Visualizing Cluster Structures and Their Changes over Time by Two-Step Application of Self-Organizing Maps

Masahiro Ishikawa

Department of Media and Communication Studies, Tsukuba Int'l University, Japan
Manabe 6-20-1, Tsuchiura, Ibaraki, 300-0051 Japan
mi@tius.ac.jp

Abstract. In this paper, a novel method for visualizing cluster structures and their changes over time is proposed. Clustering is achieved by two-step application of self-organizing maps (SOMs). By two-step application of SOMs, each cluster is assigned an angle and a color. Similar clusters are assigned similar ones. By using colors and angles, cluster structures are visualized in several fashions. In those visualizations, it is easy to identify similar clusters and to see degrees of cluster separations. Thus, we can visually decide whether some clusters should be grouped or separated. Colors and angles are also used to make clusters in multiple datasets from different time periods comparable. Even if they belong to different periods, similar clusters are assigned similar colors and angles, thus it is easy to recognize that which cluster has grown or which one has diminished in time. As an example, the proposed method is applied to a collection of Japanese news articles. Experimental results show that the proposed method can clearly visualize cluster structures and their changes over time, even when multiple datasets from different time periods are concerned.

Keywords: Clustering, Visualization, Self-Organizing Map, Cluster Changes over Time.

1 Introduction

We are in the midst of the electric era where various kinds of data, in particular, behavioral data, are originally produced electrically or converted to electric forms from original forms. For example, sensored data are produced from scientific observation equipments in electric forms minute by minute and second by second. Everyday stock prices or some socio-economic indices are also produced and recorded day to day. In social media, for example in blogs, huge amount of text data has already been produced and accumulated, and more and more texts will be continuously produced and accumulated in the future too. Not limited to these types of data, variety of data are accumulated in electric forms.

Electrically accumulated data is easy to access and process by computers. However, huge amount of data is intractable if they are not organized systematically. Clustering is one of the most fundamental processes to organize them.

L. Cao et al. (Eds.): PAKDD 2011 Workshops, LNAI 7104, pp. 160–170, 2012.

It is used to grasp the overview of the accumulated data. Though clustering results might be used to navigate users to explore the whole data, numeric expressions are not enough. Even if accurate information is contained in the numeric expressions of clustering results, it is hard to recognize them for domain experts such as market analysts who are interested in utilizing the data in their application fields. Hence, visualization is very important for human beings. In many environments, including blogs on the web, data are continuously produced and accumulated, thus cluster structures will also change over time. Therefore, visualization methods should accommodate cluster behavior changes, an important task for behavior informatics [12]. It should visualize not only cluster structures in a time period, but also changes of cluster structures over time.

In this paper, a novel method for visualizing cluster structures and their changes over time is proposed. For the visualizations, two-step application of Batch Map, which is a batch learning version of self-organizing maps[1], is used for clustering and assigning angles and colors to clusters. Then colors and angles are used in visualizations. As an example, a collection of news articles is visualized by the proposed method. The example shows that cluster structures and their changes over time are clearly visualized by the proposed method.

2 Related Work

Visualization is important task especially for interactive data mining, and self-organizing map (SOM) is a popular tool for visualization. Thus, there are many studies on SOMs for visualization. There are popular methods for visualizing a single SOM, including U-matrix[1], P-matrix[15], and U*matrix[14]. These are applied after a SOM is constructed, and SOM map is used for visualization. However, even when a single map is concerned, cluster shape is obscure in those visualizations, so we cannot clearly "see" cluster structures.

In [13], Denny et al. proposed a SOM-based method for visualizing cluster changes between two datasets. Two SOMs are constructed from the first and the second datasets. Then, cells in the first SOM is clustered by k-means clustering method producing k clusters, and clusters are assigned different colors arbitrarily. Each cell in the second SOM is assigned the color of the most similar cell in the first SOM. Thus, clusters in two SOM maps are linked by colors. However, it can be applied to only two datasets. And the coloring scheme is not systematic, thus colors do not reflect relationship among clusters systematically. Moreover, even when new clusters emerge in the second SOM, they are forced to be assigned colors of clusters in the first SOM in spite of non-existence of correctly corresponding clusters in the first SOM.

In the second step of the method proposed in this paper, ring topology one-dimensional SOM is used and plays important roles. It makes systematic color assignment and application to arbitrary number of datasets possible.

3 The Proposed Method

3.1 Batch Map

Self-organizing map (SOM)[1] is a vector quantization and clustering method which is usually used to visualize cluster structures in high dimensional numeric vectors in a lower dimensional space, typically on two-dimensional plane. Thus it is a potential candidate to use when one would like to visualize huge amount of high dimensional numeric vectors. The key property of SOM is that it "keeps" topology in the original data space on the two-dimensional plane too. Similar vectors are mapped to cells which are close to each other on the two-dimensional plane, thus the cluster structure in the original space is "reproduced" on the plane. By applying some post-processing, U-matrix construction for example, data visualization is accomplished.

Though "SOM" is usually used to refer to on-line learning version of SOMs, the word refers to the batch-learning version, called *Batch Map* in the literature[1], in this paper. Here, Batch Map is summarized. Let $D = \{d_0, d_1, \cdots, d_{n-1}\}$ ($d_i \in \mathcal{R}^m$) be a dataset to be processed, $\phi : \mathcal{R}^m \times \mathcal{R}^m \to \mathcal{R}$ be the distance(dissimilarity) function on D, and $\{c_{x,y}\}$ be the cells in two-dimensional (typically hexagonal) SOM map of $X \times Y$ grids. Each cell is associated with a reference vector $v(c_{x,y}) \in \mathcal{R}^m$. In the learning process of a Batch Map, each datum d_i is assigned to the cell associated with the reference vector which is most similar to d_i. By gradually updating the reference vectors, the Batch Map reproduces the cluster structures in D on the SOM map. Batch Map learning procedure is summarized below.

1. Initialize the reference vectors $v(c_{x,y})$.
2. Assign each datum d_i to the cell $c_{s,t}$ where

$$\phi(d_i, v(c_{s,t})) = \min_{x,y}\{\phi(d_i, v(c_{x,y}))\}.$$

3. Update reference vectors according to the following formula:

$$v(c_{x,y}) \leftarrow \frac{1}{|\mathcal{N}_{x,y}^{(R)}|} \sum_{d \in \mathcal{N}_{x,y}^{(R)}} d \tag{1}$$

   Here, $\mathcal{N}_{x,y}^{(R)}$ denotes the set of data assigned to cells in the circle centered at $c_{x,y}$ with radius R. Thus, a reference vector is updated by the average of all data assigned to its neighborhood with respect to R.
4. Repeat step 2 and 3 until it converges, gradually decreasing radius R. In the last steps, R should be zero.

Note that when R equals to zero, this procedure is identical to that of the well-known k-means clustering.

3.2 Two-Step Application of Self-Organizing Maps

In the proposed method, SOM is applied in two steps. In the first step, two-dimensional SOM of $X \times Y$ cells is applied to the input dataset. As the result, each datum is assigned to the most similar cell (i.e. the cell associated with the most similar reference vector). In a sense, $X \times Y$ clusters are formed. They are called *micro clusters* in this paper. Reference vectors are the centroids of micro clusters. One of important reasons why SOM is used in this step is that the distribution of centroids obtained by a SOM approximately follows that of the original dataset, thus the micro clusters can be used as a compact representation of the original dataset.

Second, one-dimensional SOM with p cells is applied to centroids of micro clusters obtained above. An important point is that, ring topology must be adopted for the one-dimensional SOM. Hereafter, ring-topology one-dimensional SOM is called *coloring SOM* (cSOM). As the result of cSOM application, each centroid of a micro cluster is assigned to a cSOM cell, forming *macro clusters*. Reference vectors of a cSOM are the centroids of macro clusters. In this step, p macro clusters are formed.

3.3 Assigning Angles and Colors to Clusters

Using the centroids of macro clusters, each macro cluster is assigned an angle and a color. Let c_i be the i-th cell of the cSOM. Then, each cell c_i (thus a macro cluster) is assigned an angle ω_i which is calculated by the formula below. In the formula, p is the number of cells in the cSOM.

$$\omega_i = 2\pi \frac{\sum_{k=0}^{i-1} \phi(v(c_k), v(c_{k+1}))}{\sum_{k=0}^{p-1} \phi(v(c_k), v(c_{(k+1) \bmod p}))} \tag{2}$$

Then, a triplet $\langle R_i, G_i, B_i \rangle$ is calculated from ω_i by the following formulae.

$$R_i = \frac{sin(\omega_i) + 1}{2}$$
$$G_i = \frac{sin(\omega_i + \frac{2\pi}{3}) + 1}{2}$$
$$B_i = \frac{sin(\omega_i + \frac{4\pi}{3}) + 1}{2}$$

This triplet represents a color in the RGB coloring space $[0, 1]^3$, and assigned to the cell c_i (thus, to the corresponding macro cluster, micro clusters, and cells in the first step SOM). This coloring scheme means picking up a color from the hue circle at the angle ω_i. See Figure 1. If the distance between adjacent cells' reference vectors is smaller, the difference of assigned angles will also be smaller. Thus closer adjacent clusters are assigned more similar angles and colors systematically.

The focus of this paper is how to visualize the cluster structures and their changes over time. Thus, we should think of multiple datasets from multiple

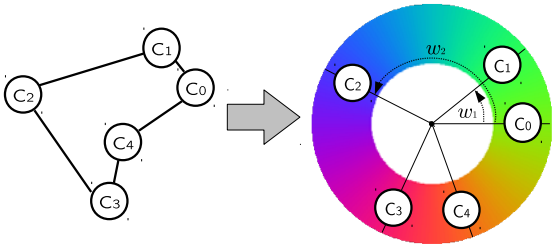

Fig. 1. Image of the Coloring Scheme: Left picture is the cSOM cells distributed in the input space. Then cells are placed on the hue circle according to the angles assigned to them. The colors under the cells are picked up.

periods of time. In such situations, each dataset is individually clustered by two dimensional SOMs, forming micro clusters in each dataset. To place similar clusters in two successive datasets at similar positions on two SOM maps, $i+1$-th SOM is initialized by the resultant reference vectors of the i-th SOM. The first SOM is randomly initialized.

Then all micro clusters of all datasets are used as input to a coloring SOM. As the result, even when they belong to different datasets, similar clusters are assigned similar angles and colors, which can be used to link similar clusters among multiple datasets.

3.4 Visualization by Colors and Angles

By using the assigned colors and angles, cluster structures are visualized in several fashions. One is based on the traditional SOM map, i.e. two-dimensional grid. Indeed two-dimensional SOM map is appealing, but there is no need to stick to it. Thus, two more fashions are introduced. They are briefly summarized here and the details are described in the next section with examples.

SOM Map Visualization. First of all, the two-dimensional SOM map is used to visualize the cluster structures. By coloring each cell the assigned color, a cluster can be seen as a continuous group of cells filled with similar colors.

Area Chart Visualization. In the area chart visualization, sizes (populations) of clusters are used. By using x-axis for time periods (datasets) and y-axis for sizes of clusters, changes of cluster sizes are visualized. The area dominated by a cluster is filled with the assigned color, and the stacking order of clusters is determined by the angles. Thus, adjacent areas are filled with similar colors, making a gradation. We can easily see which cluster (or group of similar clusters) has grown and which one has diminished.

Polar Chart Visualization. Cluster sizes can be visualized by histograms too. In a histogram, a cluster is represented by a bar, whose height reflects the cluster

size. Again, by coloring the bars, relations among clusters can be visualized. However, the coloring scheme has some deficits. The most serious one is the perceived non-uniformity of color differences. The perceived color differences do not linearly reflect the differences of angles. Moreover they depend on the part of the hue circle. The other one is the device dependency of color reproduction. Thus, to link similar clusters, angles assigned to clusters are also used directly here. Instead of the orthogonal x-y axes, the polar axis is adopted. Each cluster bar is depicted at the angle assigned to it. Thus, similar clusters are placed on similar directions. Therefore, similar clusters in different datasets can be easily found in this visualization.

4 Example: Visualization of Clusters in News Articles

As an example, a collection of news articles are visualized here. Experiment was conducted on a PC Linux system with Intel Core i7 920 CPU and 12GB memory. Programs were implemented in Python language with the numpy[10] scientific computing extension.

4.1 Target Dataset

Japanese news articles released by Sankei digital corporation as "Sankei e-text" [9] in 2005 are used. It consists of 73388 articles.

4.2 Keyword Extraction and Matrix Representation

To apply SOMs, the dataset must be represented as a set of numeric vectors. First of all, each article is represented as a bag-of-words. To extract words from an article, a morphological analyzer is applied.[1] Then words tagged as noun are extracted, excluding some exceptions.[2] Then, according to the standard vector space model commonly used in information retrieval[2], news articles are represented as an $m \times n$ term-document matrix as follows:

$$\boldsymbol{D} = \begin{pmatrix} d_{11} & d_{12} & \cdots & d_{1n} \\ d_{21} & d_{22} & \cdots & d_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ d_{m1} & d_{m2} & \cdots & d_{mn} \end{pmatrix}$$

Where, m is the number of words in the collection (vocabulary size), n the number of news articles, and $d_{i,j}$ the weight of the i-th word in the j-th article. In this example, tf-idf values given by the formula below are used.

$$d_{i,j} = -f_{i,j} \log \frac{n_i}{n}$$

[1] Mecab[11] with IPA dictionary is used.
[2] Pronouns, adjuncts, suffixes, postfixes and numerals are excluded.

In the formula, $f_{i,j}$ is the frequency of i-th word in the j-th article, and n_i is the number of articles which contains the i-th word. In the matrix, a column vector represents an article.

In the target dataset, n is 73388, and m is 108750.

4.3 Dimension Reduction by Random Projection and LSI

Usually total number of articles, n, and vocabulary size, m, are very huge, thus memory and time costs are so high in the successive process that in many cases the problem becomes intractable. Thus, dimension reduction is applied here.

First, random projection[6][5] is applied to the original matrix $\boldsymbol{D}$ for reducing the dimensionality of the column vectors. Second, LSI (Latent Semantic Indexing)[8] is applied for further reduction. In the same time, LSI incorporates the latent semantics of words. This combination could achieve better clustering quality than sole application of random projection [7]. This aspect will not be further discussed in this paper since it is out of the scope. However, random projection and LSI are briefly described below to make this paper self-contained.

Random Projection. Random projection is a dimension reduction technique for high dimensional numeric vectors. To reduce the dimension of the column vectors of matrix $\boldsymbol{D}$, randomly generated $m_1 \times m (m_1 \ll m)$ matrix $\boldsymbol{R}$ is used as follows.

$$\boldsymbol{D}_1 = \boldsymbol{R}\boldsymbol{D}$$

If $\boldsymbol{R}$ satisfies some conditions, angles among column vectors are well preserved. In this example, elements of $\boldsymbol{R}$ are generated according to the following formula proposed in [4].

$$r_{ij} = \sqrt{3} \begin{cases} 1 & (\text{with probability } \frac{1}{6}) \\ 0 & (\text{with probability } \frac{2}{3}) \\ -1 & (\text{with probability } \frac{1}{6}) \end{cases}$$

Here, $\sqrt{3}$ is omitted since relative values are suffice for clustering purpose.

In this experiment, the dimensionality is reduced to 5000 by random projection.

Latent Semantic Indexing. LSI is one of popular techniques for dimension reduction, which, in the same time, incorporates latent semantics of words. Though LSI is the best technique for dimension reduction with respect to the least square error, it is intractable when a huge matrix is concerned due to its space and time complexities. However, by applying random projection beforehand, matrix size can be reduced to which LSI can be applied. By singular value decomposition, we have $\boldsymbol{D}_1 = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}}$. Or by eigen value decomposition, we get $\boldsymbol{D}_1\boldsymbol{D}_1^{\mathrm{T}} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^{\mathrm{T}}$. Then $\boldsymbol{U}_{m_2}$ is formed by taking the leftmost m_2 ($\ll m_1$) column of $\boldsymbol{U}$. Dimension reduction is achieved by $\boldsymbol{D}_2 = \boldsymbol{U}_{m_2}^{\mathrm{T}}\boldsymbol{D}_1$.

In this experiment, the dimensionality is reduce to 3600 by LSI. Accumulative contribution rate is 0.91.

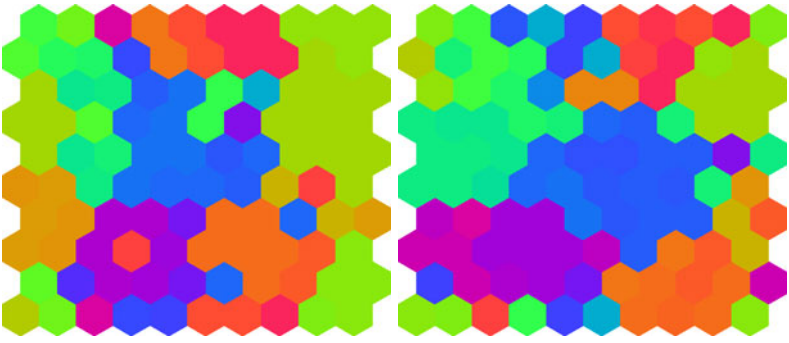

Fig. 2. SOM map Visualization: Left map is the first week and the right is the second

4.4 Final Matrix and Distance Function

After dimension reduction, column vectors of $\boldsymbol{D}_2$ are normalized to the unit length, obtaining 3600×73388 matrix $\boldsymbol{X}$. Finally, according to their release dates, $\boldsymbol{X}$ is divided into 52 sub matrices $\{\boldsymbol{X}_i\}$, each one contains articles released in each week of the year.

Similarity between articles is given by cosine measure which is popular in document retrieval and clustering. However, distance (dissimilarity) is needed in formula (2), thus the angle derived from the cosine measure is used as the distance function.

$$\phi(\boldsymbol{x}, \boldsymbol{y}) = \arccos(cos(\boldsymbol{x}, \boldsymbol{y}))$$

The range is restricted to $0 \leq \phi(\boldsymbol{x}, \boldsymbol{y}) \leq \pi$. Incidentally, learning procedure of Batch Map is modified in two points. First, in the step 1, reference vectors are normalized to the unit length. Second, in the step 3, updated reference vectors are also normalized to the unit length. As the result, Batch Map used in this example becomes the Dot Product SOM[1] which corresponds to the spherical k-means clustering[3]. Note that, these modifications are not mandatory though they are suitable for document clustering.

For the first step clustering, SOM of 10×10 cells with torus topology is used. Thus, 100 micro clusters are produced for each week and 5200 in total. For a coloring SOM, a ring of 64 cells is used, thus 64 macro clusters are produced by clustering 5200 micro clusters.

4.5 Visualization Results

SOM Maps. By successively applying SOMs, we got 52 SOM maps for 52 weeks. In Figure 2, the maps of the first and the second weeks are presented. We can easily find clusters as continuous regions of similar colors. In addition to separate maps, we can view all maps in a three-dimensional graphics at a glance. Figure 3 shows 3D views of all maps. Each cell is drawn as a cylinder. In the left view of Figure 3, 52 weeks are all visible. By controlling the transparency and the brightness of cells, we can view changes of specific clusters at a glance. In the

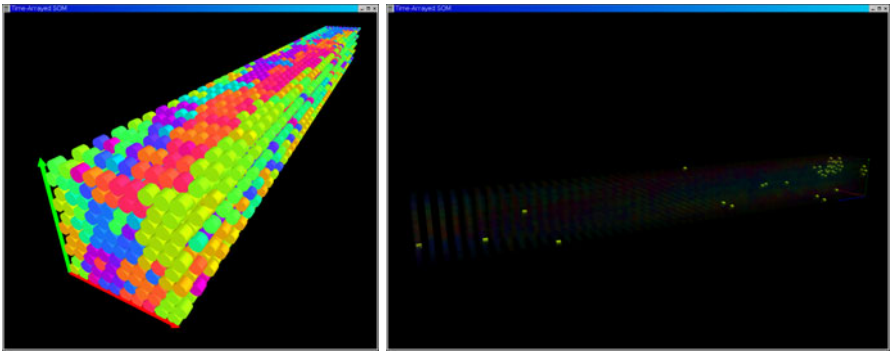

Fig. 3. 3D Visualization of SOM Maps: In the right view, only one cluster is emphasized

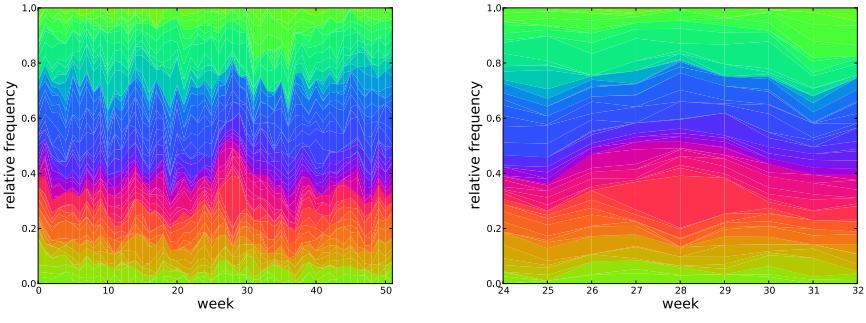

Fig. 4. Area Chart Visualization

right view of Figure 3, all clusters but one are made almost transparent. In the view, the rightmost is the first week and the leftmost is the most recent week. Recall that the distribution of reference vectors approximately follows that of the input dataset. Thus we can figure out that this cluster thrived at the beginning of the year, and rapidly diminished. This cluster mainly consists of articles about the giant tsunami in the Indian Ocean caused on the ending of the previous year.

Area Charts. Figure 4 shows area chart visualizations. In the figure, the stacking order of clusters are defined by the assigned angles. Thus, we can view not only single cluster but also a group of similar clusters as a unit. Note that the bottom and the top clusters are adjacent since a coloring SOM has ring topology.

In the right chart of Figure 4, the range from the 24th week to the 32nd week is magnified. It shows that a cluster filled with orange irregularly expanded in the 28th week. This cluster mainly consists of articles about the most popular Japanese high school baseball tournament held in August. Though we can see global structures at a glance in an area chart, it becomes intricate when the number of clusters got increased.

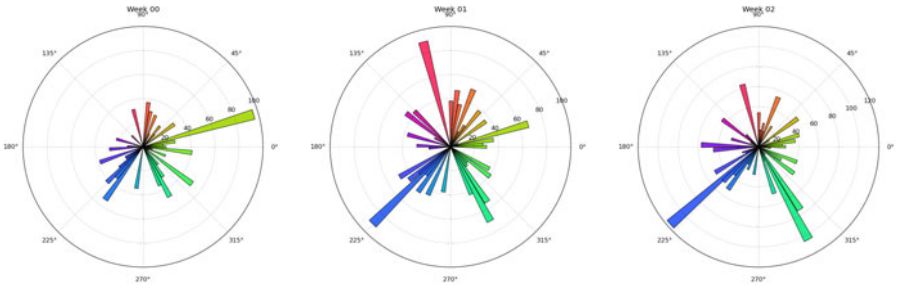

Fig. 5. Polar Chart Visualization

Polar Charts. Figure 5 shows polar charts for the first three weeks of the year. Each bar is a cluster and its length means the cluster size. Bars are filled with assigned colors. Note that a polar chart is not a pie chart. A cluster is placed at the angle assigned to it. Therefore, similar clusters are placed on similar directions, and angles between adjacent clusters reflects degrees of cluster separations. Crowded direction suggests that clusters on that direction might have to be merged into one cluster. The longest bar filled with green in the left chart of Figure 5 is the cluster about the giant tsunami in the Indian Ocean visualized in Figure 3. Even when it belongs to different weeks, the same cluster is placed on the same direction, thus we can easily find which cluster has grown and which one has diminished. These three charts show that the tsunami cluster is rapidly diminishing.

In many cases, it is difficult to determine how many clusters should be found in a dataset. Thus, for example in k-means clustering, sometimes several clustering trials with different values for k are needed. By forming sufficiently many micro and macro clusters, we might be able to figure out appropriate number of clusters by this visualization.

5 Conclusions and Future Work

In this paper, a novel method for visualizing cluster structures and their changes over time is proposed. A two-step SOM application method is introduced for clustering. Then, using the clustering results, each cluster is assigned an angle and a color systematically, which reflects relationship among clusters.

Visualization is achieved by using angles and colors assigned to clusters. In addition to the traditional two dimensional SOM map, area chart and polar chart are used for visualization. By experiments on a collection of Japanese news articles, effectiveness of the proposed method is demonstrated. Note that the proposed method can be applied to any kind of dataset as long as they are represented as numeric vectors and a distance function is defined. It should be pointed out that if SOM map visualization is not needed, the first step clustering could be done by one-dimensional SOM too. Moreover, it can be substituted for

by any clustering method as long as the distribution of the resultant clusters' centroids follows that of the input dataset.

The same dataset is visualized in three fashions. Thus, cooperative work of them is important. Interactive user interface is also needed. Each fashion has merits and demerits, thus they should complement one another.

In this paper, cSOM is applied to the set of all micro clusters, thus every time a new dataset arrives cSOM must be applied anew to the incrementally augmented set of micro clusters. This is a drawback in dynamic environments. Therefore, incremental cSOM construction method should be invented in future work.

References

1. Kohonen, T.: Self-Organizing Maps, 3rd edn. Springer, Heidelberg (2001)
2. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. Addison Wesley (1999)
3. Dhillon, I.S., Modha, D.S.: Concept decompositions for large sparse text data using clustering. Machine Learning 42(1), 143–175 (2001)
4. Achlioptas, D.: Database-friendly Random Projections. In: Proc. of the 20th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, pp. 274–281 (2001)
5. Bingham, E., Mannila, H.: Random projection in dimensionality reduction: applications to image and text data. In: Proc. of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 245–250 (2001)
6. Dasgupta, S.: Experiments with Random Projection. In: Proc. of the 16th Conference on Uncertainty in Artificial Intelligence, pp. 143–151 (2000)
7. Lin, J., Gunopulos, D.: Dimensionality reduction by random projection and latent semantic indexing. In: Proc. of SDM 2003 Conference, Text Mining Workshop (2003)
8. Papadimitriou, C.H., et al.: Latent Semantic Indexing: A Probabilistic Analysis. In: Proc. of the 17th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, pp. 159–168 (1998)
9. Sankei e-text, `https://webs.sankei.co.jp/sankei/about_etxt.html`
10. Scientific Computing Tools for Python — numpy, `http://numpy.scipy.org/`
11. MeCab: Yet Another Part-of-Speech and Morphological Analyzer, `http://mecab.sourceforge.net/`
12. Cao, L.: In-depth Behavior Understanding and Use: the Behavior Informatics Approach. Information Science 180(17), 3067–3085 (2010)
13. Denny, Squire, D.M.: isualization for Cluster Changes by Comparing Self-organizing Maps. In: Ho, T.-B., Cheung, D., Liu, H. (eds.) PAKDD 2005. LNCS (LNAI), vol. 3518, pp. 410–419. Springer, Heidelberg (2005)
14. Ultsch, A.: U*-Matrix: A Tool to visualize Cluster in high-dimensional Data. In: Proc. of the 2008 Eighth IEEE International Conference on Data Mining, pp. 173–182 (2008)
15. Ultsch, A.: Maps for the Visualization of high-dimensional Data Spaces. In: Proc. of Workshop on Self-Organizing Maps 2003, pp. 225–230 (2003)