WEBSOM FOR TEXTUAL DATA MINING

Krista Lagus, Timo Honkela, Samuel Kaski and Teuvo Kohonen
Neural Networks Research Centre
Helsinki University of Technology
P.O.Box 2200
FIN-02015 HUT, FINLAND
`http://www.cis.hut.fi/nnrc/`
`http://websom.hut.fi/websom/`

Abstract. New methods that are user-friendly and efficient are needed for guidance among the masses of textual information available in the Internet and the World Wide Web. We have developed a method and a tool called the WEBSOM which utilizes the self-organizing map algorithm (SOM) for organizing large collections of text documents onto visual document maps. The approach to processing text is statistically oriented, computationally feasible, and scalable—over a million text documents have been ordered on a single map. In the article we consider different kinds of information needs and tasks regarding organizing, visualizing, searching, categorizing and filtering textual data. Furthermore, we discuss and illustrate with examples how document maps can aid in these situations. An example is presented where a document map is utilized as a tool for visualizing and filtering a stream of incoming electronic mail messages.

Keywords: data mining, document filtering, exploratory data analysis, information retrieval, self-organizing map, SOM, text document collection, WEBSOM

1. Introduction

With the advent of the Internet and the World Wide Web endless opportunities have opened for spreading and accumulating information. However, simultaneously this poses a challenge to all information management methods and pushes them beyond their limits. There is a lack of sufficient tools for dealing with the overwhelming flood of information. Information-related stress has been reported in the media, and many have noticed that even reading through their daily incoming emails may take up a large part of the day. Searching for relevant information from the WWW is no less time-consuming: search engines, although fast and comprehensive, still present their results as lists of hits that the user must go through to distinguish the rare relevant items from the garbage. New methods are needed to bring order to the current chaos.

1.1. Needs and motives concering information

It may be useful to make a distinction between different needs and motives regarding information: (1) the explicit search for information on a specific topic, (2) the explorative attitude of finding out what is "out there", and (3) the somewhat more passive role of a receiver of, say, a flood of email where the initiative to pass the piece of information comes from elsewhere. These different cases are matched with different requirements for tools: search engines specialize in the first case, the interlinked web of HTML pages tries to serve in the second case, and "push"-like technology, including email lists, are being used in the last case.

1.2. Reasons that led to the development of the WEBSOM method

1.2.1. *Limitations of data mining by search engines*

One of the oldest methods for finding information is to index a document collection and to use search words or more complex search expressions in Boolean logic in order to find documents matching the criteria. The following three fundamental problems in applying Boolean logic to text retrieval (see e.g. Salton, 1989) make it an unsatisfactory solution. (1) Recall of retrieval is sensitive to small changes in the formulation of a query, and there is no simple way of controlling the size of the output of Boolean queries. The result may as well contain zero or one thousand hits. Moreover, the output is not ranked in the order of relevancy, which is a problem especially when the result set is large. (2) The results of a query give no indication on how many valuable documents were not retrieved, for instance, due to some relevant terms missing from the search expression. (3) If the domain of the query is not known well it is difficult to formulate the query, for example, to select the appropriate query words. These problems are alleviated somewhat in more recent search engines which accept more flexible queries and produce ranked lists of documents as their output. Even the most sophisticated search engines can, however, help in fulfilling only one of the three kinds of information needs discussed in section 1.1.

1.2.2. *Multiple sources and formats of information*

A particular problem regarding information on the Web derives from one of the main strengths of the Web: practically anyone can become an information provider, and therefore few common assumptions can be made regarding either the format or the style of the documents. Standards such as the HTML are helpful, but the fact that a document is written in HTML does not guarantee that the structures defined in

the standard are used as intended, i.e., to encode the content-related roles between different parts of text. And when it comes to emails and Usenet discussion articles, even the appealing rule of thumb "Subject line tells what a message is about" cannot be trusted. In a series of replies people seldom bother to change the "Subject:" header although the actual topic of discussion has changed long ago. Furthermore, one cannot even expect to know the whole vocabulary in advance, and even less the meanings of words. New special terms, slang words, and incorrect spellings appear, spread, and die at a speed that no dictionaries or thesauri can keep up with.

Therefore it is necessary to have available systems that make as few assumptions as possible regarding the form, content, style, and vocabulary of the input. This is in contrast with some formal AI approaches which have been ambitious in representing knowledge accurately and have not considered the general applicability of the method to be as important. Such methods are, of course, useful only if their assumptions are met by the data.

1.2.3. *Measuring similarities often suffices*

A useful tool that helps the user to cope with the information flood need not fully understand the messages. Nor does it necessarily need to be able to represent the entire meaning of a document. It is sufficient if the tool serves to direct the user's attention towards relevant information, or if it provides easily comprehensible overviews and visualizations from which the user can make the final judgement on what to read. In fact, in many cases it suffices to find those documents most similar to a given document and to do this efficiently. Then similar items can be organized close to each other, for example when categorizing email messages into folders or relevance classes such as "trash", "really interesting", and "read when plenty of time".

Many of the questions regarding the information flood problem can in fact be rephrased in terms of how to find similar documents. The "search" situation becomes "When we have a piece of interesting information how can we find pieces of similar and probably more interesting information?" The "filtering" and "automatic folder assignment" problem can be reformulated as "Can we organize text automatically so that nearby locations contain similar texts?" Even in exploration the similarity is of use: it would be advantageous to have an organization of the collection where similar documents are found near each other so that when we find an interesting document we probably find many more related documents nearby. This is exactly the approach taken by the WEBSOM system.

1.3. The WEBSOM—a new tool for organizing text

We have introduced a method called the WEBSOM (Honkela et al., 1996; Kaski et al., 1996; Kaski et al., 1998; Kohonen et al., 1996b; Lagus et al., 1996) which has shown to be able to cope with many of the different needs (discussed in section 1.1) that are typical of information processing situations. Our aim has been to avoid some of the limitations of the existing approaches. The WEBSOM method can be used to organize collections of text documents onto a visual map display where similar documents are found near each other. This organization is achieved automatically using a neural network algorithm called the self-organizing map (SOM; Kohonen, 1982). We have also designed a user interface consisting of HTML pages and CGI scripts which offers an overview of the document collection and enables its convenient exploration with any graphical WWW browser. The method is scalable—over a million documents have been organized on a single document map—and computationally feasible.

The providers of large document collections would often be the natural parties to apply the WEBSOM method. For example, news agencies or patent offices with large archives, who would like to open their archives on-line to the public might use the method to provide a convenient and intuitive means of exploring the archive. Also individual users will in time have constructed their own databases by storing personal email messages and other files, and thus they also become candidates for applying the method. In section 4 we discuss how the WEBSOM can be of help in these various situations, and illustrate its use.

2. SOM for data mining and information retrieval

2.1. Self-organizing map (SOM) algorithm

The self-organizing map (SOM) (Kohonen, 1982; Kohonen, 1995; Kohonen et al., 1996a) is a means for automatically arranging high-dimensional statistical data so that alike inputs are in general mapped close to each other. The resulting map avails itself readily to visualization, and thus the distance relations between different data items can be illustrated in an intuitive manner.

The following line of thought may help in gaining an intuitive understanding of how the self-organizing map algorithm works. Consider an information processing system that must learn to carry out various tasks. Let us assume that the system may assign different tasks to different sub-units that learn from experience. Each new task is assigned to the unit that can best complete the task. Since the units receive

tasks that they already can do well and since they learn from them they become even more competent in those tasks. This is a model of specialization by competitive learning. Furthermore, if the units are interconnected in such a way that also the (predefined) *neighbors* of the unit carrying out a task are allowed to learn some of the task, slowly the system becomes ordered: nearby units have similar abilities and the abilities change slowly and smoothly over the whole system. This is the general principle of the self-organizing map (SOM). The system is called a *map* and the task is to imitate, i.e., *represent* the input. The representations become ordered according to their similarity in an unsupervised learning process. This property makes the SOM useful for organizing large collections of data in general, including document collections.

The iterative SOM algorithm may be expressed as follows:

1. Initialize the models (in practice usually with methods such as the PCA that provide a rough initial ordering of the models).
2. Present the samples (tasks) to the system in random order until a given number of steps has passed; in most cases each sample will be presented several times:
   a) search for the model that is most similar to the sample,
   b) update the most similar model and its neighbors so that they become even more similar to the sample, and
   c) decrease the range of the neighborhood and the value of the learning rate a little.

2.2. Applications of the SOM

The SOM is one of the most widely used neural network algorithms. Studies in which the SOM has been used or analyzed have been reported in over 3000 scientific articles (Kangas and Kaski, 1998). Most of the early applications were in the fields of engineering (e.g. in visualization of process states: Goser et al., 1989; Kohonen et al., 1996c; Simula et al., 1997) but nowadays a very diverse range of applications is covered, from medicine and biology to economics. Overviews of the applications are given in (Kohonen, 1995; Kohonen et al., 1996c). A collection of recent works has been published in the proceedings of a conference devoted to the theory and applications of the SOM (WSOM'97, 1997; see also `http://www.cis.hut.fi/wsom97/`).

The usefulness of the SOM stems from its two properties: (1) It creates models or *abstractions* of different types of data in the data

set it describes, and (2) it *organizes* the abstractions onto a usually two-dimensional lattice which can be used to generate an illustrative graphical display of the data set. Especially the latter property—the ability to organize data and models—distinguishes the SOM from other neural network algorithms as well as from classical statistical clustering algorithms (Anderberg, 1973; Hartigan, 1975; Jain and Dubes, 1988; Jardine and Sibson, 1971; Tryon and Bailey, 1973).

The ability to construct illustrative displays of essential properties of data sets makes the SOM especially suitable for data exploration (data mining). Some examples of such applications of the SOM include construction of overviews of socio-economic data sets (Kaski and Kohonen, 1996) and financial analyses (Deboeck and Kohonen, 1998). Most relevant to the topic of this article are the applications of the SOM in information retrieval; one of these applications is the WEBSOM method described in the next section.

3. WEBSOM method

3.1. Background

The dominant approach in computerized language processing has been to determine and code linguistic categories and rules "by hand". The computations have been based on symbol-manipulation techniques. The statistical approaches to natural language processing which use for example artificial neural network techniques try to infer the necessary structures statistically from text corpora. When the self-organizing map is used for this task, categories emerge in the self-organizing learning process based on information of the categories that is implicitly available in the text (Honkela et al., 1995; Honkela, 1997; Ritter and Kohonen, 1989). These emergent categories may then be used in natural language processing systems. Similarly, emergent categorization of documents can also be achieved using the self-organizing map.

In an early study a small SOM of scientific documents was formed based on the words that occurred in the titles of the documents (Lin et al., 1991; Lin, 1992). Later the method has been extended to full-text documents (Lin, 1997). Scholtes has developed, based on the SOM, a neural filter and a neural interest map for information retrieval (Scholtes, 1993). In addition to encoding the documents based on their words Scholtes has used character n-grams in the encoding. This approach has also been adopted in (Hyötyniemi, 1996). Merkl (Merkl, 1993; Merkl, 1995; Merkl, 1997) has used the SOM to cluster textual descriptions of software library components based on similar principles. Document

maps have also been considered in (Zavrel, 1996). The SOM has been used in the ET-Map system in categorizing the content of Internet documents to aid in searching and exploration (Chen et al., 1996; Orwig et al., 1997).

3.2. Overview

In a nutshell the WEBSOM method aims at reducing the information overload associated with managing large document collections. The method constructs illustrative displays of document collections that can be utilized for exploring the collection, for information retrieval, and for filtering a stream of incoming messages. Detailed examples of various ways of using the WEBSOM are presented in section 4.

The map display of a document collection is constructed using the SOM algorithm (see section 2.1). The display becomes organized so that nearby locations contain similar documents. The aim is neither to represent nor to analyze the contents of the documents as faithfully as possible; that would be a much too computationally intensive or even an intractable task. We only need a description of the contents of documents that can be used for judging the similarity of the documents. In fact, one of the most important objectives in designing the WEBSOM method is that it be scalable for use with very large document collections. A scalable method cannot afford to waste too much resources on single special cases; the most important issue is to build a useful overall display. We have demonstrated the scalability of the WEBSOM method by organizing a collection of over 1,000,000 documents (Kohonen, 1997; Kaski et al., 1998). With the present methods the organization of a small collection (about 10,000 documents) takes a few hours on a workstation, whereas construction of the largest maps may take several weeks and the required amount of computer memory during construction of the map may be some hundreds of megabytes.

3.3. Document encoding

Although we do not aim at representing all of the semantical subtleties in the documents we do need some representations of their contents for building a display of the collection. Due to computational reasons we consider the documents as collections of words and neglect the order of the words. This is a traditional approach in information retrieval. We then compute certain statistical indicators of these sets of words and collect the indicators into numerical vectors, one vector for each document (called the document vector in Fig. 1 **a**). The frequency of occurrence of a given word in the document is perhaps the simplest

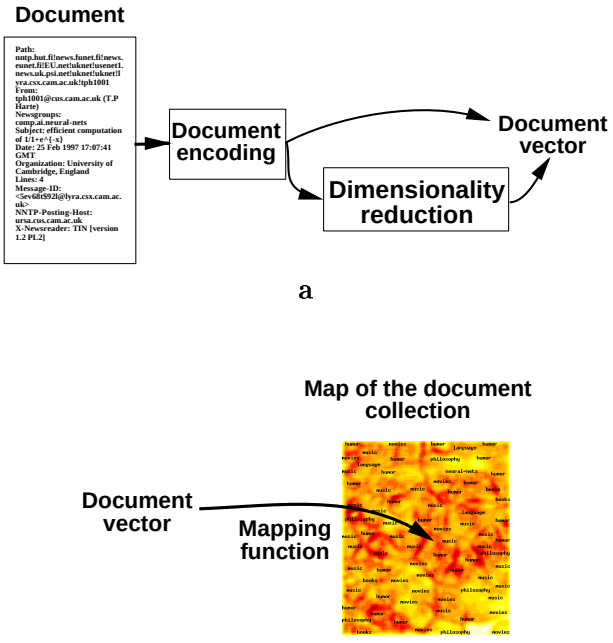

Figure 1. The basic building blocks of the WEBSOM system. (**a**) First the document is encoded into a numerical vector. The dimensionality of the vector can be reduced for computational reasons. (**b**) The numerical representation of the document is mapped onto a display of the document collection that has been generated using the SOM algorithm.

type of indicator. The constructed vector will then be used for further processing.

3.3.1. *Preprocessing*

Before encoding the documents it is useful to preprocess them by removing the parts that are not considered relevant for the organization. Examples include message headers (address, date, etc.) and signatures. Filters that are specific to the format of the documents are utilized to perform this task. Even if such filters cannot be constructed it is nevertheless possible to organize the documents. Then, however, the non-relevant information in the documents also contributes to the organization. For example the common headers in letters from a certain company could cause the letters to be grouped together. In addition, rare words (occurring fewer than, say, 50 times) are often discarded from the documents for computational reasons.

3.3.2. *Vector space model*

In a traditional document encoding method in information retrieval the frequency of occurrence of each word in a document is computed and all the frequencies are collected into a vector. The vector can be additionally normalized to reduce the effect of differing document lengths. This method, called Salton's *vector space model* (Salton and McGill, 1983), has produced good results in document retrieval. The effectiveness of the method can be increased if the importance of the words is taken into account by weighting the frequencies of occurrence. We have used for example entropy-based weighting introduced in (Kohonen et al., 1996b) which gives larger weight to words that distinguish well between different topic areas. Entropy-based weights can be computed automatically if the document collection can be divided into subparts addressing different topics: each word is weighted by (the maximal entropy minus) the Shannon entropy of the distribution of its occurrences in the subparts. When class information is not available the words can be weighted, e.g., using the inverse document frequency (IDF), where words that occur in few documents obtain larger weights. Automatic term weighting methods have been discussed in depth in (Salton and Buckley, 1987).

3.3.3. *Semantic relations*

The vector space model as such does not take into account semantic relations of the words. There exist methods which can be used to first encode the words in a manner where similar words attain similar codes. For example, in LSI (Deerwester et al., 1990) two words are encoded similarly if they co-occur often, whereas in MatchPlus (Gallant et al., 1992) words that occur generally in similar contexts are encoded similarly. In both the abovementioned methods documents are then encoded as sums of the representations of their words. When documents are encoded in this manner their similarity does not depend merely on which words appear in the document but also on whether the documents contain similar words. In our earlier studies with the WEBSOM we have first clustered the words based on their contexts. The clustering is performed using the so-called self-organizing semantic maps (Ritter and Kohonen, 1989), which cluster words into categories based on the similarities of the averaged textual contexts of each word. After the words have been clustered words belonging to the same category will be treated identically.

It has been found, however, that relatively little can be gained by considering word similarities in information retrieval systems (cf. eg. Salton and Buckley, 1987; we have also made similar observations in our recent studies). The reason is probably that the contents of many

documents are redundant enough so that the similarity of the documents does not depend crucially on the semantic relations of any single word. Furthermore, since the word clusters do not necessarily contain synonyms but merely words that have appeared in similar contexts, they may lead to false hits—especially when very short queries are used in performing searches on the map. By clustering the words we gained, however, the additional advantage that the dimensionality of the document vectors was reduced and the encoding of the documents was still extremely fast. In the experiments reported in Section 4 we used a combination of clustering the words followed by a fast dimensionality reduction method discussed next.

3.3.4. *Dimensionality reduction*

The vector space model of Salton is very effective for small document collections and we have used it in some small demonstrations (see e.g. Lagus, 1997). In large document collections, however, the vocabulary is very large. Since each word in the vocabulary requires one dimension in the document vector, the resulting vectors may have hundreds of thousands of dimensions. Computing with large amounts of such vectors is, of course, not computationally feasible, and therefore the dimensionality of the document vectors needs to be reduced before further processing.

There exist several dimensionality reduction techniques in the statistical literature (for example singular value decomposition or SVD, see e.g. Sec. 6.5 of Golub and van Loan, 1983), but unfortunately most, if not all, of them are computationally intensive when the original dimensionality of the vectors is very large. There exists, however, a very simple approach called the *random mapping* method (Kaski, 1998; Ritter and Kohonen, 1989; Kaski et al., 1998) which has been demonstrated to produce almost as good results as the original vector space model (Kaski, 1998). The random mapping consists simply of a multiplication of the original document vectors with a random matrix that produces a smaller output dimensionality. As a result each dimension of the original space becomes replaced by a random direction in the reduced-dimensional space. The original document vector can be regarded as a weighted sum of unit vectors and in the random mapping the unit vectors become replaced by random vectors. One can see an intuitive reason why the random mapping method functions very well: if the remaining dimensionality is still large, most of the random directions are sufficiently different to provide statistically accurate judgements of the similarity of the documents. When the dimensionality was reduced from about 6000 to 100 or more, the result was indistinguishable from the one obtained with PCA, and almost indistinguishable from the original one.

3.3.5. *Incorporating auxiliary knowledge*
It would be possible to incorporate various kinds of auxiliary knowledge in the document encoding process for example by using thesauri to find semantic relations between words, or by using linguistic algorithms for finding the stems of the words. We have preferred using automatic statistical procedures in order to keep the method as general as possible. We have, however, used manually generated lists of words that are neglected. The lists contain very general words that cannot be considered to differentiate between topic areas like, for example, words "that", "is", etc. In addition, we have used stemming algorithms when processing Finnish language in which a single word may have thousands of different inflected forms.

3.4. Document map

The SOM algorithm can be used to generate a graphical display of the document collection after all of the documents have been encoded as numerical vectors. Different regions of the display represent documents having different contents and the content changes smoothly on the display. The speed of change in the content can be visualized as gray levels or colors using the so-called U-matrix (Ultsch, 1993) method, where dark colors imply greater distances in the input space between neighboring map units. After the display has been formed using a representative document collection new articles can be automatically mapped onto the most suitable location on the display (Fig. 1 **b**). Examples of the exploration of document collections with the aid of the map display and of mapping new documents onto the SOM are provided in section 4.

The construction of a large SOM of a large document collection is a computationally intensive task. There exist, however, efficient computational shortcuts which can reduce the amount of computation significantly (Kohonen, 1995; Kohonen, 1996). For example, it is possible to compute first a smaller SOM and then to estimate a larger one based on the models of the smaller SOM.

4. Applications

The WEBSOM method has been used for constructing maps of articles from Usenet discussion groups (Honkela et al., 1996; Kaski et al., 1996; Kohonen et al., 1996b; Lagus et al., 1996; Kohonen, 1997), scientific abstracts (Lagus, 1997), and patent abstracts in English, as well as news bulletins in Finnish. Demonstrations of some of these document maps can be explored at the WWW address `http://websom.hut.fi/websom/`.

The sizes of the document collections have varied from 60 documents to over a million documents organized on a single document map.

The document map has many kinds of potential uses in situations where previously unknown document sets require managing. The maps can be used for visualizing and exploring an unfamiliar document collection and for finding documents similar to a given interesting document. An already familiar document map display can also be used for visualizing the relations between new documents and the collection. For example, new emails or documents retrieved using a more traditional search engine could be mapped this way. Furthermore, the familiar document map can be used to define a filter for keeping, discarding, or categorizing new information.

In visualization of information something familiar is used to illustrate something yet unfamiliar. When the document collection and the map are unknown the familiar information comes from the label words positioned on the map. Another possibility is to use familiar documents as search keys and see where they are positioned on the map display. Later, when the document map is familiar new unknown information can be portrayed by visualizing its relationship with the familiar display.

4.1. Example: Exploration and search of patent abstracts

To illustrate how a document map can be used to explore an unknown or only partially known document collection we organized 10,000 patent abstracts from the European Patent Office's collection onto a document map of 28 by 36 units seen in Fig. 2. The patents were taken from four subcategories related to machines and engines. The vocabulary after discarding the rarest and some common words was 2,075 words, and after dimensionality reduction by self-organizing semantic map the document vectors were 1,250-dimensional. The labels have been automatically selected and positioned on the map so that they (1) occur often in the documents in the area, and (2) occur relatively more often in the area than in remote areas on the map.

Exploration of the ordered map would aid a potential patent applicant to form an idea of the already patented technology and innovations in a certain field. The labels may guide the applicant to interesting map areas and give an immediate overall idea of what kind of material the collection contains. Furthermore, when a new patent arrives it can be placed on the map: the text of the patent is encoded and the map units most similar to it are searched for. The regions found in this way might be used as starting points for exploration, e.g., when a patent engineer tries to find out if there are existing patents that might overlap with the new one. It is important to note, however, that the document

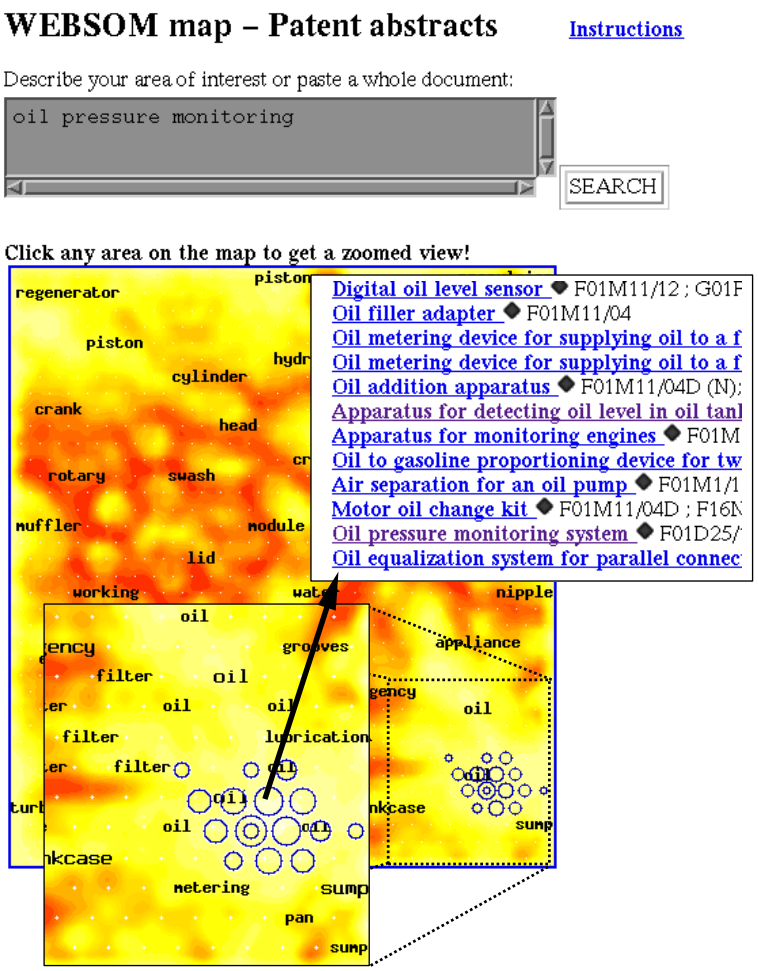

Figure 2. Exploration of a of 10,000 patent abstracts by mapping a new document or query to provide starting points for exploration. The user was interested in finding patents related to oil pressure monitoring. The system encoded this description of interest as a document and found best matching locations for it on the map, marked with circles. A detailed closeup of the region was revealed by clicking on the map and after the second click titles of individual abstracts were shown. An individual document may be viewed by clicking one of the titles. The labels which have been automatically selected and positioned on the map help in gaining an overview of the patent collection and also guide the exploration of the map.

map is neither intended nor suitable for automatically obtaining a new categorization for patents. There exist many useful visualizations of a document collection, of which a single document map provides one. Rather, the document map offers support for *exploration* of the collection, and for *finding new, related information* based on familiar information.

4.2. Example: Document map as a filter

A clear advantage of using the same document map display for a longer time is that as the user grows familiar with the map, the display can be used as a tool for interpreting new information rapidly by overlaying the new information on top of familiar structures. Naturally, the document collection used for organizing the map must be at least somewhat similar to the incoming data stream. In the extreme case, if there is no common vocabulary the visualization will not be meaningful at all.

In the following demonstration a user utilizes a familiar document map both to visualize an incoming data stream—messages from a mailing list—and to filter specific kinds of messages from the stream for immediate reading. The document map used in this example had been created earlier based on articles collected from the Usenet discussion group `comp.ai.neural-nets` (for technical details on how such maps can be constructed see, e.g., Honkela et al., 1997, Kaski et al., 1998). The incoming data stream, an electronic mailing list called `connectionists`, discusses neural networks as well, but in a clearly more academic tone.

Being already familiar with the document map the user first selects three regions of interest to concentrate on reading at the time. The areas are drawn with white ellipses on the document map of Fig. 3. The region in the lower right corner contains some interesting articles on the self-organizing map and on unsupervised learning. The large and rather unspecific region in the middle left discusses AI and intelligence in a rather philosophical and leisurely manner. The small region on the top of the map seems to have specialized particularly in advertisements of cognitive science vacancies. Let us also imagine that from this region the user had found an ideal job opportunity but unfortunately the time to apply for it had already passed.

Next, over 100 messages were taken from the connectionists mailing list, encoded as document vectors, and for each email the single best matching unit was found from the document map. The matches are marked with small circles in Fig. 3.

4.2.1. *Visualizing properties of an incoming document stream*

As a first impression of the spread of the emails on the map the user immediately notices that most of the emails are clustered in the upper right corner (Fig. 3). This region contains lots of conference calls, FAQ-lists, and other articles that mention or discuss in depth a variety of topics dealing with neural networks. To the user who is familiar with the "meaning" of this region of the map the image immediately reveals a property of the mailing list: most of its articles are indeed calls for papers, technical reports, and other elaborate treatments of neural net-

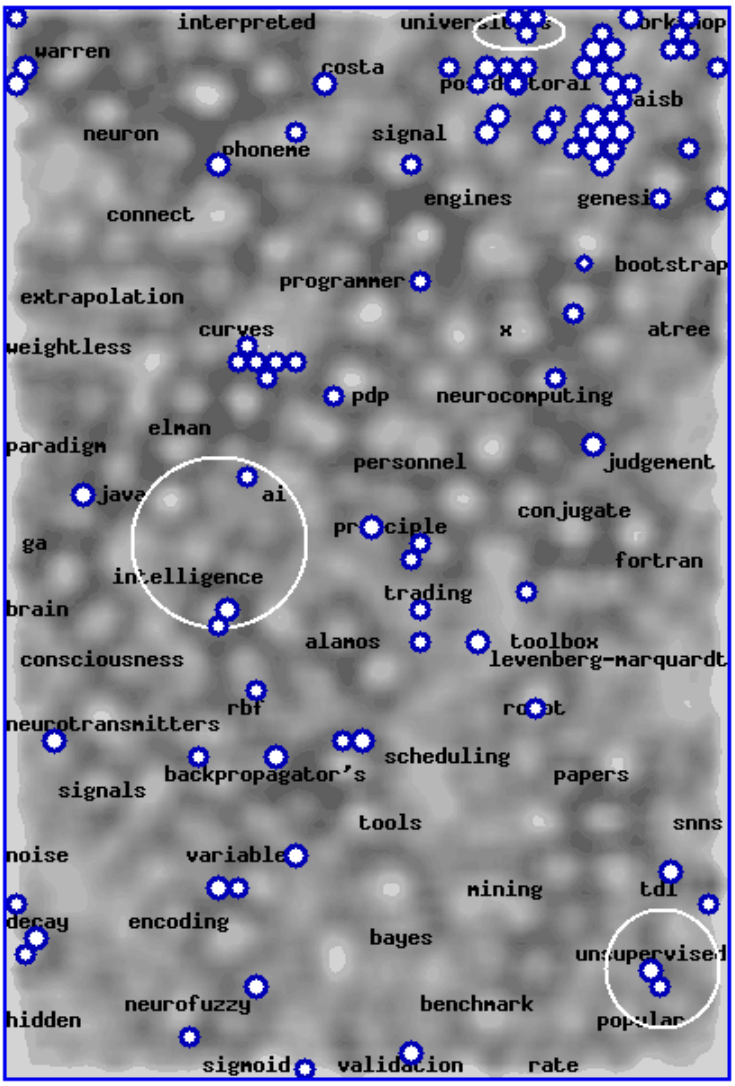

Figure 3. A document map that the user has become familiar with can be used as a filter and a tool for visualizing new documents. First the user marks the interesting regions on the map (here with white ellipses). Next the system positions the incoming stream of documents on the map (small circles). Based on the resulting display the user may get an idea of what kind of material the new documents contain, and also which individual documents would be interesting to read. In this case the contents of over 100 messages from the "connectionists" email list have been visualized on the "comp.ai.neural-nets" map. Fewer circles appear on the display than there were emails since often a single unit was the best match for several messages.

works. It also seems probable that on the mailing list there is not much discussion on topics such as artificial or natural intelligence—at least not the philosophical and speculative type of discussion found on the map region that the user circled. This supports the earlier impression that the mailing list is more strictly concentrated on scientific discussion of neural networks than the newsgroup.

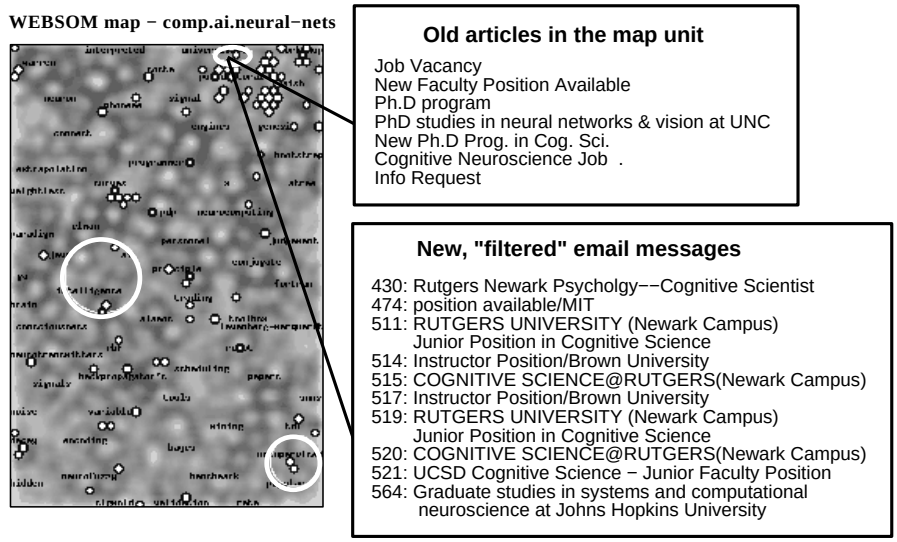

Figure 4. Contents of the region containing articles of cognitive science positions: in the upper box the titles of articles in one map unit and in the lower box the email messages that arrived on the region.

4.2.2. *Filtering a document stream*

When viewing closer the regions marked as interesting there seems to be some messages on each target area. A sample related to the cognitive science opportunity area is presented in Fig. 4. All of the emails advertise cognitive science opportunities and resemble very much the newsgroup articles found in the corresponding map units.

Naturally, when this "filter" is used interactively, the white ellipses need not be considered as strict borders but rather as approximations of the whereabouts of interesting areas. Since the contents change smoothly on the map also the borders between interesting and non-interesting information are fuzzy. For example, seeing now that quite few emails landed on the region discussing artificial intelligence (the large circle) the user may proceed to read the emails found on the closeby regions to see if they nevertheless were interesting.

5. Conclusions

In this work we have illustrated how a novel method called the WEBSOM can be used in organizing collections of documents into maps. The method performs an automatic and unsupervised full-text analysis of the document set utilizing the self-organizing map algorithm. The result of the analysis, an ordered map of the document space, visualizes the similarity relations of the contents of the documents as distance relations on the document map display. A browsing interface has been developed for exploring the maps on the World Wide Web.

In the article we have shown how a document map can be utilized as a tool for visualizing an unknown data collection, or as an automatic filter for collecting documents of interest from a data stream such as incoming electronic mails. Future research may be aimed at improving the scalability of the WEBSOM method still further as well as applying the method on texts in various languages.

6. Acknowledgements

We would like to thank the National Board of Patents and Registration of Finland for the collection of patent abstracts.

We would also like to thank the following persons for participating in the development of the WEBSOM in the later stages of the project: Mr. Antti Ahonen, Mr. Jukka Honkela, and Mr. Jarkko Salojärvi.

References

Anderberg, M. R. (1973). *Cluster Analysis for Applications.* Academic Press, New York.

Chen, H., Schuffels, C., and Orwig, R. (1996). Internet categorization and search: a machine learning approach. *Journal of Visual Communication and Image Representation*, 7(1):88–102.

Deboeck, G. and Kohonen, T., editors (1998). *Visual Explorations in Finance with Self-Organizing Maps.* Springer, London. In press.

Deerwester, S., Dumais, S. T., Furnas, G. W., and Landauer, T. K. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41:391–407.

Gallant, S. I., Caid, W. R., Carleton, J., Hecht-Nielsen, R., Pu Qing, K., and Sudbeck, D. (1992). HNC's MatchPlus system. *ACM SIGIR Forum*, 26(2):34–38.

Golub, G. H. and van Loan, C. F. (1983). *Matrix Computations.* North Oxford Academic, Oxford, England.

Goser, K., Hilleringmann, U., and Schumacher, K. (1989). Vlsi technologies for artificial neural networks. IEEE Micro. Draft.

Hartigan, J. (1975). *Clustering Algorithms.* Wiley, New York.

Honkela, T. (1997). *Self-Organizing Maps in Natural Language Processing.* PhD thesis, Helsinki University of Technology, Espoo, Finland. URL http://www.cis.hut.fi/~tho/thesis/.

Honkela, T., Kaski, S., Lagus, K., and Kohonen, T. (1996). Newsgroup exploration with WEBSOM method and browsing interface. Technical Report A32, Helsinki University of Technology, Laboratory of Computer and Information Science, Espoo, Finland. URL http://websom.hut.fi/websom/doc/websom.ps.gz.

Honkela, T., Kaski, S., Lagus, K., and Kohonen, T. (1997). WEBSOM—self-organizing maps of document collections. In *Proceedings of WSOM'97, Workshop on Self-Organizing Maps, Espoo, Finland, June 4-6*, pages 310–315. Helsinki University of Technology, Neural Networks Research Centre, Espoo, Finland. URL http://www.cis.hut.fi/wsom97/progabstracts/ps/honkela_1.ps.

Honkela, T., Pulkki, V., and Kohonen, T. (1995). Contextual relations of words in Grimm tales analyzed by self-organizing map. In Fogelman-Soulié, F. and Gallinari, P., editors, *Proceedings of ICANN-95, International Conference on Artificial Neural Networks*, volume II, pages 3–7, Paris. EC2 et Cie.

Hyötyniemi, H. (1996). Text document classification with self-organizing maps. In Alander, J., Honkela, T., and Jakobsson, M., editors, *Proceedings of Finnish Artificial Intelligence Conference – Genes, Nets and Symbols*, pages 64–72. Finnish Artificial Intelligence Society.

Jain, A. K. and Dubes, R. C. (1988). *Algorithms for clustering data.* Prentice Hall, Englewood Cliffs, NJ.

Jardine, N. and Sibson, R. (1971). *Mathematical Taxonomy.* Wiley, London.

Kangas, J. and Kaski, S. (1998). 3043 works that have been based on the self-organizing map (SOM) method developed by Kohonen. Technical Report A49, Helsinki University of Technology, Laboratory of Computer and Information Science, Espoo, Finland. See also http://www.cis.hut.fi/nnrc/refs/.

Kaski, S. (1998). Dimensionality reduction by random mapping: Fast similarity computation for clustering. In *Proceedings of IJCNN'98, International Joint Conference on Neural Networks*, volume 1, pages 413–418. IEEE Service Center, Piscataway, NJ.

Kaski, S., Honkela, T., Lagus, K., and Kohonen, T. (1996). Creating an order in digital libraries with self-organizing maps. In *Proceedings of WCNN'96, World Congress on Neural Networks, September 15-18, San Diego, California*, pages 814–817. Lawrence Erlbaum and INNS Press, Mahwah, NJ.

Kaski, S., Honkela, T., Lagus, K., and Kohonen, T. (1998). WEBSOM—self-organizing maps of document collections. *Neurocomputing.* Accepted for publication.

Kaski, S. and Kohonen, T. (1996). Exploratory data analysis by the self-organizing map: Structures of welfare and poverty in the world. In Refenes, A.-P. N., Abu-Mostafa, Y., Moody, J., and Weigend, A., editors, *Neural Networks in Financial Engineering. Proceedings of the Third International Conference on Neural Networks in the Capital Markets, London, England, 11-13 October, 1995*, pages 498–507. World Scientific, Singapore.

Kohonen, T. (1982). Self-organizing formation of topologically correct feature maps. *Biological Cybernetics*, 43(1):59–69.

Kohonen, T. (1995). *Self-Organizing Maps.* Springer, Berlin, Heidelberg. 2nd extended ed. 1997.

Kohonen, T. (1996). The speedy SOM. Technical Report A33, Helsinki University of Technology, Laboratory of Computer and Information Science, Espoo, Finland.

Kohonen, T. (1997). Exploration of very large databases by self-organizing maps. In *Proceedings of ICNN'97, International Conference on Neural Networks*, pages PL1–PL6. IEEE Service Center, Piscataway, NJ.

Kohonen, T., Hynninen, J., Kangas, J., and Laaksonen, J. (1996a). SOM_PAK: The Self-Organizing Map program package. Report A31, Helsinki University of Technology, Laboratory of Computer and Information Science. URL http://www.cis.hut.fi/nnrc/papers/som_tr96.ps.Z.

Kohonen, T., Kaski, S., Lagus, K., and Honkela, T. (1996b). Very large two-level SOM for the browsing of newsgroups. In von der Malsburg, C., von Seelen, W., Vorbrüggen, J. C., and Sendhoff, B., editors, *Proceedings of ICANN96, International Conference on Artificial Neural Networks, Bochum, Germany, July 16-19, 1996*, Lecture Notes in Computer Science, vol. 1112, pages 269–274. Springer, Berlin.

Kohonen, T., Oja, E., Simula, O., Visa, A., and Kangas, J. (1996c). Engineering applications of the self-organizing map. *Proceedings of the IEEE*, 84:1358–1384.

Lagus, K. (1997). Map of WSOM'97 abstracts—alternative index. In *Proceedings of WSOM'97, Workshop on Self-Organizing Maps, Espoo, Finland, June 4-6*, pages 368–372. Helsinki University of Technology, Neural Networks Research Centre, Espoo, Finland. URL http://www.cis.hut.fi/wsom97/progabstracts/ps/lagus.ps.

Lagus, K., Honkela, T., Kaski, S., and Kohonen, T. (1996). Self-organizing maps of document collections: A new approach to interactive exploration. In Simoudis, E., Han, J., and Fayyad, U., editors, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, pages 238–243. AAAI Press, Menlo Park, California.

Lin, X. (1992). Visualization for the document space. In *Proceedings of Visualization '92*, pages 274–81, Los Alamitos, CA, USA. Center for Comput. Legal Res., Pace Univ., White Plains, NY, USA, IEEE Comput. Soc. Press.

Lin, X. (1997). Map displays for information retrieval. *Journal of the American Society for Information Science*, 48:40–54.

Lin, X., Soergel, D., and Marchionini, G. (1991). A self-organizing semantic map for information retrieval. In *Proceedings of 14th Ann. International ACM/SIGIR Conference on Research & Development in Information Retrieval*, pages 262–269.

Merkl, D. (1993). Structuring software for reuse - the case of self-organizing maps. In *Proceedings of IJCNN-93-Nagoya, International Joint Conference on Neural Networks*, volume III, pages 2468–2471, Piscataway, NJ. IEEE Service Center.

Merkl, D. (1995). Content-based software classification by self-organization. In *Proceedings of ICNN'95, IEEE International Conference on Neural Networks*, volume II, pages 1086–1091, Piscataway, NJ. IEEE Service Center.

Merkl, D. (1997). Exploration of text collections with hierarchical feature maps. In *Proceedings of SIGIR'97, 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, New York. ACM.

Orwig, R., Chen, H., and Nunamaker, J. F. (1997). A graphical, self-organizing approach to classifying electronic meeting output. *Journal of the American Society for Information Science*, 48(2):157–170.

Ritter, H. and Kohonen, T. (1989). Self-organizing semantic maps. *Biological Cybernetics*, 61:241–254.

Salton, G. (1989). *Automatic Text Processing*. Addison-Wesley, Reading, MA.

Salton, G. and Buckley, C. (1987). Term weighting approaches in automatic text retrieval. Technical Report 87-881, Cornell University, Department of Computer Science, Ithaca, NY.

Salton, G. and McGill, M. J. (1983). *Introduction to modern information retrieval*. McGraw-Hill, New York.

Scholtes, J. C. (1993). *Neural Networks in Natural Language Processing and Information Retrieval*. PhD thesis, Universiteit van Amsterdam, Amsterdam, Netherlands.

Simula, O., Alhoniemi, E., Hollmén, J., and Vesanto, J. (1997). Analysis of complex systems using the self-organizing map. In Kasabov, N., Kozma, R., Ko, K., O'Shea, R., Coghill, G., and Gedeon, T., editors, *Progress in Connectionsist-Based Information Systems. Proceedings of the 1997 International Conference on Neural Information Processing and Intelligent Information Systems*, volume 2, pages 1313–1317. Springer, Singapore.

Tryon, R. C. and Bailey, D. E. (1973). *Cluster Analysis*. McGraw-Hill, New York.

Ultsch, A. (1993). Self-organizing neural networks for visualization and classification. In Opitz, O., Lausen, B., and Klar, R., editors, *Information and Classification*, pages 307–313, London, UK. Springer.

WSOM'97 (1997). *Proceedings of WSOM'97, Workshop on Self-Organizing Maps*, Espoo, Finland. Helsinki University of Techology, Neural Networks Research Centre.

Zavrel, J. (1996). Neural navigation interfaces for information retrieval: are they more than an appealing idea? *Artificial Intelligence Review*, 10(5-6):477–504.

Address for Offprints: Krista.Lagus@hut.fi