Adaptive Document Maps

Krzysztof Ciesielski, Michał Dramiński, Mieczysław A. Kłopotek,
Dariusz Czerski, Sławomir T. Wierzchoń

Institute of Computer Science, Polish Academy of Sciences,
ul. Ordona 21, 01-237 Warszawa,Poland
kciesiel,dcz,mdramins,klopotek,stw@ipipan.waw.pl

Abstract. As document map creation algorithms like WebSOM are computationally expensive, and hardly reconstructible even from the same set of documents, new methodology is urgently needed to allow to construct document maps to handle streams of new documents entering document collection. This challenge is dealt with within this paper. In a multi-stage process, incrementality of a document map is warranted. [1]. The architecture of the experimental system allows for comparative evaluation of different constituent technologies for various stages of the process. The quality of the map generation process has been investigated based on a number of clustering and classification measures. Some conclusions concerning the impact of incremental, topic-sensitive approach on map quality are presented.

1 Introduction

Document maps become gradually more and more attractive as a way to visualize the contents of a large document collection.

The process of mapping a collection to a two-dimensional map is a complex one and involves a number of steps which may be carried out in multiple variants. In our search engine BEATCA [9–14], the mapping process consists of the following stages (see Figure 1): (1) document crawling (2) indexing (3) topic identification, (4) document grouping, (5) group-to-map transformation, (6) map region identification (7) group and region labeling (8) visualization. At each of theses stages various decisions can be made implying different views of the document map, generated by different algorithms.

For example, the indexing process involves dictionary optimization, which may reduce the documents collection dimensionality and restrict the subspace in which the original documents are placed. Topics identification establishes basic dimensions for the final map and may involve such techniques as singular value decomposition analysis (SVD [2]), fast Bayesian network learning (ETC [15]) and other. Document grouping may involve various variants of growing neural gas (GNG) techniques [6], hierarchical SOM [4] and Artificial Immune Systems [3]. The group-to-map transformation is run in BEATCA based on SOM ideas [16], but with variations concerning dynamic mixing of local and global search, based on diverse measures of local convergence [13,12]. The visualization of GNG and AIS models involves 2D and 3D variants [12].

With a strongly parameterized map creation process, the user of BEATCA can accommodate map generation to his particular needs, or even generate multiple maps covering various aspects of document collection.

The overall complexity of the map creation process, resulting in long run times, as well as the need to avoid "revolutionary" changes of the image of the whole document collection, require an incremental process of accommodation of new incoming documents into the collection.

Within the BEATCA project we have devoted much effort to enable such a gradual growth. In this study, we investigate vertical (emerging new topics) and horizontal (new documents on current topics) growth of document collection and its effects on the map formation capability of the system.

To ensure intrinsic incremental formation of the map, all the computation-intense stages involved in the process of map formation (crawling, indexing and all the stages of map formation: GNG-based document grouping, model visualization and map region identification) need to be reformulated in terms of incremental growth.

In particular, Bayesian Network driven crawler is capable of collecting documents around an increasing number of distinct topics. The crawler learning process runs in a kind of horizontal growth loop while it

[1] Research partially supported under KBN research grant 4 T11C 026 25 "Maps and intelligent navigation in WWW using Bayesian networks and artificial immune systems"

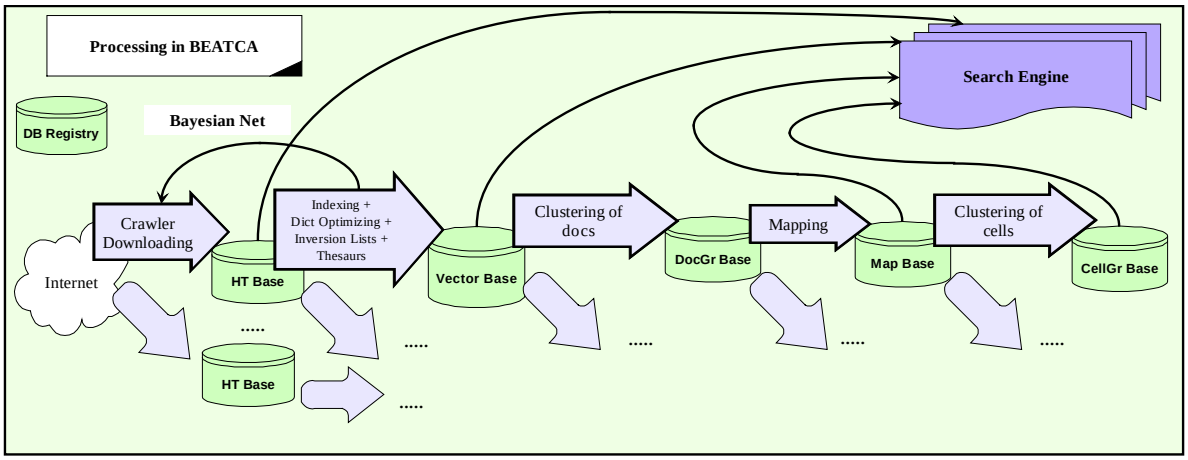

Fig. 1. BEATCA system architecture

keeps its performance with increasing number of documents collected. It may also grow vertically, as the user can add new topics of for searching.

The indexer has been constructed in order to achieve incremental growth and optimization of its dictionary with the growing collection of documents. Query extension capability of the query answering interface, based on Bayesian network and GNG derived dynamic automated thesaurus, accommodates also to the growing document collection. Though the actual clustering algorithms used in our system, like GNG, AIS or fuzzy C-means, are by their nature adaptive, nonetheless their tuning and modification was not a trivial task, especially with respect to our goal to achieve quality of incremental map comparable to the non-incremental one.

Special algorithms for thematic map initialization as well as for identification of document collection topics, based on GNG, SVD and/or Bayesian networks, lead to stabilization of the overall map. At the same time GNG detects the topic drift and so it may be appropriately visualized, due to plastic clustering approach, as new emerging map regions. It should be stressed at this point, that the map stabilization does not preclude obtaining different views of the same document collection. Our system permits to maintain several maps of the same document collection, obtained via different initializations of the map, and, what is more important, automatically tells the user which of the maps is most appropriate to view the results of his actual query.

To evaluate the effectiveness of the overall incremental map formation process, we compared it to the "from scratch" map formation in our experimental section 4. A brief discussion of related works is presented in section 5. The conclusions from our research work can be found in section 6.

2 Intelligent topic-sensitive crawling

In this section we briefly mention our efforts to create a crawler, that can collect documents from the internet devoted to a selected set of topics. The crawler learning process runs in a kind of horizontal growth loop while it improves its performance with increase of the amount of documents collected. It may also grow vertically, as the user can add new topics of for search during its run time.

The aim of intelligent crawling **??** is to crawl documents which belongs to certain topics and, obviously, to do it as efficient as it is possible. Often it is particularly useful not to download each possible document, but only that which concerns a certain subject. In our approach we use Bayesian nets (BN) and HAL algorithm to predict relevance of documents to be downloaded.

Topic-sensitive crawler begins processing from several initial links, specified by user. To describe a topic of our interest, we use query document. This special pseudo document contains descriptive terms with a priori given weights, which are later used to calculate priorities for crawled documents. During crawling first few hundred documents, crawler behavior depends only on initial query document.

2.1 Bayesian net document query expansion

To build Bayesian Net we use ETC learning algorithm [15]. In the subsequent cycles, a few BNs are built, each of those nets is assumed to be more accurate approximation of terms co-occurrence in the a priori specified topical areas. Subsequent BNs are constructed in the increasing time intervals, the number of documents between a subsequent nets is calculated as: $c * i^2$ where c is the initial number of documents and i is the net index.

After BN has been built, we use it to expand query pseudo-document and to calculate priorities for further documents links.

We expand pseudo document by adding parent and children nodes of BN terms, which are already present in query document. New terms get weights proportional to the product of the likelihood of their co-occurrence and the weight of the original term.

For each query term t_i we determine weights wz_{ij} for terms $t_j \in PC$, where PC is the set of of parent and children terms taken from BN model:

$$wz_{ij} = \frac{p_{ij}}{\sum_{k \in PC} p_{ik}} \cdot tqfidf_i \tag{1}$$

where $tqfidf_i$ is the product of query term frequency and inverse document frequency and p_{ij} is the probability of term i on the condition of occurrence of term j (taken from BN).

We can also have negative weights, to exclude some terms which are unlikely to appear, calculated on the basis of negative conditional probabilities in BN (probabilities of a term absence under condition of query term occurrence). Final document links priorities are calculated by modified cosine measure between new expanded query document and document containing those links:

$$cos(q, d) = \frac{\sum_{t \in q} wd_t \cdot wq_t}{\sqrt{(\sum_{t \in q} wq_t^2) \cdot (\sum_{t \in q} wd_t^2)}} \tag{2}$$

where wd_t is the weight of term t in document d, wq_t is the weight of term t in query q. It should be noted that all sums are restricted only to terms appearing in q.

2.2 HAL document query expansion

To expand query document we also use HAL model. HAL (Hyperspace Analogue To Language, [17]) is based on psychological theory claiming that meaning of a word is a function of contexts in which it appears; and the words sharing contexts have similar meanings. From computational perspective, HAL model can be represented as a matrix H in which cell h_{ij} corresponds to similarity measure of terms i and j. Briefly speaking, if $s = (t_1, ..., t_k)$ is a sentence (ordered list of terms), then h_{ij} is the sum (over all sentences in a collection of documents) of co-occurrences of terms i and j. The scoring value of co-occurence is defined as $max(0, K - p)$ where K is the predefined size of the scoring window and p is the number of terms which separate terms i and j in a given sentence. The main problem in this simple algorithm is obviously huge size of the matrix, which is equal to the number of distinct terms.

Similarly to the BN case, we build and iteratively update HAL term co-occurrence table. Next, we expand our pseudo document by adding k best co-occurrence terms $t_j \in W = t_1, ..., t_k$ for the terms already present in the query document. New terms get weights proportional to the product of their co-occurrence score and the weight of the original term:

$$wz_{ij} = \frac{h_{ij}}{\sum_{k \in W} h_{ik}} \cdot tqfidf_i \tag{3}$$

where $tqfidf_i$ is a product of query term frequency and inverse document frequency, h_{ij} is weight of term j taken from HAL table.

Like in Bayesian Net algorithm final document links priorities are calculated by modificationed cosine measure between new expanded query document and document containing those links.

2.3 Experiments

To evaluate effectiveness of presented topic-sensitive crawling, we conducted two experiments, one for Bayesian Net algorithm and second for HAL algorithm. In both cases, crawler starts from three seed links [`http://java.sun.com/j2ee/index.jsp`, `http://java.sun.com/products/ejb/`, `http://www.javaskyline.com/learning.html`]. Pseudo-document (a query) contains six descriptive terms with corresponding weights, treated as occurrence frequencies [java(20) documentation(30) ejb(100) application(50) server(50) J2EE(30)]. Figure 2(a) presents results for crawler based on Bayesian Net algorithm and figure 2(b) presents results for crawler based on HAL algorithm.

Quality measure is the average relevance measure, computed after each new 500 documents have been downloaded. Relevance is equal to modified cosine measure 2, but only for terms which are present in the initial user query ($Q = Q_0$).

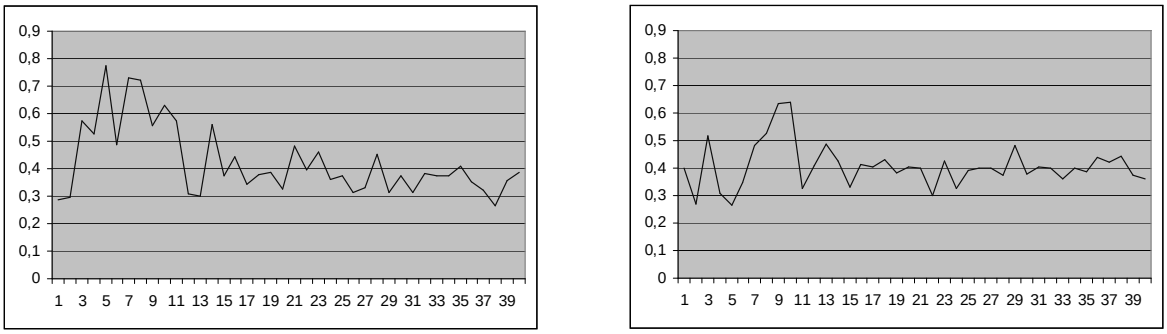

Fig. 2. Crawler evaluation (20000 documents downloaded): (a) Bayesian Net algorithm (b) HAL algorithm

Both methods gave similar results; average cosine measure was about 0.4. It appears to be satisfactory result, which shows that crawler did not lose a priori defined topic during the crawl. Bayesian Network proved to be faster of the two methods. However its disadvantage is the requirement to stop whole process in order to rebuild BN model. HAL table can be built during the crawl, but it requires more computations.

3 Topic-sensitive maps

Text documents are not uniformly distributed over the term-vector space. Characteristics of frequency distributions of a particular term depend strongly on document position in that space. In this paper we propose initial identification of groups containing similar documents as the preprocessing step in document maps formation. We argue that after splitting documents in such groups, term frequency distributions within group become much easier to analyse. In particular, it enables selection of significant and insignificant terms for efficient calculation of similarity measures during map formation step. Such document clusters we call *contextual* groups. For each contextual group, separate maps are generated. To obtain more informative maps there is a need to balance (during initial contextual clustering) size of each cluster. The number of documents presented on a map cannot be too high because the number of cells in the graph (and time required to create a map) would have to grow adequately. On the other hand, single map model should not hold only a few irrelevant documents.

Constraints on cluster size are obtained by recurrent divisions and merges of fuzzy document groups, created by a selected algorithm (e.g. EM combined with ETC or Chow-Liu Bayesian net, SVD, Fuzzy C-Means). In the case of ISODATA algorithm there is an additional modification in optimized quality criterion, that include a penalty factor for inbalanced (in sense of cluster size) splits.

In the first step, whole document set is splitted into a few (2-5) groups. Next, each of these groups is recursively divided until the number of documents inside a group meets required criteria. After such process we obtain hierarchy, represented by a tree of clusters. In the last phase, groups which are smaller than predefined constraint, are merged to the closest group. Similarity measure is defined as a single-linkage cosine angle between both clusters centroids.

Next phase of contextual document processing is the division of terms space (dictionary). In this case it is important to calculate fuzzy membership level, which will represent importance of a particular word or phrase in different contexts (and implicitly, ambiguity of its meaning). Estimation of fuzzy within-group membership of the term m_{tG} is estimated as:

$$m_{tG} = \frac{\sum_{d \in G} (f_{td} \cdot m_{dG})}{f_G \cdot \sum_{d \in G} m_{dG}} \tag{4}$$

where f_G is the number of documents in the cluster G, m_{dG} is the degree of document d membership level in group G, f_{td} is the number of occurrences of term t in document d.

Finally, vector-space representation of a document is modified to take into account document context. This representation increase weights of terms which are significant for a given contextual group and decrease weights of insignificant terms. In the boundary case, insignificant terms are ignored, what leads to the reduction of representation space dimension. To estimate the significance of term in a given context we applied following measure:

$$w_{tdG} = f_{td} \cdot m_{tG} \cdot log\left(\frac{f_G}{f_t \cdot m_{tG}}\right) \tag{5}$$

where f_{td} is the number of occurrences of term t in document d, m_{tG} is the degree of membership of term t in group G, f_G is the number of documents in group G, f_t is the number of documents containing term t.

Main idea behind the proposed approach is to replace a single model (growing neural gas, immunological net or hierarchical SOM maps) by a set of independently created contextual models. Training data for each model is a single contextual group. Each document is represented as a standard referential vector in term-document space. However, TFxIDF measure of vector components is replaced by w_{tdG}.

To represent visually similarity relation between contexts (represented by a set of contextual models), additional "global" map is required. Such model becomes a root of contextual maps hierarchy. Main map is created in a manner similar to previously created maps, with one distinction: an example in training data is a weighted centroid of referential vectors of the corresponding contextual model:

$$\overrightarrow{x_i} = \sum_{c \in M_i} (d_c \cdot \overrightarrow{v_c}) \tag{6}$$

Finally, cells and regions on the main map are labeled with keywords selected by the following contextual term quality measure:

$$Q_{tG} = ln(1 + f_{tG}) \cdot (1 - |EN_{tG} - 0.5|) \tag{7}$$

where EN_{tG} denotes normalized entropy of term frequency within the group.

Learning process of the contextual model is to some extent similar to the classic, non-contextual learning. However, it should be noted that each model can be processed independently, in particular it can be distributed and calculated in parallel. Also the incremental update of such models appears to be much easier to perform, both in terms of model quality, stability and time complexity. In the next section we present experimental results to support presented approach.

4 Experiments

To evaluate the effectiveness of the proposed map formation approach, we compared it to the "from scratch" map generation process. In this section we describe the overall experimental design, quality measures used and the results obtained.

The architecture of our system supports comparative studies of clustering methods at the various stages of the process (i.e. initial document grouping, broad topics identification, incremental clustering, model projection and visualization, identification of thematic areas on the map and its labeling). In particular, we conducted series of experiments to compare the quality and stability of GNG and SOM models for various model initialization methods, winner search methods and learning parameters [13]. In this paper we focus only on evaluation of the GNG winner search method and the quality of the resulting incremental clustering model with respect to the topic-sensitive learning approach.

4.1 Quality Measures for the Document Maps

Various measures of quality have been developed in the literature, covering diverse aspects of the clustering process (e.g. [21]). The clustering process is frequently referred as "learning without a teacher", or "unsupervised learning", and is driven by some kind of similarity measure. The term "unsupervised" is not completely reflecting the real nature of learning. In fact, the similarity measure used is not something "natural", but rather it reflects the intentions of the teacher. So we can say that clustering is a learning process with hidden learning criterion. The criterion is intended to reflect some esthetic preferences, like: uniform split into groups (topological continuity) or appropriate split of documents with known a priori categorization. As the criterion is somehow hidden, we need tests if the clustering process really fits the expectations. In particular, we have accommodated for our purposes and investigated the following well known quality measures of clustering:

- **Average Map Quantization**: the average cosine distance between each pair of adjacent nodes. The goal is to measure topological continuity of the model (the lower this value is, the more "smooth" model is):

$$AvgMapQ = \frac{1}{|N|} \sum_{n \in N} \left(\frac{1}{|E(n)|} \sum_{m \in E(n)} c(n,m) \right)$$

  where N is the set of graph nodes, $E(n)$ is the set of nodes adjacent to the node n and $c(n,m)$ is the cosine distance between nodes n and m.

- **Average Document Quantization**: average distance (according to cosine measure) for the learning set between the document and the node it was classified into. The goal is to measure the quality of clustering at the level of a single node:

$$AvgDocQ = \frac{1}{|N|} \sum_{n \in N} \left(\frac{1}{|D(n)|} \sum_{d \in D(n)} c(d,n) \right)$$

  where $D(n)$ is the set of documents assigned to the node n.

Both measures have values in the [0,1] interval, the lower values corresponds respectively to more "smooth" inter-cluster transitions and more "compact" clusters. To some extent, optimization of one of the measures entails increase of the other one. Still, experiments [13] show that the GNG models are much more smooth than SOM maps while the clusters are of similar quality.

The two subsequent measures evaluate the agreement between the clustering and the a priori categorization of documents (i.e. particular newsgroup in case of newsgroups messages).

- **Average Weighted Cluster Purity**: average "category purity" of a node (node weight is equal to its density, i.e. the number of assigned documents):

$$AvgPurity = \frac{1}{|D|} \sum_{n \in N} max_c \left(|D_c(n)| \right)$$

  where D is the set of all documents in the corpus and $D_c(n)$ is the set of documents from category c assigned to the node n.

- **Normalized Mutual Information**: the quotient of the total category and the total cluster entropy to the square root of the product of category and cluster entropies for individual clusters:

$$NMI = \frac{\sum_{n \in N} \sum_{c \in C} |D_c(n)| \ log\left(\frac{|D_c(n)| \ |D|}{|D(n)| \ |D_c|}\right)}{\sqrt{\left(\sum_{n \in N} |D(n)| \ log\left(\frac{|D(n)|}{|D|}\right)\right)\left(\sum_{c \in C} |D_c| \ log\left(\frac{|D_c|}{|D|}\right)\right)}}$$

where N is the set of graph nodes, D is the set of all documents in the corpus, $D(n)$ is the set of documents assigned to the node n, D_c is the set of all documents from category c and $D_c(n)$ is the set of documents from category c assigned to the node n.

Again, both measures have values in the [0,1] interval. Roughly speaking, the higher the value is, the better agreement between clusters and a priori categories. At the moment, we are working on the extension of the above-mentioned measures to ones covering all aspects of the map-based model quality, i.e. similarities and interconnections between thematic groups both in the original document space and in the toroid map surface space.

4.2 Experimental results

Model evaluation were executed on 2054 of documents downloaded from 5 newsgroups with quite well separated main topics (antiques, computers, hockey, medicine and religion). Each GNG network has been trained for 100 iterations, with the same set of learning parameters, using previously described winner search methods.

In the main case (depicted with the black line), network has been trained on the whole set of documents. This case was the reference one for the quality measures of adaptation as well as comparison of the winner search methods.

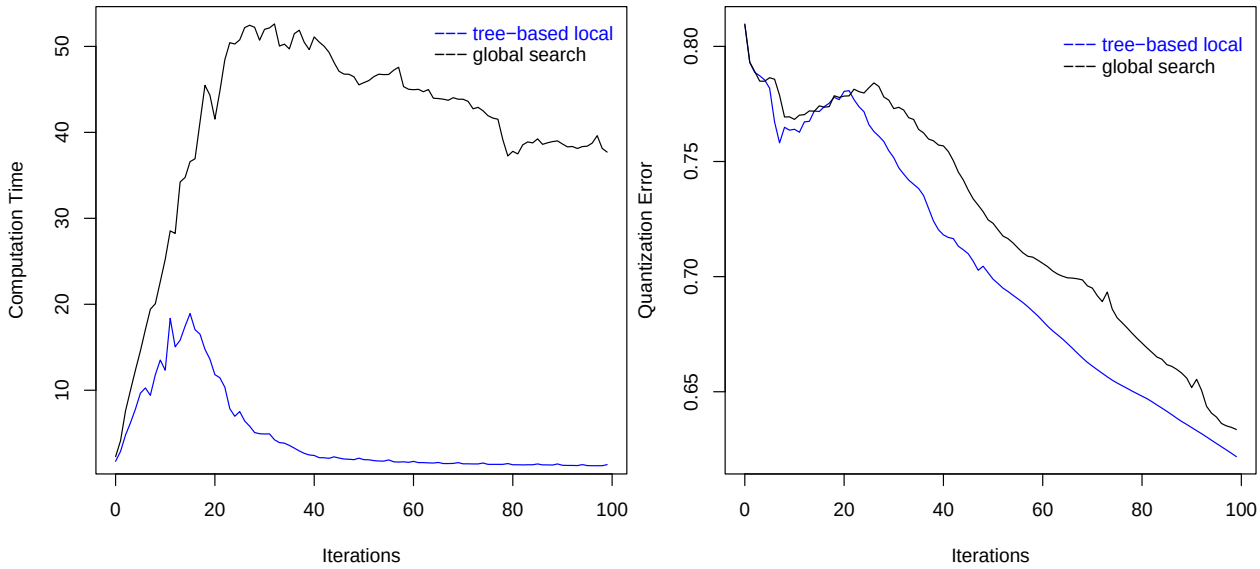

Fig. 3. Winner search methods (a) computation time (b) model quality

Figure 3 presents comparison of a standard global winner search method with our own CF-tree based approach. Local search method is not taken into consideration since, as it has already been mentioned, it is completely inappropriate in case of unconnected graphs. Obviously, tree-based local method is invincible in terms of computation time. The main drawback of the global method is that it is not scalable and depends on the total number of nodes in the GNG model.

At first, the result of the quality comparison appeared to be quite surprising. On one hand, the quality was similar, on the other - global search appeared to be worse of the two (!). We have investigated it further and it turned out to be the aftermath of process divergence during the early iterations of the training process. It will be explained on the next example.

In the next experiment, in addition to the main reference case, we had another two cases. During the first 30 iterations network has been trained on 700 documents only. In one of the cases (represented by red line) documents were sampled uniformly from all five groups and in the 33^{rd} iteration another 700 uniformly sampled were introduced to training. After the 66^{th} iteration the model has been trained on the whole dataset.

In the last case (blue line) initial 700 documents were selected only from two groups. After the 33^{rd} iteration of training, documents from the remaining newsgroups were gradually introduced in the order of their newsgroup membership. It should be noted here that in this case we had an a priori information on the document category (i.e. particular newsgroup). In the general case, we are collecting fuzzy category membership information from Bayesian Net model.

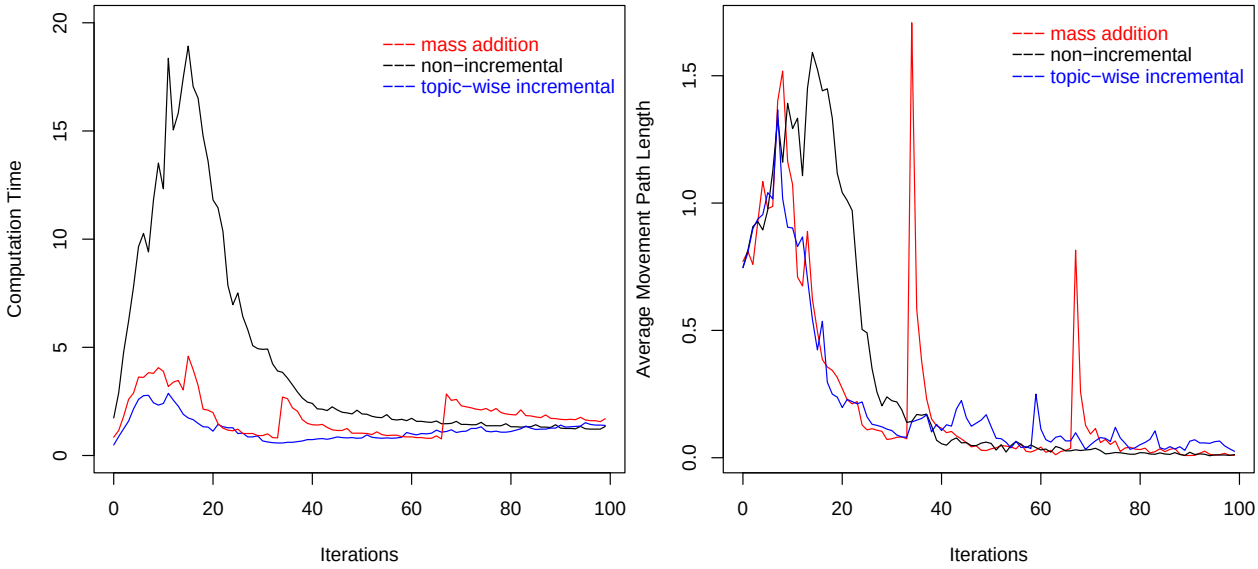

Fig. 4. Computation complexity (a) execution time of a single iteration (b) average path length of a document

As expected, in all cases GNG model adapts quite well to the topic drift. In the non-incremental and the topic-wise incremental case, the quality of the models were comparable, in terms of Average Document Quantization measure (see figure 5(a)), Average Weighted Cluster Purity, Average Cluster Entropy and Normalized Mutual Information (for the final values see table 1). Also the subjective criteria such as visualizations of both models and the identification of thematic areas on the SOM projection map were similar.

	Cluster Purity	*Cluster Entropy*	*NMI*
non-incremental	0.91387	0.00116	0.60560
topic-wise incremental	0.91825	0.00111	0.61336
massive addition	0.85596	0.00186	0.55306

Table 1. Final values of model quality measures

The results were noticeably worse for the massive addition of documents, even though all covered topics were present in the training from the very beginning and should have occupied specialized thematic areas in the model graph. However, and it can be noticed on the same plot, a complex mixture of topics can pose a serious drawback, especially in the first training iterations. In the non-incremental, reference case, the attempt to cover all topics at once leads learning process to a local minimum and to subsequent divergence (what, moreover, is quite time-consuming as one can notice on figure 4(a)). As we have previously noticed, the problem of convergence to a local minimum were even more influential in the case of global winner search (figure 3(b)).

However, when we take advantage of the incremental approach, the model ability to separate document categories is comparable for global search and CF-tree based search (Cluster Purity: 0.92232 versus 0.91825, Normalized Mutual Information: 0.61923 versus 0.61336, Average Document Quantization: 0.64012 versus 0.64211).

The figure 4(b) presents average number of GNG graph edges traversed by a document during a single training iteration. It can be seen that a massive addition causes temporal instability of the model. Also, the above mentioned attempts to cover all topics at once in case of a global model caused much slower stabilization of the model and extremely high complexity of computations (figure 4(a)). The last reason for such slow computations is the representation of the GNG model nodes. The referential vector in such node is represented as a balanced red-black tree of term weights. If a single node tries to occupy too big portion of a document-term space, too many terms appear in such tree and it becomes less sparse and - simply - bigger. On the other hand, better separation of terms which are likely to appear in various newsgroups and increasing "crispness" of thematic areas during model training leads to highly efficient computations and better models, both in terms of previously mentioned measures and subjective human reception of the results of search queries.

The last figure, 5(b), compares the change in the value of Average Map Quantization measure, reflecting "smoothness" of the model (i.e. continuous shift between related topics). In all three cases the results are almost identical. It should be noted that extremely low initial value of the Average Map Quantization is the result of the model initialization via broad topics method [9], shortly described in the section **??**.

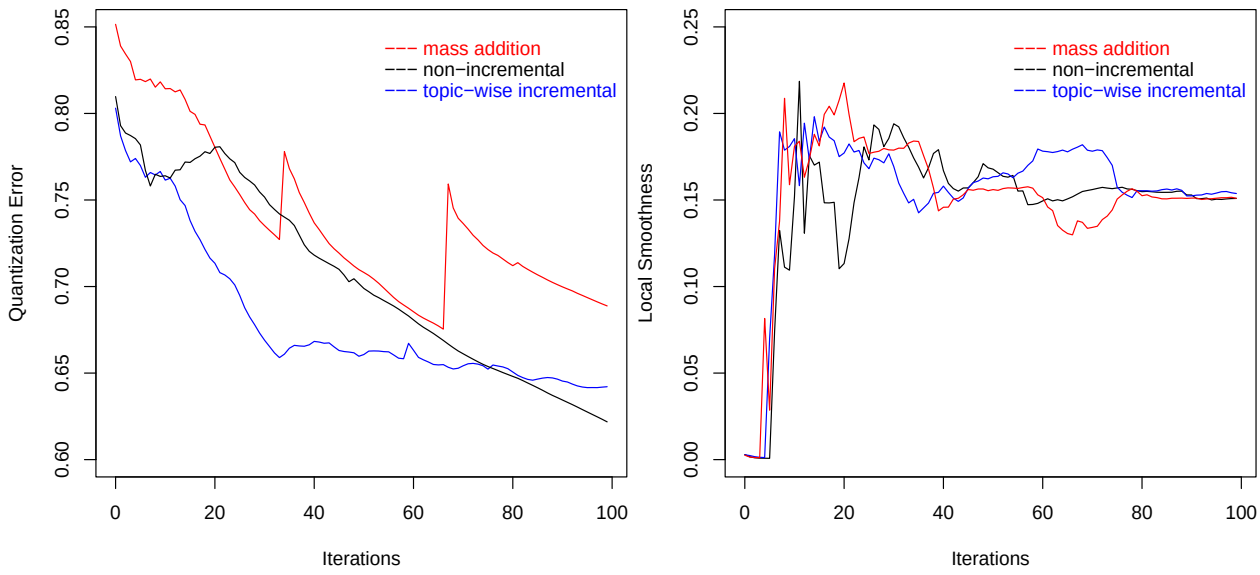

Fig. 5. Model quality (a) Average Document Quantization (b) Average Map Quantization

5 Related works

Modern man faces a rapid growth in the amount of written information. Therefore he needs a means of reducing the flow of information by concentrating on major topics in the document flow. Grouping documents based on similar contents may be helpful in this context as it provides the user with meaningful classes or clusters. Document clustering and classification techniques help significantly in organizing documents in this way. A prominent position among these techniques is taken by the WebSOM (Self Organizing Maps) of Kohonnen and co-workers [16]. However, the overwhelming majority of the existing document clustering and classification approaches rely on the assumption that the particular structure of the currently available static document collection will not change in the future. This seems to be highly unrealistic, because both the interests of the information consumer and of the information producers change over time.

A recent study described in [8] demonstrated deficiencies of various approaches to document organization under non-stationary environment conditions of growing document quantity. The mentioned paper pointed to weaknesses among others of the original SOM approach (which itself is adaptive to some extent) and proposed a novel dynamic self-organizing neural model, so-called Dynamic Adaptive Self-Organising Hybrid (DASH) model. This model is based on an adaptive hierarchical document organization, supported by human-created concept-organization hints available in terms of WordNet.

Other strategies like that of [18,4], attempt to capture the move of topics, enlarge dynamically the document map (by adding new cells, not necessarily on a rectangle map).

We take a different perspective in this paper claiming that the adaptive and incremental nature of a document-map-based search engine cannot be confined to the map creation stage alone and in fact engages all the preceding stages of the whole document analysis process.

6 Concluding remarks

As indicated e.g. in [8], most document clustering methods, including the original WebSOM, suffer from their inability to accommodate streams of new documents, especially such in which a drift, or even radical change of topic occurs.

Though one could imagine that such an accommodation could be achieved by "brute force" (learning from scratch whenever new documents arrive), but there exists a fundamental technical obstacle for a procedure like that: the processing time. But the problem is deeper and has a "second bottom": the clustering methods like those of WebSOM contain elements of randomness so that even re-clustering of the same document collection may lead to a change of the view of the documents. The results of this research are concerned with both aspects of adaptive clustering of documents.

The important contribution of this paper is to demonstrate, that the whole incremental machinery not only works, but it works efficiently, both in terms of computation time, model quality and usability. For the quality measures we investigated, we found that our incremental architecture compares well to non-incremental map learning both under scenario of "massive addition" of new documents (many new documents, not varying in their thematic structure, presented in large portions) and of scenario of "topic-wise-increment" of the collection (small document groups added, but with new emerging topics). The latter seemed to be the most tough learning process for incremental learning, but apparently the GNG application prior to WebSOM allowed for cleaner separation of new topics from ones already discovered, so that the quality (e.g. in terms of cluster purity and entropy) was higher under incremental learning than under non-incremental learning.

The experimental results indicate, that the real hard task for an incremental map creation process is a learning scenario where the documents with new thematic elements are presented in large portions. But also in this case the results proved to be satisfactory.

A separate issue is the learning speed in the context of crisp and fuzzy learning models. Apparently, separable and thematically "clean" models allow for faster learning as the referential vectors in the model nodes are smaller (contain fewer non-zero components).

From the point of view of incremental learning under soft-competitive scenario, a crucial factor for the processing time is the winner search method for assignment of documents to neurons. We were capable to

elaborate a very effective method of stable, context-dependent winner search which does not deteriorate the overall quality of the final map [12]. At the same time, it comes close to the speed of local search and is not directly dependent on the size of the model.

References

1. CC Aggarwal, F Al-Garawi, and PS Yu. Intelligent crawling on the World Wide Web with arbitrary predicates. In Proc. 10th International World Wide Web Conference, pages 96–105, 2001.
2. M.W.Berry, Large scale singular value decompositions, International Journal of Supercomputer Applications, 6(1), 1992, pp.13-49
3. L.N. De Castro, F.J. von Zuben, An evolutionary immune network for data clustering, SBRN'2000, IEEE Computer Society Press, 2000
4. M. Dittenbach, A. Rauber, D. Merkl, Discovering Hierarchical Structure in Data Using the Growing Hierarchical Self-Organizing Map, Neurocomputing, Elsevier, ISSN 0925-2312, 48 (1-4)2002, pp. 199-216
5. B.Fritzke, Some competitive learning methods, draft available from http://www.neuroinformatik.ruhr-uni-bochum.de/ini/VDM/research/gsn/JavaPaper
6. B.Fritzke, A growing neural gas network learns topologies, in: G. Tesauro, D.S. Touretzky, and T.K. Leen (Eds.) Advances in Neural Information Processing Systems 7, MIT Press Cambridge, MA, 1995, pp. 625-632
7. B.Fritzke, A self-organizing network that can follow non-stationary distributions, in: Proceedings of the International Conference on Artificial Neural Networks '97, Springer, 1997, pp.613-618
8. C.Hung, S.Wermter, A Constructive and Hierarchical Self-Organising Model in A Non-Stationary Environment, International Joint Conference in Neural Networks, 2005
9. M.Kłopotek, M.Dramiński, K.Ciesielski, M.Kujawiak, S.T.Wierzchoń, Mining document maps, in Proceedings of Statistical Approaches to Web Mining Workshop (SAWM) at PKDD'04, M. Gori, M. Celi, M. Nanni eds., Pisa, 2004, pp.87-98
10. K. Ciesielski, M. Dramiński, M. Kłopotek, M. Kujawiak, S. Wierzchoń: Architecture for graphical maps of Web contents, in Proceedings of WISIS'2004, Warsaw, 2004
11. K. Ciesielski, M. Dramiński, M. Kłopotek, M. Kujawiak, S. Wierzchoń: Mapping document collections in non-standard geometries. B. De Beats, R. De Caluwe, G. de Tre, J. Fodor, J. Kacprzyk, S. Zadrożny (eds): Current Issues in Data and Knowledge Engineering. Akademicka Oficyna Wydawnicza EXIT Publishing, Warszawa, 2004, pp.122-132
12. K.Ciesielski, M.Dramiński, M.Kłopotek, M.Kujawiak, S.T.Wierzchoń, On some clustering algorithms for Document Maps Creation, to appear in: Proceedings of the Intelligent Information Processing and Web Mining (IIS:IIPWM-2005), Gdansk, 2005
13. M. Kłopotek, S. Wierzchoń, K. Ciesielski, M. Dramiński, D. Czerski, M. Kujawiak: Understanding Nature of Map Representation of Document Collections Map Quality Measurements, to appear in Proceeding of International Conference on Artificial Intelligence, Siedlce, September 2005
14. K.Ciesielski, M.Dramiński, M.Kłopotek, M.Kujawiak, S.T.Wierzchoń, Crisp versus Fuzzy Concept Boundaries in Document Maps, to appear in: Proceedings of DMIN-05 Workshop at The 2005 World Congress in Applied Computing, Las Vegas, 2005
15. M.Kłopotek: A New Bayesian Tree Learning Method with Reduced Time and Space Complexity, Fundamenta Informaticae, 49(no 4)2002, IOS Press, pp. 349-367
16. T. Kohonen, Self-Organizing Maps. Springer Series in Information Sciences, Vol. 30, Springer, Berlin, Heidelberg, New York, 2001. Third Extended Edition, 501 pages. ISBN 3-540-67921-9, ISSN 0720-678X
17. B. C., K. Livesay, and K. Lund. Explorations in context space: Words, sentences, discourse. Discourse Processes, 25(2-3):211–257, 1998
18. A.Rauber, Cluster Visualization in Unsupervised Neural Networks, Diplomarbeit, Technische Universitt Wien, Austria, 1996
19. J.Timmis, aiVIS: Artificial Immune Network Visualization, in: Proceedings of EuroGraphics UK 2001 Conference, Univeristy College London 2001, pp.61-69
20. T.Zhang, R.Ramakrishan, M.Livny, BIRCH: Efficient Data Clustering Method for Large Databases, in: Proceedings of ACM SIGMOD International Conference on Data Management, 1997
21. Y. Zhao, G. Karypis, Criterion functions for document Clustering: Experiments and analysis, available at URL: http://www-users.cs.umn.edu/ karypis/publications/ir.html