Supporting Information Management in Digital Libraries with Map-based Interfaces

Rudolf Mayer, Angela Roiger and Andreas Rauber
Institute of Software Technology and Interactive Systems
Vienna University of Technology, Vienna, Austria
Email: mayer@ifs.tuwien.ac.at, angela@roiger.at, rauber@ifs.tuwien.ac.at

***Abstract*—The *Self-Organising Map* (SOM) has been proposed as an alternative interface for exploring Digital Libraries (DL), in addition to conventional search and browsing. With advanced visualisations assisting the user in understanding the contents of the map and its structure, as well as advanced interaction modes as zooming, panning and area selection, the SOM becomes a feasible alternative to classical search and browse interfaces. Several applications show the SOMs utility for this task. However, there are still shortcomings in helping the user understanding the map – there are insufficient methods developed for describing the map to support the user in the analysis of the map contents. In this paper, we give an overview of existing techniques and applications of SOMs in Digital Libraries, and present recent work in assisting the user in exploring the map by automatically describing maps using advanced labelling and summarisation of map regions. Therewith, the SOM becomes an attractive tool for Information Management.**

I. Introduction

The Self-Organising Map (SOM) [1] is a popular unsupervised neural network model that provides a mapping from a high-dimensional input space (for example text documents described in a vector space model) to a low, often two-dimensional, output space. The mapping of the SOM is topology preserving – elements close in the input space will also be close in the output space. Due to its interesting properties, the SOM has been used in several applications to automatically organise documents in a Digital Library by their content. Examples are text, as in the SOMLib Digital Library system [2], or in a map of news texts [3], or music in the SOMeJB system [4]. As a recent example also the Digital Library Management System (DLMS) developed by the DELOS Network of Excellence [5] reckons the possibilities of using the SOM as an interface to a Digital Library's content, as it offers the user support in analysing and exploring the content. With advanced visualisations and interaction possibilities, the user can exploit the full potential of the SOM. However, we still lack techniques to adequately help the user in analysing the contents of the map. For large maps, containing several tens of thousands documents describing various different topics, it becomes increasingly difficult to quickly analyse the map.

In this paper, we give an overview of existing applications of the Self-Organising Map in Digital Libraries and techniques to explore and interact with the map. Furthermore, we present recent work in making the SOM more usable for Information Management by automatically describing regions in the map through adding semantic labels to the SOM, using clustering methods to identify topical areas and selecting representative labels for those regions. Moreover, we present work on automatically summarising the content of those regions on the SOM.

The remainder of this paper is organised as follows: Section II gives a brief overview of the Self-Organising Map and its application in the context of Digital Libraries. Section III describes our work in labelling and summarising regions, while Section IV presents the experiments conducted. Section V presents conclusions and future work.

II. Self-Organising Map

The *Self-Organising Map* (SOM) has been successfully used for clustering various kinds of data. It provides a mapping from a high-dimensional input space to a lower-dimensional output space. In the context of Digital Libraries, the input space is mostly a vector-space model representation of the documents the Digital Library holds, may it be text, images, audio, video, or any other media that can be represented in vectorial form. Although many different architectures exist, the output space is in many applications organised as a two-dimensional rectangular grid of units, a representation that is easily understandable for users due to its analogy to 2-D maps. Each of the units on the map is assigned a *weight vector*, which is of the same dimensionality as the vectors in the input space. During the training process, vectors from the input space are presented to the Self-Organising Map, and the unit with the most similar weight vector to this input vector is determined. The weight vector of this unit, and, to a lesser extent, of the neighbouring units, are adapted towards the input vector, i.e. their distance in the input space is reduced. At the end of the training process, the output space will be arranged in a way to represent the input space as closely as possible. For more details on the SOM training process, please refer e.g. to [1].

An important property of the SOM is that the mapping is topology preserving – elements which are located close to each other in the input space will also be closely located in the output space, while dissimilar patterns will be mapped on opposite regions of the map. The SOM therefore provides a sort of clustering of the data, however, without explicitly

assigning data items to clusters. It neither identifies cluster boundaries as opposed to, e.g, the k-Means method. The generated map can help the user in getting a quick overview of the patterns in the input space. With fitting visualisations highlighting boundaries, it also allows an easier interpretation of the cluster structures and correlations in the content.

A. Self-Organising Maps in Digital Libraries

The Self-Organising Map as an interface to digital document collections has already been proposed in the WEBSOM project [6], where the contents of a newsgroup collection containing a million of articles was clustered on the map. The application provides the user with a map of the document collection which she can zoom into by pre-defined levels and navigate in. On the most detailed zooming level, a list of the documents mapped onto that region of the map is provided. The map itself, and the exact position of the documents on the map, is not available anymore. To add semantic meaning, the map gets automatically labelled by the names of the most dominant corresponding newsgroups, which is a feasible approach when some kind of categorisation is available for the documents. The labels are however not determined by topical clusters, but rather on a unit basis.

The SOM as an interface to Digital Libraries has further been demonstrated in the SOMLib Digital Library System [2]. The SOMLib system utilises the SOM and other techniques to support the user by employing as many concepts as possible which she already knows from a conventional library. Similar to a map of a conventional library, depicting the arrangement of the shelves where books on certain topics are located, the SOM gives the overview over the contents of the Digital Library. Similar to finding related books in the same shelf, once the user has found a specific document of interest, she can find documents that are related in content in the neighbourhood of that document. A symbolic visualisation of bookshelves using the LIBViewer method further supports this metaphor. The SOMLib system utilises a labelling algorithm to automatically add semantic descriptors to the single map units, without having the need for any category information being available. This method is described in more detail in Section III-B, where we will also present an extended version.

In [3], the SOM is used to create a web-based knowledge map of news articles. The application supports hierarchical zooming into the map in pre-defined levels of zoom. However, the map layout changes on different levels of zooming. Besides the map, also a hierarchical list of topics is displayed as an alternative for users who prefer a one-dimensional visualisation. [7] uses clustering on the map to apply labels to regions, which are generated based on term and document frequencies, using a $tf \times idf$-based approach. The focus in this work is on the visualisation, which tries to resemble geographical maps as closely as possible. This is achieved using a separate GIS software system. The interaction possibilities for the user are limited -

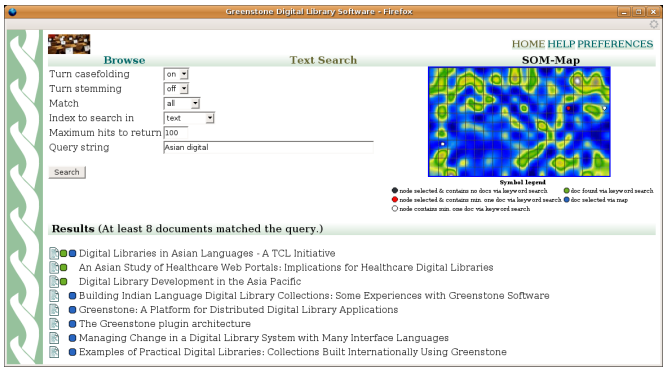

Fig. 1. Enhancing the traditional Greenstone query search with a SOM map

zooming is in predefined levels, the labels cannot be changed interactively, and only one type of visualisation is available.

Based on the concept of the SOMLib system, a sophisticated desktop client software has been developed, which allows for various ways of user interaction with the Digital Library content [8]. With zooming and panning functionalities, the user can analyse the content at any desired detail level, from viewing the whole map at once down to viewing single documents. Tools for selecting rectangular regions or units along a path allow the user to select several documents at once, and open them for example in a text viewer or in an audio application for playlist generation.

One important aspect of SOMs as interfaces to Digital Libraries is that they should not be meant to replace traditional query and retrieval techniques, but rather be complementary to it. This approach is presented in [9], where a Self-Organising Map is integrated into the popular open-source system Greenstone as a new service, based on the existing query services such as search or list browsing. That way, the user can still use all the basic functionalities provided by Greenstone, but she will also be able to use the wealth of additional information the SOM mapping provides about the documents matching the query results and the whole collection itself. A screenshot of the system is depicted in Figure 1, where the left side figures the traditional Greenstone search interface, the lower part the document result listing, and the right part holds the map. The map can be used in two different ways. First, results of the Greenstone search will be highlighted on the map by markers. The user can immediately see which documents have a topical similarity, as these documents will usually all be located close to each other and form a cluster on the map. This way, distinguishing between different topics found on an ambiguous word as e.g. 'jaguar' becomes easily – the documents referring to the sports car will be clearly separated from those talking about the animal because they appear in a different context together with other words. Additionally, outliers found via the search become visible as isolated spots on the map. Secondly, the user can explore the map – she can select nodes, upon which

the documents lying on that nodes will be added to the result list of the Greenstone search. This allows the user to retrieve potentially relevant documents for a specific information need, even if they were not initially retrieved by the (usually rather short) query issued. Documents that have been matched both by the map selection and the search result will be marked especially, as they may be of higher importance. The user can get additional information on the content of the collection by mouse-over popups displaying terms that best describe the documents on a certain node. A small user study suggested that this interface is suitable for Digital Libraries once the user has become a bit familiar with the map.

The SOM has also been used for other types of media besides text. In the SOMeJB project [4], the concept of the SOMLib system has been extended to audio and music documents. Similar to the SOMLib system, the SOMeJB arranges musical pieces, described by a set of feature vectors extracted solely from the audio content, into topical clusters by the sound characteristics, as the user is familiar with from a traditional record store. In the PicSOM project [10], the SOM has been used for Information Retrieval in image databases, incorporating methods of relevance feedback.

III. Describing the Self-Organising Map Regions

In this section we present our work on identifying and describing regions in the mapping generated by the Self-Organising Map. As the SOM does not generate a partition of the map into separate clusters, we utilise another clustering algorithm on the weight vectors of the units to identify the regions (Section III-A). Applying the LabelSOM method (Section III-B), we create semantic labels for those regions (Section III-C) that assist the user in getting a first glance overview of the contents of the map. To further support the analysis phase, we additionally provide summarisation of documents of the regions using Automatic Text Summarisation methods (Section III-D).

A. Clustering

Clustering is an unsupervised process of finding natural groupings amongst unlabelled objects. The members of a cluster are similar in some way, and are dissimilar to members of other clusters.

To identify topical regions, we are clustering the units of a SOM applying an agglomerative, hierarchical clustering algorithm on the units' weight vectors. In the beginning of an agglomerative clustering process, every unit lies in its own cluster. In each subsequent step, the two nearest clusters are merged, until finally only one cluster remains. Specifically, we use Ward's linkage [11] (also known as minimum variance clustering) as one of the most performant within the linkage clustering families. In this algorithm, the distance of each pair of clusters is defined by the increase in the "error sum of squares" (ESS) if the two clusters are to be combined. The ESS of a cluster X of $|X|$ values is defined as:

$$ESS(X) = \sum_{i=1}^{|X|} |x_i - \frac{1}{|X|} \sum_{j=1}^{|X|} x_j|^2 \qquad (1)$$

and the distance D between two clusters X and Y is defined by

$$D(X,Y) = ESS(XY) - [ESS(X) + ESS(Y)] \qquad (2)$$

where XY is the union of clusters X and Y.

The result of the Ward's algorithm is a hierarchy of clusters which the user can browse through. Increasing the number of displayed clusters means splitting existing clusters into two new ones, while reducing the number of clusters is achieved by merging two clusters into one. This is advantageous over a non-hierarchical clustering algorithm, where changing the number of clusters might completely change the layout of clusters, which obviously is not a desired behaviour when we want to allow the user to interactively analyse the map contents by changing the number of clusters. Moreover, hierarchical clustering allows us to at the same time display multiple layers with a different number of clusters. In contrast to other clustering algorithms, the clusters of a layer with more clusters can never be cut by clusters of a layer with less clusters.

B. Labelling units with LabelSOM

To assist the user in interpreting the regions of the SOM, we automatically generate labels for the clusters we identified previously. The cluster labels are based on the units labels generated by the LabelSOM method [12], which assigns labels to the units of the SOM describing the features of the data points mapped onto the respective unit. This is done by utilising the *quantisation error* q_{i_k} of the vector elements, which is the sum of the distances for a feature between the unit's weight vector m_i and all the input vectors $x_j \in C_i$, i.e. the vectors mapped onto the unit i.

$$q_{i_k} = \sum_{x_j \in C_i} \sqrt{(m_{i_k} - x_{j_k})^2} \qquad k = 1 \ldots n \qquad (3)$$

This means that a low quantisation error characterises a feature that is similar in all input vectors to the weight vector. Thus the assumption is made that this feature describes the unit well. If the input vector however contains features which are non existent and therefore have the value of 0, those attributes often also have a quantisation error of almost 0 for a unit. Such features are in most cases not appropriate for labelling the unit, since this would describe what the unit does not contain. Therefore, we require a feature to also have a minimum average value, calculated from all the input vectors mapped to the unit.

C. Labelling Regions

To choose a label for a region, we consider only the unit labels present in that cluster, as the unit labels are already a selection of features describing the contents of each unit. This

method is faster in computation than checking all possible features.

Depending on the data, it is preferable to choose the region label based upon a low average quantisation error, a high mean value, or a combination of both. Therefore we offer the user the possibility to interactively assign priority weights for those two measures, to achieve more meaningful labels. Making use of the properties of the hierarchical clustering as described in Section III-A, we can also display two or more different levels of labels, some being more global, some being more local.

In the visualisation of the SOM, the labels are placed in the centroid of the cluster, which may result in some overlapping labels. To achieve a clear arrangement, the labels can be manually moved on the map, or adjusted in their size and rotation. For some labels, it might also be useful to edit their text, for example if the label text is only a word stem as in our experiment described later in this paper.

D. Region Summarisation

Even though labelling the map regions assists the user in quickly getting a coarse overview of the topics, labels can still be ambiguous or not conveying enough information. Therefore, we also employ methods from the Automatic Text Summarisation field. Based on the regions identified from the clustering process, we automatically provide a short summary of the contents of the documents mapped onto those regions, allowing the user to get a deeper insight into the contents.

Automatic Text Summarisation [13] tries to automatically generate a summary of one or more texts to present the main ideas of the contents in a short and compact form. It can in principle be divided into two areas: single and multi-document summarisation. Single document summarisation deals with providing a summary of a single document, may it be by extraction of the most relevant sentences, or the more sophisticated approach of generating an abstract of the text. Multi-Document summarisation, on the other hand, deals with generating summaries of a whole collection of documents. Simple approaches would just treat each single document separately, generate the summary of it, and then present all the summaries to the user. This approach of course does not consider redundancy in the extracted sentences. More advanced techniques would treat the whole document collection at once, and extract the sentences which are most important concerning all documents. Redundant sentences are eliminated first by applying a measure for overlapping words, and remove sentences with too high similarity.

Although we also provide summaries of single documents in our application, the main focus is to assist the user in analysing the contents of the Digital Library by providing summaries of the previously identified regions using multi-document summarisation. The application allows the user to select whole regions, or manually any other rectangular shape or units a long a path. For the chosen documents, the user can choose from several different summarisation algorithms using different weighting schemes to determine the importance of sentences for the summaries, and can also specify the desired length of the summary, measured in percent of the original sentences.

IV. Experiments

The following experiments were performed using the 20 newsgroups data set[1], a big benchmark corpus which has become very popular for text experiments in the field of machine learning. The data set consists of newsgroup postings from the 20 newsgroups listed in Table I. Each newsgroup contains 1,000 articles from the year 1993; each text consists of the message body and in addition the 'Subject' and the 'From' header lines.

TABLE I
The 20 Newsgroups data set

alt.atheism	rec.motorcycles	soc.religion.christian
comp.graphics	rec.sport.baseball	talk.politics.guns
comp.os.ms-windows.misc	rec.sport.hockey	talk.politics.mideast
comp.sys.ibm.pc.hardware	sci.crypt	talk.politics.misc
comp.sys.mac.hardware	sci.electronics	talk.religion.misc
comp.windows.x	misc.forsale	sci.med
rec.autos	sci.space	

In our experiments we used a standard bag-of-words indexing approach. Porter's stemming algorithm [14] was applied to remove prefixes and suffixes to obtain word stems. From the remaining word stems, the features for the input vectors were selected according to their document frequency, and the weights are computed using a standard $tf \times idf$ weighting scheme. This resulted in a 3151 dimensional feature vector for each document, from which the maps were trained. The specific map we will use in the remainder of this paper to illustrate our results is of 75x55 units in size.

A. Cluster Hierarchy Browsing and Labelling Regions

In our application it is possible to explore the clustered SOM interactively: to view the different levels of clustering and to zoom into the map to view the single postings. The user can browse the clustering levels either viewing only the cluster borders, or also highlighting each cluster in a different colour. The former is shown in Figure 2, illustrating the steps from one to eight clusters. There is a special cluster in the lower right-hand corner with the a label also used on other clusters – 'god' in the fourth step, and 'gun' in the steps five to seven. This cluster is, however, not a separate one - when viewing the clustering with colours, it becomes apparent that this area is part of the disjoint clusters 'god' and 'gun', respectively, in the upper part of the map. With this interactive exploration of the clusters, the user can gain valuable information about the structure of the documents in the collection.

The upper image in Figure 3 shows nine clusters with larger labels, and in addition 67 clusters with smaller labels. The two clusters labelled 'david' are in fact one disjointed

[1] http://people.csail.mit.edu/jrennie/20Newsgroups

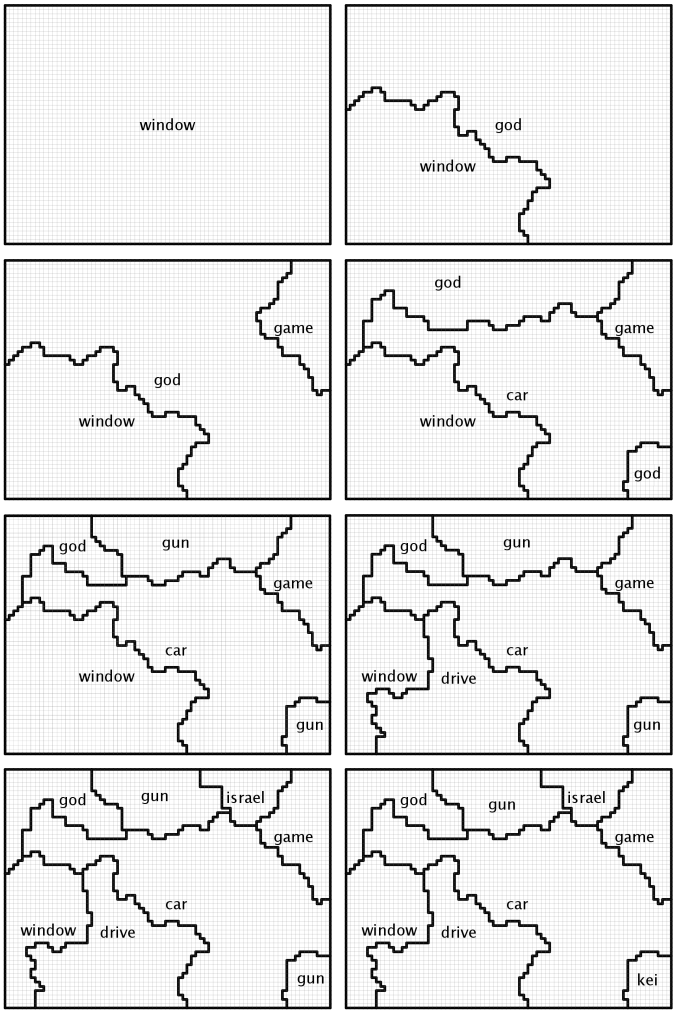

Fig. 2. 1 – 8 Clusters With Labels

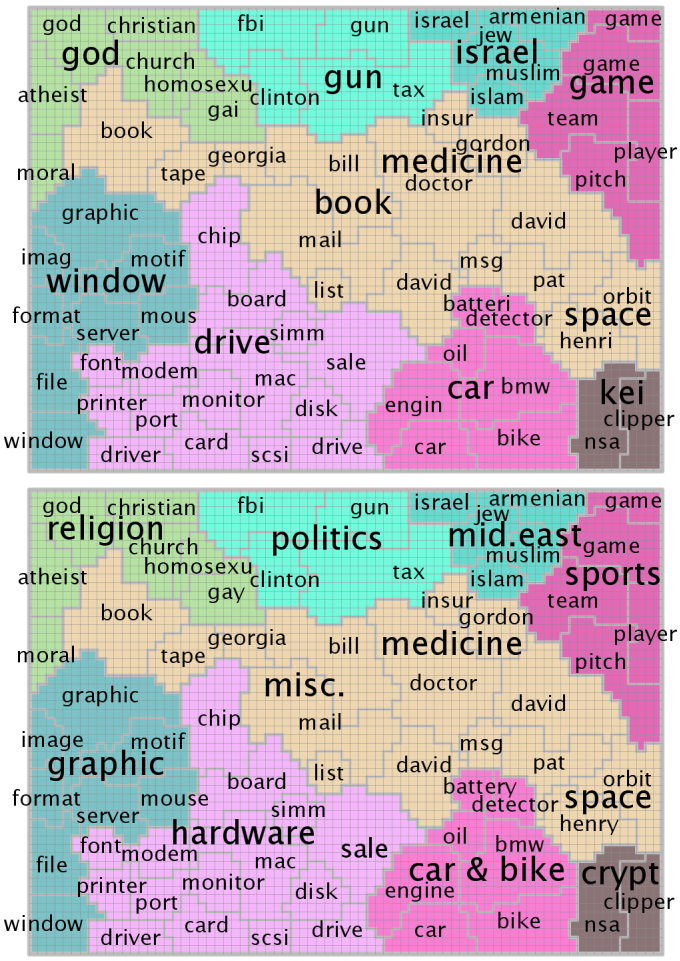

Fig. 3. Nine coloured and labelled clusters and 67 smaller clusters with labels

cluster. As a result of the stemming algorithm, words ending with an *y* now end with an *i*, for example the labels 'kei' (containing most of sci.crypt) or 'batteri' (various postings e.g. from sci.electronics or rec.motorcycles). There are also some labels where obviously the suffixes of the original words have been removed, as in the labels 'imag' or 'insur'. The labels were not edited to show the terms based on which the map has been created.

In the top right-hand corner is a large cluster labelled 'game' containing most postings from the two sports related newsgroups. The large cluster next to it labelled 'israel' contains mainly postings from talk.politics.mideast. In the upper left-hand corner there is a cluster labelled 'god' containing all the newsgroups dealing with religion, i.e. alt.atheism, soc.religion.christian and talk.religion.misc. It is interesting to note, that in contrast to the newsgroup hierarchy, where these groups lie in three different top level hierarchies, they are combined into one cluster here.

The large cluster in the middle labelled 'book' contains many of small clusters of which only a few have meaningful labels. The small clusters labelled 'insur' and 'doctor' suggest that they contain postings from the sci.med newsgroup and the cluster label 'orbit' relates to the newsgroup sci.space. The labels containing names such as 'gordon', 'david' or 'bill' do not help in identifying the underlying topics in those areas of the map. However, names cannot be easily automatically removed, as some common names as Mark or Bill are also a verb or noun respectively. Furthermore names can sometimes be useful labels, for example if they refer to a famous person. The small cluster labelled 'drive' lies in the cluster with the hardware topics but also directly next to the cluster labelled 'car'. It imples that in this area lies an transition of the word *drive* being used in the meaning of *hard disk drive* or in the meaning of *to drive a car*.

To enhance the comprehensibility of the map some labels are manually edited, which is shown in the lower image of Figure 3: word endings have been added and the labels of the larger areas have been edited to better suit the diverse topics. For example the cluster automatically labelled 'car' is extended to 'car & bike' to point out both newsgroups contained in this cluster. The cluster previously labelled 'gun' containing the newsgroups talk.politics.guns and parts

of talk.politics.misc and talk.politics.mideast is adapted to 'politics'. The large cluster in the middle is manually described with three labels to point out the various topics inside.

The map thus created can now be used to interactively present the contained information of the Digital Library in an intuitive way.

B. Region Summarisation

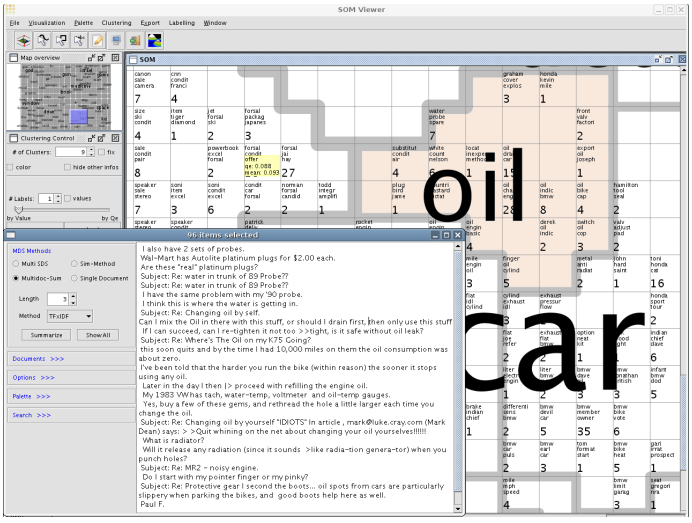

Fig. 4. Automatic Summary of the cluster 'oil'

Figure 4 shows the summarisation of one of the regions in the map, namely the cluster labelled 'oil'. The lower-left part of the interface shows the summarisation module, which allows the user to select a summarisation method, and the desired length of the summary. In our example, we use a multi-document summarisation extracting sentences considering their importance for the whole collection of documents selected, and chose 3% of the selected documents as desired summarisation length. A small user study on the summarisation showed that users find the summaries acceptably comprehensible and useful, and that generally a summary of regions can help in understanding the map better.

V. Conclusion

In this paper we presented the usage of the Self-Organising Map as an interface to Digital Libraries. On top of this well-known approach, we presented recent work on methods to assist the user in interacting with the map. We employ clustering of the SOM to reveal hierarchical structures which can be explored by the user to get a rough overview of the structure of the data on the map. The clustering identifies regions, which we describe on the one hand very concisely by single descriptive words extracted from the document contents, and secondly by applying automatic text summarisation techniques to generate executive summaries of the contents. All methods are integrated into a single application, that provides additional features such as visualisations and advanced interaction via zooming and panning, and selection of arbitrary regions of the map.

With these tools available, the user can be greatly assisted in analysing the SOM generated from the contents of the Digital Library, and therefore getting a quick overview of the contents of the Digital Library itself, and the structure and relationship of the documents it contains, even if the number of documents is huge and their topics diverse. Therewith, the SOM becomes an attractive tool to support Information Management.

Future work will focus on user studies on the region identification, labelling and summarisation, and the interaction possibilities with the map.

References

[1] T. Kohonen, *Self-Organizing Maps*, ser. Springer Series in Information Sciences. Berlin, Heidelberg: Springer, 1995, vol. 30.

[2] A. Rauber and D. Merkl, "The SOMLib digital library system," in *European Conference on Digital Libraries (ECDL 1999)*. Paris, France: Springer, September 22-24 1999, pp. 323–342.

[3] T.-H. Ong, H. Chen, W. Sung, and B. Zhu, "Newsmap: a knowledge map for online news," *Decision Support Systems*, vol. 39, no. 4, pp. 583–597, 2005.

[4] A. Rauber and M. Frühwirth, "Automatically analyzing and organizing music archives," in *Proceedings of the 5. European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2001)*. Darmstadt, Germany: Springer, Sept. 4-8 2001.

[5] M. Agosti, S. Berretti, G. Brettlecker, A. del Bimbo, N. Ferro, N. Fuhr, D. Keim, C.-P. Klas, T. Lidy, M. Norrie, P. Ranaldi, A. Rauber, H.-J. Schek, T. Schreck, H. Schuldt, B. Signer, and M. Springmann, "Delos-DLMS - the integrated DELOS digital library management system," in *Proceedings of the DELOS Conference on Digital Libraries*, Pisa, Italy, February 13-14 2007.

[6] T. Kohonen, S. Kaski, K. Lagus, J. Salojrvi, V. Paatero, and A. Saarela, "Organization of a massive document collection," *IEEE Transactions on Neural Networks, Special Issue on Neural Networks for Data Mining and Knowledge Discovery*, vol. 11, no. 3, pp. 574–585, May 2000.

[7] A. Skupin, "A cartographic approach to visualizing conference abstracts," *IEEE Computer Graphics and Applications*, vol. 22, no. 1, pp. 50–58, 2002.

[8] R. Neumayer, M. Dittenbach, and A. Rauber, "PlaySOM and PocketSOMPlayer: Alternative interfaces to large music collections," in *Proceedings of the Sixth International Conference on Music Information Retrieval (ISMIR 2005)*, London, UK, September 11-15 2005, pp. 618–623.

[9] R. Mayer and A. Rauber, "Adding SOMLib capabilities to the Greenstone Digital LibrarySystem." in *Proceedings of the 9th International Conference on Asian Digital Libraries (ICADL)*. Springer, November 27-30 2006, pp. 486–489.

[10] J. Laaksonen, M. Koskela, S. Laakso, and E. Oja, "PicSOM-content-based image retrieval with self-organizing maps," *Pattern Recogn. Lett.*, vol. 21, no. 13-14, pp. 1199–1207, 2000.

[11] J. Ward, "Hierarchical grouping to optimize an objective function," *Journal of the American Statistical Association*, vol. 58, no. 301, pp. 236–244, March 1963.

[12] A. Rauber and D. Merkl, "Automatic labeling of Self-Organizing Maps for Information Retrieval," *Journal of Systems Research and Inf. Systems (JSRIS)*, vol. 10, no. 10, pp. 23–45, December 2001.

[13] I. Mani, *Advances in Automatic Text Summarization*, M. T. Maybury, Ed. Cambridge, MA, USA: MIT Press, 1999.

[14] M. Porter, "An algorithm for suffix stripping," *Program*, vol. 14, no. 3, pp. 130–137, 1980.