shows the estimated entropy value and W-Ratio value of four algorithms. The estimated entropy of the Markov Chain clustering (MCL) algorithm is 3832.066, and hclust1 and hclust2 have larger entropies. The entropy-based algorithm, the expected entropy value of K disjoint clusters is

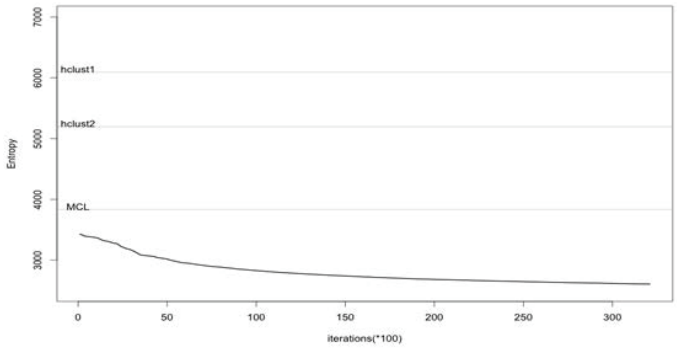

Figure 3: *Estimated entropy values of graph clustering algorithms. MCL = Markov chain clustering; Entropy= entropy-based clustering; hclust1 = hierarchical clustering algorithms based on Jaccard coefficient; hclust2 = hierarchical clustering algorithms based on Simple matching coefficient.*

decreasing on iterations and we can see that it converges to the smallest amount, 2604.859 (Figure 3). MCL and hclust1 algorithms have relatively high W-Ratio values. Note the entropy-based algorithm that makes clustering groups relatively uniform, which does not consider the number of papers containing the same MSC codes, has a relatively low W-Ratio value. However, the hclust2 algorithm having one or two dominantly big clusters has a relatively high W-Ratio value despite not considering the effectiveness of the number of papers containing the same MSC codes.

5. Concluding remarks

In this paper, we compare appropriate two graph clustering methods for grouping Mathematical Subject Classification (MSC) codes according to similarities. The hclust1 and hclust2 clustering algorithms have one or two dominantly big clusters; therefore, we compare the Markov chain and entropy-based clustering algorithms making clustering groups relatively uniform. The MCL clustering algorithm has a higher entropy expectation value than the entropy-based clustering algorithm minimizing the expected entropy of the partition. The MCL clustering algorithm has a larger W-Ratio value than the entropy-based clustering algorithm since the MCL algorithm considers the number of papers containing the same MSC codes; however, the entropy-based clustering algorithm does not. Therefore, we conclude that the MCL clustering algorithm is suitable for clustering the MSC codes.

Acknowledgments

This work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. 2017R1A6A3A04005963; 2018R1D1 A1B07050910) and by Ministry of Science, ICT and Future Planning (2019R1A6A1A11051177).

References

Chen K and Liu L (2005). The best K for entropy-based categorical data clustering. In *Proceedings of the 17th International Conference on Scientific and Statistical Database Management*, 253–262.

De Robbio A, Maguolo D, and Marini A (2002). Mathematics Subject Classification and related schemes in the OAI framework. In: Find and Post Mathematics in the Web A workshop on Electronic Information and Communication in Mathematics, *Lange Springer-Lecture Notes in*

Computer Sciences

Kenley EC and Cho YR (2011). Entropy-Based Graph Clustering: Application to Biological and Social Networks. In *Proceedings of the 11th IEEE International Conference on Data Mining*, 1116–1121.

Lange C, Ion P, Dimou A, Bratsas C, Sperber W, Kohlhase M, and Antoniou I (2012). Bringing mathematics to the web of data: the case of the mathematics subject classification, *Extended Semantic Web Conference*, 763–777.

Li T, Ma S, and Ogihara M (2004). Entropy-based criterion in categorical clustering. In *Proceedings of the 21st International Conference on Machine Learning*, 68–75.

Van Dongen S (2008). Graph clustering via a discrete uncoupling process, *SIAM Journal on Matrix Analysis and Applications*, **30**, 121–141.

Received May 22, 2020; Revised July 3, 2020; Accepted July 8, 2020