8/10

2012 - 16pages - 2021.12.29

- some basic concept
- not indepth enough.

Chapter 10

Application of Self-Organizing Maps in Text Clustering: A Review

Yuan-Chao Liu, Ming Liu and Xiao-Long Wang

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/50618

1. Introduction

Text clustering is one of the most important text mining research directions. Despite the loss of some details, clustering technology simplifies the structure of data set, so that people can observe the data from a macro point of view.

After clustering process, the text data set can be divided into some different clusters, making the distance between the individuals in the same cluster as small as possible, while the distance between the different categories as far away from each other as possible.

Similar as text classification, text clustering is also the technology of processing a large number of texts and gives their partition.What is different is that text clustering analysis of the text collection gives an optimal division of the category without the need for labeling the category of some documents by hand in advance, so it is an unsupervised machine learning method. By comparison, text clustering technology has strong flexibility and automatic processing capabilities, and has become an important means of effective organization and navigation of text information. Jardine and van Rijsbergen made the famous clustering hypothesis: closely associated documents belong to same category and the same request [1]. Text clustering can also act as the basic research for many other applications. It is a preprocessing step for some natural language processing applications, e.g., automatic summarization, user preference mining, or be used to improve text classification results. YC Fang, S. Parthasarathy, [2] and Charu [3] use clustering techniques to cluster users' frequent query and then the results to update the FAQ of search engine sites.

Although both text clustering and text classification are based on the idea of class, there are still some apparent differences: the classification is based on the taxonomy, the category distribution has been known beforehand. While the purpose of text clustering is to find the top-

ic structure of documents [4] [5] [6] [7] [8] [9] [10]. Yasemin Kural [11] made a lot of experiments and compared the clustering mode and linear array mode for search engine, the results show that the former can indeed increase information access efficiency greatly.

Although there are many clustering methods, SOM has attracted many researchers in recent years. In this chapter, we reviewed the application of Self-Organizing Maps in Text Clustering. Our recent works on SOM based text clustering are also introduced briefly. The remaining of this chapter is organized as follows. Section 2 gives a review about the advances in text clustering and SOM; section 3 presents our recent work on application of self-organizing maps in text clustering. Then in section 4 some conclusions and discussions are given.

2. The Advances In Text Clustering And SOM

2.1. Text Clustering And Its Recent Research And Development

Text clustering is an unsupervised process that is not dependent on the prior knowledge of data collection, and based solely on the similarity relationship between documents in the collection to separate the document collection into some clusters. The general mathematical description of text clustering can be depicted as follows:

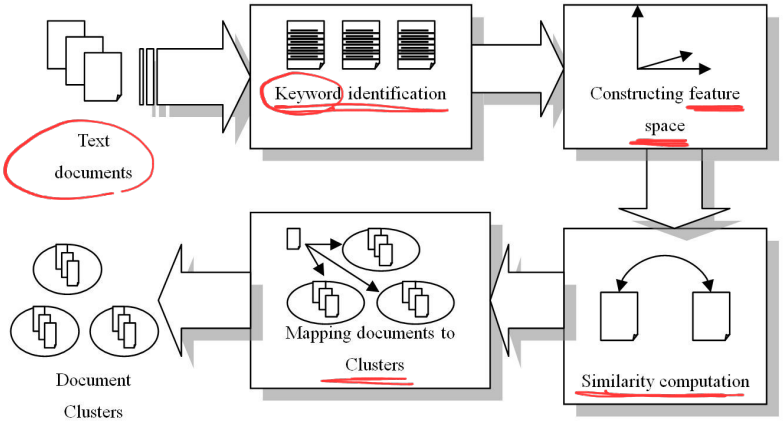

Figure 1. The main framework for text clustering system.

Suppose $C=\{d_1, d_2, \ldots, d_n\}$ is a collection of documents to be clustered, each document d_i can be represented as high-dimensional space vector $d_i=\{w_1, w_2, \ldots, w_i\}$ by the famous vector space model (VSM), where w_i means the weight of d_i on feature j. The purpose of text clustering is to divide C into $C_1, C_2, \ldots, C_x$, $C_1 \cup C_2 \cup \ldots \cup C_x = C$, here $1 \le i \ne j \le k$. For hard clustering, each document can belong to only one class, i.e. $C_i \cap C_j = \Phi$. Whereas for soft clus-

http://dx.doi.org/10.5772/50618

tering, one document may belong to multiple clusters. Membership degree μ_{ij} can be used to denote how much d_i belongs to cluster C_j.

Compared with other data types, text data is semi-structured. This makes man database-based algorithms does not apply to text clustering.

One important preprocessing step for text clustering is to consider how the text content can be represented in the form of mathematical expression for further analysis and processing. The Common method is Salton's vector space model [12] (Vector Space Model, VSM). The basic idea is: one feature space are constructed firstly, each dimension means one term, which comes from the key words of each document. Then each document is represented as one vector in this feature space. The document vector is usually a sparse vector as the dimension is very huge.

Dimensionality reduction is an essential step in text clustering. There are several techniques to reduce the dimension of the high-dimensional feature vector. PCA (Principal Component Analysis) method is one of the widely used dimension reduction techniques. Given an n × m-order document-term matrix, the k eigenvectors of the PCA with an m × m-order covariance matrix is used to reduce the dimension of the word space, and ultimately resulted in a k-term space dimension, which is much smaller than m.

LSI (Latent Semantic the Indexing) method is also widely used in the field of information retrieval, dimensionality reduction. It is in essence similar with the PCA. LSI make singular value decomposition not on covariance matrix, but on the initial n × m-order document–term matrix, and then selecting these singular eigenvectors as representative, thereby reduces the dimension.

Another problem is how to extract important features from documents. Mark P. Sinka and David W. Corne [13] argue that stop word removal will improve the text clustering effect. They also pointed out that after obtaining all unique words in the collection, you can only keep some high-frequency words to construct the space. Anton V. Leouski and W. Bruce Crof demonstrated that for each document, it is necessary to select only some important words to represent the document, and can basically meet the needs of the cluster without impacting clustering results. Literature [14] proposed a method to extract the key words in the document as features Literature [15] use latent semantic indexing (LSI) method to compress the dimension of the clustering feature space. Besides, ZhengYu Niu [16] and STANISŁAW OSIŃSKI [17], etc also performed research on feature selection.

Assume there are five documents doc1 doc2, doc3, doc4, and doc5. For each document, the first steps are segmenting, stop word removal, and word frequency counting. In order to improve the clustering efficiency, only the words which frequency is above a certain threshold value are used to construct the feature space. Studies have shown that such a treatment will not have an adverse impact on the clustering quality. Then the feature space can be constructed by using the term set which comes from all these terms. Each document is represented as a vector in the feature space. Fig.2. depicts the preprocessing steps for text clustering.

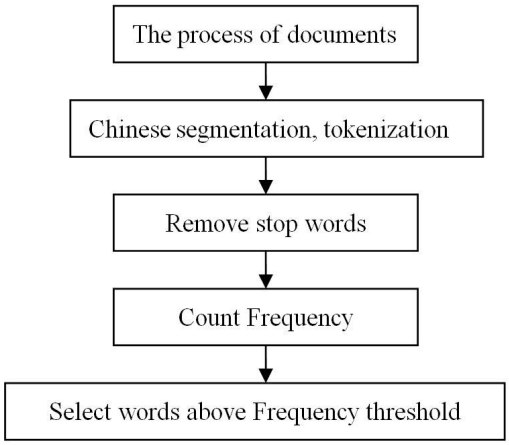

Figure 2. the preprocessing steps of text document for text clustering.

Suppose the feature space is (apple, banana in the cat, window), and feature words frequency threshold is 2, then the following example document-term matrix can be formed:

	(Apple,	banana,	cat,	window)
doc1 = (	5,	3,	0,	4)
doc2 = (	4,	6,	0,	3)
doc3 = (	0,	3,	7,	5)
doc4 = (	8,	0,	9,	0)
doc5 = (	5,	0,	0,	3)

As all documents are represented as the vector in the same feature space, thus it is more convenient for computing the document similarity. In fact, the similarity calculation is very frequent for most clustering algorithms. In addition, as there are usually many common words in different documents, the actual dimension of the feature space is less than the sum of the number of words selected from each document.

The evaluation of word importance. Take a science paper as an example, it is shown that about 65% to 90% author-marked keywords can be found in the main content in the original paper[18]. This means that by importance evaluation, the key words can be extracted from documents to represent the main content. Basically, keyword extraction can be seen as a supervised machine learning problems; this idea is first proposed by Turney [19]. Turney also make a comparative study based on genetic algorithms and decision tree-based keywords extraction algorithm. Factors which can denote the word importance includes word frequency, word location (title, caption and etc.). Many researches showed that high-frequency words are the more important words. Some typical keyword extraction system has been listed in table 1.

name	websites
NRC's Extractor	http://ai.iit.nrc.ca/II_public/extractor/
Verity's Search 97	http://www.verity.com/
KEA	http://www.nzdl.org/Kea/
GenEX	http://extractor.iit.nrc.ca/
Microsoft office 2003	http://www.microsoft.com/
Eric Brill's Tagger	ftp://ftp.cs.jhu.edu/pub/brill/Programs/

Table 1. Some Classical Keyword Extraction Systems.

2.2. Two Clustering strategies in Text Clustering: whole clustering and incremental clustering

There are two common Clustering strategies, and both need to measure the similarity of the document.

The first strategy is the "complete" strategy, or called "static" strategy. During the clustering process, the documents collection did not change neither adding documents, nor removing documents. At the beginning of clustering, the documents in the collection are fixed. In the clustering Method based on this policy, an N*N similarity matrix can be generated from the beginning and there are $N(N-1)/2$ similarity values in the matrix. As it will compare the similarity among any documents, the computation is very costly.

The second strategy is the strategy of "incremental"[20]. In many occasions, the document collection can be increased at any time in the clustering process. When adding a document, it will be merged into the existing cluster, or you can separate it as a new category. While increasing documents, it may be necessary to perform re-clustering.

There are some methods to calculate the similarity or distances between different clusters: 1) the shortest distance method (single link method). If G_p, G_q are two different clusters, $D_s(p, q)=\min\{d_{ij} \mid i \in G_p,\ j \in G_q\}$; 2) the longest distance method. If G_p, G_q are two different clusters, $D_s(p, q)=\max\{d_{ij} \mid i \in G_p,\ j \in G_q\}$;3) Group average method. $D_s^2(p, q)=\frac{1}{n_p n_q}\sum_{\substack{i\in G_p \\ j\in G_q}} d_{ij}^2$;4) The centric method. $\bar{x}_G=\frac{1}{L}\sum_{i=l}^{L} x_i$ Mean Quantization Error (abbreviated as MQE) is adopted as convergence condition as performed by Ref. [10-12]. Since MQE can measure the average agglomeration degree of clustering results, when its value is less than a threshold such as 0.01 (which is adopted by Kohonen in Ref. [21]), this dynamic algorithm stops.

$$MQE=\frac{\sum_{j=1}^{C}\sum_{Di\in Cj}\frac{|Di-Nj|^2}{|Cj|}}{C} \tag{1}$$

Where, C represents the quantity of clusters. Nj represents one neuron. Cj represents the cluster, which includes the data that are more similar to Nj than to other neurons. |Cj| represents the quantity of the data included by Cj. Di represents one datum among Cj.

2.3. SOM And Its Application For Text Clustering

Self-organizing map network (SOM, for abbreviation) is first proposed by T.Kohonen Professor in University of Helsinki in Finland, also known as the Kohonen network [22]. Kohonen believes that a neural network will be divided into different corresponding regions while receiving outside input mode, and different regions have different response characteristics for corresponding input mode, and this process can be done automatically. SOM network has the following main properties: 1) The cluster center is the mathematical expectation of all the documents in this cluster; 2) "cluster" of input data, and maintaining the topological order. Fully trained SOM network can be viewed as a pattern classifier. By inputting a document, the neurons representing the pattern class-specific in the output layer will have the greatest response.

The self-organizing map is proposed based on this idea, which is similar to the self-organization clustering process in human brain[23] [24]. SOM clustering method has been successfully used in the field of digital libraries, text clustering and many other applications [25] [26] [27] [28].

The running process of the SOM network can be divided into two stages: training and mapping. In the training phase, the samples were input randomly. For a particular input pattern, there will be a winning node in the output layer, which produces the greatest response. At the beginning of the training phase, which node in the output layer will generate the maximum response is uncertain. When the category of the input pattern is changed, the winning node of the two-dimensional plane will also change. Due to the lateral mutual excitatory effects, Nodes around the winning node have a greater response, so all the nodes of the winning node and its neighborhood will both perform different levels of adjustment.

SOM adjust the weights of the output layer nodes with a large number of training samples, and finally each node in the output layer is sensitive to a specific pattern class. When the class characteristics of the two clusters are close, the nodes on behalf of these two clusters are also close in position.

After the training of the SOM network, the relation between output layer nodes and each input pattern can be determined, then all the input patterns can be mapped onto the nodes in the output layers, which is called mapping steps.

SOM method usually requires pre-defining the size and structure of the network. There are some methods which can achieve this purpose [29][30][31]. The basic idea is to allow more rows or columns to be dynamically added to the network, make the network more suitable for the simulation of the real input space.

SOM method requires the definition of neighborhood function and learning rate function beforehand. There is no fixed pattern in Kohonen model on the choice of neighborhood

function and learning rate function, they are generally selected based on the heuristic information [32][33]. H.Yin proposed BSOM, which is SOM method based on Bayesian [34]. The basic idea is to minimize the KL distance of the data density and neural models. KL distance can measure the distance or deviation between the environment probability density and real probability density, its value is generally a positive number. Learning process can be done within a fixed range of the winner neuron. The BSOM therefore gives a new perspective on the role of the conventional SOM neighborhood function. In addition, Filip, Mulier and Vladimir Cherkassky studied the learning rate function strategy in SOM [35]. The experimental results show that the location of the neurons may be over affected by the last input data. Filip, Mulier, Vladimir Cherkassky has improved the learning rate function and neighborhood function, to make impact of the input training data on the neuron location more uniform.

2.4. The Comparison Of SOM With Other Text Clustering Methods

Besides from SOM, There are also two widely used text clustering methods: AHC clustering method and K-means clustering method. The basic steps of AHC for text clustering method are as follows:

1. Calculate the document similarity matrix;
2. Each document is seen as a cluster firstly;
3. Merge the nearest two clusters into one;
4. Update the similarity matrix, i.e, re-calculating of the similarity of the new cluster with the current cluster; if there are only one cluster, then go to step 5), otherwise go to step 3);
5. End.

Researchers often use two different methods to cut the hierarchical relationships. One is to use the number of clusters as segmentation standard; another method is using the similarity as the segmentation standard, that is, when the similarity between two clusters is lower than a given threshold, the clustering algorithm will stop. Besides, it has been shown that the clustering entropy [36] can be used as the termination conditions of the hierarchical clustering method:

$$En = (\sum_{j=1}^{k} \sum_{i=1}^{n} e(p_i^{(j)}, p_0^{(j)})) + \sum_{j=1}^{k} e(p_0^{(j)}, c_0) \quad (2)$$

The first expression in the right side of the formula is the intra-cluster entropy; the second means the inter-cluster entropy. When En is smallest, the clustering result achieves optimum value. c_0 is the center of all the samples. $p_i^{(j)}$ is the i documents for cluster j. $p_0^{(j)}$ is the center of the jth clusters. K is the number of clusters, n_j is the number of documents in cluster j.

K-means clustering algorithm is the typical dynamic partition method [37] [38] [39] [40]. The basic steps [41] are as follows:

1. Randomly select K documents, which represent initial cluster centroids.
2. Assign each document to the cluster that has the closest centroid.
3. When all documents have been assigned, recalculate the K centroids.
4. Repeat Steps 2 and 3 until the centroids no longer change.
5. Output the separation of these documents, i.e. different clusters.

For K-means, if the k value selected is inappropriate or the choice of initial accumulation point is uneven, the clustering process will be delayed and the clustering results are also adversely affected. Traditionally, there are mainly two methods to select the initial cluster center: 1) randomly select k points; 2) use empirical method to select the initial cluster centers. In addition, the researchers also made some of the more complex but very effective method: 1) the gravity center method. The basic idea is: first calculate the gravity center of all the samples as the first point; then select a positive number as the minimum critical distance. Input all the samples in turn, if the input sample has distance greater than d, it will be deemed as a new clustering point; 2) the density method. Two positive numbers d1 and d2 ($d1d2$) are first set, form the ultra-dimensional ball using d1 as the radius, which density is calculated as the number of samples in that ball. Select the sample with the maximum density as the first center; select the sample with the second maximum density.

Generally, SOM has proven to be the most suitable document clustering method. It can map documents onto two-dimensional diagram to show the relationship between the different documents. SOM can depict text in more figurative and better visual way. High-dimensional space can be transformed into two-dimensional space, and the similarity between the input data in the multi-dimension space is well maintained in the two-dimensional discrete space, the degree of similarity between the high dimensional spatial data can also be transformed into the location proximity of representation space, which can maintain the topological order. SOM also has the following advantages: 1) noise immunity; 2) visualization; 3) parallel processing.

Text Clustering is a high-dimensional application and closely related to the semantic features. The above characteristics of SOM make it very suitable for text clustering.

2.5. Dynamic clustering of SOM

Self-Organizing-Mapping (abbreviated as SOM) is one of the most extensively applied clustering algorithm for data analysis, because of its characteristic that its neuron topology is identical with the distribution of input data. However, the inconvenience, that it needs to predefine two parameters of cluster quantity and neuron topology, prevents it from prevailing in online situation.

As indicated by Ref. [42][43][44], many methods have been proposed to cluster dynamic data. For example, Dhillon et al. [45] proposed a dynamic clustering algorithm to help analyze the transfer of information. Unfortunately, this algorithm is time-consuming and impractical, since it needs to run several times. Ghaseminezhad and Karami [46] improve this algorithm by employing SOM structure, which forms an initial neuron topology at first and then

dynamically tunes its topology once input data are updated. However, its neuron topology is fixed in advance and too rigid to be altered.

In order to enable neuron topology easily to be altered, some self-adaptive algorithms have been proposed. The prominent merit of them is that they don't need to set any assumption about neuron topology in advance. For example, Melody in Ref. [47] initializes a neuron topology of small scale at first and then gradually expands it following the update of input data. Tseng et al in Ref. [48] improve this algorithm by tuning neuron topology in virtue of dynamically creating and deleting the arcs between different neurons.

Unfortunately, aforementioned self-adaptive algorithms have two defects. One is that, when neuron topology isn't suitable for current input data, they will insert or split neurons, whereas, these newly created neurons may locate out of the area where input data distribute. The other is that, they fail to preserve topology order. Therefore, they can't perform competitive learning as transitional SOM based algorithms, which will generate some dead neurons and they will never be tuned. The detailed discussions are indicated in Ref. [49][50].

For avoiding predefining cluster quantity, some scalable SOM based clustering algorithms are proposed, such as GSOM in Ref. [51] and GHSOM in Ref. [52]. Nevertheless, neuron topologies of them are fixed as liner, cycle, square or rectangle in advance. These kinds of topologies are too rigid, and hardly to be altered.

In order to solve this problem, some topology adaptive algorithms have been proposed, such as GNG in Ref. [53], PSOM in Ref. [54], and DASH in Ref. [55]. These algorithms free of predefining neuron topology and can automatically construct it to let it conform to the distribution of input data.

3. Our Recent Work On Application Of Self-Organizing Maps In Text Clustering

3.1. The Conceptual SOM Model For Text Clustering

Most of the existing text clustering methods simply use word frequency vector to represent the document, with little regard to the language's own characteristics and ontological knowledge. When documents are clustered using conventional "SOM plus VSM" way, it is hard to grasp the underlying semantic knowledge and consequently the clustering quality may be adversely affected. However, we notice that the documents in same cluster are very relevant to each other even though there are few common words shared by these documents, so the relevance calculation among documents can be simplified by the relevance calculation of words in documents.

Y.C. Liu et al. have proposed a conceptional self-organizing map model (ConSOM) [56]for text clustering, in which neurons and documents are represented by the vector in extended concept space and that in traditional feature space. It has been shown that by importing concept relevance knowledge, SOM can achieve better performance than traditional mode due

to its semantic sensitivity. Figure 3 give the basic principle for ConSOM. After both extended concept space and traditional feature space are constructed, all documents and neurons are represented by two vectors: traditional vector VF purely formed by word frequency and extended concept vector VC, as shown in Fig. 3. Table 2.presents Concept Representation of Word in HowNet.

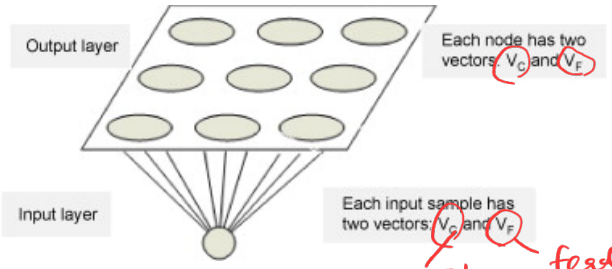

Figure 3. The basic principle of ConSOM.

NO.=124922
W_C=医生；G_C=N；E_C=NULL；W_E=Dr.；G_E=N；E_E=NULL；DEF={human|人:HostOf={Occupation|职位},domain={medical|医},{doctor|医治:agent={~}}}
NO.=048622
W_C=患者；G_C=N；E_C=NULL；W_E=patient；G_E=N；E_E=NULL；DEF={human|人:domain={medical|医},{SufferFrom|罹患:experiencer={~}},{doctor|医治:patient={~}}}
NO.=124949
W_C=医院；G_C=N；E_C=NULL；W_E=hospital；G_E=N；E_E=NULL；DEF={InstitutePlace|场所:domain={medical|医},{doctor|医治:content={disease|疾病},location={~}}}

Table 2. Concept Representation of Word in HowNet.

3.2. Fast SOM Clustering Method For Large-Scale Text Clustering

Conventional data clustering methods frequently perform unsatisfactorily for large text collections due to 3 factors:1) there are usually large number of documents to be processed; 2) the dimension is very huge for text clustering; 2) the computation complexity is very high. So it is very necessary to improve the computation speed.

As similarity computation is very crucial for text clustering, and has much impact on clustering efficiency, Y. liu and etc[57], propose one novel feature representation and similarity computation method to make SOM text clustering much faster. Each document is coded as the collection of some keywords extracted from the original document, and will directly be input to SOM, whereas each output layer node of SOM are coded as numerical vector as that of most Kohonen Networks.

http://dx.doi.org/10.5772/50618

In order to directly separate documents into different groups, ring topology is adopted as our SOM structure, thus the number of groups can be any integral values. Like Kohonen Networks, it consists of two layers, input layer and output layer; each node in output layer corresponds to one cluster. Only neurons need to be represented as high-dimension vector, whereas the document will be coded as indexes of keywords.

3.3. The Variant Of SOM Model For Dynamic Text Clustering

Figure 5 shows the ring output layer topology of V-SOM [58]. The advantage of this topology is that sector number (node number) can be any integers, and it will be possible to reflect topic distribution of the input documents more finely and make full use of neurons. Besides, the number of neighboring neurons for each neuron is same, thus it can help avoid edge effect which usually happens by using rectangular or hexagonal topology. Neurons can be inserted gradually to avoid lack-of-use phenomenon of neurons. R^2 cluster criterion is used to find suitable network size which can reflect topic distribution of input documents.

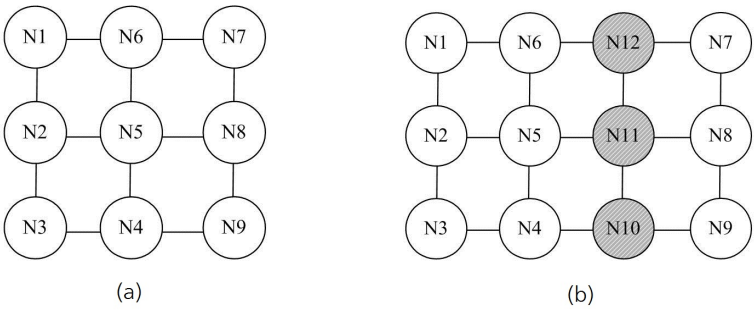

Figure 4. The rectangular topology of GHSOM (N10, N11, N12 in Figure1. (b) are the newly inserted neurons).

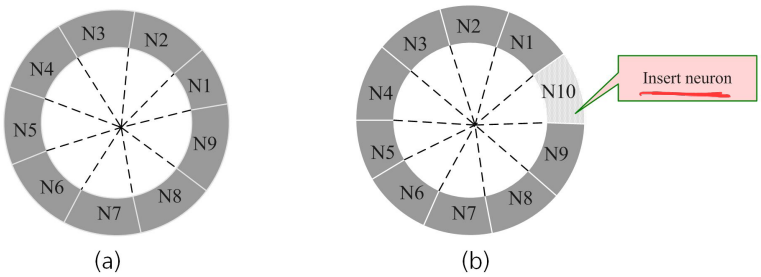

Figure 5. The Ring Topology of V-SOM. N10 Is The Inserted Node In Figure (b).

4. Conclusions and discission

In conclusion, SOM has obvious advantage in terms of topology preserving order, anti-noise ability. By using self-organizing map network as the main framework of the text clustering, semantic knowledge can also be easily incorporated so as to enhance the clustering effect.

First, SOM can better handle the dynamic clustering problem through various kinds of dynamic vari-structure model. E.g. V-SOM model, which combine the decomposition strategy and neuronal dynamic expansion, under the guidance of clustering criterion function, dynamically and adaptively adjust the network structure, thus the clustering results can better reflect the topic distribution of input documents.

Second, semantic knowledge can be easily integrated into the SOM. Due to the diversity and complexity of language, same concept may also have different forms of expression. The traditional "VSM+SOM" mode rely solely on the frequency of feature words, and cannot grasp and embody semantic information. We use HowNet as a source of conceptual knowledge and perform effective integration with statistical information in order to enhance the sensitive ability of the clustering. if there are clusters with hidden common concept, they will be merged into one cluster, even if they are less common words shared by these documents.

Finally, the SOM's unique training structure provides convenience for the realization of parallel clustering and incremental clustering, thus contributing to improve the efficiency of clustering. Incremental clustering also makes it more suitable for dynamic clustering of web documents.

Author details

Yuan-Chao Liu*, Ming Liu and Xiao-Long Wang

*Address all correspondence to: lyc@insun.hit.edu.cn

School of Computer Science and Technology, Harbin Institute of Technology, China

References

[1] Jardine, N., & van Rijsbergen, C. J. (1971). The use of hierarchic clustering in information retrieval. *Information Storage and Retrieval*, 7(1), 217-240.

[2] Fang, Y. C., Parthasarathy, S., & Schwartz, F. (2002). Using Clustering to Boost Text Classification. *In Proceedings of the IEEE ICDM Workshop on Text Mining*, 101-112.

[3] Charu. (2004). On using Partial Supervision for Text Categorization. *IEEE Transactions On Knowledge And Data Engineering*, 16(2), 245-258.

[4] Croft, W. B. (1978). Organizing and searching large files of documents. *Ph.D. Thesis*, University of Cambridge.

[5] Hearst, M. A., & Pedersen, J. O. (1996). Reexamining the cluster hypothesis: Scatter/Gather on retrieval results. *Proceedings of the 19th International ACM SIGIR Conference on Research and Development in InformationRetrieval (SIGIR'96)*, 76-84.

[6] Leouski, A. V., & Croft, W. B. (1996). An evaluation of techniques for clustering search results. *Technical Report IR-76*, Department of Computer Science, University of Massachusetts, Amherst.

[7] Allen, R. B., Obry, P., & Littman, M. (1993). An interface for navigating clustered document sets returned by queries. *Proceedings of the ACM Conference on Organizational Computing Systems*, 166-171.

[8] Cutting, Douglass R., Karger, David R., & etc. Scatter/Gather. (1992). A Cluster-based Approach to Browsing Large Document Collections. *SIGIR'92*, 318-329.

[9] Voorhees, E. M. (1986). The efficiency of Inverted Index and Cluster Searches. *In: Proceedings of the ACM Conference on R&D in IR. Pisa*, 1986, 164-174.

[10] El -Hamdouchi, A., & Willett, P. (1989). Comparison of Hierarchic Agglomerative Clustering Methods for Document Retrieval. *The Computer Journal*, 32(3), 220-227.

[11] Kural, Yasemin, Robertson, Steve, & Jones, Susan. (2001). Clustering Information Retrieval Search Outputs. *Information Processing & Management*, 1630-1700.

[12] Salton, G., Wong, A., & Yang, C. (1975). A vector space model for automatic indexing. Communications of the ACM ., 18(11), 613-620.

[13] Seung-Shik, Kang. (2003). Keyword-based Document Clustering. *The 6th International Workshop on Information Retrieval with Asian Languages*, 132-137.

[14] Lerman, K. (2004). Document Clustering in Reduced Dimension Vector Space. *Proceedings of CSAW'04*.

[15] Niu, Z. Y., Ji, D. H., & Tan, C. L. (2004). Document clustering based on cluster validation. *13th Conference on Information and Knowledge Management. CIKM 2004*, Washington DC, USA, 501-506.

[16] Osiński, S. (2004). Dimensionality Reduction Techniques for Search Results Clustering. *MSc. thesis*, University of Sheffield, UK.

[17] Wang, B. B., Mc Kay, R. I., Hussein, A., Abbass, , et al. (2003). A comparative study for domain ontology guided feature extraction1. Darlinghurst, Australia. *In Proc of 26th Australian Computer Science Conference (ACSC2003)*, Australian Computer Society Inc., 69-78.

[18] Witten, I. H., Paynter, G. W., Frank, E., Gutwin, C., & Nevill-Manning, C. G. (1999). KEA: Practical automatic keyphrase extraction. *Proceedings of DigitalLibraries 99 (DL'99)*, ACM Press, 254-256.

[19] Turney, P. (2002, July). Mining the Web for Lexical Knowledge to Improve Keyphase Extraction: Learning from Labeled and Unlabeled Data. *P. Source: NRC/ERB-1096*, NRC Publication Number: NRC 44947.2002.

[20] Zamir, O., & Etzioni, O. (1999). Grouper: A dynamic clustering interface to web search results. *Computer networks*, 31, 1361-1374.

[21] Herbert, J. P., & Yao, J. T. (2009). A Granular Computing Framework for Self-Organizing Maps. *Neurocomputing*, 72, 2865-2872.

[22] Niklasson, L., Bodén, M., & Ziemke, . (1998). Self-organization of very large document collections: State of the art. *Proceedings of ICANN98, the 8th International Conference on Artificial Neural Networks*, 65-74.

[23] Lin, X., Soergel, D., & Marchionini, G. (1991). A self-organizing semantic map for information retrieval. *Proceedings of the annual international ACM SIGIR conference on research and development in information retrieval*, 262-269.

[24] Su, K. Lamu-Chun, Chang, Hsiao-te, & Chou, Chien-hsing. (1996). Approach to interactive exploration. *In proc int'l conf knowledge discovery and data mining(KDD'96)*, 238-243.

[25] Miikkulainen, R. (1990). Script recognition with hierarchical feature maps. *Connection science*, 2, 83-101.

[26] Merkl, D. (1993). Structuring software for reuse: the case of self-organizing maps. Piscataway, NJ, IEEE Service Center. *Int. Joint Conf. on Neural Networks*, III, 1993, 2468-2471.

[27] Roussinov, D., & Ramsey, M. (1998). Information forage through adaptive visualization. *The Third ACM Conference on Digital Libraries*, 303-304.

[28] Rauber. (1999). LabelSOM: On the labeling of self-ofganizing maps. *In proc int'l joint conf neural networks(IJCNN'99)*.

[29] Martinetz, T. M., & Schulten, K. J. (1991). A "neural-gas'' network learns topologies' in Kohonen. *Artificial neural networks*, 397-402.

[30] Fritzke, B. (1995). Growing grid-a self-organising network with constant neighbourhood range and adaptation strength. *Neural Process. Letters*, 2, 9-13.

[31] Bauer, Ha., & Villmann, T. (1997). Growing a hypercubical output space in a self-organising feature map. *IEEE Transactions on Neural Networks*, NN-8(2), 218-226.

[32] Ritter, H., Martinetz, T., & Schulten, K. (1992). *Heurd Computation and Self-Organizinp Maps: Introduction*, Addison-Wesley.

[33] Kohonen, T. (1990). The self-organizing map. *Proc. of the IEEE*, 9, 1464-1479.

[34] Yin, H., & Allinson, N. M. (1990). Bayesian self-organising map for gaussian mixtures. *IEEE Proceedings: Vision, Image and Signal Processing*, 148(4), 234-240.

[35] Mulier, F., & Cherkassky, V. (1994). *Learning rate schedules for self-organizing maps*, *In Proceedings of 12th International Conference on Pattern Recognition*, 2, 224-228.

[36] Jung , Yunjae. (2001). Design and Evaluation of Clustering Criterion for Optimal Hierarchical Agglomerative Clustering. *Phd. thesis*, University of Minnesota.

[37] Steinbach, M., Karypis, G., & Kumar, V. (2000). A comparison of document clustering techniques. *In KDD Workshop on Text Mining*.

[38] Larsen, Bjornar, & Chinatsu, Aone. (1999). Fast and effective text mining using linear-time document clustering. *In Proc. of the Fifth ACM SIGKDD Int'l Conference on Knowledge Discovery and Data Mining*, 16-22.

[39] Aggarwal, Charu C., Gates, Stephen C., & Yu, Philip S. (1999). On the merits of building categorization systems by supervised clustering. *In Proc.of the Fifth ACM SIGKDD Int'l Conference on Knowledge Discovery and Data Mining*, 352-356.

[40] Cutting, D. R., Pedersen, J. O., Karger, D. R., & Tukey, J. W. (1992). Scatter/gather: A cluster-based approach to browsing large document collections. Copenhagen. *In Proceedings of the ACM SIGIR*, 318-329.

[41] Luke, Brian T. (1999). *K-Means Clustering*, http://fconyx.ncifcrf.gov/~lukeb/kmeans.html.

[42] Martin, S., & Detlef, N. (2006). Towards the Automation of Intelligent Data Analysis. *Applied Soft Computing*, 6, 348-356.

[43] Zhou, X. Y., Sun, Z. H., Zhang, B. L., & Yang, Y. D. (2006). Research on Clustering and Evolution Analysis of High Dimensional Data Stream. *Journal of Computer Research and Development*, 43, 2005-2011.

[44] Huang, S., Chen, Z., Yu, Y., & Ma , W. Y. (2006). Multitype Features Coselection for Web Document Clustering. *IEEE Transactions on Knowledge and Data Engineering*, 18, 448-459.

[45] Dhillon, I. S., Guan, Y. Q., & Kogan, J. (2002). Iterative Clustering of High Dimensional Text Data Augmented by Local Search. *In: Proceedings of the Second IEEE International Conference on Data Mining*, 131-138, IEEE Press, Japan.

[46] Ghaseminezhad, M. H., & Karami, A. (2011). A Novel Self-Organizing Map (SOM) Neural Network for Discrete Groups of Data Clustering. *Applied Soft Computing*, 11, 3771-3778.

[47] Melody , Y. K. (2001). Extending the Kohonen Self-Organizing Map Networks for Clustering Analysis. *Computational Statistics & Data Analysis*, 38, 161-180.

[48] Tseng, C. L., Chen, Y. H., Xu, Y. Y., Pao, H. T., & Fu, H. C. (2004). A Self-Growing Probabilistic Decision-Based Neural Network with Automatic Data Clustering. *Neurocomputing*, 61, 21-38.

[49] Tsai, C. F., Tsai, C. W., Wu, H. C., & Yang, T. (2004). ACODF: A Novel Data Clustering Approach for Data Mining in Large Databases. *Journal of Systems and Software*, 73, 133-145.

[50] Lee, S., Kim, G., & Kim, S. (2011). Self-Adaptive and Dynamic Clustering for Online Anomaly Detection. *Expert Systems with Applications*, 38, 14891-14898.

[51] Alahakoon, D., , S., Halganmuge, K., & Srinivasan, B. (2000). Dynamic self-organizing maps with controlled growth for knowledge discovery. *IEEE Transactions on Neural Networks*, 11(3), 601-614.

[52] Merkl, Rauber D., & Dittenbach, M. (2002). The growing hierarchical self-organizing map: exploratory analysis of high-dimensional data. *IEEE Transactions on Neural Networks*, 13(6), 1331-1341.

[53] Qin, A.-K., & Suganthan, P.-N. (2004). Robust growing neural gas algorithm with application in cluster analysis. *Neural Networks*, 17(8-9), 1135-1148.

[54] , L., Robert, K., & Warwick, K. (2002). The plastic self organising map. Hawaii. *Proceedings of the 2002 International Joint Conference on Neural Networks, IEEE*, 727-732.

[55] Hung, C., & Wermter, S. (2003). A dynamic adaptive self-organising hybrid model for text clustering. Melbourne. *Proceedings of the Third IEEE International Conference on Data Mining, IEEE*, Florida, USA, 75-82.

[56] Liu, Yuanchao, Wang, Xiaolong, & Wu, Chong. (2008, January). ConSOM: A conceptional self-organizing map model for text clustering. *Neurocomputing*, 71(4-6), 857-862.

[57] Liu, Yuan-chao, Wu, Chong, & Liu, Ming. (2011, August). Research of fast SOM clustering for text information. *Expert Systems with Applications*, 38(8), 9325-9333.

[58] Liu, Yuanchao, Wang, Xiaolong, & Liu, Ming. (2009). V-SOM: A Text Clustering Method based on Dynamic SOM Model. *Journal of Computational Information Systems*, 5(1), 141-145.