https://doi.org/10.20965/jaciii.2014.p0896

Paper:

Visualizing Fuzzy Relationship in Bibliographic Big Data Using Hybrid Approach Combining Fuzzy *c*-Means and Newman-Girvan Algorithm

Maslina Zolkepli, Fangyan Dong, and Kaoru Hirota

Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology
G3-49, 4259 Nagatsuta, Midori-ku, Yokohama 226-8502, Japan
E-mail: {maslina, tou, hirota}@hrt.dis.titech.ac.jp
[Received December 15, 2013; accepted August 15, 2014]

Bibliographic big data visualization method is proposed by incorporating a combination of fuzzy *c*-means clustering and the Newman-Girvan clustering algorithm, where clustered results are displayed in a network view by grouping objects with similar cluster memberships. As current bibliographic visualizations focus on the crisp relationship among data, fuzzy analysis and visualization may offer insights to bibliographic big data, enabling faster decision making by improving displayed information precision. The proposed method is applied to the DBLP citation network dataset. Results show that merging two clustering algorithms and visualization using fuzzy techniques enables the user to converge a few target papers within an average of 5 minutes from 1.5 million papers stored in the DBLP. Users targeted for the proposed method include researchers, educators, and students who hope to use real-world social and biological networks. The proposal is planned to be opened to the public through the Internet.

Keywords: visualization, bibliographic big data, fuzzy *c*-means, Newman-Girvan algorithm, DBLP

1. Introduction

Bibliographic visualization is developed to visually capture the relationship among bibliographic big data consisting of journal/conference papers, especially in the computer science field. Bibliographic visualization is used by researchers, educators, and students to find appropriate scientific papers. Conventional bibliographic visualization methods include CiteWiz [1], Biblioviz [2], and BibRelEx [3], where the main concern is to visually convey crisp relationships among bibliographic big data in different ways without focusing on the fuzzy relationship among big data. Therefore fuzzy analysis and visualization may offer deeper insights into big data.

Several hybrid fuzzy *c*-means clustering methods, such as the fuzzy *c*-means hybrid approach to the clustering of supply chain [4], have been introduced to integrate fuzzy *c*-means, genetic algorithms, and tabu searches for determining optimal clustering parameters that individual fuzzy *c*-means cannot produce. Another hybrid fuzzy clustering algorithm that combines fuzzy *c*-means and multivariate adaptive regression splines (MARS) [5] is introduced in bankruptcy forecasting. In it, clusters are created using fuzzy *c*-means and classified into two groups. A MARS model is then created using data from the two groups as part of input information. Basically, many hybrid approaches to combining fuzzy *c*-means with other methods aim to overcome fuzzy *c*-means limitations and enhance the advantages of integrated methods.

By incorporating a hybrid combination of self-adapted fuzzy *c*-means clustering [6] and the Newman-Girvan clustering algorithm [7], a method is presented to search for relationships in bibliographic big data by applying fuzzy concept. It accepts results from self-adapted fuzzy *c*-means clustering as part of the input information used for the Newman-Girvan clustering algorithm to produce in-depth results for providing quantitative information on how much data belongs to each cluster.

The proposed method uses visualization techniques in which the membership value of each dataset is displayed so as to retain fuzziness and thus prevent the loss of useful information. Visualization techniques provide an interactive network view by grouping objects with similar cluster memberships, showing connections between objects in each cluster and the strength of the relationships between objects by applying fuzzy concepts. The proposed method thereby increases the level of detail per retrieved result that conventional methods [1–3] without fuzzy logic cannot focus on positively. It assists users in making faster decisions by increasing the precision of the information displayed. The level of detail in visualization is critical to users because they either require information on specific items or simply must view the general characteristics of their searches [8].

The proposed method uses the DBLP computer science bibliography citation network dataset.[1] The two clustering algorithms, i.e., self-adapted fuzzy *c*-means and the Newman-Girvan algorithm, are applied to the dataset in Java. The proposed method implements the Java Univer-

1. http://arnetminer.org/DBLP_Citation [9, 10]

sal Network/Graph framework [11] for interactive visualization to provide functions that enable users to explore and manipulate search results.

Section 2 presents two target clustering algorithms, Section 3 touches on dataset DBLP, and Section 4 proposes visualization. Experiment results are shown in Section 5.

2. Hybrid Approach to Combining Self-Adapted Fuzzy *c*-Means Clustering and the Newman-Girvan Clustering Algorithm

A number of clustering algorithms are suitable for bibliographic big data. In the proposed method, two clustering algorithms are chosen for clustering purposes, namely self-adapted fuzzy *c*-means clustering and the Newman-Girvan clustering algorithm.

2.1. Self-Adapted Fuzzy *c*-Means Clustering

The fuzzy *c*-means algorithm is a powerful unsupervised method for data clustering [12]. In fuzzy *c*-means, data points on boundaries between clusters are not forced to fully belong to one cluster. They are instead assigned membership degrees between 0 and 1 to indicate their partial membership of each cluster. The membership degree is assigned to individual data points corresponding to each cluster center based on the distance between the cluster and the data point. The closer data is to the cluster center, the higher membership degree it has toward the particular cluster center. Fuzzy *c*-means clustering gives results for overlapped datasets, a feature not possible with other crisp clustering methods. In hard *k*-means, for example, individual data points must belong exclusively to one cluster center. In fuzzy *c*-means, however, individual data points is assigned a membership degree for each cluster center, so it can belong to more than one cluster center – hence giving more precise results than the *k*-means clustering algorithm.

Compared to other fuzzy clustering methods such as Gustafsone-Kessel [13] and Gath-Geva [14] algorithms, fuzzy *c*-means performs better by creating better-separated meaningful clusters with high compactness [15].

A big advantage of fuzzy *c*-means clustering is that it does not decide the absolute membership of a data point to a given cluster. Absolute membership is not calculated, which makes it extremely fast because the number of iterations required to achieve a specific clustering exercise corresponds to the required accuracy.

In the proposed method, membership degree information for individual data points resulting from fuzzy *c*-means clustering is suitable for use as a weighing factor when combined with the Newman-Girvan clustering algorithm. A sample of programming code is shown in **Fig. 1**.

One issue with fuzzy *c*-means is that the a priori specification of the number of clusters must be determined. To eliminate this issue, self-adapted fuzzy *c*-means has been introduced in which a new validity function is used where intercluster distances are as long as possible and intracluster distances are as short as possible.

```
public void calculate_centre_vectors(){
  int i,j,k;
  double numerator,denominator;
  double t[][] = new
double[MAX_DATA_POINTS][MAX_CLUSTER];
  for (i=0; i<num_data_points;i++){
      for(j=0;j<num_clusters; j++){
      t[i][j] =
    Math.pow(degree_of_memb[i][j],
    fuzziness);
      }
  }
  for(j=0;j<num_clusters;j++){
      for(k=0;k<num_dimensions;k++){
      numerator=0.0;
      denominator=0.0;
      for(i=0;i<num_data_points;i++){
      numerator +=
    t[i][j]*data_point[i][k];
      denominator += t[i][j];
      }
      cluster_centre[j][k] =
    numerator/denominator;
      }
  }
}
```

Fig. 1. Method for calculating vector centers in the fuzzy *c*-means clustering method.

The advantage of the self-adapted feature is that the initial number of clusters does not need to be determined prior to the clustering process. This feature is strongly useful for the proposed method because the number of clusters generated by the clustering algorithm is the stop criterion for the edge removal process in the Newman-Girvan clustering algorithm.

2.2. Newman-Girvan Clustering Algorithm

The second algorithm used in the clustering combination is the Newman-Girvan clustering algorithm. It has been hailed as the algorithm that marks the beginning of a new era in community detection field [16]. It works by selecting edges based on values of edge centrality measures and removing edges with the highest betweenness and splitting the network into isolated subgraphs until the network is broken into isolated single nodes. The steps of the algorithm are as follows:

Step 1 Calculate centrality for all edges.

Step 2 Remove the edge with the largest centrality. In case of ties with other edges, pick one edge randomly.

Step 3 Recalculate centrality on the running graph.

Step 4 Iterate the cycle from Step 2 until desired graphs are produced.

Newman-Girvan clustering predominates over other clustering methods because numerical studies have shown that its recalculation phase in Step 3 is important in detecting meaningful communities [17].

The Newman-Girvan algorithm has been applied many ways, and it is now integrated into many network analysis programs, such as Java Universal Network/Graph Framework,[2] and igraph.[3] It is chosen for the proposed method because it has succeeded in many applications to social and biological networks [18]. One issue arising from the Newman-Girvan algorithm is how to decide when to stop removing edges when a suitable number of clusters have been found. To do so, self-adapted fuzzy *c*-means finds sum of clusters automatically, thus giving this sum information to Newman-Girvan clustering as the criterion for stopping the iteration process in Newman-Girvan that removes edges with the largest centrality.

2.3. Combining Clustering Algorithms

A combination of clustering algorithms is proposed because different single clustering methods have their advantages and disadvantages, and combining clustering algorithms may produce better results than the best individual clustering algorithms.

The Newman-Girvan algorithm is deterministic, meaning that nodes lying at boundaries between communities may not be classified clearly. To eliminate this issue, self-adapted fuzzy *c*-means is introduced to classify data points using membership degrees. Membership degree values are used as weight criteria that give individual data points more value. The higher the membership degree of a data point, the more it belongs to a cluster. Automated cluster number generation by self-adapted fuzzy *c*-means clustering is the input for the criterion for stopping removing edges in the Newman-Girvan clustering algorithm.

To get these two valuable inputs, the self-adapted fuzzy *c*-means algorithm is first applied to the dataset. Membership matrix values and cluster numbers output from self-adapted fuzzy *c*-means clustering are used as additional input for the Newman-Girvan clustering algorithm, which is used in turn to classify data points into several clusters until the stop criterion is met and results are visualized at the end user interface. **Fig. 2** describes the architecture of bibliographic big data visualization.

3. DBLP Bibliographic Big Data

Bibliographic big data used to test the proposed method is the DBLP dataset. DBLP is a computer science bibliography website hosted by Universität Trier in Germany that provides bibliographic information on major computer science journals and proceedings. The DBLP dataset was selected for the proposed method because it is available for free and is one of the datasets most recently released for research purposes.

2. http://jung.sourceforge.net/
3. http://igraph.org/

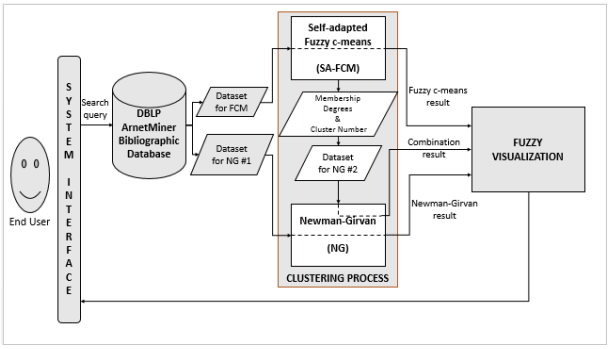


Fig. 2. Bibliographic big data visualization architecture.

```
#*QoSMap: Achieving Quality and Resilience through Ove...
#@Jawwad Shamsi,Monica Brockmeyer
#t2009
#cICIW
#index1259773
#%
#!

#*A Navigation over XML Documents through Linear Algeb...
#@Adriana Georgieva,Bozhidar Georgiev
#t2009
#cICIW
#index1259774
#%
#!

#*Towards a Disciplined Engineering of Adaptive Servic...
#@Nasreddine Aoumeur,Kamel Barkaoui
#t2009
#cICIW
#index1259775
#%
#!

#*A P2P Collaborative Bibliography Recommender System
#@Rushed Kanawati,Hager Karoui
#t2009
#cICIW
#index1259776
#%
#!
```

Fig. 3. Raw DBLP dataset prepared by ArnetMiner.

3.1. Citation Network Dataset by ArnetMiner

The dataset is provided by ArnetMiner, an online service used to index and search academic social networks [9, 10]. Located at the website of ArnetMiner,[1] it contains citation relationships between DBLP papers where each node is a paper from DBLP, and is further associated with abstract and citation relationships. **Fig. 3** shows the raw DBLP dataset.

A total of 492,550 unique authors have been identified from the 1,511,035 entries in the DBLP citation network dataset. There are at least 66,801 connections among these unique authors.

The DBLP dataset accessed on April 12, 2013, is Version 5 of the ArnetMiner dataset for DBLP citations up to February 21, 2011. It thus does not contain information on papers published after the release date.

Self-adapted fuzzy *c*-means expects an input data file in the format shown in **Fig. 4**. An example of a valid fuzzy *c*-means input file is shown in **Fig. 5**.

To use the Newman-Girvan clustering algorithm, the DBLP dataset must be converted to the Pajek NET for-

```
  Line 1:<number of data-points> <number of
dimensions>
  Line 2:<fuzziness coefficient>
<termination criterion>
  Line 3:<data points> ...
```

Fig. 4. Dataset format for fuzzy *c*-means clustering.

```
11 2
2.0 5.0E-4
1118466 773276 1118466 42 599096
642966 42 599096 642966 773276
```

Fig. 5. Dataset sample for fuzzy *c*-means clustering.

```
*Nodes
1 "Jose A. Blakeley"
2 "Yuri Breitbart"
3 "Stavros Christodoulakis"
4 "Umeshwar Dayal"
5 "Angelika Kotz Dittrich"
…
*Edgeslist
281 95 567
320 80 1656 1873 2922
333 1656 68 84
339 2719 587
355 237240 2041
358 63504
372 1656
…
```

Fig. 6. Dataset for Newman-Girvan clustering without weight information.

mat [19] shown in **Fig. 6**. Pajek files are text files whose individual lines are elements and whose list of edges follows the list of nodes. In Pajek format, nodes basically have one unique identifier and a label. The definition of nodes starts with the chain *Nodes N where N is the number of nodes following.

Edges are represented in one of two ways. In the first way of representing edges, the first identifier is the source node and all identifiers that follow are destination nodes. The dedicated marker is *Edgeslist. This way of representing edges is used in individual cases of Newman-Girvan clustering without weight information to enable a result comparison with the individual fuzzy *c*-means algorithm and the combination of both clustering algorithms. The dataset in **Fig. 6** consists of an entire DBLP dataset of 492,550 nodes in which labels are quoted directly after the node identifier.

In the second way of representing edges, edges are defined as a pair of node identifiers. The *Arcs marker goes before the pairs list. Weight is added by a third column, as shown in **Fig. 7**. In the proposed method, this type of dataset is used in the combination of both clustering algorithms. Membership degree results of self-adapted fuzzy

```
*Nodes 42
124812
158591
…
*Arcs
1       13      0.004125401364886321
1       14      0.004125401364886321
1       15      0.004125401364886321
…
```

Fig. 7. Dataset for Newman-Girvan algorithm with membership degrees as weights for individual data points.

c-means clustering are used as weights for individual data points for the Newman-Girvan clustering algorithm.

3.2. Bibliographic Big Data

Big data is useful in obtaining critical insights from data processing and behavior. The challenge when working with big data lies in finding a way to perform automated processing that is both extremely fast and produces meaningful results to users.

The large volume of DBLP citation network datasets is ideal for the proposed method because bibliographic data are by nature big and strongly mutually connected. The proposed method requires first that the user enter a keyword. The proposed method then quickly identifies which data matches user queries. To ensure that the clustering process returns results as quickly as possible, only relevant data is included in the clustering process. Irrelevant data is kept in the database for later use.

4. Fuzzy Visualization Using the Java Universal Network/Graph

4.1. Java Universal Network/Graph

The visualization of the clustering result is developed using the Java Universal Network/Graph Framework (JUNG) [11]. JUNG is a free open-source software library written in Java that provides a common, extendible language for manipulating, analyzing, and visualizing data is represented as a graph or network.

JUNG is used in developing the visualization part of the proposed method because it offers a simple, yet colorful and attractive way to construct tools for interactively exploring network data. It also enables visualization to be portrayed in an interactive network view. Nodes are clustered based on their membership degree, and connections are shown as edges between related nodes.

The proposed method uses two main classes in the JUNG library – the EdgeBetweennessClusterer and the WeakComponentClusterer [11]. The EdgeBetweennessClusterer class calculates clusters for a graph based on the betweenness property of edges The WeakComponentClusterer class finds all weak components in a graph. A weak component is defined as a subgraph in which each pair of nodes is connected by at least one undirected

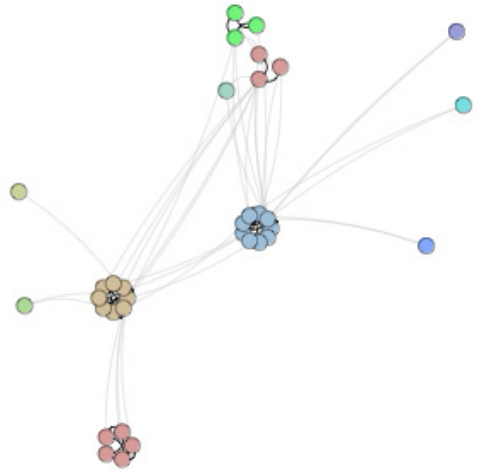


Fig. 8. Example of JUNG visualization.

graph.

Rendering clustering visualization uses node and edge color functions to create a visual contrast between elements. JUNG also enables users to focus on specific portions of a graph. This, in turn, enables users to interactively explore and manipulate search results to enhance their understanding of target results.

4.2. Fuzzy Visualization Requirements

Fuzzy-based clustering has visualization requirements different from crisp clustering because it brings more complexity to the visualization aspect. It is important for the user of such fuzzy systems to visualize and interpret information and its propagation effectively. The objective of applying fuzzy information visualization is to retain valuable and higher quality knowledge resulting from the fuzzy clustering method [20]. Showing membership degrees through visualization techniques enables search results to be displayed in ways that can help users focus more on target data and less on other data [21].

In the proposed method, search results are displayed in a graph view based on the Fruchterman-Reingold algorithm [22] as shown in **Fig. 8**. It incorporates two principles for graph drawing, i.e., that nodes connected by an edge should be drawn near each other and these nodes should not be drawn too close to each other. The Fruchterman-Reingold algorithm is good at distributing nodes evenly, making edge lengths uniform, and reflecting symmetry. It is chosen due to its simplicity and fast implementation speed. Graphs drawn using the Fruchterman-Reingold algorithm were drawn in less than a second, which is crucial to the proposed method because it must achieve its results within a time limit of 5 minutes of less.

In the proposed method, nodes represent the search category, such as the author, paper title, year, or publication venue. Edges represents the connections between individual nodes.

One desirable feature of visualization for fuzzy clustering results is the seamless integration of results and their fuzzy values. The visualization technique used here realizes this feature by applying different colors and brightness as intrinsic representations of nodes. Multiple dark colors are used to differentiate nodes based on their cluster. Light colors are used for nodes that do not belong to any cluster.

To further represent the relation of each node, different edge thicknesses show the strength of connections. This is determined by the membership degrees of individual nodes.

To facilitate decision making by users ideally, interactive functionalities that help users get more information would be offered. The proposed method enables user to click on individual nodes to find more detailed information.

5. Experiments in Journal Paper Retrieval from DBLP

5.1. Performance of Combination of Self-Adapted Fuzzy *c*-Means and the Newman-Girvan Algorithm

Experiments are conducted to confirm that combining self-adapted fuzzy *c*-means and the Newman-Girvan clustering algorithm performed better than individual clustering algorithms. To do this, self-adapted fuzzy *c*-means, the Newman-Girvan clustering algorithm and their combination applied to the DLBP citation network dataset prepared by ArnetMiner.

5.1.1. Comparisons in Fast Decision Making

The massive nature of bibliographic big data requires that the combination of the two clustering algorithms be optimized to ensure that several matched results based on user queries are obtained within 5 minutes or less.

Gelernter et. al. (2009) proposed a method that integrates fuzzy classification algorithms with an interface to visualize fuzzy results [23]. This method enables users to enter keyword(s) or to browse from categories of preclassified data. The classification algorithm is optimized to speedily return results when indexing is simplified by classifying individual items upon entry to the database rather than when a query is input. Upper levels of a hierarchy are saved together with the item to be speedmatched at query entry.

The proposed method, in dealing with bibliographic big data, needs a mechanism to reduce the number of nodes and edges to ensure that the visualization of results is not so overwhelming that users cannot comprehend it. It is crucial to ensure that only relevant information is displayed. When a user enters a keyword, the proposed method first selects matching data from the database. Data retrieval focuses only on data that matches the keyword, and retrieved data is converted to a format suitable for clustering purposes. Using this mechanism

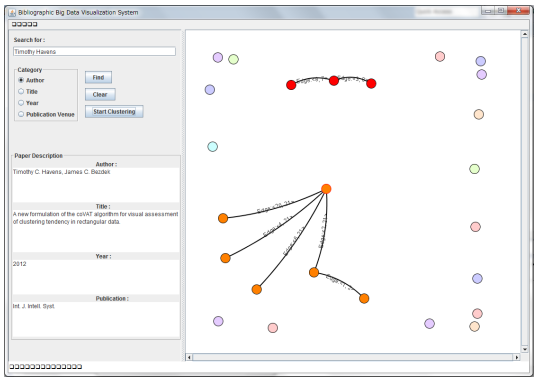


Fig. 9. Bibliographic big data retrieval method' user interface.

```
42 2
2.0 5.0E-4
1 1 1 1 1 1 3 3 3 3 3 3 3 3 3 3 …
13 14 15 16 17 18 1 18 19 20 …
```

Fig. 10. Dataset for self-adapted fuzzy *c*-means clustering based on the search keyword Andreas Neumann.

helps ensure that clustering algorithms are applied only to retrieved data instead of to the whole dataset.

5.1.2. Experiment Setup

To conduct the experiments, a software program was developed to get keyword input information from users and their preferred categories and to show the visualization of clustering results to users. The program was developed in Java and experiments were conducted in Eclipse IDE 4.2.2 [24] using a Dell Latitude E5430 laptop with Intel (R) Core (TM) i5-3210M at 2.50 GHz. A prototype of the software program is shown in **Fig. 9**.

5.1.3. Applying Clustering Algorithms to a DBLP Citation Network Dataset

To start the retrieval process, users first enter a keyword for the papers that they want to find in bibliographic big data. They must choose a search category, i.e., by author, title, publication year, or publication venue. Next, users they click on the Find button, then the method searches for related papers containing keywords. After all related papers have been gathered from the database, users select the Start Clustering button to begin the clustering process. Visualization results of the clustering process are shown at right in the user interface.

The five steps in the clustering process are as follows:

Step 1: Creating a dataset for self-adapted fuzzy *c*-means clustering

The keyword Andreas Neumann generates the dataset shown in **Fig. 10**. This dataset contains 42 data found in the DBLP citation network dataset related to the keyword Andreas Neumann.

Step 2: Applying the self-adapted fuzzy *c*-means algorithm

```
Membership matrix :
0.004125401364886321
0.004125401364886321
0.002428400884306878
…
```

Fig. 11. Membership matrix result from fuzzy *c*-means.

```
*Nodes 42
1       124812
2       158591
…
*Arcs
1       13      0.0041254013648863
1       14      0.0041254013648863
…
```

Fig. 12. Newman-Girvan dataset with weight from self-adapted fuzzy *c*-means membership degrees.

The self-adapted fuzzy *c*-means algorithm is applied to the dataset. A fuzzy *c*-means Java program is used in the process with self-adapted functions added to ensure that the number of clusters is set automatically. Program results are displayed in a membership matrix as shown in **Fig. 11**.

The membership degree of individual data points is used as a weight as added information in the dataset used in the Newman Girvan clustering algorithm. With weight information added, a more precise result is obtained when the second algorithm is applied to the dataset.

Step 3: Creating datasets for Newman-Girvan clustering

The dataset for the Newman-Girvan clustering algorithm was created after results had been produced by self-adapted fuzzy *c*-means. The membership matrix was used as weight for individual edges, and the number of clusters created was used to indicate when the Newman-Girvan algorithm should stop dividing graphs into smaller clusters.

Step 4: Applying the Newman-Girvan algorithm

The dataset in **Fig. 12** is used for the Newman-Girvan algorithm procedure. A Java program using the JUNG EdgeBetweennessClusterer library is used to process the algorithm. It calculates clusters in graphs based on edge betweenness in which the betweenness of an edge measures the extent to which that edge lies along the shortest paths between all pairs of nodes.

Edges that are least central to communities are progressively removed until communities have been adequately separated. It works by iteratively following a 2step process:

1. Calculating edge betweenness for all edges in current graphs
2. Removing edges with the highest betweenness

The iteration stop criterion is the number of cluster generated by the self-adapted fuzzy *c*-means algorithm. When the number of cluster generated using the Newman-Girvan algorithm is smaller than the number of clusters

generated by the self-adapted fuzzy c-means algorithm, iteration stops.

Step 5: Visualizing clustering results using fuzzy visualization

After both clustering algorithms have been applied to the dataset, visualization of results is realized by implementing Jung libraries in Java using the Frutcherman-Reingold algorithm, with fuzzy visualization features added to give the user more in-depth information on search results as compared to the usual crisp visualization features of current bibliographic visualizations.

5.1.4. Measures for Evaluating the Effectiveness of the Proposed Method

The dataset used for each experiment depends on keyword input information and then category selected by a user, ensuring that clustering results include almost all data that a user wants. To measure the effectiveness of the clustering combination, the precision, recall, and f-measure [24] of each clustering results are calculated using

$$\text{Precision}(P) = \frac{\#(\text{relevant papers retrieved})}{\#(\text{relevant papers})} \quad . \quad (1)$$

$$\text{Recall}(R) = \frac{\#(\text{relevant papers retrieved})}{\#(\text{retrieved papers})} \quad . \quad . \quad (2)$$

$$\text{F} - \text{measure} = \frac{2PR}{P+R} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (3)$$

Precision is defined as the proposed method's ability to retrieve papers that are mostly relevant. Recall is the ability of the proposed method to find all relevant papers in the database. F-measure is a harmonic mean that trades off precision versus recall. F-measure is also used to measure the method's performance because it gives an even weight to both precision and recall.

5.2. Clustering Results and Visualization

5.2.1. Comparison of Clustering Results by Using Self-Adapted Fuzzy c-Means Clustering, the Newman-Girvan Clustering Algorithm, and a Combination of Both Algorithms

Table 1 shows the number of clusters found using each of the individual clustering methods and their combination for the keyword Andreas Neumann. When the Newman-Girvan algorithm is applied the Andreas Neumann dataset, 6 clusters were found in 1 minute and 50 seconds. Self-adapted fuzzy c-means algorithms generate 2 clusters from the same dataset in 3 minutes and 24 seconds. When both algorithms combined are applied to the same dataset, it took 4 minutes and 8 seconds to find one cluster that strongly matches the search keyword by the user. **Table 2** shows experiment results for self-adapted fuzzy c-means clustering based on 9 search cases from 3 search categories by author, by title, and by publication venue.

Table 1. Experiment results for self-adapted fuzzy c-means clustering, the Newman-Girvan clustering algorithm, and the proposed method for the author search keyword Andreas Neumann.

Algorithm	*Clusters*	*Nodes*	*Time*
Newman-Girvan	6	24	1m 50s
Self-adapted fuzzy c-means	2	9	3m 24s
Combination	1	5	4m 08s

Table 2. Experiment results for self-adapted fuzzy c-means clustering algorithms.

Self-adapted fuzzy c-means algorithm			
Keywords	Time(s)	Clusters	Nodes
Author search			
#1: "Andreas Neumann"	204	2	9
#2: "Edward Omiecinski"	189	5	12
#3: "William Kent"	165	4	12
Title search			
#4: "Big Data"	61	3	11
#5: "Fuzzy Clustering"	259	8	36
#6: "Community Network"	94	3	18
Publication search			
#7: "Information Technology Management"	171	5	18
#8: "Wireless Communications Mobile Computing"	130	3	9
#9: "Scalable Computing: Practice and Experience"	156	2	9

Table 3. Experiment results for the Newman-Girvan clustering algorithm.

Newman-Girvan clustering algorithm			
Keywords	Time(s)	Clusters	Nodes
Author search			
#1: "Andreas Neumann"	110	6	24
#2: "Edward Omiecinski"	200	8	25
#3: "William Kent"	152	4	8
Title search			
#4: "Big Data"	63	4	16
#5: "Fuzzy Clustering"	265	11	58
#6: "Community Network"	98	5	31
Publication search			
#7: "Information Technology Management"	173	7	26
#8: "Wireless Communications Mobile Computing"	132	3	10
#9: "Scalable Computing: Practice and Experience"	159	3	11

Table 3 shows experiment results for the Newman-Girvan clustering algorithm and **Table 4** experiment results for the combination of both algorithms.

(1) Time Evaluation of Clustering Processes

Table 5 shows the mean and standard deviation of re-

Table 4. Experiment results for the combination of both clustering algorithms.

Combination of both algorithms			
Keywords	Time(s)	Clusters	Nodes
Author search			
#1: "Andreas Neumann"	248	1	5
#2: "Edward Omiecinski"	209	2	6
#3: "William Kent"	178	3	7
Title search			
#4: "Big Data"	70	2	9
#5: "Fuzzy Clustering"	271	6	32
#6: "Community Network"	102	2	12
Publication search			
#7: "Information Technology Management"	176	4	14
#8: "Wireless Communications Mobile Computing"	136	2	7
#9: "Scalable Computing: Practice and Experience"	161	1	4

Table 5. Response time comparison for the two clustering algorithms and their combination.

Clustering Algorithm	Mean	Standard Deviation
Self-adapted fuzzy *c*-means algorithm	158.78	64.92
Newman-Girvan clustering algorithm	150.22	59.72
Combination of both algorithms	172.33	214.75

Table 6. Precision, recall, and F-measure compared among algorithms and their combination.

Clustering Algorithm	***Precision***		***Recall***		***F-measure***	
	Avg	***Std***	***Avg***	***Std***	***Avg***	***Std***
Newman-Girvan	0.592	0.242	0.530	0.266	0.530	0.236
SA-FCM	0.633	0.109	0.706	0.104	0.658	0.058
Combination	0.751	0.103	0.711	0.107	0.724	0.073

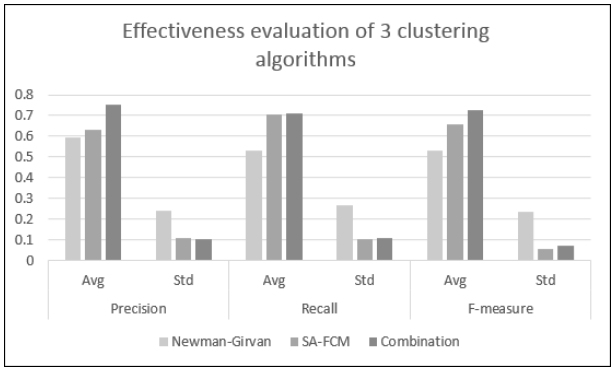


Fig. 13. Comparison of algorithms and their combination.

sponse times for the two clustering algorithms and their combination. The response time of the hybrid combination of self-adapted fuzzy *c*-means and the Newman-Girvan algorithm has the highest mean, 214.75 compared to 158.78 for self-adapted fuzzy *c*-means and 150.22 for the Newman-Girvan algorithm. This means that the combination tend to take longer to be completed than individual clustering algorithms. This is due to the extra processes that the combination must perform to get final results. The combination, however, also has a higher standard deviation of 214.72, compared to 64.92 for fuzzy *c*-means and 59.72 for the Newman-Girvan algorithm. This further means that time required to complete the algorithm combination varies from case to case due to the dataset size each time a search is done. The greater the volume of search results, the more time it takes gather all related data and process the algorithms. In a title search, for example, if keyword input information is related to a well-known research area such as fuzzy clustering, the keyword yields a massive amount of data requiring more time to process compared that for a lesser known research area.

Even though the mean time of the algorithm combination is the highest for the algorithms and their combination, the important point is that the combination focuses on several nodes that really match keyword input information. Individual clustering algorithms yielded larger numbers of clusters and total numbers of nodes, thus requiring more time and effort from users in examining each. This point reaches the target of the proposed method, that is to obtain a few target papers in an average 5 minutes or less from more than 1.5 million papers stored in the DBLP.

(2) Clustering Process Performance Evaluation

Table 6 shows mean and standard deviations for precision, recall, and f-measure among algorithms and their combination. **Fig. 13** compares the evaluation.

The average precision of the algorithm combination clearly takes the highest value, of 0.751, over the two individual algorithms. The combination also has the lowest standard deviation for precision, which means that the precision value for individual search cases varies only minimally.

The algorithm combination also takes the highest average recall value, 0.711, followed closely by self-adapted fuzzy *c*-means at 0.706. The standard recall deviation for the algorithm combination is slightly higher than that for self-adapted fuzzy *c*-means clustering. This means that the recall value for the algorithm combination varies slightly more than that for the self-adapted fuzzy *c*-means algorithm.

F-measure measures the proposed method's performance while taking into account both precision and recall. The results show that the average f-measure for the algorithm combination takes the highest value, 0.724, compared to 0.658 for self-adapted fuzzy *c*-means and 0.530 for the Newman-Girvan algorithm.

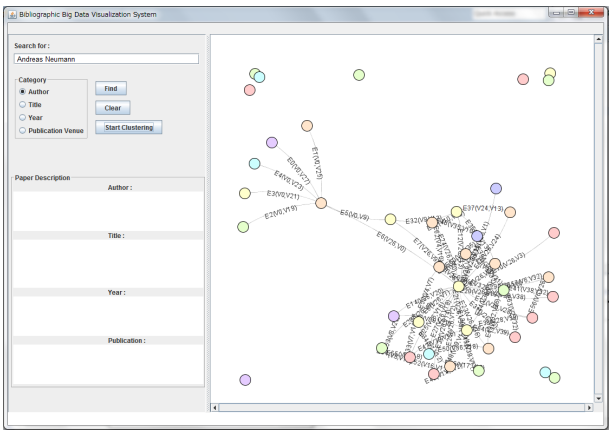


Fig. 14. Crisp relationship of the dataset based on the keyword Andreas Neumann.

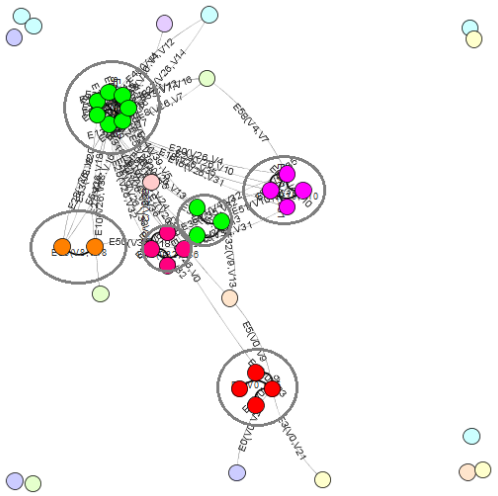


Fig. 15. Visualization of Newman-Girvan clustering results for the keyword Andreas Neumann.

Overall results show significant numerical improvement in the algorithm combination over the two separate algorithms.

From practical use point of view, this improvement is sufficient because the algorithm combination helps users focus on the strongly related results of their search. Less precise results are not clustered but are still available in visualization. This means that users can still select unclustered results and get the information they need, as desired.

To ensure that clustering results appealing to users, a prototype of the proposed method is now in planning to be tested using a user-based evaluation. This will test the prototype by selecting participants to perform a set of predetermined tasks on the prototype. A questionnaire given to participants after tasks are performedwill ask participants for their opinions on the prototype. The practical usability of the proposed method will then be determined from feedback results. This evaluation process is to be conducted when the prototype is completed.

5.2.2. Comparison of Visualization by Using Crisp Techniques and Fuzzy Techniques

Figure 14 shows the visualization relationship of a dataset based on the keyword Andreas Neumann. At left are search and paper description areas. At right is visualization of the clustering results area.

In the search area, users search for information by author, by title, by publication year, or by publication venue.

The paper description area at bottom left displays information on a user-selected node. The visualization area at right displays clustering results in a network view.

Visualization in **Fig. 14** displays the crisp relationship of papers related to the author Andreas Neumann. The search yielded 42 papers related to keyword input. This included papers by the author and that cites papers written by the author.

Initially, results show crisp connections among papers relevant to the search keyword before clustering is performed. Nodes are not yet clustered, so they are displayed in variety of light colors without uniformity.

Users press the Start Clustering button to perform clustering with algorithms to be applied to results.

In **Fig. 15**, clustering is based on the Newman-Girvan algorithm. After half of the number of edges is removed to find a desirable number of clusters, 6 clusters are generated from results. Five dark colors – green, red, magenta, pink, and orange – are used to paint nodes in individual clusters, The 2 clusters painted in green are located far from each other, indicating that they do not belong to the same cluster. Nodes not belonging to any cluster are painted in different light colors. Based on **Fig. 15**, the Newman-Girvan algorithm was found to be suitable for finding a desired number of clusters but was unable to focus on a few important clusters useful to the user.

The self-adapted fuzzy c-means clustering algorithm in **Fig. 16** is applied to the Andreas Neumann dataset. Two clusters were created automatically by applying this algorithm. The thickness of edges between nodes represents the strength between two nodes. Since the edges connecting the nodes are thin, these nodes are not strongly related to each other. The colors of the nodes are also nonuniform. Based on these two visual representations, users are able to get information on connections between individual nodes that are not strongly related.

Applying the combined self-adapted fuzzy c-means and the Newman-Girvan algorithm resulted in one main cluster consists of five nodes in dark red as shown in **Fig. 17**. These results clearly show that 5 papers are strongly related to keyword input information. Edges connecting nodes are thick, giving users visual information that papers are strongly related to each other. Two nodes connected to the cluster but shown in light inconsistent colors and the thin edges connecting them indicate to the user that the two nodes are not strongly related, unlike the five red nodes.

Figure 18 shows visualization results for the author

Fig. 16. Visualization of self-adapted fuzzy *c*-means clustering results for the keyword Andreas Neumann.

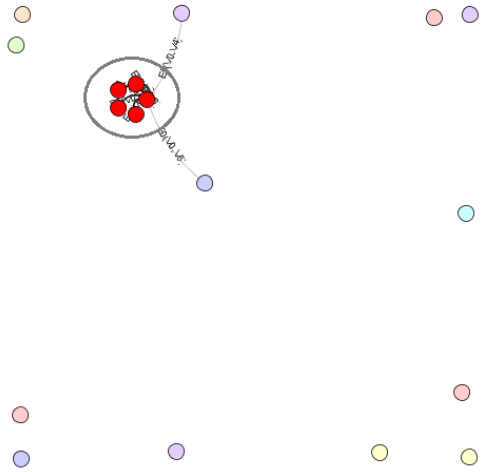


Fig. 17. Visualization of the algorithm combination for the keyword Andreas Neumann.

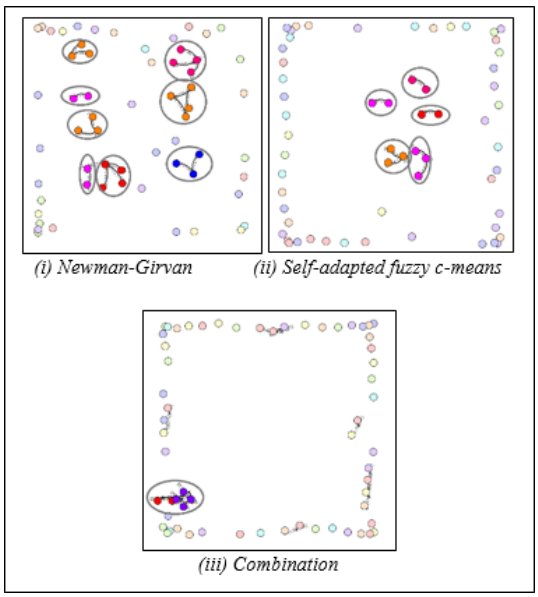


Fig. 18. Clustering results for the second author search using the keyword Edward Omiecinski.

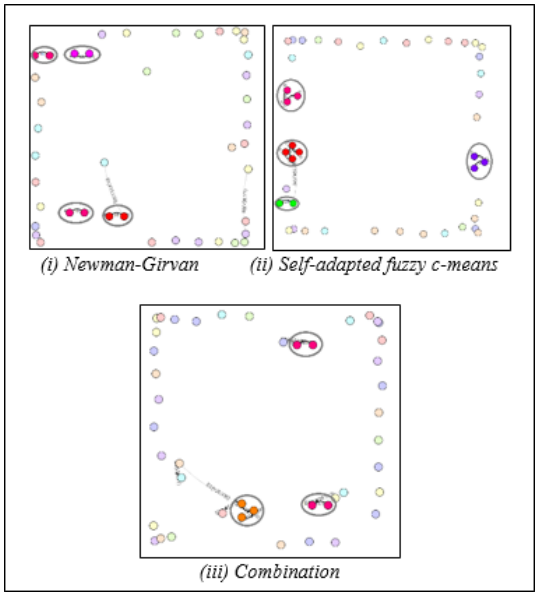


Fig. 19. Clustering results for the third author search using the keyword William Kent.

search using the keyword Edward Omiecinski using the Newman-Girvan algorithm, the self-adapted fuzzy *c*-means algorithm, and a combination of the two. **Fig. 19** visualizes clustering results for an author search using the keyword William Kent.

Clicking on individual nodes gives the user detailed information on papers such as the paper title, authors, publication year, publication venue, citation count, references, and an abstract. This detailed information helps users to make decisions on for further action, if any.

6. Conclusions

Results of the experiments have confirmed that combining two clustering algorithms and visualizing search results using fuzzy techniques enables users to converge a few target papers in an average 5 minutes from the 1.5 million papers stored in the DBLP.

Based on keyword input information from the user, search results are shown as a small network in which nodes indicate papers that match a keyword and edges indicates connections between nodes.

Nodes strongly related to the keyword are displayed so as to distinguish them from nodes having less relevance by using colors and different levels of the darkness of colors. The proposed visualization helps users focus on nodes that match the keyword most closely.

Selecting individual nodes gives the a more detailed ex-

planation of the papers displayed, such as the paper title, authors, publication year, publication venue, citation count, references, and an abstract of the paper.

The combination of two fuzzy clustering algorithms is intended to overcome the limitations of individual clustering algorithms. This does not necessarily mean, however, that the algorithm combination will produce the fastest result every time. To further improve the proposed method, it should be able to automatically compare results for individual algorithms as well as the algorithm combination to find which yields the fastest result.

Bibliographic big data sets challenges for creating visualization that display results of exploration in a way that does not overwhelm users but helps them explore data easily. Results are also made available quickly to speed up decision making. Modeling fuzzy information on visual elements has been introduced to solve problems in visualizing relationships across a variety of knowledge domains.

Users targeted to apply the proposed method are researchers, educators, and students using realworld social and biological network. The proposed method is planning to be opened to the public over the Internet.

References:

[1] N. Elmqvist and P. Tsigas, "CiteWiz: a tool for the visualization of scientific citation networks," J. of Information Visualization, Vol.6, No.3, pp. 215-232, 2007.

[2] Z. Shen, M. Ogawa, S. T. Teoh, and K. Ma, "BiblioViz: A System for Visualizing Bibliography Information," Proc. of the Asia Pacific Symp. on Information Visualization (APVIS '06), Vol.60, pp. 93-102, 2006.

[3] A. Brüggemann-klein, R. Klein, and B. Landgraf, "BibRelEx: Exploring Bibliographic Databases by Visualization of Annotated Contents-Based Relations," Int. Conf. on Information Visualization, Vol.5, No.11, pp. 19-24, 2000.

[4] X. F. Yin, L. P. Khoo, Y. T. Chong, "A fuzzy *c*-means based hybrid evolutionary approach to the clustering of supply chain," Comput. Ind. Eng., Vol.66, No.4, pp. 768-780, 2013.

[5] J. D. Andrés, P. Lorca, and F. J. Cos Juez, "Bankruptcy forecasting: A hybrid approach using Fuzzy *c*-means clustering and Multivariate Adaptive Regression Splines (MARS)," Expert Syst. Appl., Vol.38, No.3, pp. 1866-1875, 2011.

[6] J. Jin, Y. Liu, L. T. Yang, N. Xiong, and F. Hu, "An Efficient Detecting Communities Algorithm with Self-Adapted Fuzzy *C*-Means Clustering in Complex Networks," IEEE 11th Int. Conf. on Trust, Security and Privacy in Computing and Communications (TrustCom), Vol.1988, No.1993, pp. 25-27, 2012.

[7] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks, Proceedings of the National Academy of Sciences," Vol.99, No.12, pp. 7821-7826, 2002.

[8] A. S. Ehikioya, "A Characterization of Information Quality Using Fuzzy Logic," Fuzzy In-formation Processing Society, NAFIPS. 18th Int. Conf. of the North American, pp. 635-639, 1999.

[9] J. Tang, J. Zhang, L. Yao, L. Li, L. Zhang, and Z. Su, "ArnetMiner: Extraction and Mining of Academic Social Networks," Proc. of the 14th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (SIGKDD'2008), pp. 990-998, 2008.

[10] J. Tang, D. Zhang, and L. Yao, "Social Network Extraction of Academic Researchers," Proc. of 2007 IEEE Int. Conf. on Data Mining (ICDM'2007), pp. 292-301, 2007.

[11] Java Universal Network Graph http://jung.sourceforge.net

[12] J. C. Bezdek, "Pattern Recognition with fuzzy objective functions algorithms," New York: Plenum Press, 1981.

[13] E. E. Gustafson, and W. C. Kessel, "Fuzzy clustering with a fuzzy covariance matrix," pp. 761-766, IEEE CDC, 1979.

[14] I. Gath and A. B. Geva, "Unsupervised optimal fuzzy clustering," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.11, No.7, pp. 773-781, 1988.

[15] A. T. Azar, S. A. El-Said, and A. E. Hassanien, "Fuzzy and hard clustering analysis for thyroid disease," Computer Methods Programs Biomed, Vol.111, No.1, pp. 1-16, 2013.

[16] S. Fortunato, "Community Detection in Graphs," Physics Reports, Vol.486, No.3-5, pp. 75-175, 2010.

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

[18] R. Guimerà and L. A. Nunes Amaral, "Functional cartography of complex metabolic networks," Nature, Vol.433, No.7028, pp. 895-900, 2005.

[19] V. Batagelj and A. Mrvar, "Pajek datasets," http://vlado.fmf.uni-lj.si/pub/networks/data/, 2006.

[20] B. Pham, A. Streit and R. Brown, "Visualization of Information Uncertainty: Progress and Challenges," Trends in Interactive Visualization, pp. 19-48, 2009.

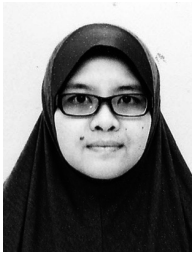
[21] B. Pham and R. Brown, "Analysis of Visualization Requirements for Fuzzy Systems," Proc. of the 1st Int. Conf. on Computer Graphics and Interactive Techniques in Australasia and South East Asia, Vol.1, No.212, pp. 181-187, 2003.

[22] T. M. J. Fruchterman and R. M. Reingold, "Graph Drawing by Force-directed Placement," Software Practice and Experience, Vol.21, No.11, pp. 1129-1164, 1991.

[23] J. Gelernter, D. Cao, R. Lu, E. Fink, and J. G. Carbonell, "Creating and visualizing fuzzy document classification," Proc. of the 2009 IEEE Int. Conf. on Systems, Man and Cybernetics (SMC'09), pp. 672-679, 2009.

[24] Eclipse IDK 4.2.2, http://www.eclipse.org

[25] C. D. Manning, P. Raghava, and H. Schutze, "Introduction to Information Retrieval," pp. 151-161, Cambridge University Press, 2008.



Name:
Maslina Zolkepli

Affiliation:
Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology

Address:
G3-49, 4259 Nagatsuta, Midori-ku, Yokohama, Kanagawa 226-8502, Japan

Brief Biographical History:
2008-2010 M. Sc. in Computer Science, Universiti Putra Malaysia
2012- Ph.D. Student, Tokyo Institute of Technology

Main Works:
• M. Zolkepli, F. Dong, and K. Hirota, "Visualization of Fuzzy Relationships using Clustering Algorithms on Bibliographic Big Data," Int. Symposium on Advanced Intelligent Systems, Daejeon, Korea, 2013.

Membership in Academic Societies:
• Japan Society for Fuzzy Theory and Intelligent Informatics (SOFT)

Name:
Fangyan Dong

Affiliation:
Associate Professor, Education Academy of Computational Life Sciences, Tokyo Institute of Technology

Address:
G3-49, 4259 Nagatsuta, Midori-ku, Yokohama 226-8502, Japan

Brief Biographical History:
2006-2014 Assistant Professor, Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology
2014- Associate Professor, Education Academy of Computational Life Sciences, Tokyo Institute of Technology

Main Works:
• F. Dong, K. Chen, E. M. Iyoda, H. Nobuhara, and K. Hirota, "Solving Truck Delivery Problems Using Integrated Evaluation Criteria Based on Neighborhood Degree and Evolutionary Algorithm," J. of Advanced Computational Intelligence and Intelligent Informatics, Vol.8, No.3, pp. 336-345, 2004.
• F. Dong, K. Chen, and K. Hirota, "Computational Intelligence Approach to Read-world Cooperative Vehicle Dispatching Problem," Int. J. of Intelligent Systems, Vol.23, pp. 619-634, 2008.

Membership in Academic Societies:
• Japan Society for Fuzzy Theory and Intelligent Informatics (SOFT)
• The Japanese Society for Artificial Intelligence (JSAI)

Name:
Kaoru Hirota

Affiliation:
Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology

Address:
G3-49, 4259 Nagatsuta, Midori-ku, Yokohama 226-8502, Japan

Brief Biographical History:
1982-1995 Professor, College of Engineering, Hosei University
1995- Professor, Tokyo Institute of Technology

Main Works:
• M. L. Tangel, C. Fatichah, M. R. Widyanto, F. Dong, and K. Hirota, "Multiscale Image Aggregation for Dental Radiograph Segmentation," J. of Advanced Computational Intelligence and Intelligent Informatics, Vol.16, No.3, pp. 388-396, May 2012.
• M. Dai, F. Dong, and K. Hirota, "Fuzzy Three-Dimensional Voronoi Diagram and its Application to Geographical Data Analysis," J. of Advanced Computational Intelligence and Intelligent Informatics, Vol.16, No.2, pp. 191-198, March 2012.

Membership in Academic Societies:
• The Institute of Electrical and Electronics Engineers (IEEE)
• International Fuzzy Systems Association (IFSA), Fellow, Immediate-Past-President
• Japan Society for Fuzzy Theory and Intelligent Informatics (SOFT), Past-President