Node Similarity-based Graph Clustering and Visualization

Miklós Erdélyi, János Abonyi

Department of Process Engineering, University of Pannonia
P.O. Box 158, H-8201 Veszprém, Hungary, e-mail: abonyij@fmt.uni-pannon.hu

Abstract: The basis of the presented methods for the visualization and clustering of graphs is a novel similarity and distance metric, and the matrix describing the similarity of the nodes in the graph. This matrix represents the type of connections between the nodes in the graph in a compact form, thus it provides a very good starting point for both the clustering and visualization algorithms. Hence visualization is done with the MDS (Multidimensional Scaling) dimensionality reduction technique obtaining the spectral decomposition of this matrix, while the partitioning is based on the results of this step generating a hierarchical representation. A detailed example is shown to justify the capability of the described algorithms for clustering and visualization of the link structure of Web sites.

1 Introduction

Complex networks are getting more important and by time they get more complex also, thus in order to be able to gain insight into these sophisticated structures we need to somehow visualize them. There have been many graph drawing algorithms developed in recent years [1, 2], however, they alone cannot be efficiently used to visualize large graphs which have hundreds or thousands of nodes. The problem with traditional graph drawing algorithms is that because of the complexity of the networks which have to be visualized the resulting drawing is difficult to interpret for humans. In order to encounter this problem clustering of these kinds of graphs can be used to reduce the visual complexity and help the data miner discover the intrinsic features.

Examples of complex networks are scale-free networks. A network is named scale-free if its degree distribution, i.e. the probability that a node selected uniformly at random has a certain number of links (degree), follows a particular mathematical function called a power law. The power law implies that the degree distribution of these networks has no characteristic scale.

Albert-László Barabási is one of the most well-known researchers of scale-free networks. In the Barabási-Albert preferential attachment model the probability of

adding a new edge between an existing and a new node is proportional to the degree of the existing node [3]. In [4] it is concluded that the Web forms a small-world network, which characterizes social or biological systems, such that two randomly chosen documents on the Web are on average 19 clicks away from each other. That is, despite its huge size, the Web is a highly connected graph with an average diameter of only 19 links. In the well-known book [5], though in a less scientific manner, more examples and explanation of the workings of scale-free networks are given.

Because of the large-scale property of the Web and its high growth rate, finding information on it is becoming more challenging. A well-known algorithm for extracting relationships between Web pages is PageRank [6] which creates a transition matrix for the Markov chain of a theoretical infinitely dedicated Web surfer browsing the Web by randomly clicking on links to obtain the 'authority' of individual Web pages.

Another solution to easing the location of information is clustering the Web pages. In [7] a way of web content mining is introduced by performing relational clustering of Levenshtein distances. Relational alternating cluster estimation (RACE) is applied to automatically extract meaningful keywords from documents and then these keywords are used to automatically classify (previously unknown) documents. In order to speed up the web mining process a new graph representation, the graph matrix, which combines the adjacency matrix with the linked lists allowing for the fastest possible access to different types of information on a graph is shown in [8]. The graph representation is increasingly important for a high search performance, for instance, for rapidly extracting information from the link structure in a hub and authority graph of the Web. An application of this data structure arises from categorical data clustering defining proximity and similarity of data through their patterns of co-occurrence. Another sophisticated document clustering is presented by [9]. It is concluded that the novel normalized-cut method using a new approach of combining textual information, hyperlink structure and co-citation relations into a single similarity metric provides an efficient way of clustering documents. Graph-theoretic clustering methods include [10] in which a structure called scale-free minimum spanning tree is used. In [11] a spectral method is described which can be used to partition graphs into non-overlapping subgraphs along with how the Fiedler-vector of the Laplacian matrix can be used to decompose graphs into non-overlapping neighbourhoods that can be used for the purposes of clustering. There is also a growing research interest in complex networks from the perspective of bioinformatics. Primarily for this field Vicsek et al. developed the application CFinder [12] for locating and visualizing overlapping, densely interconnected nodes in an undirected graph. This application is able to locate the cliques of large sparse graphs efficiently and allows the user to navigate between the original graph and the web of interconnected node groups. In a letter Vicsek et al [13] discussed clique percolation in Erdős-Rényi random graphs, a novel and efficient

approach for discovering the overlapping communities in large networks. It was obtained that the percolation transition takes place when the probability of the connection of two vertices reaches a threshold, and at this transition point the scaling of the giant component with the number of vertices in the random graph is highly non-trivial and is related to the size of the inspected cliques.

The most related work to this paper is presented in [14]. A novel metric of node similarity is proposed which is used for clustering the graph and with the help of which the linkage pattern of the graph is encoded into the similarity matrix. The hierarchical abstraction of densely linked subgraphs is obtained by applying the k-means algorithm to this matrix with a heuristic method to overcome the inherent drawbacks of the k-means algorithm. For the resulting clustered graphs a multilevel multi-window approach is presented to hierarchically draw them in different abstract level views with the purpose of improving their readability.

Visualization of large graphs is very important because humans are better at pattern recognition in the two-dimensional space. In this paper the main emphasis is on the visualization and clustering of Web graphs but the methods presented here apply generally to other complex networks also. On a related note another method of visualization for the Web has to be mentioned. This is based on Kohonen's self-organizing map (SOM) algorithm which is able to automatically categorize a large Internet information space into manageable sub-spaces. However, because of the ever increasing information on the Web, the size of the map has to be increased and thus the visual load of the SOM increases also, making it difficult to clearly recognize local details. Fisheye views and fractal views have been investigated [15] in order to support the visualization of SOM.

This paper describes a new approach to hierarchically clustering graphs and the visualization of them. The key idea behind this approach is to use the results of the dimensionality reduction technique multidimensional scaling (MDS) [16] not for only visualization but for clustering too. This is achieved by first constructing a node similarity matrix based on a novel node similarity metric and then applying the dimensionality reduction technique on it. The obtained two-dimensional data points are then used as the input to the traditional single-linkage clustering algorithm, from the results of which a dendrogram and the Visual Assessment of Cluster Tendency (VAT) [17] figures can be generated.

The remainder of this paper is as follows. In the next section the definitions needed for the graph clustering problem are presented, then the node similarity matrix is described in Section 3 along with the proposed clustering and visualization algorithms. Section 4 presents the experimental results including an example of visualization and clustering of a small Web graph followed by the conclusions and possible future work.

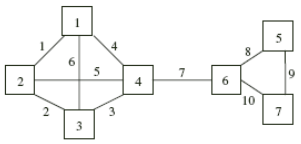
2 Definitions

A graph $G = (V,E)$ is a set V of vertices and a set E of edges such that an edge joins a pair of vertices. In this paper G will be always a general undirected or binary graph.

The *adjacency matrix A* of G is a matrix with rows and columns labeled by graph vertices, with a 1 or 0 in position (v_i,v_j) according to whether v_i and v_j are neighbours or not. For an undirected graph, the adjacency matrix is symmetric. If a graph G is a large graph it is important to note that its adjacency matrix A can be characterized by high dimensionality and sparsity.

The *incidence matrix R* of G is a matrix defined as $R=(r_{ij})_{|E|x|V|}$ such that r_{ij} equals to 1 or 0 whether node v_j is incident with edge e_i or not.

An example graph and its matrices A and R are shown in Fig. 1.



$$A = \begin{vmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{vmatrix}$$

$$R = \begin{vmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{vmatrix}$$

Figure 1
An example of a simple graph and its adjacency and incidence matrices

As the basis of the later described similarity measure a so-called *node vector* has to be defined. The node vectors are represented by the columns of R, thus they simply define to which edges a node belongs. In other words, the column space of R represents nodes and each row of R characterizes an edge. Note that node vectors of an undirected graph are binary vectors since they are derived from the adjacency matrix of the graph.

The *k-clique* of an undirected graph G is a complete subgraph of G with k number of vertices.

3 Clustering and Visualization Algorithms

3.1 Node Similarity Matrix

For the purpose of clustering a graph, a node metric has to be defined which quantifies the abstract features of the nodes, and then clustering can then be done by assigning the nodes to a group according to their metric values. In this paper a node structural metric has been chosen 14 making use of the number of shared edges. The similarity degree of two nodes is partly determined by the number of shared edges between them. The more shared edges two nodes have, the more similar they are, and conversely, the more edges they do not share, the less similar they are. In order to quantify these features a good choice is the Jaccard coefficient among the other most used measures in the literature such as the Euclidean distance, the Minkowsky distance and the dot product. The Jaccard coefficient is a good choice because it is able to measure the degree of overlap, which is defined as

$$sim(\boldsymbol{a}, \boldsymbol{b}) = \frac{\#(a_i = b_i = 1)}{\#(a_i = 1) + \#(b_i = 1) - \#(a_i = b_i = 1)} \tag{1}$$

where $\boldsymbol{a}$ and $\boldsymbol{b}$ are binary vectors. For example, the numerator in the above equation denotes the number of common attributes i (i.e., edge) of $\boldsymbol{a}$ and $\boldsymbol{b}$.

According to the above definition (1) the similarity degree of two nodes are calculated using their node vectors defined in Section 2. Substituting a, b with the node vectors in the incidence matrix R yields the following equation:

$$sim(\boldsymbol{r}_i, \boldsymbol{r}_j) = \frac{\boldsymbol{r}_i^{\mathrm{T}} \boldsymbol{r}_j}{\boldsymbol{r}_i^{\mathrm{T}} \boldsymbol{r}_i + \boldsymbol{r}_j^{\mathrm{T}} \boldsymbol{r}_j - \boldsymbol{r}_i^{\mathrm{T}} \boldsymbol{r}_j} = \frac{(R^{\mathrm{T}} e_i)(e_j^{\mathrm{T}} R)}{(R^{\mathrm{T}} e_i)(e_i^{\mathrm{T}} R) + (R^{\mathrm{T}} e_j)(e_j^{\mathrm{T}} R) - (R^{\mathrm{T}} e_i)(e_j^{\mathrm{T}} R)} \tag{2}$$

where i and j is from the set $\{1, 2, ..., |V|\}$, and e_i (e_j) denotes the ith (jth) canonical vector of dimension e, i.e., $e = (1, 1, \ldots, 1)^{\mathrm{T}}$.

Note that the more similar two nodes, the less links that connect them. The degree of similarity of two nodes will reach its maximum, i.e., 1, when the two nodes are connecting a sole edge.

The problem with using this metric alone is that the similarities of all pairs of non-neighbour nodes are zero, which is inadequate in real applications. For example, if a Web page represented by node v_1 links to another one represented by node v_2, which in turn links to node v_3, then v_1 should be somewhat related to v_3. To solve this problem, a transitive similarity has to be used. Thus $sim(\boldsymbol{r}_1, \boldsymbol{r}_3)$ becomes $sim(\boldsymbol{r}_1, \boldsymbol{r}_2)*sim(\boldsymbol{r}_2, \boldsymbol{r}_3)$ assuming that an existing path between v_1 and v_3 is (v_1, v_2) and (v_2, v_3).

The shortest paths between non-neighbour nodes, that is, the paths which have the fewest edges can be found by the well-known Dijkstra or Floyd's algorithm. The products of sequentially multiplying similarity values of node pairs of the resulting paths can then be calculated. Finally, the minimum value among those products is chosen as the degree of similarity between two non-neighbour nodes. Thus the node similarity matrix is of the following form: $S = [sim(\boldsymbol{r}_i, \boldsymbol{r}_j)]_{|V|x|V|}$.

The symmetirc node similarity matrix of the example graph in Fig. 1 is shown below.

S =

1.000	0.200	0.200	0.167	0.007	0.028	0.007
0.200	1.000	0.200	0.167	0.007	0.028	0.007
0.200	0.200	1.000	0.167	0.007	0.028	0.007
0.167	0.167	0.167	1.000	0.042	0.167	0.042
0.007	0.007	0.007	0.042	1.000	0.250	0.333
0.028	0.028	0.028	0.167	0.250	1.000	0.250
0.007	0.007	0.007	0.042	0.333	0.250	1.000

3.2 Visualization of the Graph with Dimensionality Reduction

Let $\mathbf{X} = \{x_1, x_2, \ldots, x_N\}$ be a set of the observed data, denote x_i the *i*-th observation. Each data object is characterized by *D* dimension, so $x_{i;j}$ yields the *j*-th ($j = 1, 2, .., D$) attribute of the *i*-th ($i = 1, 2, \ldots, N$) data object. The goal of dimensionality reduction is to map a set of observations from a high-dimensional space (*D*) into a low-dimensional space (d, $d << D$) preserving as much as possible the intrinsic structure of data. In the reduced space many data analysis tasks (e.g. classification, clustering, image recognition) can be carried out more faster than in the original data space. Dimensionality reduction methods can be performed in two ways: (i) feature selection or (ii) feature extraction.

Feature selection methods keep the most important dimensions of the data and eliminate the unimportant or noisy factors. Features extraction methods take all attributes into account and they provide reduced representation by feature combination and/or transformation. The Principal Component Analysis (PCA) is one of the well-known linear feature extraction methods. PCA represents data as linear combinations of a small number of basis vectors. The method finds the projection that stores the largest variance possible in the original data.

Multidimensional scaling (MDS) [16] refers to a group of methods, composing by widely used unsupervised data visualization techniques. The classical MDS discovers the underlying structure of data set by preserving similarity information (pair wise distance) among the data objects. Given a set of data in a high-dimensional feature space, MDS maps them into a low-dimensional (generally 2-dimensional) data space in a way that objects that are very similar to each other in

the original space are placed near each other on the map, and objects that are very different from each other, are placed far away from each other. There are two types of MDS: (i) metric MDS and (ii) non-metric MDS. Multidimensional scaling based on measured proximities is called metric multidimensional scaling. While metric MDS preserves the distances among the objects, non-metric MDS methods attempts to preserve the rank order among the dissimilarities. The exact MDS algorithm used for the purposes of visualization in this paper is described in detail in [19].

3.3 Clustering

Clustering is the unsupervised process of grouping information to achieve a more recognizable presentation of the original data. The computation of a clustering generally requires a metric on the data to determine the closeness of data points. The clustering of graphs can be based on either the graph structure itself, or on some kind of properties suitable for the application domain.

In this paper the metric used for graph clustering is a traditional one which is not obtained directly from the graph structure. The two-dimensional results of the dimensionality reduction algorithm described in the previous section are used to do the agglomerative hierarchical clustering [20] based on the traditional single-linkage algorithm. The dissimilarity of the data points obtained by MDS is calculated using the Euclidean distance metric. Note that the single-linkage algorithm corresponds to Kruskal's minimal spanning tree algorithm and is basically the greedy approach to find a minimal spanning tree.

3.4 Visualization of the Clustering Results

For the visualization of the clustering results a dendrogram and a VAT is used. Based on the distance matrix obtained from the two-dimensional results of MDS a dendrogram can be drawn to visualize and hierarchically cluster the nodes in the original graph (for examples, see Section 4). Using this diagram, the human data miner can get a conception how similar the clusters are in the original space and is able to determine which clusters should be merged if needed.

Visual Assessment of Cluster Tendency (VAT) method was proposed in [17]. Its aim is similar to one of cluster validity indices, but it tries to avoid the 'massive aggregation of information' by scalar validity measures. Instead of a scalar value or a series of scalar values by different number of clusters, an N x N intensity image is proposed by Hathaway and Bezdek. It displays the reordered form of the dissimilarity data $\mathbf{D} = [d(\mathbf{x}_i, \mathbf{x}_j)]_{N\text{x}N}$, where $d(\mathbf{x}_i,\mathbf{x}_j)$ is the dissimilarity of the ith and jth samples (not necessarily distance, but in this paper the distance of the projected data points is used as the dissimilarity measure). For the exact VAT algorithm used here please refer to [21].

3.5 Implementation

The implementation is written in MATLAB. To obtain the Web connectivity graph for the example Web site and for testing a robot was used which was written for MATLAB also.

It has to be noted that since the implementation is not fully mature certain functions which are planned in the future are not incorporated into it yet. One of these is the comfortable user interface which now completely relies on MATLAB's figure-viewing tools which have to be used in order to navigate through the visualization of the graph. Since the drawing of the whole graph is shown in a small window when the figure window opens, many nodes cannot be distinguished from the neighbouring ones and their URL labels cannot be read. To remedy this problem the user has to zoom in and use panning to scrutinize the drawing of the graph. Reducing the similarity threshold above which the edges between connected nodes are drawn could also help.

4 Application Examples

This section contains examples of the results yielded with the presented method. Visualization examples are given along with a short explanation of them. A simple graph is presented first which shows the good clustering capability of the method described in this paper. Then a more sophisticated example of a Web graph is analyzed.

4.1 Simple Graph

The visualization of the graph depicted in Fig. 1 is shown below along with the accompanying dendrogram for the clustering.

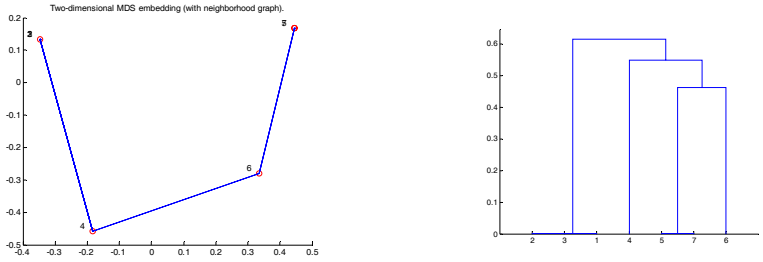


Figure 2
Visualization of the two-clique graph and the dendrogram illustrating the clustering

As it can be seen on the dendrogram, the nodes labeled **1**, **2** and **3** are completely identical, just like the nodes **5**, **7**. They are symmetric to each other because they have the same kind of connections: they are fully connected to each other and to another node which connects this 4-clique with the 3-clique. The situation is the same with the nodes **5**, **7**. The nodes connecting the 3-clique and 4–clique subgraphs are the ones labeled **4** and **6**.

4.2 Real-Life Example: Link Structure of a Web Portal

The Web graph of ORIGO (http://www.origo.hu) was obtained with a maximum link depth of 500. This is a comprehensive Hungarian Web portal with a large collection of news channels and articles for different topics.

The visualization of the link structure of this Web portal is shown in Fig. 3 with the similarity threshold for shown-up edges set to *0.025*, that is, edges between nodes with a similarity below the mentioned threshold remain hidden.

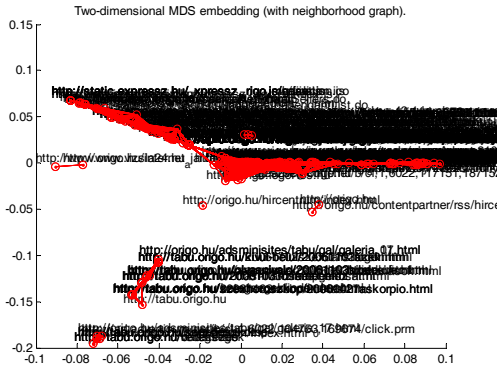


Figure 3
MDS embedding of the Web graph

Since the figure representing the full graph is too crowded two zoomed-in views are provided to show that clusters can be recognized indeed. In Fig. 4 a portion of the Web portal related to advertisements (vehicles, real estate, etc.) can be seen which justifies the effectiveness of the node similarity described in Section 3 along with a view of nodes from *tabu.origo.hu*, the subdomain of *origo.hu*, which can be clearly separated from the rest of the graph.

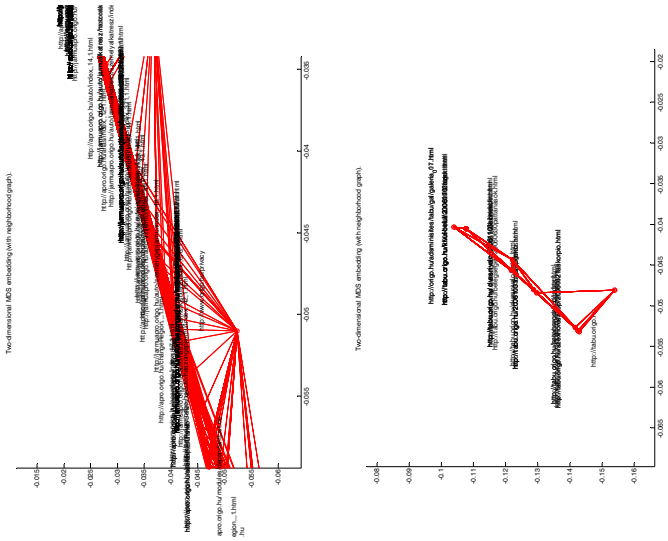


Figure 4
An example of highly related pages and a clearly identifable cluster in the Web graph

The clustering results of the aforementioned Web graph can be seen in Fig. 5.

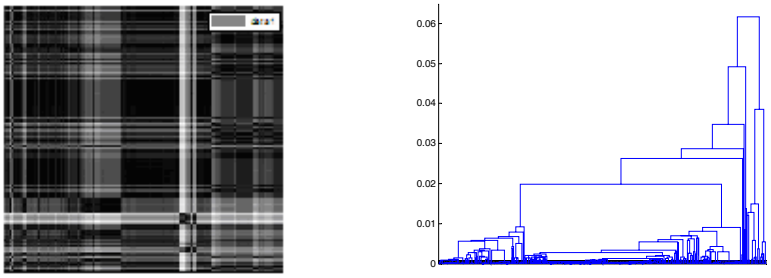


Figure 5
VAT and dendrogram for the Web graph

Many distinct clusters can be identified on the VAT also. For better usability the user interface has to be extended so that the user can check what kind of nodes correspond to certain areas of the VAT and the dendrogram.

Conclusions

The analysis of graphs is very important in our linked world. The visualization and clustering of them deserves particular attention. This paper described a novel algorithm and program prototype for these two aforementioned tasks. The basic idea mentioned here was the construction of a similarity metric which can be used

to describe not only the nodes themselves in the graph but the type of their connections also. Using this similarity metric the graph can be visualized in two dimensions. This way a map of one part of the World Wide Web can also be obtained in which the navigation is easy with the help of MATLAB's figure-viewing tools. The resulting two-dimensional graph can be easily partitioned with the help of the aforementioned similarity metric. An example is shown for clustering in the form of a dendrogram. To visualize the hidden clusters of the graph a VAT figure is generated which shows very well how many and what kind of clusters can be found in the graph. The example in this paper illustrated the connections of the ORIGO Web portal. The presented solution can also be used efficiently for web mining as the preprocessing step. Of course, other types of graphs, such as social networks could be analyzed using the described method.

Acknowledgement

The authors would like to acknowledge the support of the Cooperative Research Centre (VIKKK) (project 2004-I) and Hungarian Research Found (OTKA T049534). János Abonyi is grateful for the support of the Bolyai Research Fellowship of the Hungarian Academy of Sciences.

References

[1] D. Harel, Y. Koren: A Fast Multi-Scale Method for Drawing Large Graphs, in *Graph Drawing: 8th International Symposium (GD'00)*, 2000, pp. 183-196

[2] G. D. Battista, P. Eades, R. Tamassia, I. G. Tollis: Graph Drawing: Algorithms for the Visualization of Graphs, Prentice Hall, 1999

[3] A.-L. Barabási, E. Bonabeau: Scale-Free Networks, Scientific American 288, 2003, pp. 60-69

[4] R. Albert, H. Jeong, A.-L. Barabási: Diameter of the World Wide Web, Nature 401, 1999, pp. 130-131

[5] A.-L. Barabási: Linked: The New Science of Networks, Perseus Books Group, 2002

[6] L. Page, S. Brin, R. Motwani, T. Winograd: The Pagerank Citation Ranking: Bringing Order to the Web, Technical Report, Computer Science Department, Stanford University, 1998

[7] T. A. Runkler, J. C. Bezdek: Web Mining with Relational Clustering, International Journal of Approximate Reasoning 32, 2003, pp. 217-236

[8] J. Błazewicz, E. Pesch, M. Sterna: A Novel Representation of Graph Structures in Web Mining and Data Analysis, Omega 33, 2005, pp. 65-71

[9] X. Hea, H. Zhaa, C. H. Q. Ding, H. D. Simon: Web Document Clustering Using Hyperlink Structures, Computational Statistics & Data Analysis 41, 2002, pp. 19-45

[10] N. Päivinen: Clustering with a Minimum Spanning Tree of Scale-Free-Like Structure, Pattern Recognition Letters 26, 2005, pp. 921-930

[11] H. Qiu, E. R. Hancock: Graph Matching and Clustering Using Spectral Partitions, Pattern Recognition 39, 2006, pp. 22-34

[12] CFinder application. http://angel.elte.hu/cfinder/DS.html

[13] I. Derényi, G. Palla, T. Vicsek: Clique Percolation in Random Networks, Physical review letters 94, 2005, pp. 160202.1-160202.4

[14] X. Huang, W. Lai: Clustering Graphs for Visualization via Node Similarities, Journal of Visual Languages and Computing 17, pp. 225-253, 2006

[15] C. C. Yang, H. Chen, K. Hong: Exploring the World Wide Web with Self-Organizing Map, 2002

[16] J. Kruskal, M. Wish: Multidimensional Scaling, SAGE publications, Beverly Hills, 1978

[17] J. Bezdek, R. Hathaway: VAT: Visual Assessment of (Cluster) Tendency, Proceedings of IJCNN, 2002, pp. 2225-2230

[18] S. Roweis, L. Saul: Nonlinear Dimensionality Reduction by Locally Linear Embedding, Science 290, No. 5500, 2000, pp. 2323-2326

[19] A. Vathy-Fogarassy, B. Feil, A. Kiss, J. Abonyi: Visualization of Topology Representing Networks, submitted

[20] S. C. Johnson: Hierarchical Clustering Schemes, Psychometrika 2, 1967, pp. 241-254

[21] J. Abonyi, B. Feil: Aggregation and Visualization of Fuzzy Clusters based on Fuzzy Similarity Measures, Advances in Fuzzy Clustering and its Applications, John Wiley & Sons, edited by J. V. de Oliveira and W. Pedrycz, accepted