TAF Journal of Advances in Technology and Engineering Research JATER
2019, **5**(1): 25-36

PRIMARY RESEARCH

Website evaluation using cluster structures

Kiyoshi Nagata*

Faculty of Business Administration, Daito Bunka University, Tokyo, Japan

Keywords

Kernel *k*-means clustering
Link-based similarity
Term-based similarity
Spectral analysis

Received: 5 December 2018
Accepted: 9 January 2019
Published: 20 February 2019

Abstract

Since the early 1990s, when frequent commercial use of the Internet started, academic researchers and website practitioners have actively conducted research on websites. The research's three broad categories are Web content mining, Web structure mining, and Web usage mining; apart from those, some research on coloring, placement technique of images, texts, and links on each page. This paper focuses on the difference between two cluster structures of websites, one induced from the link-based property and the other from the term-based property. The link-based property is stable until a new link is added, but the term-based one varies depending on the items for searching. We propose an evaluation method for the website by comparing the structures of clusters resulted from these properties. Here we adopt kernel k-means method as the clustering method and compare partial clusters derived from term-based property depending on the given sequence of particular terms to definite partial clusters from the link-based property. To distinguish them, we try to adopt spectral analysis.

I. INTRODUCTION

Since commercial use of the internet began early in 1990s, information dissemination centering on websites has been done in many fields. In about 30 years, various techniques for Internet-based environment is developed and put to practical use, which gives websites the leading role in worldwide information dissemination. However, these years are also the period when unnecessary and meaningless information is accumulated on the website. Like as a database system that is not well maintained, websites within an organization may still keep information that needs to be expired or deleted, and there may be inappropriate links to them or from them. It is because the network administrator of the organizational website and the person in charge of each department's home-page management can not properly manage pages, and there is no system for looking over and improving the whole site.

Regarding website links and described content, active research on search engines has been conducted and put to practical use. For example, as a hierarchical network search engine, HyPersuit proposed by [1], is well known and pioneering software in which content-link hyper-text clustering is exploited to obtain effective search results. The content-link hypertext clustering organizes documents into clusters of related documents based on the hybrid function of hyperlink and terms related structures. The hyperlink structure is described as the value of link similarity function of each of two nodes, and the term structure is described as the value of term-based similarity function using term frequency, size of document, and the term attributes. Chaomei Chen published a paper for analyzing the structure of a large hypermedia information space based on three types of similarity measures such as hypertext linkage, content similarity, and usage patterns [2, 3]. He tried to describe the network or its sub-networks as a graphical image of their Pathfinder Networks models, [4], based on pairwise integrated similarity by applying the vector space model [5, 6]. The primary purpose of our research is to develop an analyzing system for website from various perspectives with which website managers or webpage designers could improve the whole website by removing or adding proper links and pages. Of course, there is a problem as to what kind of website is good. Although excluding unnecessary links or information might be one of good website require-

*Corresponding author: Kiyoshi Nagata
†email: nagata@ic.daito.ac.jp

ments, it is not always true from the user's point of view. We already proposed in [7, 8] website evaluation system by combining some of website indexes and user evaluation indexes by applying user's perspective information quality measures, [9, 10, 11]. In that research, we tried to construct a formula describing some of information quality measures as a formula with link or term based indexes. As an illustrative example, some coefficients of linear regression model between index values from a few dozens of pages in each website and information quality index obtained from the result of the factor analysis of questionnaire survey responses. However, the data volume was insufficient for both explanatory variables and objective variables, and the obtained result was not acceptable for practical use.

In this paper, while the user's perspective evaluation is not involved, we create an application software written in Java language for collecting data from real website and analyzing link and term related properties. The rest of the paper organized as follows. Some of link and term related indexes are introduced in the next section, and analyzing method implemented or will be implemented in our application program are explained in the following. Some experimental results are described, then discuss them. The last part is the conclusion and about the future works.

II. INDEXES ON WEBSITE

Here we refer to two types of indexes on website, one is those related to the link structure and the other is those related to the term structure.

A. Link Related Index

Among various link related indexes proposed so far, Compactness and Stratum represent site-wide characteristics, Hub Weight and Authority Weight as characteristics of each node, and Complete Hyper Link Similarity (CHLS) represents the similarity of two nodes in terms of characteristics of links.

For an explanation of these indexes above, here we give assumptions and notations. Let $\{P_1, ..., P_N\}$ be a set of entire pages as nodes at the target site, and put a_{ij} = 1 if there is a direct link from node P_i to node P_j , otherwise put a_{ij} = 0, then consider the adjacent matrix *A* = (a_{ij}). Applying an appropriate method to this *A*, find c_{ij} the shortest distance from P_i to P_j, i.e., the minimum number of edges that must be traversed to reach. In case of being no route to reach, set a predefined sufficiently large number *K* for the value of c_{ij}.

1) Compactness and stratum: As two indicators of site-wide status, here we refer to Compactness (*Cp*) and Stratum (*St*). Compactness is close to 1 if the nodes are densely connected, and is close to 0 otherwise, and Stratum is close to 1 when the whole node has close serial connections and is close to 0 otherwise. Therefore, these are complementary indexes, but as shown in (1) and (2), it is difficult to understand the clear relationship from their defining expressions.

$$C_p = \frac{K}{K-1} - \frac{\sum_{i,j} c_{ij}}{(N^2-N)(K-1)} \tag{1}$$

$$St = \frac{\sum_i |OD_i - ID_i|}{LAP} \quad \text{where } OD_i = \sum_{j=1}^N c_{ij}, ID_i = \sum_{j=1}^N c_{ji}, \tag{2}$$

$$LAP = \begin{cases} \frac{N^3}{4} & (\text{ if } N \text{ is even }) \\ \frac{N^3-N}{4} & (\text{ if } N \text{ is odd }) \end{cases}$$

2) Hub weight and authority weight: For the importance value of each node as a referring or referred one in a directed graph, Kleinberg's hub and authority weights are very popular [12]. Each component of the principal eigenvectors (eigenvector for the greatest positive real eigenvalue) of the matrices AA^t and A^tA represents the Hub Weight and the Authority Weight of the corresponding node.

In a homepage site, the number of nodes is several hundred to several thousand, or it exceeds 10,000 in some cases. These calculations are not so easy, thus we have not implemented them in our application software yet.

3) CHLS: The index proposed by Weiss et al. expresses the similarity between given two nodes by the link relation with other nodes and can be obtained by taking the weighted average of the following three values, [1].

$$\begin{aligned} S_{ij}^{spl} &= \frac{1}{2^{c_{ij}}} + \frac{1}{2^{c_{ji}}} \\ S_{ij}^{anc} &= \sum_{x \in A_{ij}} \frac{1}{2^{\left(c_{xi}^{\bar{j}} + c_{xj}^{\bar{i}}\right)}} \\ S_{ij}^{dsc} &= \sum_{x \in D_{ij}} \frac{1}{2^{\left(c_{ix}^{\bar{j}} + c_{jx}^{\bar{i}}\right)}} \end{aligned} \tag{3}$$

Here, A_{ij} is a set of nodes such that there is at least one path to both P_i and P_j, D_{ij} is a set of nodes such that there is at least one path from both P_i and P_j, and $c_{xi}^{\bar{j}}$ is the shortest distance from P_x to P_i not passing P_j .

From these values and weights w_s, w_a, w_d, the CHLS index S_{ij}^{link} of each node pair (P_i, P_j) is defined as follows.

$$S_{ij}^{link} = w_s S_{ij}^{spl} + w_a S_{ij}^{anc} + w_d S_{ij}^{dsc}. \tag{4}$$

4) SG scores and graph spectrum: Shoda et al[13] considered all the connected sub-graph and calculate the total weight of each of them, then proposed to visually evaluate the similarity by graphing their frequency of appearance as the spectrum.

For a weighted non-directed graph *G* and the set of weight $\{w(P)\}$, they considered all the connected sub-graphs $\{SG \in 2^G; SG\}$ is connected and calculate the total weight of all node in *SG* as $\boldsymbol{w(SG)} = \sum_{P \in SG} \boldsymbol{w}(P)$ for each *SG*. Then, the graph spectrum is defined as the vector with component values of numbers of *SGs* whose weight are corresponding to the index number. In the paper, the graph spectrum is used to calculate the structural similarity of clusters and apply *k*-means method to find a good cluster decomposition.

Here we propose the in- and out-weight of connected subgraphs in a directed graph. For a fixed weights (w_1, w_2) with $w_1 < w_2$, the in-weight of a subgraph of only two nodes $\{P_1, P_2\}$ is calculated as the weighted value $\boldsymbol{w}_{P_1 \to P_2} = \boldsymbol{w}_1 \boldsymbol{w}\left(P_1\right) + \boldsymbol{w}_2 \boldsymbol{w}\left(P_2\right)$ if there is a direct path from P_1 to P_2. Then define the in-weight for *SG* of two nodes, called twin in-weight, to be the average of $w_{P_1 \to P_2}$ and $w_{P_2 \to P_1}$. When *SG* has more than two nodes, the in-weight can be defined as the average of all the twin in-weights for connected twin subsets. However, the calculation efforts increase exponentially proportion to the number of nodes in *SG*.

B. *Term Related Index*

In order to define term related indexes for a set of fixed query terms *q* = $\{Q_1, ..., Q_M\}$, we need the occurrence of Q_j in a page *P* denote by X_{PQj} $(= \ 0$ or $1)$, the term frequency, the number of $Q_j S$ appear in a page *P*, denoted by TF_{PQj}, , and the maximum number of TF_{P,Q_j}, through all terms $Q_j (j = 1, ..., M)$ denoted by $TF_{P,max}$.

1) Boolean and vector spread activation: For some fixed values k_1 and k_2 satisfying 0 < k_2 < k_1, the Boolean spread Activation $R_{P,q}$ is defined:

$$R_{P,q} = \textstyle\sum_{j=1}^{M} I_{P,Q_j}$$

where

$$I_{P,Q_j} = \begin{cases} k_1 & \text{if } X_{P,Q_j} = 1 \\ k_2 & \text{if } X_{P'}, Q_j = 1 \text{ for some } P' \neq P \\ & \text{and } c_{PP'} (< K \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

The vector spread activation is defined by

$$RV_{P,q} = S_{P,q} + \textstyle\sum_{P' \neq P} \alpha a_{PP'} S_{P',q'}$$

where the reduced version of TFxIDF ($S_{P',q}$) is defined as follows,

$$S_{P,q} = \sum_{j=1}^{M} w_{P,Q_j}^{TF} \times IDF_{Q_j},$$

With w_{P,Q_j}^{TF} and the "Inverse document frequency" IDF_{Qj},

$$w_{P,Q_j}^{TF} = \frac{1}{2}\left(1 + \frac{TF_{P,Q_j}}{TF_{P,max}}\right)$$

$$IDF_{Qj} = log\left(\frac{N}{\sum_{P'} X_{P',Q_j}}\right)$$

2) Term similarity index: In order to measure a similarity of two pages P_i and P_j for a particular set of search terms, the following index is defined as the term-based similarity index,

$$S_{P_i,P_j}^{terms} = \frac{\sum_{l=1}^{M} w_{P_i,Q_l} w_{P_j,Q_l}}{\sqrt{\sum_{l=1}^{M} w_{P_i,Q_l}^2 \sum_{l=1}^{M} w_{P_j,Q_l}^2}} \tag{6}$$

where $w_{P_i,Q_l} = w_{P_i,Q_l}^{TF} w_{P_i,Q_l}^{at}$ with

$$w_{P,Q_j}^{at} = \begin{cases} 10 & \text{if } Q_j \text{ is in title of } P, \\ 5 & \text{if } Q_j \text{ is in headers or keywords} \\ & \text{or address in } P, \\ 1 & \text{otherwise,} \end{cases}$$

III. CALCULATING AND ANALYZING METHOD

Although the matrix *A* can be obtained by examining the direct link relationship between nodes, we must find the shortest distance c_{ij} from the node P_i to the node P_j for carrying out the calculation of the values of Cp, St, and S^{link}.

For clarifying a given website as a directed graph, clustering methods are usually applied. Two types of clustering methods using both link-and term-related similarities are implemented in our program, and some others are planned to implement.

A. *Shortest Pass Calculation*

The problem of finding the shortest distance from a given node to another node is called the Single Source Shortest Pass (SSSP) problem, and the Dijkstra method [14] is well known as an algorithm to this problem. The original method required $O(n^2)$ execution time, and now improved to $O(e + nlog(logn))$ (*e* is the number of edges), [15].

On the other hand, the problem of finding the shortest distance for all node pairs at once is called the problem of All Pairs Shortest Pass (APSP). The Floyd-Warshall method

[16, 17] requires $\Theta\left(n^3\right)$ complexity and a memory area proportional to $n2$ are required. The algorithm is simple and easy to implement to the program and has the advantage of being able to keep the shortest path. Thus, we use the Floyd-Warshall Method of the following procedure.
1. Let $C^{(0)}(=A)$ be a matrix representing (weighted) direct link relationships.
2. Node $P_k(k=1,...,N)$ is sequentially added as a transit node, and the (weighted) shortest distance changed for each node pair (P_i,P_j) is a component of the matrix $C^{(k)}$. When the shortest distance changes, P_k is stored as a predecessor node from P_i to P_j .
Furthermore, the calculation of the remaining two index values in Equation (4) requires the conditional shortest distance $\boldsymbol{c}^{\bar{j}}_{x\boldsymbol{i}}$. Simple way for this task is to apply the algorithm to the matrix $A^{(j)}$ which is obtained by excluding the row and column corresponding to the node P_j from the whole, but processing takes time in the case of a large number of nodes. In [18] we proposed a method for finding the conditional shortest distance when the shortest distance matrix $C=(c_{ij})$ is given for all nodes of a directed graph whose edge weights are all 1.
1) Proposition 1: Suppose that a shortest distance matrix $C=(c_{xz})$ for a directed graph (*G,E*) with edge weights all 1. For any node pair (x,z) $(x,z\in G)$, put $c_{xz}=\infty$, when there is no path from x to z.
For any $x,y,z\in G$ with $c_{xz}<\infty$, let $c^{\bar y}_{xz}$ be the shortest distance from x to z that does not pass through y. Then following holds.
(i) If $c_{xz}{\neq}c_{xy}+c_{yz}$, then $c^{\bar y}_{xz}$ = c_{xz}.
(ii) If $c_{xz}=c_{xy}+c_{yz}$, and any y' different from *y* satisfies $c_{xy'}{\neq}c_{xy}$, then $c_{xz}=\infty$.
(iii) When there exists $c_{xz}=c_{xy}+c_{yz}$, for y' giving $m=min\{c_{y'z}-c_{yz};y'{\neq}y,c_{xy'}=c_{xy}\}$,
• If $c_{y'y}>m$, then $c^{\bar y}_{xz}$ = $c_{xz}+m$
• If $c_{y'y}$ = m, then necessary and sufficient condition for $c^{\bar y}_{xz}$ = $c_{xz}+m$ is that there is a y'' different from *y* satisfying $c_{y'y''}=m$ and $c_{y''z}=c_{yz}$
(Proof).
If there is a shortest distance through *y*, then *y* will be its relay node, so $c_{xz}=c_{xy}+c_{yz}$, so (i) is obvious.
When $c_{xz}=c_{xy}+c_{yz},y$ is a node on the path giving the shortest distance. Since the weight of each edge is 1 there exists a node y' for which $c_{xy'}=c_{xy}$ on the shortest path not passing *y*, thus (ii) is also valid. When there is a node y' such that $c_{xz}=c_{xy}+c_{yz}$ and $c_{xy'}=c_{xy}$, then $c_{y'z}{\geq}c_{yz}$. If y is on the shortest path from y' to z, then $c_{y'y}=m$ holds from the condition $m=c_{y'z}-c_{yz}$. Together with $c_{y'y}>m$, we have $c^{\bar y}_{xz}$ = $c_{xz}+m$. In case that $c_{y'y}=m$, we can determine $c^{\bar y}_{xz}$ = $c_{xz}+m$ depending on whether there is a node or not different from *y* on the route giving the shortest distance $c_{y'z}$ and distance from y' is *m*.

B. *Clustering Methods*

The clustering is a very important concept for the data structure analysis as data mining, and there are many proposals for general purpose algorithms and for specific data. Depending on the difference in data structure that is output as a result, they are divided into hierarchical algorithm and non-hierarchical one. The former outputs dendrogram and the latter outputs cluster set.
When a distance is well-defined in data space, clustering is performed by the distance, but when it is difficult to define the distance like nodes on a website, the similarity between two nodes as seen in the previous section is used.
Here we refer to two of non-hierarchical clustering algorithms, one is the kernel version of the popular k-means algorithm and the other is SCAN.
1) Kernel K-means and SCAN: We quote MacQueen's commentary on *k*-means procedure in [19], "Stated informally, the k-means procedure consists of simply starting with *k*-groups each of which consists of a single random point, and thereafter adding each new point to the group whose mean the new point is nearest. After a point is added to a group, the mean of that group is adjusted in order to take account of the new point. Thus, at each stage the *k*-means are, in fact, the means of the groups they represent (hence the term *k*-means)."
Dhillon et al.,[20], consider the weighted kernel k-means clustering and show the connection between it and spectral clustering algorithm. Before describing a formula for the kernel *k*-means with feature map ϕ from a set of nodes *G* to the Hilbert space H of functions on *G* over *R*, let us review the definition of a kernel.
A positive definite kernel is a map

$$k:G\times G\rightarrow R$$

satisfying
• $k(x,y)=k(y,x)$ for any $x,y\in G$
• For any $\{c_i\}\subset R$ and any $\{x_i\}\subset G$,

$$\sum c_i c_j k\left(x_i,x_j\right)\geq 0$$

Now go back to the kernel k-means algorithm. For a given number of clusters k, we try to find a set of clusters $C_1,...,C_k$ minimizing the following value.

$$\sum_{i=1}^{k}\sum_{x\in C_i}\left\|\phi(x)-\mu_i\right\|^2, \tag{7}$$

ISSN: 2414-4592
DOI: 10.20474/jater-5.1.3

Where

$$\begin{aligned}\|\phi(x)-\mu_i\|^2 =&\left\|\phi(x)-\frac{1}{|C_i|}\sum_{x'\in C_i}\phi\left(x'\right)\right\|^2\\ =&k(x,x)-\frac{2}{|C_i|}\sum_{x'\in C_i}k\left(x,x'\right)\\ &+\frac{1}{|C_i|^2}\sum_{x''\in C_i}\sum_{x'\in C_i}k\left(x'',x'\right)\end{aligned}$$

Xu et al.,[21], proposed SCAN (Structural Clustering Algorithm for Networks), in [21] which, outputs three types of clusters such as "hub", "outlier", and ordinal clusters, by using structural similarity and two parameters 0 "≤e ≤1" and $\mu \in \mathbf{Z}^+$.

The structural similarity of two nodes x and x' is a number between 0 and 1 with value 1 only when $x = x'$. A node x is called a core with respect to e and μ, if x has at least μ number of nodes to which the structural similarity is greater than e. Two nodes x and x' are connected when there exists one core x'' from which they can be reached by following cores. Then output clusters are maximum subset of nodes any two of which are connected with each other. The hub is a node not belonging to any cluster and there are at least two distinct clusters each of which has a node with edge to it. When a node not belonging to any cluster does not satisfy the condition for the hub, it is called an outlier.

IV. OUTLINE OF OUR APPLICATION PROGRAM

Now we describe an outline of the developing application implementing the index calculation and the analyzing methods explained in the previous sections. The initial window is shown in the Figure 1. below composed of three essential panes and the menu bar.

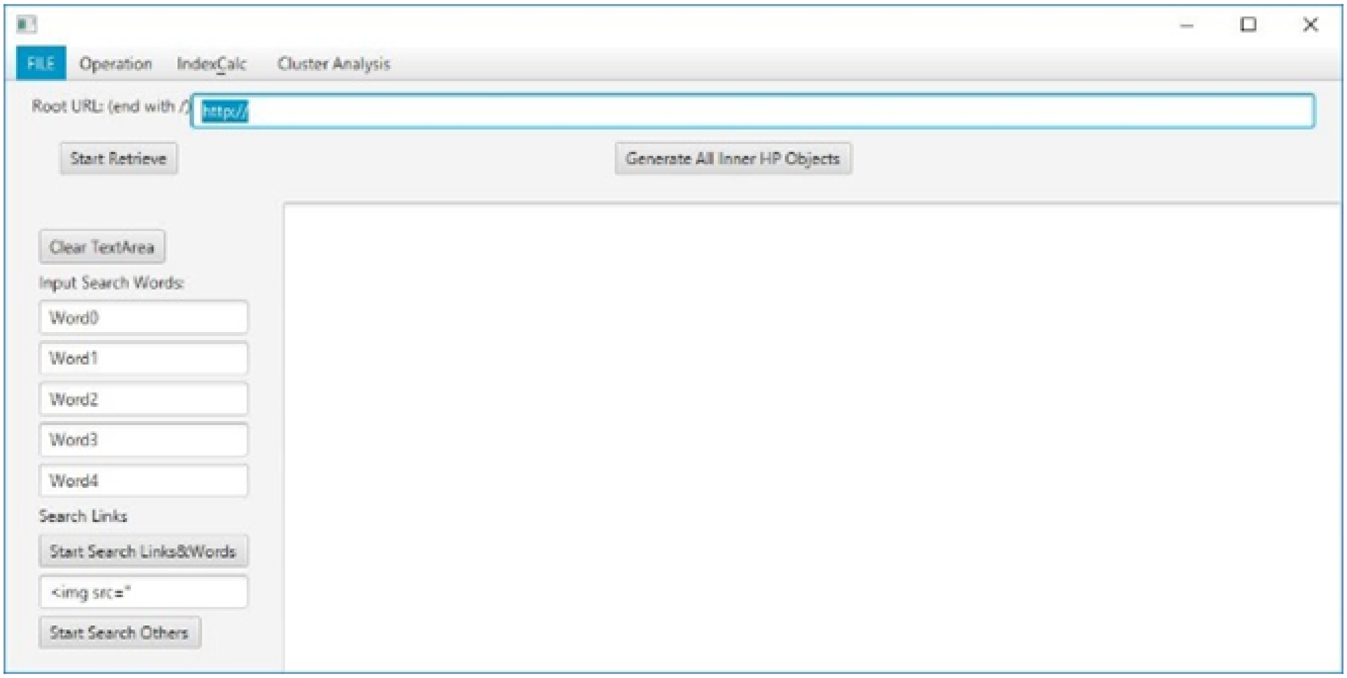


Fig. 1. Initial window

A. *Main Window*

First of all, we submit the target website in the text field on the top of the main window, then starts retrieving the page by clicking the button below left. All the inner linked pages are retrieved while displaying the status in the text area below right with sequentially numbered page names. The left pane is for the key words or particular type of file search, and the results are displayed in the text area.

B. *Menus*

We set four menu items so far such as "FILE", "Operations", "Index Calc(ulation)", and "Cluster Analysis", as shown in Figure 2. Our program creates two types of data set, one is a text type file of home page written in html or xml language, the other is JAVA's class object type data listed below the Figure 2. The second type of data set may require a lot of memory, so clearing memories are sometime necessary when new object data is reloaded.

ISSN: 2414-4592
DOI: 10.20474/jater-5.1.3

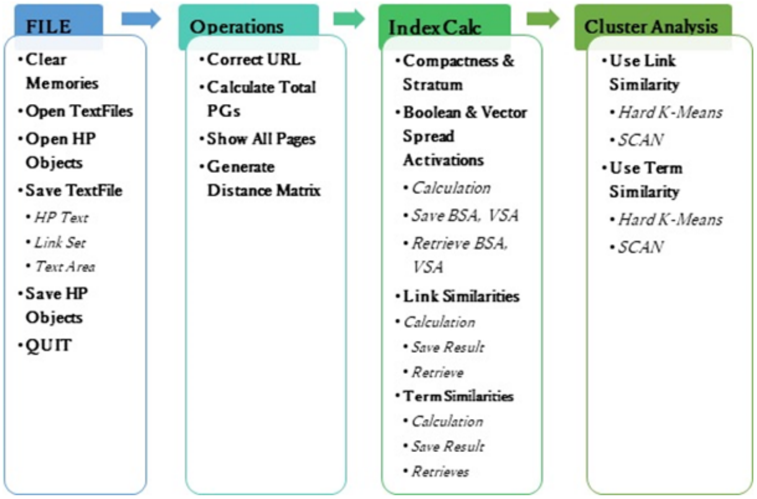


Fig. 2. Menu list

Class Objects defined in Initial Page
static Floyd_Warshall
static InnerWebPG inPG;
static ArrayList<InnerWebPG> innerPGs;
static HashSet<String> totalOutLinkS;
static Floyd_Warshall dPG;
static int[] bsa;
static float[] vsa;
double [][] sim_Link;
double [][] sim_Term;

In the “Operations” item, the last sub-item for calculation of a distance matrix from an adjacent matrix by Floyd-Warshall method is essential to proceed a following step. The item “Index Calc” includes sub-items for calculation of indexes explained in the section 2, and it is saving or retrieving resulted data files are put here. The method of Prop. 1 is applied when calculating the complete hyperlink similarities. Here noticing that some search words should be input in the left pane of the main pane when calculating Boolean or vector spread activation indexes or term similarities.

The last menu item is “Cluster Analysis” including two clustering methods depending on types of similarity measures. When choosing “Hard K-Means” method using link similarity measure, new window appears with two regions in the main pane as shown in the Figure 3. Enter two numbers in left text fields, one is the number of clusters and the other is the number of iterations, then k-means clustering method is applied using similarity measures as a kernel value by clicking “Start Cluster Calculation” button. The resulted clusters are shown in the text area, and a graph displaying each node placed on the circumference and direct links represented by lines appears in the right side of the text area by clicking the button below. To know the link status for each node, input the node number in text fields just above the “→Lines from the Nodes” or “ ←Lines into the Nodes” button. When vector spread activation values are given with some search key words, a circle with a radius in order of the magnitude of the score value is displayed near each of the node number.

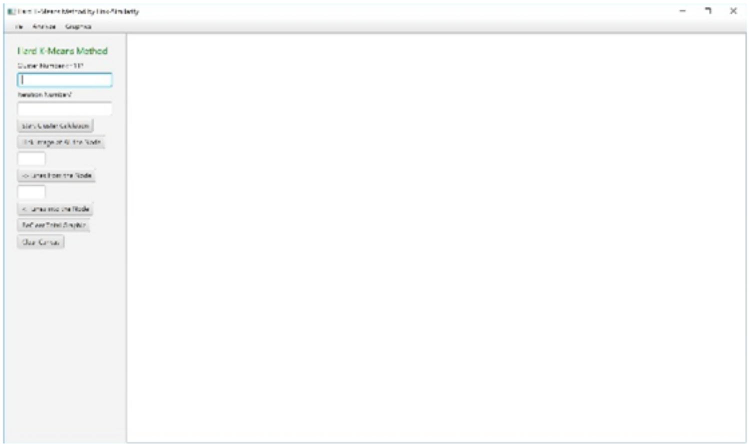


Fig. 3. Cluster analyzing window

ISSN: 2414-4592
DOI: 10.20474/jater-5.1.3

In the cluster window, we set three menu items so far such as “File”, “Analyze”, and “Graphic", as shown in Figure 4. By clicking the cluster wise analysis sub-item, cluster number buttons appear in the left pane by which adjacent link relations in the chosen cluster are graphically displayed. By the cluster pair analysis, Welch test for each of possible pairs of clusters using vector spread activation score values with significance level 1% and 5% are performed. In the Graphic item, we only have spectral chart sub-item, by which in- or out- spectral charts of chosen pair of clusters are displayed instead of direct link image.

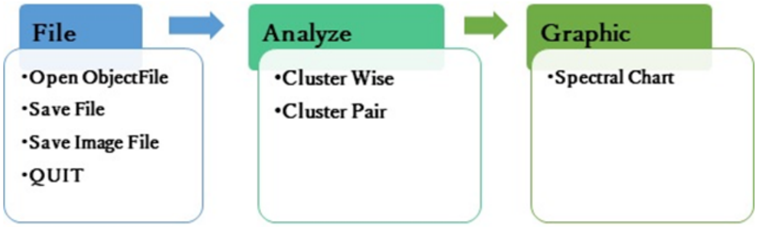
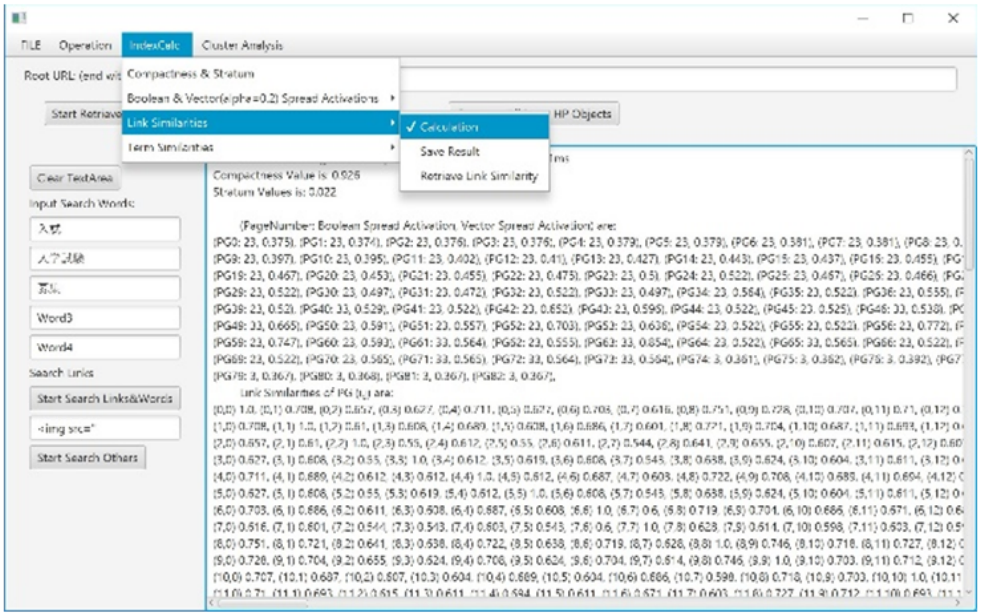


Fig. 4. Cluster menu list

V. EXPERIMENTAL RESULTS

As experimental applications of our developing program, several websites of Japanese College are examined in October 2018 with searching key words on recruitment of students Figure 5. represents some of indexes values for one of example website consisting of 82 webpages.

The resulted values of compactness 0.926 and stratum 0.022 imply that pages in the website are densely connected with each other. Pairs of Boolean spread activation and vector spread activation are displayed under the stratum value with the page number from 0 to 82. Although pages with higher Boolean spread activation value tend to have higher vector spread activation value, they are not always in proportion. For instance, (PG0: 23, 0.375), (PG46: 33, 0.538), (PG56: 23, 0.772), and (PG76: 3, 0.392). The following data set is that of link similarities ranging from 0 to 1, which are used for clustering.



```
Compactness Value is: 0.926
Stratum Values is: 0.022

        (PageNumber: Boolean Spread Activation, Vector Spread Activation) are:
(PG0: 23, 0.375), (PG1: 23, 0.374), (PG2: 23, 0.376), (PG3: 23, 0.376), (PG4: 23, 0.379), (PG5: 23, 0.379), (PG6: 23, 0.381), (PG7: 23, 0.381), (PG8: 23, 0.
(PG9: 23, 0.397), (PG10: 23, 0.395), (PG11: 23, 0.402), (PG12: 23, 0.41), (PG13: 23, 0.427), (PG14: 23, 0.443), (PG15: 23, 0.437), (PG16: 23, 0.455), (PG1
(PG19: 23, 0.467), (PG20: 23, 0.453), (PG21: 23, 0.455), (PG22: 23, 0.475), (PG23: 23, 0.5), (PG24: 23, 0.522), (PG25: 23, 0.467), (PG26: 23, 0.466), (PG2
(PG29: 23, 0.522), (PG30: 23, 0.497), (PG31: 23, 0.472), (PG32: 23, 0.522), (PG33: 23, 0.497), (PG34: 23, 0.564), (PG35: 23, 0.522), (PG36: 23, 0.555), (P
(PG39: 23, 0.52), (PG40: 33, 0.529), (PG41: 23, 0.522), (PG42: 23, 0.652), (PG43: 23, 0.595), (PG44: 23, 0.522), (PG45: 23, 0.525), (PG46: 33, 0.538), (PG
(PG49: 33, 0.665), (PG50: 23, 0.591), (PG51: 23, 0.557), (PG52: 23, 0.703), (PG53: 23, 0.636), (PG54: 23, 0.522), (PG55: 23, 0.522), (PG56: 23, 0.772), (P
(PG59: 23, 0.747), (PG60: 23, 0.593), (PG61: 33, 0.564), (PG62: 23, 0.555), (PG63: 33, 0.854), (PG64: 23, 0.522), (PG65: 33, 0.565), (PG66: 23, 0.522), (P
(PG69: 23, 0.522), (PG70: 23, 0.565), (PG71: 33, 0.565), (PG72: 33, 0.564), (PG73: 33, 0.564), (PG74: 3, 0.361), (PG75: 3, 0.362), (PG76: 3, 0.392), (PG7
(PG79: 3, 0.367), (PG80: 3, 0.368), (PG81: 3, 0.367), (PG82: 3, 0.367),
        Link Similarities of PG (i,j) are:
(0,0) 1.0, (0,1) 0.708, (0,2) 0.657, (0,3) 0.627, (0,4) 0.711, (0,5) 0.627, (0,6) 0.703, (0,7) 0.616, (0,8) 0.751, (0,9) 0.728, (0,10) 0.707, (0,11) 0.71, (0,12) 0.
(1,0) 0.708, (1,1) 1.0, (1,2) 0.61, (1,3) 0.606, (1,4) 0.689, (1,5) 0.608, (1,6) 0.686, (1,7) 0.601, (1,8) 0.721, (1,9) 0.704, (1,10) 0.687, (1,11) 0.693, (1,12) 0.
(2,0) 0.657, (2,1) 0.61, (2,2) 1.0, (2,3) 0.55, (2,4) 0.612, (2,5) 0.55, (2,6) 0.611, (2,7) 0.544, (2,8) 0.641, (2,9) 0.655, (2,10) 0.607, (2,11) 0.615, (2,12) 0.60
(3,0) 0.627, (3,1) 0.606, (3,2) 0.55, (3,3) 1.0, (3,4) 0.612, (3,5) 0.619, (3,6) 0.608, (3,7) 0.543, (3,8) 0.638, (3,9) 0.624, (3,10) 0.604, (3,11) 0.611, (3,12) 0.
(4,0) 0.711, (4,1) 0.689, (4,2) 0.612, (4,3) 0.612, (4,4) 1.0, (4,5) 0.612, (4,6) 0.687, (4,7) 0.603, (4,8) 0.722, (4,9) 0.706, (4,10) 0.685, (4,11) 0.694, (4,12) 0
(5,0) 0.627, (5,1) 0.606, (5,2) 0.55, (5,3) 0.619, (5,4) 0.612, (5,5) 1.0, (5,6) 0.608, (5,7) 0.543, (5,8) 0.638, (5,9) 0.624, (5,10) 0.604, (5,11) 0.611, (5,12) 0.
(6,0) 0.703, (6,1) 0.686, (6,2) 0.611, (6,3) 0.608, (6,4) 0.687, (6,5) 0.608, (6,6) 1.0, (6,7) 0.6, (6,8) 0.719, (6,9) 0.701, (6,10) 0.686, (6,11) 0.671, (6,12) 0.6
(7,0) 0.616, (7,1) 0.601, (7,2) 0.544, (7,3) 0.543, (7,4) 0.603, (7,5) 0.543, (7,6) 0.6, (7,7) 1.0, (7,8) 0.628, (7,9) 0.614, (7,10) 0.598, (7,11) 0.603, (7,12) 0.5
(8,0) 0.751, (8,1) 0.721, (8,2) 0.641, (8,3) 0.638, (8,4) 0.722, (8,5) 0.638, (8,6) 0.719, (8,7) 0.628, (8,8) 1.0, (8,9) 0.746, (8,10) 0.718, (8,11) 0.727, (8,12) 0
(9,0) 0.728, (9,1) 0.704, (9,2) 0.655, (9,3) 0.624, (9,4) 0.706, (9,5) 0.624, (9,6) 0.704, (9,7) 0.614, (9,8) 0.746, (9,9) 1.0, (9,10) 0.703, (9,11) 0.712, (9,12) 0
(10,0) 0.707, (10,1) 0.687, (10,2) 0.607, (10,3) 0.604, (10,4) 0.689, (10,5) 0.604, (10,6) 0.686, (10,7) 0.598, (10,8) 0.718, (10,9) 0.703, (10,10) 1.0, (10,11
(11,0) 0.71, (11,1) 0.693, (11,2) 0.615, (11,3) 0.611, (11,4) 0.594, (11,5) 0.611, (11,6) 0.671, (11,7) 0.603, (11,8) 0.727, (11,9) 0.712, (11,10) 0.693, (11,1
```

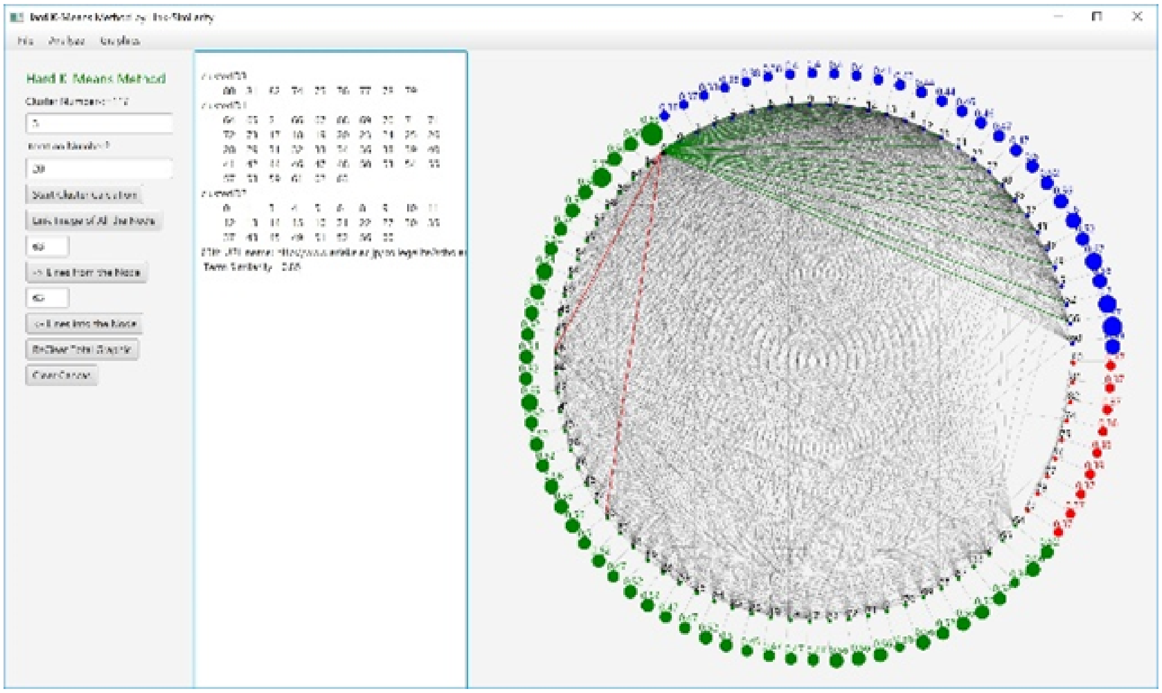
Fig. 5. Index calculation

In the k-means clustering window, setting 3 as the cluster number and 20 as the number of iterations, we have clusters shown in the figure 6 where colored bubbles represent clusters and their vector spread activation score values as the size of the radius. Moreover, multiple lines coming out of page number 63 to the right represent the existence of direct link from the page of number in the input space to the others, and two lines to the left down are opposite direction

ISSN: 2414-4592
DOI: 10.20474/jater-5.1.3

links from other pages to the page number 63.

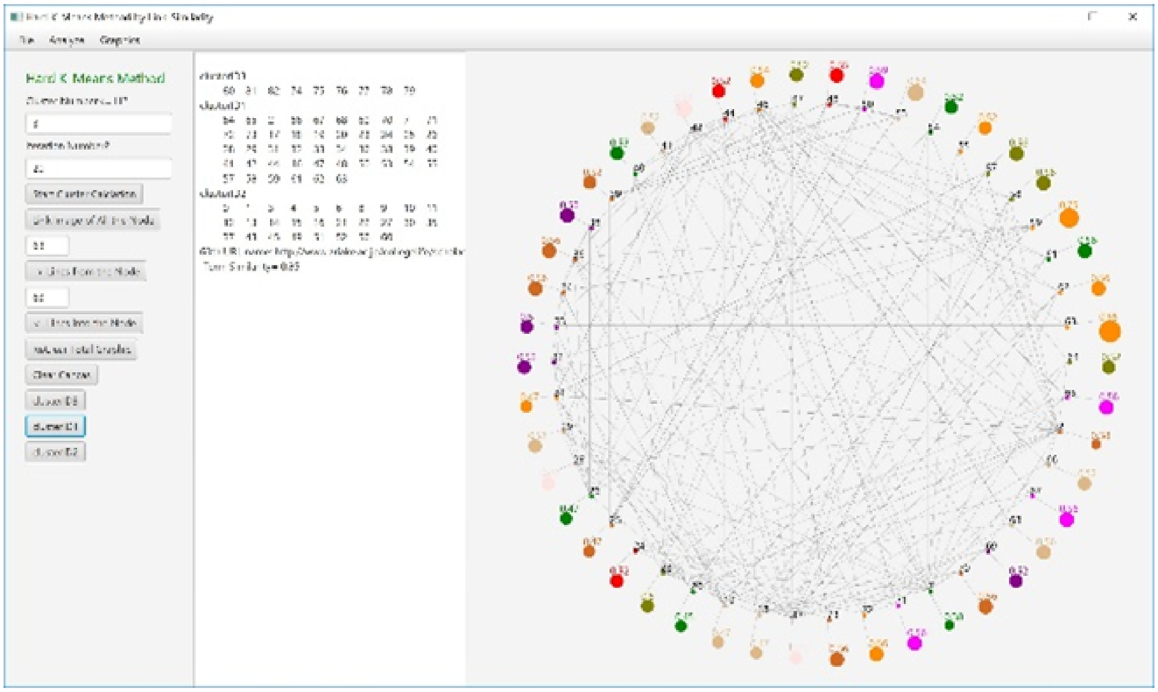
We can see that there are few direct links from pages in cluster 3, and this cluster is the set of pages with least vector activation scores values.



```
cluster[0]
    60  61  62  74  75  76  77  78  79
cluster[1]
    64  65  2   66  67  68  69  70  7   71
    72  73  17  18  19  20  23  24  25  26
    28  29  31  32  33  34  36  38  39  40
    41  42  44  45  47  48  50  53  54  55
    57  58  59  61  62  63
cluster[2]
    0       1   3   5   6   8   9   10  11
    12  13  14  15  16  21  22  27  30  35
    37  43  45  46  51  52  56  [illegible]
63th URL name: http://www.aiit.ac.jp/[illegible]
 Term Similarity: 0.05
```

Fig. 6. *K*-means result

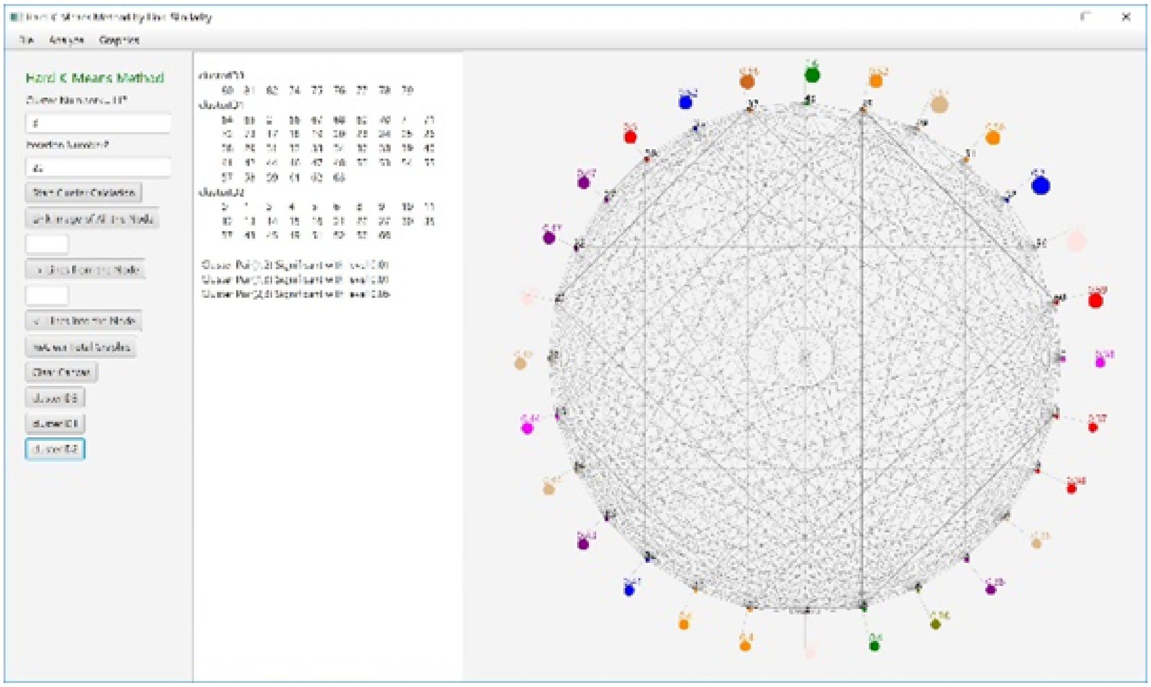
The graphical images of each cluster are shown in the Figure 7, Figure 8, and Figure 9, from which we can see that the link relation only in the cluster 2 has relatively symmetry. From the results of Welch test, shown in the center pane of Figure 8, the average values of vector spread activation scores are differ by at least 5% significance level.



```
cluster[0]
    60  61  62  74  75  76  77  78  79
cluster[1]
    64  65  2   66  67  68  69  70  7   71
    72  73  17  18  19  20  22  24  25  26
    28  29  31  32  33  34  35  36  39  40
    41  42  44  45  47  48  50  53  54  55
    57  58  59  61  62  63
cluster[2]
    0   1   3   4   5   6   8   9   10  11
    12  13  14  15  16  21  22  27  30  35
    37  41  43  46  51  52  56  66
63th URL name: http://www.aiit.ac.jp/[illegible]
 Term Similarity = 0.83
```

Fig. 7. Cluster 1 with link similarity

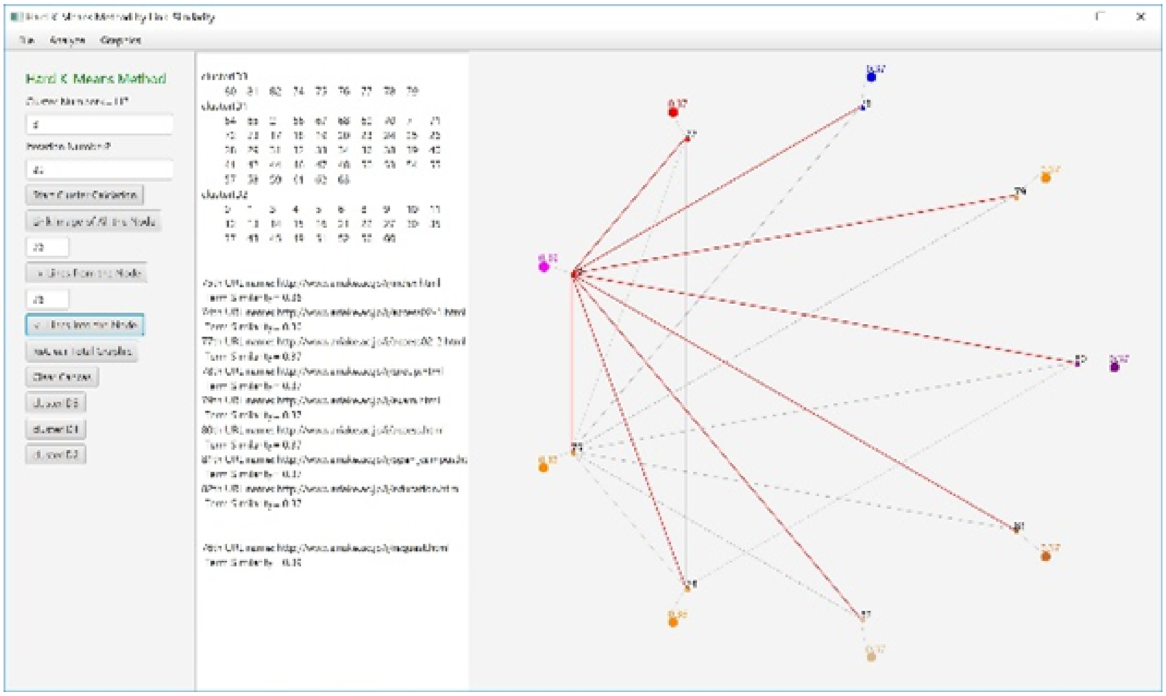
ISSN: 2414-4592
DOI: 10.20474/jater-5.1.3



```
cluster[0]
[illegible]
cluster[1]
[illegible]
cluster[2]
[illegible]

Cluster Pair(1,2) Significant with [illegible]
Cluster Pair(1,3) Significant with [illegible]
Cluster Pair(2,3) Significant with [illegible]
```

Fig. 8. Cluster 2 with link similarity

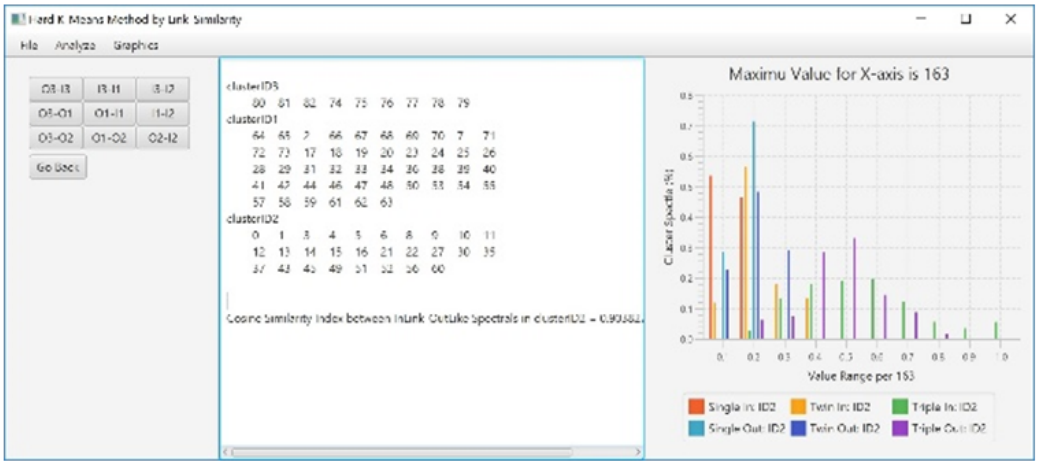


```
cluster[0]
[illegible]
cluster[1]
[illegible]
cluster[2]
[illegible]

[illegible] URL named http://www.[illegible]
 Term Similarity = [illegible]
```

Fig. 9. Cluster 3 with link similarity

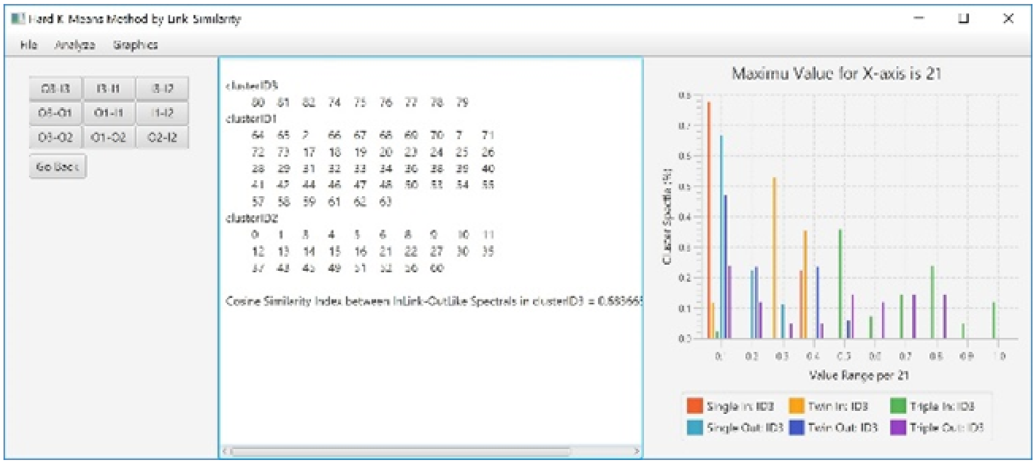
For the spectral calculation, we implemented up to only triple type of spectral so far. Figure 10 and Figure 11 describe each of single, twin, and triple types of in- and out-spectral which are normalized to have value between 0 and 1 for the cluster 2 and for the cluster 3 respectively. And the Figure 12 shows the comparison of out-spectral between cluster 2 and cluster 3. The cosine similarity indexes of in- and out-spectral vectors for each of cluster are also shown in the middle pane.



```
clusterID3
    80  81  82  74  75  76  77  78  79
clusterID1
    64  65  2   66  67  68  69  70  7   71
    72  73  17  18  19  20  23  24  25  26
    28  29  31  32  33  34  36  38  39  40
    41  42  44  46  47  48  50  53  54  55
    57  58  59  61  62  63
clusterID2
    0   1   3   4   5   6   8   9   10  11
    12  13  14  15  16  21  22  27  30  35
    37  43  45  49  51  52  56  60

Cosine Similarity Index between InLink-OutLike Spectrals in clusterID2 = 0.90382
```

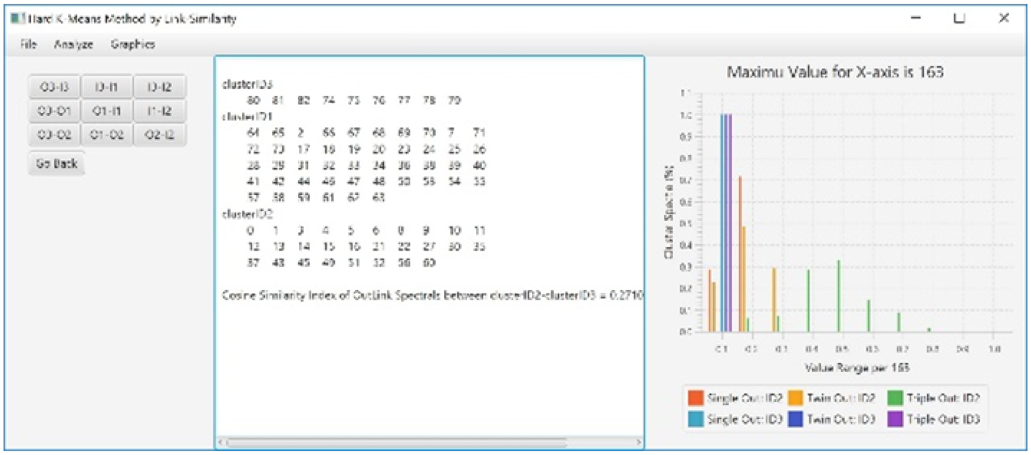
Fig. 10. Spectral chart of cluster 2



```
clusterID3
    80  81  82  74  75  76  77  78  79
clusterID1
    64  65  2   66  67  68  69  70  7   71
    72  73  17  18  19  20  23  24  25  26
    28  29  31  32  33  34  36  38  39  40
    41  42  44  46  47  48  50  53  54  55
    57  58  59  61  62  63
clusterID2
    0   1   3   4   5   6   8   9   10  11
    12  13  14  15  16  21  22  27  30  35
    37  43  45  49  51  52  56  60

Cosine Similarity Index between InLink-OutLike Spectrals in clusterID3 = 0.683665
```

Fig. 11. Spectral chart of cluster 3



```
clusterID3
    80  81  82  74  75  76  77  78  79
clusterID1
    64  65  2   66  67  68  69  70  7   71
    72  73  17  18  19  20  23  24  25  26
    28  29  31  32  33  34  36  38  39  40
    41  42  44  46  47  48  50  53  54  55
    57  58  59  61  62  63
clusterID2
    0   1   3   4   5   6   8   9   10  11
    12  13  14  15  16  21  22  27  30  35
    37  43  45  49  51  52  56  60

Cosine Similarity Index of OutLink Spectrals between clusterID2-clusterID3 = 0.2710
```

Fig. 12. Out-spectral chart of cluster 2 and 3

VI. CONCLUSION

We have investigated several indexes on website and analyzing method originated in the graph theory and clustering. Then developed an application software implementing some of them for finding link and term related properties in order to lead to an improvement of the website. We also give an experimental example with some discussion.

The application software has not completed yet, and we intend to implement some other functions such as fuzzy clustering, pathfinder networks, etc. We need gather much more website information to compare with each other to see better website properties. Further consideration is needed on some indicators, especially the interpretations of in- and out- spectral might be a big problem. As we refer in the 1st section, evaluation from users are also necessary to see what kind of indexes are related to so-called a good website.

REFERENCES

[1] R. Weiss, B. Vélez, and M. A. Sheldon, ``Hypursuit: A hierarchical network search engine that exploits content-link hypertext clustering,'' in *Proceedings of the the seventh ACM conference on Hypertext,* New Dehli, India, 1996.

[2] C. Chen, ``Generalised similarity analysis and pathfinder network scaling,'' *Interacting with Computers*, vol. 10, no. 2, pp. 107-128, 1998. doi: https://doi.org/10.1016/s0953-5438(98)00015-0

[3] J. P. L. Relacion, ``Patient management information system for the university of the immaculate conception college department clinic,'' *International Journal of Technology and Engineering Studies*, vol. 3, no. 5, pp. 213-223, 2017. doi: https://doi.org/10.20469/ijtes.3.40005-5

[4] R. W. Schvaneveldt, D. Dearholt, and F. Durso, ``Graph theoretic foundations of pathfinder networks,'' *Computers & Mathematics with Applications*, vol. 15, no. 4, pp. 337-345, 1988. doi: https://doi.org/10.1016/0898-1221(88)90221-0

[5] G. Salton, A. Wong, and C.-S. Yang, ``A vector space model for automatic indexing,'' *Communications of the ACM*, vol. 18, no. 11, pp. 613-620, 1975. doi: https://doi.org/10.1109/icectech.2011.5941988

[6] A. Al-Canaan and A. Khoumsi, ``Towards designing high-performance restful multime- dia web services on FPGA,'' *Journal of Advances in Technology and Engineering Studies*, vol. 4, no. 3, pp. 111-117, 2018. doi: https://doi.org/10.20474/jater-4.3.2

[7] G. Liang and K. Nagata, ``A study on e-business website evaluation formula with variables of information quality score,'' in *Proceedings of the 12th Asia Pacific Industrial Engineering and Management Systems Conference,* Istanbul, Turkey, 2011.

[8] R. Suryanita, H. Maizir, and H. Jingga, ``Prediction of structural response based on ground acceleration using artificial neural network,'' *International Journal of Technology and Engineering Studies*, vol. 3, no. 2, pp. 74-83, 2017. doi: https://doi.org/10.20469/ijtes.3.40005-2

[9] Y. W. Lee, D. M. Strong, B. K. Kahn, and R. Y. Wang, ``AIMQ: A methodology for information quality assessment,'' *Information & Management*, vol. 40, no. 2, pp. 133-146, 2002. doi: https://doi.org/10.1016/s0378-7206(02)00043-5

[10] N. M. Cherl and R. J. F. Locsin, ``Neural networks application for water distribution demand-driven decision support system,'' *Journal of Advances in Technology and Engineering Studies*, vol. 4, no. 4, pp. 162-175, 2018. doi: https://doi.org/10.20474/jater-4.4.3

[11] A. N. Noorzad and T. Sato, ``Multi-criteria fuzzy-based handover decision system for heterogeneous wireless networks,'' *International Journal of Technology and Engineering Studies*, vol. 3, no. 4, pp. 159-168, 2017. doi: https://doi.org/10.20469/ijtes.3.40004-4

[12] J. M. Kleinberg, ``Authoritative sources in a hyperlinked environment,'' *Journal of the ACM (JACM)*, vol. 46, no. 5, pp. 604-632, 1999. doi: https://doi.org/10.1145/324133.324140

[13] R. Shoda, T. Matsuda, T. Yoshida, H. Motoda, and T. Washio, ``Graph clustering with structure similarity,'' in *Proceedings of the 17th Annual Conference of the Japanese Society for Artificial Intelligence,* New York, NY, 2003.

[14] E. W. Dijkstra, ``A note on two problems in connexion with graphs,'' *Numerische Mathematik*, vol. 1, no. 1, pp. 269-271, 1959. doi: https://doi.org/10.1007/bf01386390

[15] M. Thorup, ``Integer priority queues with decrease key in constant time and the single source shortest paths problem,'' *Journal of Computer and System Sciences*, vol. 69, no. 3, pp. 330-353, 2004. doi: https://doi.org/10.1016/j.jcss.2004.04.003

[16] R. W. Floyd, ``Algorithm 97: Shortest path,'' *Communications of the ACM*, vol. 5, no. 6, pp. 345-350, 1962. doi: https://doi.org/10.1145/367766.368168

[17] S. Warshall, ``A theorem on boolean matrices,'' in *Proceedings of the ACM,* Berlin, Germany, 1962.

[18] K. Nagata, ``Complete hyper link similarity calculation using distance matrix,'' in *Proceedings of the 3rd Annual Conference of International ICT Application Research Society,* Tokyo, Japan, 2018.

[19] J. MacQueen *et al.*, ``Some methods for classification and analysis of multivariate observations,'' in *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability,* Oakland, CA, 1967.

[20] I. S. Dhillon, Y. Guan, and B. Kulis, ``Kernel k-means: Spectral clustering and normalized cuts,'' in *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* California, CA, 2004.

[21] X. Xu, N. Yuruk, Z. Feng, and T. A. Schweiger, ``Scan: A structural clustering algorithm for networks,'' in *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* Jakarta, Indonesia, 2007.