International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 06, June-2019

Data Dimension Reduction for Clustering Semi-Structured Documents using QR Fuzzy C-Mean (QR-FCM)

Hsu-Kuang Chang
Department of Information Engineering
I-Shou University
Kaohsiung, Taiwan

***Abstract*—The rapid growth of XML adoption has created an urgent need for a proper representation for semi-structured documents, where the document semantic structural information has to be taken into account in order to support a more precise document analysis. In order to efficiently analyze the information represented in XML documents, research on XML document clustering is actively in progress. The key issue is how to devise a similar measure to be used for clustering between XML documents and, since XML documents have a hierarchical structure, it is not appropriate to cluster them by using a general document similarity measure. Data dimension reduction (DDR) plays an important role in handling a massive quantity of high dimensional data such as mass semantic structural documents. Having projected XML documents to the lower dimensional space obtained from the DDR/QR, our proposed method of QR Fuzzy C-Mean is to execute the document-analysis clustering algorithms (we call QR-FCM). The DDR can substantially reduce the computing time and/or memory requirement of a given document-analysis clustering algorithm, especially when we need to run the document analysis algorithm many times for estimating parameters, or to search for a better solution.**

Keywords—QR; DDR; PESSW; FCM; QR-FCM

I. INTRODUCTION

An XML document, which is semi-structured data, has a hierarchical structure. Therefore, rather than using the similarity measure of the general document clustering techniques as it is, a new similarity measure which considers the semantic and structural information of an XML document must be investigated. However, some XML clustering methods used the similarity measure which only takes the structural information of XML documents into account. Hwang proposes a clustering method which extracts a typical structure of the maximum frequency pattern n using *PrefixSpan* algorithm [1] on XML documents [2, 3]. However, since such a typical structure extracted from XML documents is not the only structure which represents the XML document itself, it cannot be the representative of the whole document corpus, since there is an accuracy issue of similarity. Lian summarizes XML documents into an *S-graph* which is a structural graph, and proposes that the calculation method of the distance between *S-graphs* is used for clustering [4]. However, they do not consider semantic information on XML documents since they only focus on structural information. Since dimension reduction is one of the fundamental methods of data analysis, there have been a great many studies on effective and efficient dimension reduction algorithms. There are linear dimension reduction algorithms including principal component analysis (PCA) [5] and multidimensional scaling (MDS) [6]. There are also nonlinear dimensional reduction algorithms (NLDR) including an Isomap [7], locally linear embedding (LLE) [8], [9], Hessian LLE [10], Laplacian eigenmaps [11], local tangent space alignment (LTSA) [12] and a distance preserving dimension reduction based on a singular value decomposition (DPDR/QR) [13]. These dimensions cover a variety of areas such as biomedical image recognition, biomedical text data mining, and biological data analysis.

The rest of this paper is organized as follows. In Section II, we introduce the prepared XML documents on a vector space model. In Section III, we show the DDR based on the QR factorization and the DDR QR-FCM. Section IV presents the experimental results illustrating properties of the proposed DDR methods. A summary is made in Section V.

II. PREPARATION OF SEMANTIC-BASED XML DOCUMENTS

In this section, we first introduce the pre-processing steps for the incorporation of hierarchical information in encoding the XML tree's paths. This is based on the preorder tree representation (PTR) [14] and will be introduced after a brief review of how to generate an XML tree from an XML document.

Figure 1 illustrates an example of structural summary. By applying the phase of the nested reduction on tree T_1, we derived tree T_2 where there are no nested nodes. By applying the repetition reduction on tree T_2, we derived tree T_3, which is the structural summary tree without nested and repeated nodes. Once the trees have been compacted using structural summaries, the nesting and repetition are reduced.

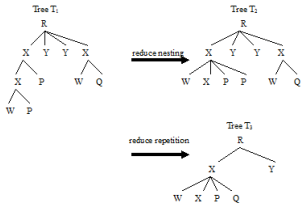


Figure 1 nested and repeated nodes extraction

Now the XML document is modeled as a XML tree $T=(V,E)$ where $V=\{ v_1, v_2, ...\}$ as a set of vertices and $v_1 \in V$,

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 06, June-2019

$v_2 \in V$, $(v_1, v_2) \in E$ as a set of edges. As an example, Figure 2 depicts a sample XML tree containing some information about the collection of books. The *book* consists of *intro* tags, each comprising *title*, *author* and *date* tags. Each *author* contains *fname* and *lname*, each *date* includes *year* and *month* tags. Figure 2 left shows only the first letter of each tag for simplicity.

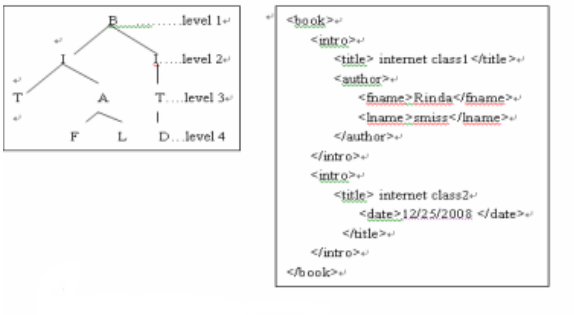


Figure 2 Example of XML Document

The XML document has a hierarchical structure and this structure is organized with tag paths, to represent document characteristics which can predict the contents of the XML document. Strictly speaking, this shows the semantic structural characteristics of the XML document. In this paper, we propose a new method for calculating the similarity using all of the tag paths of the XML tree representing the semantic structural information of the XML document. From now on, a tag path is termed *path element*. Table I shows path elements obtained from the XML document in Figure 3. The $PE_{L\text{-}i}$ represents the extracted path elements on the XML document tree from the i^{th} tree level to the leaf node. For example, the $PE_{L\text{-}1}$ means the path element from the root level (level 1) to the leaf node, and $PE_{L\text{-}2}$ means the path element from the level 2 to the leaf node respectively.

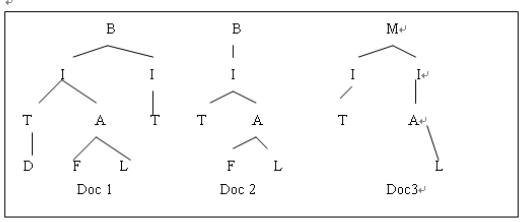


Figure 3 XML Documents Example

Table I. Path elements example

$PE_{L\text{-}1}$	$PE_{L\text{-}2}$	$PE_{L\text{-}3}$	$PE_{L\text{-}4}$
/B/I/T/D	/I/T/D	/T/D	D
/B/I/A/F	/I/A/F	/A/F	F
/B/I/A/L	/I/A/L	/A/L	L
/M/I/A/L	/I/T	/T	
/B/I/T/	/I/A	/A	
/B/I/A	/I		
/B/I/			
/B			
/M/I/T			
/M			
/M/I			

III. PATH ELEMENT OF THE VECTOR SPACE MODEL (PEVSM)

Vector model represents a document as a vector whose elements are the weights of the path elements within the document. To calculate the weight of each path element within the document, a Term Frequency and IDF (Inverse Document Frequency) method is used [15].

A. Path Element Structural Semantic Weight (PESSW)

We define the PESSW (Path Element Structural Semantic Weight) which calculates the weight of the path element in an XML document. The PESSW is PEWF (Path Element Weighted Frequency) multiplied by the PEIDF (Path Element Inverse Document Frequency). The $PESSW_{ij}$ of i^{th} path element in the j^{th} document is shown in equation (1). In this paper, we use the PESSW and D_{PESSW} interchange.

$$PESSW_{ij} = PEWF_{ij} \times PEIDF_{ij} \quad (1)$$

$PEWF_{ij}$ is shown in equation (2).

$$PEWF_{ij} = freq_{ij} \times \frac{1}{x^n} \quad (2)$$

$freq_{ij}$ is a frequency of j-th path element in a i-th document and it is multiplied by level weight $\frac{1}{x^n}$ in order to consider the semantic importance of a path element in a document. X refers to the level number of the highest tag of a tag path. The level number of the root tag is 1, and that of a tag under the root tag is 2, and so on. N is a real number larger than 1, and in this paper, 1 is chosen for the value of n. $PEIDF_{ij}$ is shown in equation (3). $PEIDF_{ij}$ is shown in equation (3).

$$PEIDF_{ij} = \log \frac{N}{DF_j} \quad (3),$$

where N is the total number of documents and DF_j is the number of documents in which the j^{th} path element appears. The PESSW is prudently calculated to correctly reflect the structural semantic similarity. Table II shows the PEWF, PEIDF, and PESSW on sample trees in Figure 3.

TABLE II. AN EXAMPLE OF *PTWF*, *PTIDF* AND *PESSW*

Path Element	PEWF			PEIDF			PESSW		
	doc_1	doc_2	doc_3	doc_1	doc_2	doc_3	doc_1	doc_2	doc_3
/B/I/T/D	1.0	0.0	0.0	1.1	0.0	0.0	1.1	0.0	0.0
/B/I/A/F	1.0	1.0	0.0	0.41	0.41	0.0	0.41	0.41	0.0
/B/I/A/L	1.0	1.0	0.0	0.41	0.41	0.0	0.41	0.41	0.0
/M/I/A/L	0.0	0.0	1.0	0.0	0.0	1.1	0.0	0.0	1.1
/B/I/T/	2.0	1.0	0.0	0.41	0.41	0.0	0.81	0.41	0.0
/B/I/A	1.0	1.0	0.0	0.41	0.41	0.0	0.41	0.41	0.0
/B/I/	2.0	1.0	0.0	0.41	0.41	0.0	0.81	0.41	0.0
/B	1.0	1.0	0.0	0.41	0.41	0.0	0.41	0.41	0.0
/M/I/T	0.0	0.0	1.0	0.0	0.0	1.1	0.0	0.0	1.1
/M	0.0	0.0	1.0	0.0	0.0	1.1	0.0	0.0	1.1
/M/I	0.0	0.0	2.0	0.0	0.0	1.1	0.0	0.0	2.2
/I/T/D	0.5	0.0	0.0	1.10	0.0	0.0	0.5	0.0	0.0
/I/A/F	0.5	0.5	0.0	0.41	0.41	0.0	0.21	0.21	0.0
/I/A/L	0.5	0.5	0.5	0.0	0.0	0.0	0.0	0.0	0.0
/I/T	1.0	0.5	0.5	0.0	0.0	0.0	0.0	0.0	0.0
/I/A	0.5	0.5	0.5	0.0	0.0	0.0	0.0	0.0	0.0
/I	1.0	0.5	1.0	0.0	0.0	0.0	0.0	0.0	0.0
/T/D	0.33	0.0	0.0	1.1	0.0	0.0	0.33	0.0	0.0

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 06, June-2019

/A/F	0.33	0.33	0.0	0.41	0.41	0.0	0.14	0.14	0.0
/A/L	0.33	0.33	0.33	0.0	0.0	0.0	0.0	0.0	0.0
/T	0.67	0.33	0.33	0.0	0.0	0.0	0.0	0.0	0.0
/A	0.33	0.33	0.33	0.0	0.0	0.0	0.0	0.0	0.0
D	0.25	0.0	0.0	1.1	0.0	0.0	0.27	0.0	0.0
F	0.25	0.25	0.0	0.41	0.41	0.0	0.1	0.1	0.0
L	0.25	0.25	0.25	0.0	0.0	0.0	0.0	0.0	0.0

Let d_x and d_y be two vectors which represent an XML document doc_x and doc_y. Cosine similarity is defined as being the angle between two vectors and is quantified by equation (4) and (5).

$$\cos\theta = \frac{d_x \cdot d_y^{\,T}}{|d_x| \cdot |d_y|}, \text{ that is} \qquad (4)$$

$$sim(doc_x, doc_y) = \frac{d_x \cdot d_y^{\,T}}{|d_x| \times |d_y|} = \frac{\sum_{k=1}^{t} w_{kx} \times w_{ky}}{\sqrt{\sum_{k=1}^{t} w_{kx}^2} \times \sqrt{\sum_{k=1}^{t} w_{ky}^2}} \qquad (5)$$

$d_x = (w_{1x}, w_{2x}, ..., w_{tx})$, $d_y = (w_{1y}, w_{2y}, ..., w_{ty})$ and $(w_{1x}, w_{2x}, ..., w_{tx})$ is weight of d_x, $(w_{1y}, w_{2y}, ..., w_{ty})$ is weight of document of d_y, and t is the total number of path elements in d_x, d_y respectively [16].

B. Data Dimension Reduction (DDR) Via QR Factorization

Let us deal with n XML documents the dimension of which is m s.t. $m \succ n$. We compute the QR factorization of the XML document matrix $D \in \Re^{m \times n}$:

$$D = QR = Q\begin{pmatrix} R_1 \\ 0 \end{pmatrix} = \begin{pmatrix} Q_1 & Q_2 \end{pmatrix}\begin{pmatrix} R_1 \\ 0 \end{pmatrix} = Q_1 R_1,$$

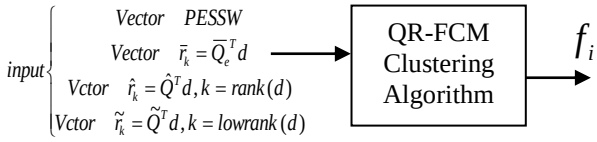
$Q \in \Re^{m \times m}$,is an orthogonal matrix and $R_1 \in \Re^{n \times n}$ is an upper triangular matrix. Then, $Q_1 \in \Re^{m \times n}$ can be considered as being a dimensionality transformation matrix when $m > n$ and the lower dimensional representation $\hat{x} \in \Re^{n \times 1}$ of a vector $x \in \Re^{m \times 1}$ can be computed as $\hat{x} = Q_1^T x$. Thus, the lower dimensional representation d_i of each XML document $\hat{d}_i = Q_1^T d_i = r_i$, where d_i is the i-th column of D and r_i is the i-th column of R_1. We can obtain cosine similarities between the two XML documents (d_i and d_j) in the original dimensional space from R_1 as

$$\cos(d_i, d_j) = \frac{d_i^T Q_1 Q_1^T d_j}{\left\|Q_1^T d_i\right\|_2 \left\|Q_1^T d_j\right\|_2} = \frac{r_i^T r_j}{\|r_i\|_2 \|r_j\|_2},$$

where r_j is the j-th column of $R_1 \in \Re^{n \times n}$. We refer this method to as DDR-QR.

C. Our proposed DDR SVD-QR Algorithm

As described in the previous section, from the QR factorization, we have the originated document vector PESSW, document vector $\bar{Q}^T D = \bar{r}$ (refer to economic QR factoring), document vector $\hat{Q}^T D = \hat{r}$ (refer to QR factorization of rank PESSW), and document vector $\tilde{Q}^T D = \tilde{r}$ (refer to QR factorization of efficient low rank PESSW) based on the XML documents, which is taken as the QR-FCM input data and then goes through the clustering.



F(I)=[f_1,f_2,…..,f_c] , where $f_i = \sum_{j=1}^{N} \mu_{ij} P_j = \frac{1}{N} \sum_{j=1}^{N} \mu_{ij}$.

Fuzzy C-Means (FCM) is a method of clustering which allows one piece of data to belong to two or more clusters. This method developed by [17] and improved by [18] [19] is frequently used in pattern recognition. It is based on the minimization of the following objective function:

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} \mu_{ij}^m \times \left\|d_i - c_j\right\|^2, \; 1 \le m \le \infty$$

where *m* is any real number greater than 1, u_{ij} is the degree of membership of d_i in the cluster *j*, d_i is the *i*th of d-dimensional measured data, c_j is the d-dimension center of the cluster, and ||*|| is any norm expressing the dissimilarity between any measured data and the center.

Fuzzy partitioning is carried out by means of an iterative optimization of the objective function shown above, with the update of membership u_{ij} and the cluster centers c_j by:

$$\mu_{ij} = \frac{1}{\sum_{k=1}^{C} \left(\frac{\left\|d_i - c_j\right\|}{\left\|d_i - c_k\right\|} \right)^{\frac{2}{m-1}}}, \quad c_j = \frac{\sum_{i=1}^{N} \mu_{ij}^m \times d_i}{\sum_{i=1}^{N} \mu_{ij}^m}$$

This iteration will stop when $\max_{ij} \{ \left| \mu_{ij}^{(k+1)} - \mu_{ij}^{(k)} \right| \} < \xi$, where ξ is a termination criterion between 0 and 1, whereas *k* are the iteration steps. This procedure converges to a local minimum or a saddle point of J_m.

IV. EXPERIMENT RESULT

In Figure 4, we show the occupied space (k) on the variant document vectors D_{PESSW}, $\bar{Q}^T D$, $\hat{Q}^T D$, and $\tilde{Q}^T D$ from 400, 800, 1200, 1600, and 2000 XMLs separately. When comparing the D_{PESSW} with the $\tilde{Q}^T D$ on 2000 XMLs, we could save a lot of space, and importantly the clustering result where we used the QR-FCM would be unaffected.

In Figure 5, we show the CPU executing time (ms) to run QR-FCM on the variant document vectors D_{PESSW} , $\bar{Q}^T D$,

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 06, June-2019

$\hat{Q}^T D$, and $\tilde{Q}^T D$ from 400, 800, 1200, 1600, and 2000 XMLs separately. When comparing the D_{PESSW} with the $\tilde{Q}^T D$ on 2000 XMLs, we could save a great deal of time, and importantly the clustering result where we used the QR-FCM still remains the right result.

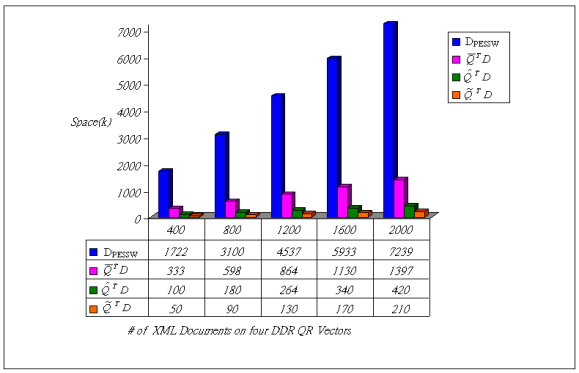


Figure 4: Space on the variant vector from different # XMLs

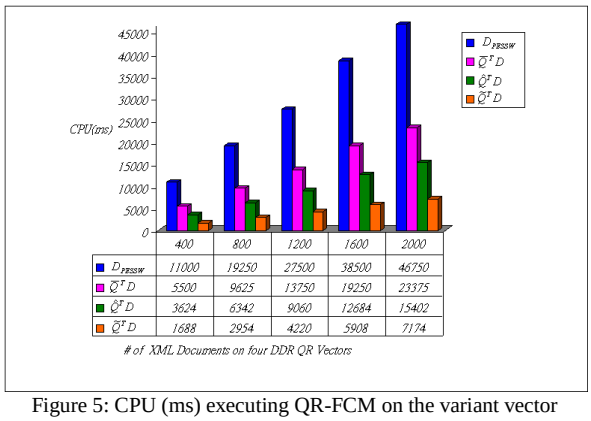


Figure 5: CPU (ms) executing QR-FCM on the variant vector

In Table III, we show the percentage of space saved when comparing $\bar{Q}^T D$ with $\hat{Q}^T D$, $\bar{Q}^T D$ with $\tilde{Q}^T D$, $\hat{Q}^T D$ with $\tilde{Q}^T D$,on running QR-FCM using 2000 XMLs and 1600 XMLs from 5 DTDs. We also show the percentage of the CPU executing time when comparing $\bar{Q}^T D$ with $\tilde{Q}^T D$, $\bar{Q}^T D$ and $\hat{Q}^T D$, $\hat{Q}^T D$ with $\tilde{Q}^T D$, on running QR-FCM using 2000 XMLs and 1600 XMLs from 5 DTDs. On comparing $\bar{Q}^T D$ with $\hat{Q}^T D$, we find that using $\hat{Q}^T D$ instead of $\bar{Q}^T D$ on 2000 XMLs for the QR-FCM, saving % of space in k and % of CPU time in ms. In Table IV, we show the percentage of the space saved and CPU time (ns) when comparing the original document vector D_{PESSW} with $\bar{Q}^T D$, $\hat{Q}^T D$, and $\tilde{Q}^T D$ on running QR-FCM using 2000 XMLs and 1600 XMLs from 5 DTDs. When comparing D_{PESSW} with $\hat{Q}^T D$, we find that using $\hat{Q}^T D$ instead of D_{PESSW} on 2000 XMLs for the QR-FCM saves % of space in k and % of CPU running time in ms.

Table III. Percentage of space and CPU saved in DDR-QR.

Space(k)/CPU(ns)	2000 XMLs		1600 XMLs	
	% saved		% saved	
Variant Vectors	Space	CPU	Space	CPU
$\bar{Q}^T D \sim \hat{Q}^T D$	70%	34%	70%	34%
$\bar{Q}^T D \sim \tilde{Q}^T D$	85%	69%	85%	69%
$\hat{Q}^T D \sim \tilde{Q}^T D$	50%	53%	51%	53%

Table IV. Percentage of saved in D_{PESSW} / thin-QR.

Space(k)/CPU(ns)	2000 XMLs		1600 XMLs	
	% saved		% saved	
Variant Vectors	Space	CPU	Space	CPU
$D_{PESSW} \sim \bar{Q}^T D$	81%	50%	81%	50%
$D_{PESSW} \sim \hat{Q}^T D$	94%	67%	94%	68%
$D_{PESSW} \sim \tilde{Q}^T D$	97%	85%	97%	86%

V. CONCLUSION

The original XML documents $D_N=[d_1,d_2,..,d_N]$ are modeled on the vector space model according to the path element of each document, that is D_{PESSW}=PESSW, then a QR factorization was conducted on the D_{PESSW}. We derived the D_{PESSW}=QR ($D = \bar{Q}_k \bar{R}_k$), or $\bar{Q}_k{}^T D = \bar{R}_k$, and then took the rank on the $\hat{r}_k = \hat{Q}_k^T d$ with the rank of D_{PESSW}, and finally $\tilde{r}_k = \tilde{Q}_k^T d$ with the low-rank of QR on D_{PESSW}. We passed the 4 resulting vectors (D_{PESSW}, $\bar{R}_k$, $\hat{R}_k$, and $\tilde{R}_k$) into the QR-FCM clustering algorithm to attain the clustering result. In terms of clustering result of the section experiment, we found the same clustering result as from the variant D_{PESSW}, $\bar{R}_k$, $\hat{R}_k$ and $\tilde{R}_k$ vectors. From the practical experiment results, we conclude that using the low-rank vector $\tilde{R}_k$ instead of the PESSW original document not only saved space on the input vector but also took less time to cluster the documents.

REFERENCES

[1] J. Pei, J. Han, B. M. Asi, H. Pinto, “PrefixSpan : Mining Sequential Pattern efficiently by Prefix-Projected Pattern Growth”, Int. Conf. Data Engineering(ICDE), 2001.

[2] J. H. Hwang, K. H. Ryu, XML A New XML clustering for Structural Retrieval, International Conference on Conceptual Modeling, 2004.

[3] Jwong Hee Hwang, Keun ho Ryu, Clustering and Retrieval of XML documents by Structure, Computational Science and Its Applications-ICCSA 2005.

[4] Wang Lian, David Wai-lok, An Efficient and Scalable Algorithm for Clustering XML Documents by Structure, IEEE Computer Society Technical Committee on Data Engineer-ing , 2004.

[5] W. F. Massay, Principal components regression in exploratory statistical research, J. Amer Statist. Assoc., vol. 60, pp. 234-246, 1965.

[6] W. S. Torgerson, Theory & Methods of Scaling. New York: Wiley,1958.

[7] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," Science, vol. 290, no. 5500, pp. 2319-2323, 2000.

[8] S. T. Roweis and L. K. Saul, Nonlinear dimensionality reduction by locally linear embedding Science, vol. 290, pp. 2323-2326, 2000.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 06, June-2019

[9] L. K. Saul and S. T. Roweis, "Think globally, fit locally: Unsupervised learning of low dimensional manifolds," Journal of Machine Learning Research, vol. 4, pp. 119-155, 2003.

[10] D. L. Donoho and C. E. Grimes, "Hessian eigenmaps: locally embedding techniques for high-dimensional data," Proc. Natl Acad. Sci. USA, vol. 100, pp. 5591-5596, 2003.

[11] M. Belkin and P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, Neural Computation, vol. 15, no. 6, pp. 1373-1396, 2003.

[12] Z. Zhang and H. Zha, "Principal manifolds and nonlinear dimension reduction via tangent space alignment," SIAM Journal of Scientific Computing, vol. 26, no. 1, pp. 313-338, 2004.

[13] Hyunsoo Kim, Haesun Park, and Hongyuan Zha, Distance preserving dimension reduction using the QR factorization or the Cholesky factorization, *Proceedings of the 2007 IEEE 7th International Symposium on Bioinformatics & Bioengineering (BIBE 2007)*, vol I, pages 263-269.

[14] Theodore Dalamagas, Tao Cheng, Klaas Jan Winkel, Timos Sellis, A Methodology for Clustering XML Documents by Structure, Information Systems, 31(3): 187-228, 2006.

[15] Gao J. and Zhang J. (2005): *Clustered SVD strategies in latent semantic indexing*.—Inf. Process. Manag., Vol. 41, No. 5, pp. 1051–1063.

[16] Berry M.W. and Shakhina A.P. (2005): *Computing sparse reduced-rank approximation to sparse matrices*. — ACM Trans. Math. Software, 2005, Vol. 31, No. 2, pp. 252–269.

[17] J. C. Dunn, Well Separated Clusters and Optimal Fuzzy Partitions, *Journal Cybern*., 4, 1974,. 95-104.

[18] J. C. Bezdek, Numerical Taxonomy with Fuzzy Sets, *J. Math. Biol.*, 1, 1974, 57-71.

[19] J. C. Bezdek, *Fuzzy Mathematics in Pattern Classification*, (Ph.D Thesis, Cornell University, *1973).*