A novel self-organising clustering model for time-event documents

Chihli Hung[1], Stefan Wermter[2]

[1]Department of Management Information Systems,

Chung Yuan Christian University, Taiwan, R.O.C.

[2]School of Computing and Technology, University of Sunderland, UK

[1]chihli@cycu.edu.tw, [2]stefan.wermter@sunderland.ac.uk

Category of the paper: Research paper

Abstract

Purpose

Neural document clustering techniques, e.g., self-organising map (SOM) or growing neural gas (GNG), usually assume that textual information is stationary on the quantity. However, the quantity of text is ever-increasing. We propose a novel dynamic adaptive self-organising hybrid (DASH) model, which adapts to time-event news collections not only to the neural topological structure but also to its main parameters in a non-stationary environment.

Design/methodology/approach

Based on features of a time-event news collection in a non-stationary environment, we review the main current neural clustering models. The main deficiency of them is a need of pre-definition of the thresholds of unit-growing and unit-pruning. Thus, the dynamic adaptive self-organising hybrid (DASH) model is designed for a non-stationary environment.

Findings

We compare DASH with SOM and GNG based on an artificial jumping corner data set and a real world Reuters news collection. According to our experimental results, the DASH model is more effective than SOM and GNG for time-event document clustering.

Practical implications

A real world environment is dynamic. This paper provides an approach to present news clustering in a non-stationary environment.

Originality/value

Text clustering in a non-stationary environment is a novel concept. We have demonstrated DASH, which can deal with a real world data set in a non-stationary environment.

Keywords: Text Clustering, Knowledge Engineering, Self-Organising Map, Time-Event Text Processing

Introduction

In an era of Internet, a vest amount of textual information can overwhelm users. By grouping similar concepts of documents, an organised structure quickly reduce the search space and help users to access relevant documents (van Rijsbergen, 1979). Many document clustering approaches, including statistical solutions and artificial neural networks, have been proposed for these tasks (e.g. Chang and Chen, 2006; Chen and Chen, 2006; Hung et al., 2004; Pullwitt, 2002; Jain et al., 1999). Particularly, in the field of artificial neural networks, self-organising maps (SOMs) have been proposed for document clustering (Kohonen, 1984). Documents containing a similar concept are grouped into the same unit on a map and units representing a similar concept are located nearby on the map. Therefore, documents are self-organised to an ordered map, which can be treated as an Internet browsing interface such as the WebSOM project (Honkela et al., 1997).

Real-world textual information such as news is dynamic and continuously growing. In a news collection, some specific events occur over a specific period. In other words, the news topic is changing over time. However, most of document clustering approaches are based on a common assumption that the documents are organised as a stationary collection (i.e., a fixed number of documents). Thus, although the particular structure of the current stationary document collection has been identified, this becomes outdated for new information.

Therefore, motivated by the need for non-stationary organisation of information, we propose a novel neural clustering model, the dynamic adaptive self-organising hybrid (DASH) model for time-event documents in a non-stationary environment. We use the Reuters corpus volume one, RCV1 (Rose et al., 2002), to model time-event documents and evaluate the DASH model based on classification accuracy and average quantization error (AQE).

Background and purpose

Traditionally, the factor of time is not involved in an artificial learning environment for clustering. However, documents, e.g. news articles, usually have some relationship with time. Similar articles related to the same specific event are presented in a specific time period. Topics of news articles are gradually changed over time and the latest event generally attracts more attention (Smeaton et al., 1998). For the word usage, some specific terms have a strong relationship with some specific topics. Thus, it can be anticipated that the word usage is also slightly changing over time and more co-occurring words are used for two news articles if they happened recently.

The size of the document population is always increasing in a real world. Thus an organised structure of documents built based on a traditional static training set is inevitable to produce a wrong decision for an ever-changing test set. A time-event document clustering model is a model which is able to cluster documents changed over time and reflect the latest knowledge from the latest time-event documents. In other words, a time-event document clustering model is able to learn continuously, stay up-to-date and provide the results at any time.

A time-event document clustering model shown in Figure 1 associates document time-stamps based on time. Each data set contains several news articles issued within a short period. The incremental learning model adapts to the latest data set by continuously adjusting the learned structure. When the knowledge from the existing data set in the model is outdated, the existing clustering results are replaced gradually by a new structure based on a new data set.

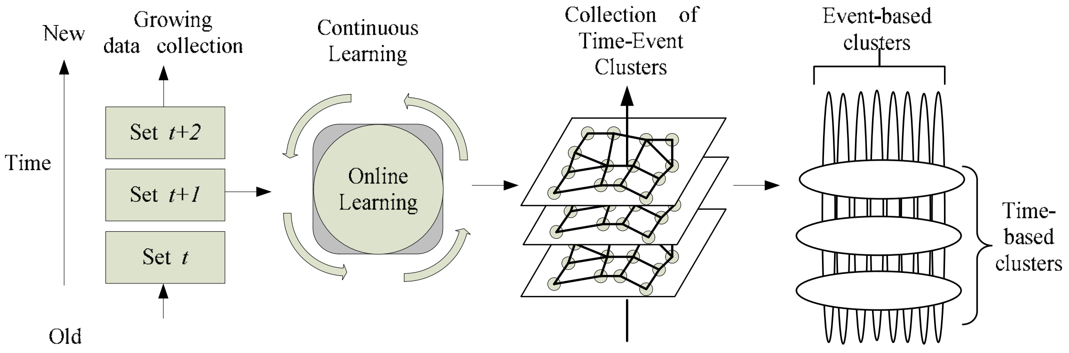


Figure 1: A conceptual architecture of time-event document clustering

Self-organising map

Inspired by the biological concept, in which neurons with similar functions are placed together, Kohonen proposed a self-organising map (SOM) using a time-based decaying learning rate and a pre-defined topological structure of units such that adjacent units contain similar weights so units self-organise into an ordered map (Kohonen, 1984). The SOM is a powerful algorithm for the visualisation of high-dimensional data. It is able to project the high-dimensional data onto a low-dimensional map, usually a two-dimensional grid of units. These geometric relationships between units in a grid represent the relationships between high-dimensional data. In other words, the SOM is able to abstract the most important data relationships for them to be visualised on a two-dimensional map. These features, i.e. visualisation and abstraction, present the SOM as a robust tool for many tasks such as document clustering, information visualisation, pattern recognition, data mining, image analysis and so on (e.g. Honkela et al., 1997; Kohonen, 2001).

The SOM model is usually designed for a static data collection due to its pre-defined topology and time-based decaying learning rate. It is trained by a static training set and tested by an unseen test set. Thus, the model can generalise well under the assumption that the unseen test set is similar to the training set. However, the real world information is continuously growing and often changes over time, which means that the boundary of the unseen test set is hard to be defined and therefore the unseen test set is usually different from the training set. Therefore, it is hard to presuppose the suitable learning length and inner structure of data in a non-stationary environment.

Related neural clustering models

Many neural clustering models have been proposed for a non-stationary clustering task. These models are focused on the ability of continuous learning in a non-stationary environment. For example, the growing cell structure (GCS) (Fritzke, 1994), growing neural gas (GNG) (Fritzke, 1995), incremental grid growing (IGG) (Blackmore and Miikkulainen, 1993), growing neural gas with utility criterion (GNG-U) (Fritzke, 1997) and grow when required (GWR) (Marsland et al., 2002), contain unit-growing and unit-pruning functions which are analogous to biological functions of remembering and forgetting in a non-stationary environment. These models also depend on the pre-definition of several thresholds which are used as guidance of neural behaviours for specific data sets. However, it is not trivial to determine those thresholds in a non-stationary environment. A set of better parameters often requires several iterations of trial and error or rules of thumb from experience (Hsu and Halgamuge, 2003). Even though a proper threshold has been found, this threshold may not be suitable for the future in a non-stationary environment. Therefore, it is not a good idea to use such a constant threshold for a big data set. Unfortunately, the GCS, GNG, IGG, GNG-U and GWR apply a constant threshold for detection of unsuitable units. We argue that a unit-pruning or connection-trimming threshold should be automatically adjusted to suit different data sets during training.

The proposed time-event document clustering model

By inspecting limitations of existing dynamic neural models, such as GNG, we propose the dynamic adaptive self-organising hybrid model (DASH). DASH follows the neural self-organising rule and can be treated as an extension of GNG in a non-stationary environment for time-event document clustering. In this paper, we only stress on the different features of DASH for conciseness. DASH adapts not only its main parameters (mainly including the connection trimming threshold and the growing frequency) but also its architecture to input samples. The DASH algorithm is divided into five main stages, which are initialisation, learning, growing, pruning and self-adjusting (Figure 2).

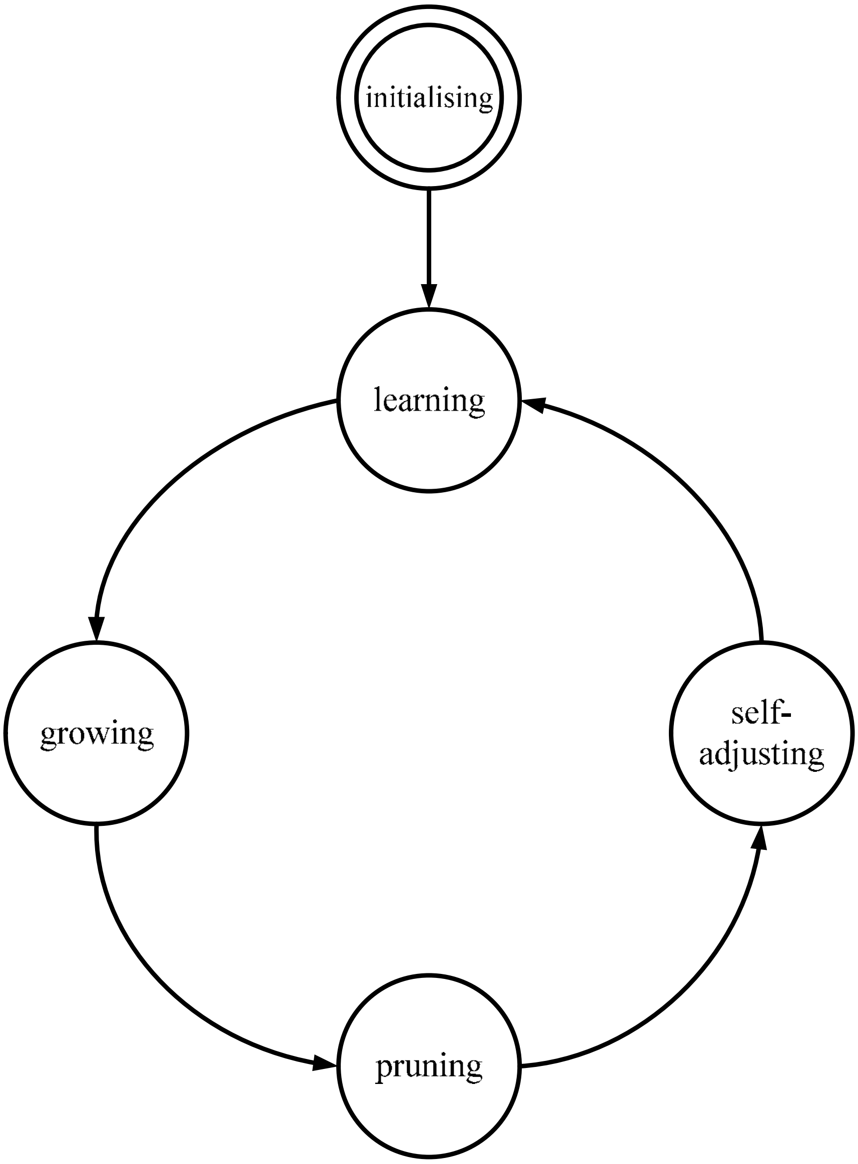


Figure 2: Five stages of the dynamic adaptive self-organising hybrid model

In the initialising stage, we define a map quality index, τ, which decides the objective average quantization error (AQE) for a child map. The AQE is the average of the Euclidean distance between every input vector and its best matching unit. Like GNG, we define an age threshold, β, for a connection between output units *i* and *j*. Unlike GNG, which uses a constant age threshold, the β of DASH cooperates with the current highest age of connection to decide whether a connection is too old.

In the learning and growing stages, like GNG, the DASH model starts with two units, uses fixed learning rates and applies the competitive Hebbian learning principle to connect the best matching unit and second best matching unit for an input stimulus (Martinetz, 1993). Unlike GNG, which increases the age only for connections of BMU's neighbours, DASH increases the age for all connections except the BMU. Furthermore, GNG grows every pre-defined constant cycle which is determined by trial and error. In contrast, this cycle is a part of the DASH model, which is mutually decided by the objective AQE and the number of input samples in the current map. The growing behaviour of the DASH model is illustrated in Figure. 3a-3h.

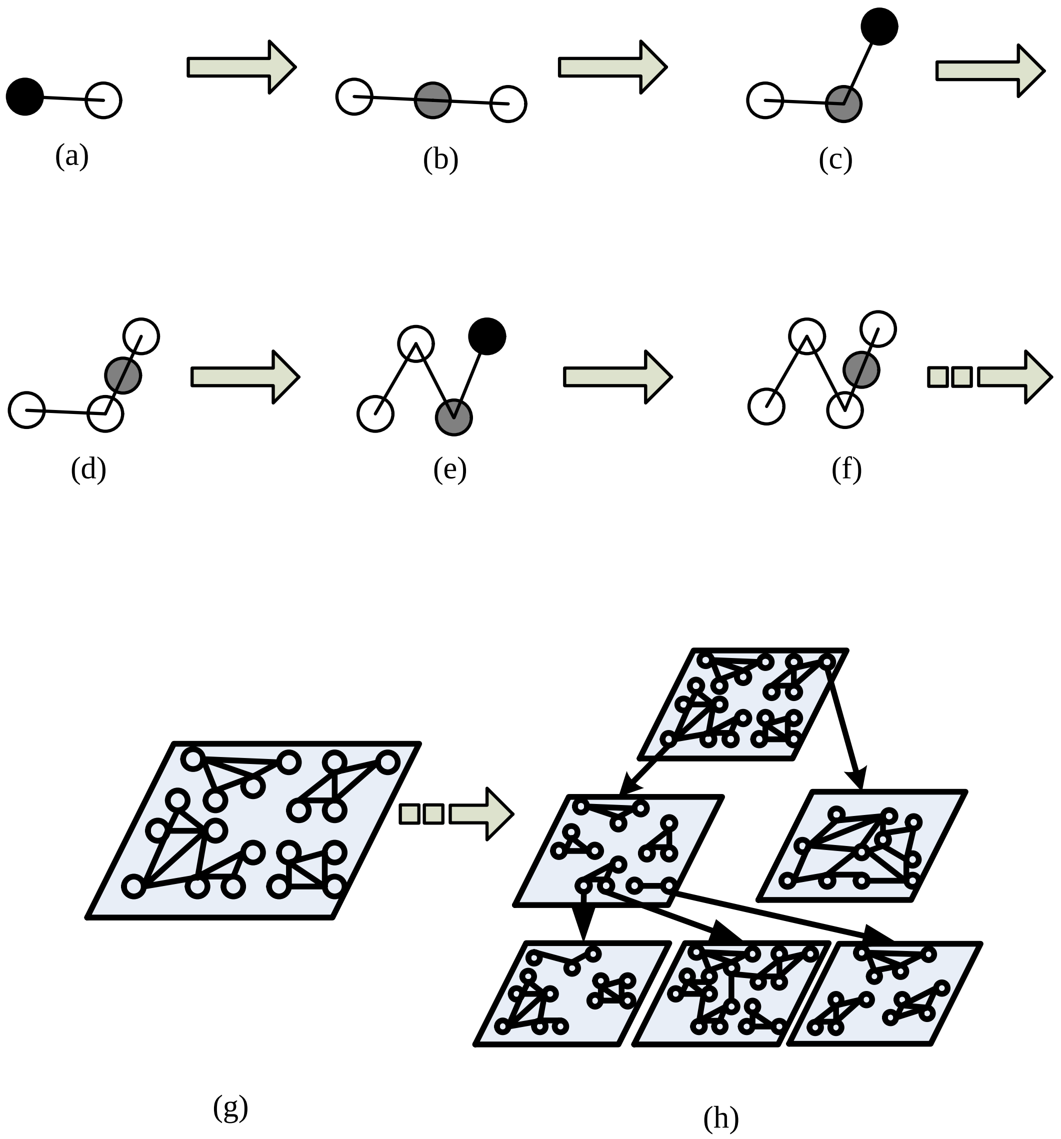


Figure 3: The growing processes for the DASH model. Units are represented as circles and lateral connections between units are represented as lines.

For a non-stationary data set, a trained unit or training unit should be updated by a unit which is trained with new input samples. This is performed by the unit-pruning or connection-trimming function in the pruning stage. A connection between output units *i* and *j* is trimmed if it is relatively old compared to other connections, i.e. $\frac{age_{ij}}{age_{\max}} > \beta$. This is a quasi-global connection-trimming function since the maximum age is got from connections built by a current data set in the model only. The connection-trimming function used by GNG is sub-optimal because a local age variable of a connection does not grow when units of this connection are not activated. That is, the aged connection may be kept forever so that the capability of self-adjustment for a model to new stimuli is diminished.

DASH uses a self-adjusted connection-trimming variable based on input vectors. Thus, in self-adjusting stage, this threshold is increased if units are not growing (Eq. 1) and is decreased if the number of units has reached the reference number of units in a map (Eq. 2). The reference number of units is a temporal maximum unit number for the current map and is also increased when this number is reached (Eq. 3).

$$\beta(t+1) = \beta(t) \times (2 - J_{\beta}), \qquad (1)$$

where *ß* is a connection age threshold, t indicates time, *J*$_{ß}$ is the *ß* adjusting parameter which is between 0.5 and 1.

$$\beta(t+1) = \beta(t) \times J_{\beta}. \qquad (2)$$

$$O_{I}(t+1) = O_{I}(t) \times (2 - J_{O}), \qquad (3)$$

where O_l is the reference number of units in a map and J_O is its adjusting parameter which is between 0.5 and 1.

Experiments on an artificial non-stationary data set

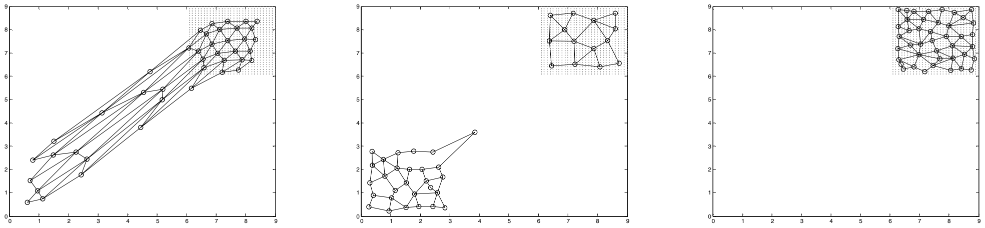
To demonstrate the ability of DASH in a non-stationary environment, we design a jumping-corner data set and compare with SOM (Kohonen, 1984) and GNG (Fritzke, 1995) because they are typical models in the static neural clustering group and dynamic neural clustering group respectively. In the beginning, an existing time-stamped data set contains 900 two dimensional vector inputs which are between (0, 0) and (3, 3) at intervals of 0.01 from the bottom left corner of a 9 by 9 grid. A new time-stamped data set in the top right corner substitutes for the existing data set in the bottom left corner at iteration 4 500.

We use the same parameters for all models if applicable. For example, the training length for all models is ten epochs. The learning rate for BMU and neighbours of BMU is 0.1 and 0.001 respectively for GNG and DASH as suggested by Ahrns et al. (1995). The learning rate for SOM is decayed from 0.1 to 0.001. The number of units is 91 for GNG and DASH. The number of units is 100 for the SOM because we use a square SOM topology.

All models learn well in the beginning because the data set is a uniform distribution. However, when the existing data set is replaced by the new data set, some units of the SOM cannot be re-trained since the learning rate is decayed. Thus, dead units, which represent no associated input samples, are inevitable for a SOM in a non-stationary environment (Figure 4a).

The GNG model traces the data set by modifying its neural topology. It forms two separate areas after the existing data set is replaced by the new data set. The topological structure keeps tracing the new data set in the top right corner but it is still in the bottom left corner. This is not because these output units in the bottom left corner still represent their associated input samples well but because none of these units in this area has been activated after some iterations. Thus, there are many dead units for a GNG in a non-stationary environment (Figure 4b).

Conversely, the DASH model removes unsuitable existing units while new input vectors occur and finally represents the new data set well without any dead unit (Figure 4c). This experimental result shows that our DASH outperforms a typical static neural clustering model, i.e, the SOM, and a typical dynamic neural clustering model, i.e., the GNG, in a non-stationary environment.



(a) SOM (b) GNG (c) DASH

Figure 4: Three convergence maps for SOM, GNG and DASH. The small dots are the input vectors and the circles are the output units.

Evaluation criteria

Even though it is possible to see clusters through the SOM-like maps, human qualitative judgements should not be the only evaluation criterion. Unlike qualitative assessment, quantitative criteria can be divided into two types: internal and external (Steinbach et al., 2000). The internal quantitative measure is data-driven and the average quantization error (AQE) is applied in this paper.

The quantization error is suggested by Kohonen as a measurement used in the vector quantization technique and is an indicator of the quality of the model (Kohonen, 2001). The AQE is defined as the average of the Euclidean distance between every input vector and its best matching unit (BMU). Given a data set X containing input vectors x_i, the AQE is described as Eq. 4.

$$AQE = \frac{1}{N}\sum_{i=1}^{N}\left\| x_i - w_i \right\|, \qquad (4)$$

where w_i is the weight vector of BMU for input sample i and N is the total number of input vectors.

The external quantitative measure evaluates how well the clustering model matches some prior knowledge which is usually provided by humans. The most common form of such external information is human manual classification knowledge, so classification accuracy (CA) is used in this paper. Kohonen et al. (2000) defined the classification error thus: "all documents that represented a minority newsgroup at any grid point were counted as classification errors … the node and the abstracts belonging to the other subsections were considered as misclassifications.” That is, each document has a pre-defined newsgroup label. After the training process, the category of a map unit is assigned according to the highest number of pre-defined labels of documents. Therefore, every unit represents its major article labels. The pre-defined label of each document which is mapped into this unit will be replaced by the unit label. Thus, if the unit label of each document matches its pre-defined label, it is a correct mapping. The classification accuracy is calculated from the number of correct mappings relative to the number of input articles.

Experimental design on new Reuters news corpus volume one

We work with the current version of the Reuters news corpus, RCV1 (Rose et al., 2002), and concentrate on the eight most dominant topics (Table 1) for our data sets. Since a news article can be pre-classified as more than one topic, we consider the multi-topic as a new combination of topics. Thus, the 8 chosen topics are expanded into 40 combined topics for the first 10 000 news articles (Table 2).

Table 1: The description of chosen topics and their distribution over the whole new Reuters corpus

Topic	Description	Distribution
c15	Performance	149 359
c151	Accounts/Earnings	81 201
c152	Comment/Forecasts	72 910
ccat	Corporate/Industrial	372 099
ecat	Economics	116 207
gcat	Government/Social	232 032
m14	Commodity markets	84 085
mcat	Markets	197 813

Table 2: The distribution of topic composition for the first 10 000 full-text news data set and the second 10 000 full-text news data set. The meanings of topics are described in Table 1.

The 10 000 full-text news data set			
No	Topic composition	Existing set	New set
1	ecat/mcat	155	104
2	ccat	1 780	2 033
3	c15/c151/ccat/ecat/gcat	6	2
4	c15/c151/ccat	999	916
5	m14/mcat	877	846
6	ecat	771	672
7	ccat/gcat	293	392
8	ccat/ecat/gcat	162	174
	……	……	……
39	c15/c151/ccat/gcat	1	3
40	c15/c152/ccat/ecat/mcat	1	0
Total number of news articles		10 000	10 000

Two data sets are used for the simulation of a non-stationary environment. The first one is to treat the first 10 000 full-text news articles as the existing data set and the second one is to treat the following 10 000 full-text news articles as the new data set. Besides using two data sets, three scenarios whose new data set is introduced at a different time are applied. The training length for all models is 42 000, 46 000 and 62 000 iterations for scenario 1, scenario 2 and scenario 3 respectively. The existing data set is used for all scenarios in the beginning and the new data set is introduced in scenario 1 at iteration 10 000, scenario 2 at iteration 30 000 and scenario 3 at iteration 50 000. In other words, there are 32 000 (42 000 – 10 000), 16 000 (46 000 – 30 000) and 12 000 (62 000 – 50 000) iterations to train models based on the new data set for scenarios 1, 2 and 3 respectively.

We use a traditional vector space model (VSM) such as TFxIDF to represent a full-text document as a numeric vector (Salton, 1989). We remove the stop words, allow only words shown in WordNet (Miller, 1985), which only contains open-classed words, i.e. nouns, verbs, adjectives and adverbs, and lemmatise each word to its base form. We further pick up the 1 000 most frequent words from the word master list to represent the original 15 760 words since this method is as good as some dimensionality reduction technique (Chakrabarti, 2000).

We compare the DASH model with SOM and GNG in a non-stationary environment. We use the same training length and the similar number of units of the DASH model for the SOM and GNG. The learning rate for BMU and neighbours of BMU is 0.1 and 0.001 respectively for GNG and DASH. The learning rate for SOM is decayed from 0.1 to 0.001. GNG makes use of two pre-defined parameters to control its topographic structure, i.e. the growing frequency and the connection-trimming threshold. Different values are used in different tasks in GNG by Fritzke (1995), because they are data-driven. In our experiments, the connection-trimming threshold for GNG for all scenarios is 62. Based on this threshold, the growing frequency cooperates with the controlled training length to produce the controlled number of units. In our experiments, the growing frequency threshold is 406, 407 and 520 for scenarios 1, 2 and 3 respectively. In the DASH model, the map quality index, τ , should be set before training. In our experiments, the value of τ , is 0.9 to control the shape of the DASH map. There are two main parameters which need to be set before training. The initial values of β and O_l are 0.95 and 100 respectively. However, these two parameters are self-adjusted and adapted to the current data set in the model.

Results on new Reuters news corpus volume one

We evaluate our model by AQE and classification accuracy, which have also been used in the work of Kohonen et al. (2000). AQE and classification accuracy for each scenario are shown in Tables 3 and 4. According to these results, the DASH outperforms other models with a higher classification accuracy and a lower AQE for all scenarios.

Table 3: A comparison of SOM, GNG and DASH evaluated by classification accuracy in a non-stationary environment

	Scenario 1	Scenario 2	Scenario 3
SOM	67.82%	65.75%	62.22%
GNG	66.16%	65.11%	64.70%
DASH	68.69%	69.63%	68.05%

Table 4: A comparison of SOM, GNG and DASH evaluated by AQE criterion in a non-stationary environment

	Scenario 1	Scenario 2	Scenario 3
SOM	0.937	0.948	0.956
GNG	0.869	0.867	0.870
DASH	0.812	0.818	0.815

The results can be further examined by analysing the variations of AQE based on time (Figure 5). In scenario 2, when the existing data set is replaced by the new data set at iteration 30 000, the AQE of all models are much higher. The SOM has a higher AQE, compared with the GNG and DASH models because a fixed topographic structure is unsuitable for the current data samples.

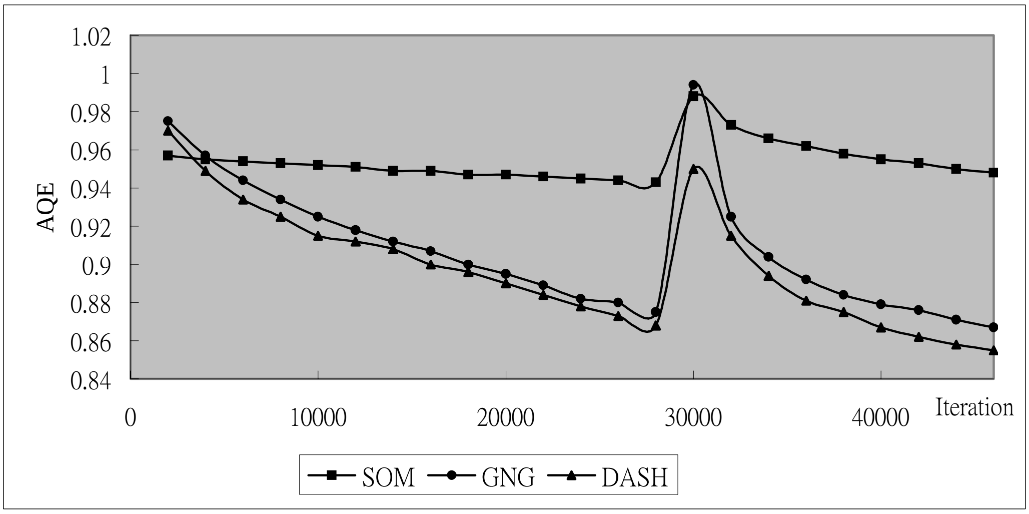
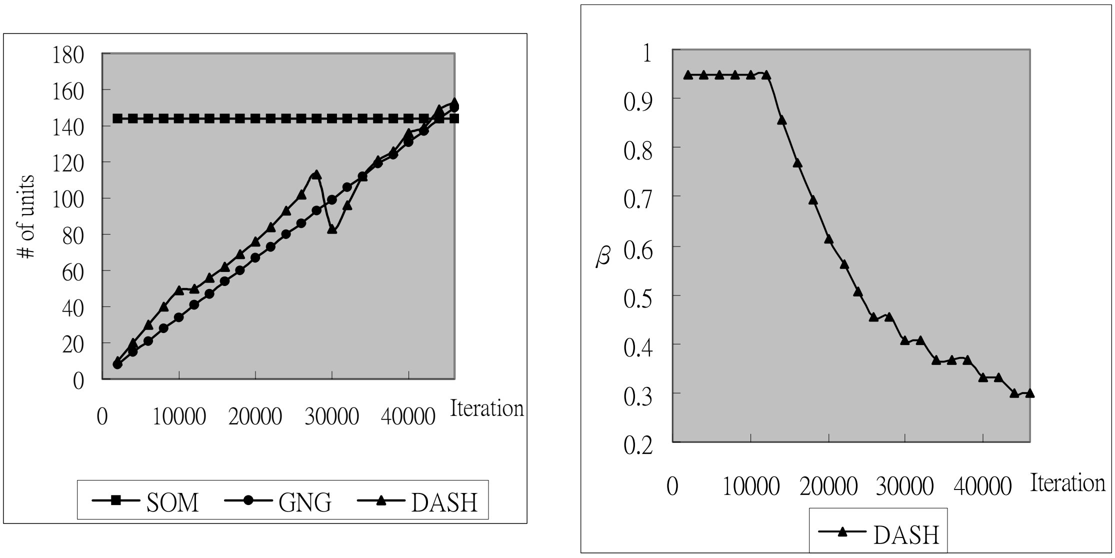


Figure 5: AQE for SOM, GNG and DASH in scenario 2.

The relationship of unit number to training length is shown in Figure 6a. The SOM uses a pre-defined structure with 144 units, while GNG and DASH have a dynamic structure. In general, the number of units for the GNG and DASH models is continuously growing. When the new data set is introduced at iteration 30 000, many unsuitable units are removed by the DASH model in a short period (Figure 6a). This is performed by the connection-trimming variable, i.e. β. It is adjusted automatically based on the current data set and the final value of β is about 0.3 in scenario 2 (Figure 6b). However, the removal of units is not evident for GNG (Figure 6a).

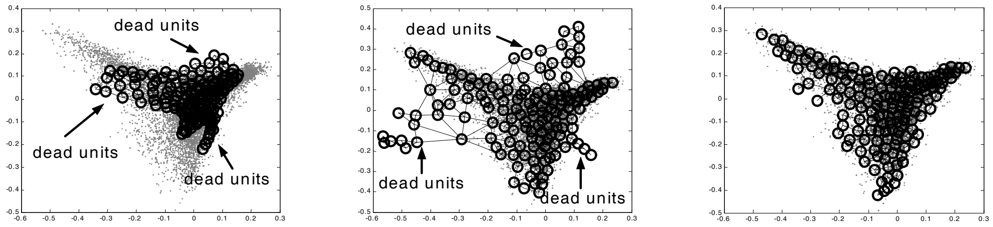


(a) (b)

Figure 6: (a) The units of SOM, GNG and DASH in scenario 2. (b) The β parameter of DASH in scenario 2.

SOM suffers from a non-stationary environment when the new data set is introduced at a later training stage. This is mainly because its time-based learning rate has decayed to a very small value. Another reason is that the SOM does not equip a unit-removing or connection-trimming function to remove unsuitable outdated units. In other words, the training length in scenarios 2 and 3 is not enough for the new data set for the SOM to keep the same performance in scenario 1.

GNG also suffers from a non-stationary environment due to its local connection-trimming function. It is very hard to pre-determine the connection-trimming threshold and growing frequency for GNG in a non-stationary environment. Conversely, the DASH model self-adjusts its parameters and performs better than SOM and GNG. Various initial settings of the connection trimming parameter, β, have been tried. DASH is more stable than GNG due to the function of self-adjustment in a non-stationary environment. We show their convergence maps in scenario 2 and the DASH model does not contain any dead unit on a map (Figure 7).



(a) SOM (b) GNG (c) DASH

Figure 7: Three convergence maps for SOM, GNG and DASH in scenario 2. The small dots are the input vectors and the circles are the output units.

Conclusions and future work

In the non-stationary environment, a clustering model runs continuously since the new document set is formed consecutively for training while the old document set is still at the training stage. Thus, output units of the map learned from the old data set are continuously adjusted to reflect the new data set. Based on the same or very similar resources (i.e. training length and the number of units), the DASH model outperforms SOM and GNG in a non-stationary environment by a greater classification accuracy and a lower average quantization error.

Even though the DASH model is designed for document clustering, it is appropriate for use in different applications, which may provide other interesting research directions. For example, one possible area is the novelty detector, which can detect surrounding stimuli, and is suitable for the characteristics of the DASH model. Image or multimedia content-based organisation is another area, in which DASH could be useful for content-based image or multimedia information retrieval. By combining text and multimedia organisation using recursive training, the DASH model could be an alternative approach to the building of a hierarchical digital library.

Acknowledgements:

This work was partially supported by the National Science Council of Taiwan No. NSC 93-2416-H-237-002.

Reference

- Ahrns, I., Bruske, J and Sommer, G. (1995), "On-line learning with dynamic cell structures", *Proceedings of ICANN-95, the International Conference on Artificial Neural Networks*, pp. 141-146.

- Blackmore, J. and Miikkulainen, R. (1993), "Incremental grid growing: encoding high-dimensional structure into a two-dimensional feature map", *Proceedings of the IEEE International Conference on Neural Networks (ICNN'93).*

- Chakrabarti, S. (2000), "Data mining for hypertext: a tutorial survey", *ACM Special*

Interest Group on Knowledge Discovery and Data Mining (SIGKDD) Explorations, Vol. 1 No. 2, pp. 1-11.

- Chang, C.-C. and Chen, R.-S. (2006), "Using data mining technology to solve classification problems- A case study of campus digital library", *The Electronic Library*, Vol. 24, No. 3, 2006, pp. 307-321.
- Chen, A.-P. and Chen, C.-C. (2006), "A new efficient approach for data clustering in electronic library using ant colony clustering algorithm", *The Electronic Library*, Vol. 24, No. 4, 2006, pp. 548-559.
- Fritzke, B. (1994), "Growing cell structures – a self-organizing network for unsupervised and supervised learning", *Neural Networks*, Vol. 7 No. 9, pp. 1441-1460.
- Fritzke, B. (1995), "A growing neural gas network learns topologies", in Tesauro, G., Touretzky, D.S. and Leen, T.K. (Eds.), *Advances in Neural Information Processing Systems 7*, MIT Press, Cambridge MA, pp. 625-632.
- Fritzke, B. (1997), "A self-organizing network that can follow non-stationary distributions", *Proceedings of ICANN-97, International Conference on Artificial Neural Networks*, pp. 613-618.

- Honkela, T., Kaski, S., Lagus, K. and Kohonen, T. (1997), “WEBSOM-self-organizing maps of document collections”, *Proceedings of Workshop on Self-Organizing Maps 1997 (WSOM'97)*, Espoo, Finland, pp. 310-135.
- Hsu, A.L. and Halgamuge, S.K. (2003). “Enhancement of topology preservation and hierarchical dynamic self-organising maps for data visualization”, *International Journal of Approximate Reasoning*, Vol. 32, No.2-3, 2003, pp. 259-279.
- Hung, C., Wermter, S, and Smith, P. (2004), “Hybrid neural document clustering using guided selforganisation and WordNet”, *IEEE-Intelligent Systems,* Vol. 19, No. 2, pp. 68-77.
- Jain, A.K., Murty, M.N. and Flynn, P.J. (1999), “Data clustering: a review”, *ACM Computing Surveys*, Vol. 31, No. 3, pp. 264-323.
- Kohonen, T. (1984), *Self-Organization and Associative Memory*, Springer-Verlag, Berlin.
- Kohonen, T. (2001), *Self-Organizing Maps*, Springer-Verlag, Berlin, Heidelberg, New York.
- Kohonen, T., Kaski, S., Lagus, K., Salojärvi, J., Honkela, J., Paatero, V. and Saarela, A. (2000), “Self organization of a massive document collection”, *IEEE Transactions on Neural Networks*, Vol. 11 No. 3, pp. 574-585.

- Marsland, S., Shapiro, J., and Nehmzow, U. (2002), “A self-organising network that grows when required”, *Neural Networks*, Vol. 15, pp. 1041-1058.
- Martinetz, T.M. (1993). “Competitive Hebbian learning rule forms perfectly topology preserving maps”, *Proceedings of ICANN-93, the International Conference on Artificial Neural Networks*, Amsterdam, pp. 427-434.
- Miller, G.A. (1985), “WordNet: a dictionary browser”, *Proceedings of the First International Conference on Information in Data.*
- Pullwitt, D. (2002), “Integrating contextual information to enhance SOM-based text document clustering”, *Neural Networks,* Vol. 15, pp.1099-1106.
- Rose, T., Stevenson, M. and Whitehead, M. (2002), “The Reuters corpus volume 1-from yesterday’s news to tomorrow’s language resources”, *Proceedings of the Third International Conference on Language Resources and Evaluation.*
- Salton, G. (1989), *Automatic Text Processing: the Transformation, Analysis, and Retrieval of Information by Computer*, Addison-Wesley.
- Smeaton, A.F., Burnett, M., Crimmins, F. and Quinn, G. (1998), “An architecture for efficient document clustering and retrieval on a dynamic collection of newspaper texts”, *Proceedings of the 20th BCS-IRSG Colloquium*, Springer-Verlag Workshops in Computing, Grenoble, France.

- Steinbach, M., Karypis, G. and Kumar, V. (2000), “A comparison of document clustering techniques”, *KDD Workshop on Text Mining*, http://citeseer.nj.nec.com/steinbach00comparison.html.

- Van Rijsbergen, C.J. (1979), *Information Retrieval*, Butterworths, London.

- Wermter, S. (2000), “Neural network agents for learning semantic text classification”, *Information Retrieval*, Vol. 3 No. 2, pp. 87-103.