Reduction of the Dimension of a Document Space Using the Fuzzified Output of a Kohonen Network

Vicente P. Guerrero
Library and Information Science Faculty, University of Extremadura, C/José Maréa Alcaráz y Alenda 1 (pasaje), 06011 Badajoz, Spain. E-mail: vicente@alcazaba.unex.es

Félix de Moya Anegón
Library and Information Science Faculty, University of Granada, Campus Cartuja, Granada, Spain. E-mail: felix@goliat.ugr.es

The vectors used in IR, whether to represent the documents or the terms, are high dimensional, and their dimensions increase as one approaches real problems. The algorithms used to manipulate them, however, consume enormously increasing amounts of computational capacity as the said dimension grows. We used the Kohonen algorithm and a fuzzification module to perform a fuzzy clustering of the terms. The degrees of membership obtained were used to represent the terms and, by extension, the documents, yielding a smaller number of components but still endowed with meaning. To test the results, we use a topological classification of sets of transformed and untransformed vectors to check that the same structure underlies both.

Introduction

Many information retrieval techniques are based on the vector space model, in which documents and terms are represented by high-dimensional vectors. The algorithms used in the said techniques usually increase very rapidly with respect to the dimension (as a power law or even exponentially).

Therefore, the reduction of the dimension in these spaces has always been of great interest. Three methods used to this end are: principle component analysis, backpropagation neural networks, and the WCM (Word Category Map). In none of them do the components that emerge have any great significance. We shall briefly comment on the fundamentals of the last two, as these have the greater interest for us.

The *backpropagation networks* (Muñoz, 1994) used for this purpose have a hidden layer with very few nodes, which have simply been trained to reproduce the input. If the final layer manages to reconstruct the inputs from the information in the output generated by a few of the hidden layer inputs, there has been no information loss in the output pattern from the hidden layer, and this can thus be used to represent the initial inputs. The costliness of the associated learning process is one of the method's greatest drawbacks.

The *Word Category Map* (Honkela, Kaski, Lagus, & Kohonen, 1996; Kaski, Honkela, Lagus, & Kohonen, 1996; Lagus, Kaski, Honkela, & Kohonen, 1996) has been used in *Websom* (Kaski, 1999; Kohonen et al., 1999; Lagus & Kaski 1999). This system was designed to classify large numbers of documents from the Internet. The following are the steps in the representation of each document:

(1) Elimination of the high- and low-frequency words to reduce the computational load and the noise.
(2) A word category map is formed from the remaining words (Honkela, 1995). To this end, each term is first assigned a code formed of 90 random components between 0 and 1. To represent each term, vectors are created from the code of the term in question and those of the immediately preceding and following terms. These 270-component vectors are used to train a Kohonen network that yields a category map. For the English language, in some cases quite distinct zones were obtained where nouns, verbs, or functional words were the winners (Honkela, Pulkki, & Kohonen, 1995). There is no reason, however, for this result to be carry over to languages with a different structure. The real innovation of the WCM is the way a small context is included in the representation of each word.
(3) Each document is represented by counting the responses of the WCMs to each of its terms.

Kohonen's algorithm is neural in nature. A neural network is a structure that is able to perform basic intellectual tasks. Although some researchers have, therefore, seen these

Received July 17 2000; Revised February 6, 2001; Accepted June 25, 2001

networks as a new paradigm of artificial intelligence, recent studies have shown that works in one or the other of these two fields do not usually share many references, and neither are they cited by the same works (Van der Besselaar & Leydesdorff, 1996). They are able to assign outputs to multidimensional inputs in a training phase, which in most networks is carried out off-line. This training may be supervised or unsupervised. Supervised training uses a series of input/output pairs so that the network can adapt to the outputs. The unsupervised network only uses inputs that it groups into clusters (as is often done in IR).

Methods and Data

We here also use Kohonen's neural algorithm to reduce the dimension, but now adding a module that fuzzifies the output. We represent the documents using the vector space model. To test the goodness of the reduction, we determine the difference between the topological organization after applying a Kohonen self-organizing map (SOM) to the two representations. We, therefore, divide this section into two subsections—the first relative to Kohonen's algorithm, and the second to the selection and representation of the data.

Kohonen's Algorithm

Despite the enormous complexity of the cerebral cortex microscopically, macroscopically its structure is uniform, even when one is comparing one brain with another. The centers corresponding to such specific activities as thought, vision, hearing, motor functions, etc., are located in specific zones of the cortex, and these zones have a certain relative location with respect to each other. This map may be to a great degree predestined genetically. Nevertheless, interest in discovering how an organization of this type might have arisen led Kohonen to investigate the subject (Kohonen, 1982, 1984, 1995). The product of these researches was the network model that bears his name, with networks that are capable of topologically organizing the inputs. The model's main contribution is a new competitive layer that may be used as a part of other networks or independently.

Kohonen networks differ essentially in the influence that each neuron has on its neighbors. Although in other networks each neuron has a negative influence on the rest in its own layer, in a Kohonen network the influence will be a function of the distance between them: each neuron exerts a positive influence on itself and on the topologically close neurons, but the influence decreases with increasing distance to reach negative values that finally again become positive for the most distant neurons. The consequence is that a bubble of activity is created from all those units that are close to the winner, and that share in the reinforcement corresponding to the learning process. The simulation hardware requires the creation of a competitive layer of a certain complexity. The learning process that it performs each time that an input is presented can be summed up, however, in the following steps: (a) Select as the winning node (each node of the network represented by a different weight vector of the same dimension as the input vectors) that which is closest (typically using the Euclidean distance) to the presented input vector; (b) adjust the weight vectors of the winning node and the nodes corresponding to its neighborhood by adjusting them towards the input vector values (in some cases the reinforcement is the same for the whole neighborhood, and in others it falls off with distance from the winner).

During this phase, the training vectors are presented repeatedly to the network at random, simultaneously with a gradual reduction of the neighborhood and of the learning rate to force the network's stability. After this phase the configuration is one in which the neurons that are topologically close in the network (arranged as a two-dimensional lattice) are winners with respect to clusters of vectors that are close to each other in the input space. Occasionally, as in the present case, the neurons are endowed with a "conscience." This is a mechanism that reduces the probability that a neuron will win a competition as the number of competitions that it has already won rises, which is sometimes used to avoid the problem of the stuck vector (Freeman & Skapura, 1991; Guerrero Bote, Moya Anegón, & Herrero Solana, 2001; Muñoz Garcia, 1994), allowing the victories to be shared out over the whole network.

Because of this topological organization, at times the only thing that is of interest is the clustering performed by the hidden layer, and the entire set of vectors is selected for the training just to see the resulting topological organization. There are two possibilities in this last application: if there are more hidden units than training vectors, the result is an optimal projection onto the chosen topology; if the number of units is less than the number of vectors, the result is a layer that clusters its inputs and orders each cluster topologically, with the number of clusters being equal to the number of neurons making up the hidden layer.

One thus achieves iteratively but simply not only a good cluster analysis but also a good topological organization. Nonetheless, as with other algorithms, there are certain characteristics that are not very satisfying mathematically: the cutoff forced by the number of iterations, the lack of guarantee of convergence, the dependence on the data input order, the attainment of stability by reducing the speed of learning, the generation of a classical instead of a fuzzy partition, etc. There have been attempts at improvement and fuzzification such as the *Fuzzy Kohonen Clustering Networks* of Bezdek, Chen-Kuo Tsao, & Pal (1992), which is aimed at incorporating some of the characteristics of the *c-mean method*, yielding a fuzzy output. Although these attempts result in certain improvements, they all usually have the common denominator of losing the topological organization that is characteristic of this algorithm.

We chose to add on an output fuzzification module that allows each input to be assigned a degree of membership to each cluster. During learning, the algorithm then functions in accord with Kohonen's original idea, varying only in the output phase, where instead of yielding the winning (closest) cluster it generates a fuzzy output that indicates the degree of membership of the presented input in each cluster. This fuzzy output is computed in the same way as does the *c-mean* algorithm (Bezdek, 1981) in the iterative process for $m = 1$, i.e., in a summarized form:

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} (D_{ij}/D_{kj})^2}$$

where u_{ij} is the degree of membership of term t_j in cluster i, D_{ij} is the distance between the vector corresponding to term j, and the weight vector of neuron i (which is just the centroid of the cluster it gives rise to).

Experimental Environment

To test the capacities of this algorithm under conditions that are as general and close to reality as possible, we generated a test database. With a view to being able to contrast the results of the topological organization of the documents, we opted for the use of records from the LISA bibliographic database. We would then be able to compare the classifications that we obtain with that resulting from the assigned subject terms. We considered each abstract in the records as an independent document. Of the remaining fields, we only took into account the subject terms for the subsequent comparison. To maintain as much generality as possible, instead of retrieving information by topics, we selected the last 954 records of LISA (version Summer 96), which were those with an *Accession number* greater than or equal to 9605000. These records have a summary field, and it was this that we used to generate the documentary representation as if it were a full document. The total of these summaries contained 7,758 different words.

The next step after creating the database was to transform these documents into vectors that could be used as inputs for our algorithms or networks. We chose a weighting scheme very close to the classical *IDF*, which was one of those that gave the best results in the study of Noreault et al. (1981). Each component's weight is given by:

$$a_{ij} = t_{ij} \log \frac{F}{f_j}$$

where a_{ij} is the weight assigned to the term t_j in document D_i, t_{ij} is the frequency of appearance of the term t_j in document D_i, f_j is the frequency of appearance of the term t_j in the whole database, F is the total number of words (repeated or not, tokens) in the whole database.

We decided to use this scheme instead of the classical IDF because the latter assigns the greatest weights to the terms that appear in only a single document because it uses (the reciprocal of) *nj*, the number of documents in which the term appears, instead of (the reciprocal of) *fj*, the frequency of appearance of the term *tj* in the whole database. In IDF, these terms are translated into differences between their corresponding document and the rest, and do not allow similarities to be found (Guerrero et al., 2001). Because we wish to make a topological organization to evaluate the dimensional reduction that was attained, it is just such similarities between documents that interest us.

In the study of Noreault, McGill, and Koll (1981), the best similarity measures were those that made angular comparisons. We, therefore, next normalized the resulting vectors by dividing them by their Euclidean norm.

The number of terms in the database was 7,758. This was computationally unfeasible for us, so that it became necessary to determine a set of terms that we could use to describe all the documents of the database. We performed the reduction using the discrimination value introduced by Salton (Moya, 1994; Salton & McGill, 1983). The entire process carried out on the terms can be summed up in three phases:

(1) *Elimination of stop words:* this first phase deals with eliminating functional words of the language that have no meaning. Because the language was English, we used the frequency dictionary of Kucera and Francis (1967), with which we generated a list of 200 stop words corresponding to the most frequent words of that language. This reduced the number of different terms in the database from 7,758 to 7,577.
(2) *Stemming*: we used the *Porter Stemmer*, which is the most commonly used stemmer in English (Frakes, 1992; Porter, 1980). This reduced the number of terms to 5,052.
(3) *Extraction of the terms with the greatest discrimination value:* we calculated the discrimination values using the centroid approximation (El-Hamdouchi & Willett, 1989; Moya Anegon, 1994; Salton & McGill, 1983), and extracted the 1,200 words with the greatest value.

We also needed a vector representation of the terms similar to that of the documents. In a binary representation, the components of the two representations would coincide. In *IDF* weighting, however, each component consists of two parts: one local, indicating the importance of the term in each document (and which varies with the frequency in each document), and the other global, indicating the importance of the term in the database as a whole and that will be the same for all a term's components. As the best similarity functions in the study of Noreault et al. (1981) were found to be those that used angular measures, the global component of the *IDF* (which only modifies the modulus of the vector corresponding to each term) is irrelevant, so that the situation is very close to the use of the frequency in the document for each component. As was done for the document vectors, different document weights can be generated for the terms, based, for instance, on the number of terms the documents contain (Kantor, 1994). It is logical that documents that contain few terms have more weight. One can, hence, use a *IDF*-like weighting scheme for the terms:

TABLE 1. Subject terms used to reduce the document set and the number of documents in which each subject term appears.

Acquisitions	37
Artificial intelligence	28
Business management	20
Computerized information storage and retrieval	23
Conferences	20
Periodicals	22
World Wide Web	58

$$a_{ij} = t_{ij}\log\frac{D}{ND_i}$$

where a_{ij} is the weight assigned to document D_i in term t_j; t_{ij} is the frequency of appearance of term t_j in document D_i; ND_i is the number of terms that appear in document D_i; and D is the total number of terms of the whole database.

Finally, to test the resulting reduction, we shall generate topological classifications with the original document vectors and those resulting from the reduction. Given the dimension (1,200) and the number (954) of the vectors, however, the process becomes computationally burdensome. There is the additional difficulty that one would have in subsequently contrasting the results due to the great number of documents and of subject terms to represent. We, therefore, decided to reduce the number of vectors to process as a function of the subject terms listed in Table 1, which also gives the number of documents in which each subject term occurs. As was described in a previous work (Guerrero et al., 2001), the subject terms were chosen so that (a) they had not been assigned together very often so as to avoid forcing them to appear at the same node, and (b) they corresponded to topics that were markedly either strongly related or weakly related. Also, two subject terms of form were added to check whether the distribution was different. This reduces the computation time and the difficulty in interpretation. It does not lead to any loss in generality, however, because in obtaining the said vectors, we used all the documents and, hence, all the terms that they contributed.

Results

The vectors generated to represent the terms were applied to a 20 × 20 Kohonen network with a fuzzification module. In this way we topologically clustered the terms at the same time as obtaining output vectors formed by the degrees of membership in the different clusters.

To illustrate the differences between the input and the output vectors, we can depict the vectors as grey-scale images, where each component corresponds to a pixel of the image. The said pixel will be brighter or dimmer according to the value of the said component. Figure 1 shows the input vector corresponding to the term *accept*, generated as described in the data section from the database documents, and therefore, with 954 components (one for each document). The image is made up of 31 × 31 pixels for a total of 961, with the last seven being left dark. The vector is normalized so that all of the components lie between zero and unity. The white pixel does not mean that a component has the value unity, because the normalization would mean that all the other components were zero. As there are normally several nonzero components, however, their values will be much less than unity, so that if white is assigned to the value unity most of the points would be indistinguishably dark. We, therefore, assigned white to the largest component, and the other shades in proportion to each value. In the vector corresponding to this image only 21 of the 954 components are nonzero. The white pixel corresponds to component number 552 whose value of 0.4829978 does not even reach 50% of unity.

Figure 2 shows the vector formed by the degrees of membership yielded by the aforementioned Kohonen network. The image is a square with the same dimensions as the network, i.e., formed of the same number of pixels as there are neurons.

One sees that there are now no totally dark pixels, i.e., no component is zero. This is a property of the fuzzification function that was used. Also, one sees a bright zone around the largest component, although smaller than would have been expected.

Thus, we have performed a coordinate transformation of the terms from the original 954-dimensional space to one of 400 dimensions.

Although this is the case for the terms, the documents can be expressed as a function of these terms so that we will be able to transform their components to these coordinates. One may express the required transformation vectorially by

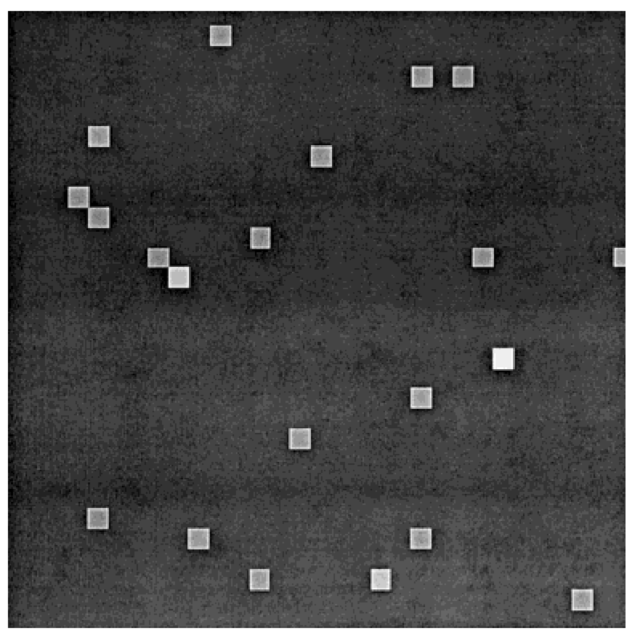

FIG. 1. Grey-scale image corresponding to the input vector that represents the term "accept." Each component is represented by a pixel of the image, and the pixel is brighter or dimmer according to the value of the component.

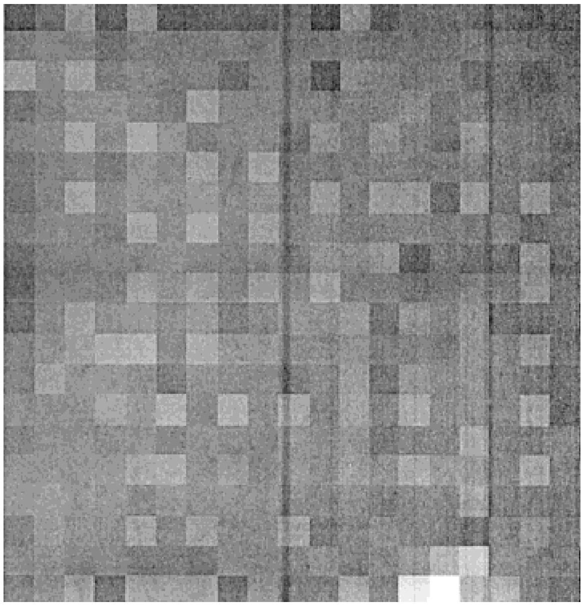

FIG. 2. Grey-scale image corresponding to the degree membership vector for the term "accept." As in Figure 1, each component is represented by a pixel of the image, and the pixel is brighter or dimmer according to the value of the component.

means of a matrix ***T***, whose columns will be the new term vectors:

$$D' = \mathbf{T}D$$

Figure 3 shows examples of document vectors. Those on the left are generated with the IDF scheme, and the corresponding vectors on the right after the transformation.

One sees that their characteristics are similar to the term vectors: the initial vectors only have some isolated bright pixels, while those resulting from the transformation have more continuous values. Some of the transformed vectors are darker because their components vary more, and others are brighter because they have more uniform values.

As we indicated above, to test the goodness of the transformation, we performed a topological classification of the document vectors before and after the transformation using another SOM for the purpose. This type of network has recently been used in the analysis of domains (White, Lin, & McCain, 1998), for *textual data mining* (Lagus, Honkela, Kaski, & Kohonen, 1999), and specifically for the generation of topological maps of a document set, even labeling the zones of influence of each word or term (Chen, Houston, Sewell, & Schatz, 1998; Guerrero Bote, 1997; Guerrero Bote, 2001; Moya, Herrero, & Guerrero, 1998; Lin, 1997; Moya Anegón, Moscoso, Olmeda, Ortiz-Repiso, Herrero Solana, & Geurrero Bote, 1999; Orwig, Chen, & Nunamaker, 1997). They allow the documents to be classified into a two-dimensional array, so that topically related documents appear at nearby nodes of the network (or even at the same node) and vice versa. The idea is to extract the underlying structure of the two sets of vectors and, by comparing them, see whether the reduced vectors still conserve the characteristics that allow their organization as documents.

For this purpose, we used a 20×20 neuron network and a square topology. The result of applying the document vectors obtained from the IDF scheme is shown in Figure 4. We have marked the nodes where the documents were classified with (redundantly) both a color and a letter associated to the subject term assigned to the documents, and with the number of documents classified at the said node. At node (19, 19) for instance, there were two documents classified that had the subject term *Business management* assigned them. One sees in the figure how nodes with documents containing the same subject terms cluster together. In some of the apparent confusions that exist, it was found that the documents had subject terms different from those shown in the figure, so that the analysis performed by the network is more thorough than that which comes from observing some of the subject terms associated with the documents. The most disaggregated subject terms are *Conferences* and *Periodicals*. The explanation might be that they are both subject terms of form, so that it would be natural to expect a topical dispersion of the documents that contain them. Another noteworthy aspect is the placement of the different subject terms' winning zones on the map. For example, the zones corresponding to the subject terms *World Wide Web, Computerized information storage and retrieval,* and *Artificial Intelligence*, which are the most technology related, border each other, etc.

The result of applying the transformed vectors is shown in Figure 5. Observation of the figure shows that the same structure underlies the topological organization as in the previous experiment, although now the subject terms are slightly more mixed. It is worth noting at this point the difficulty that exists in evaluating quantitatively the result of applying this kind of network. Despite the great number of occasions on which they have been used in IR applications, there have been no attempts to evaluate the resulting topology (except that based on user opinion in Chen et al., 1998, which evaluated the interface that was generated rather than the topological organization). We have recently applied the same method of using the subject terms of a bibliographic database to evaluate the resulting topological organization (Guerrero Bote et al., 2001).

We next applied both sets of vectors to a 30x30-dimensional Kohonen network. The results are shown in Figures 6 and 7. They clearly have the same aspect.

Conclusion

The reduction of the dimension has been a much sought-after goal in this vector space model for the representation of full texts. Different methods have been used, most of them giving uninterpretable coordinates.

We have here used a Kohonen Self-organizing Map to cluster the terms present in a collection of documents. To this we coupled a fuzzification module that allowed us to

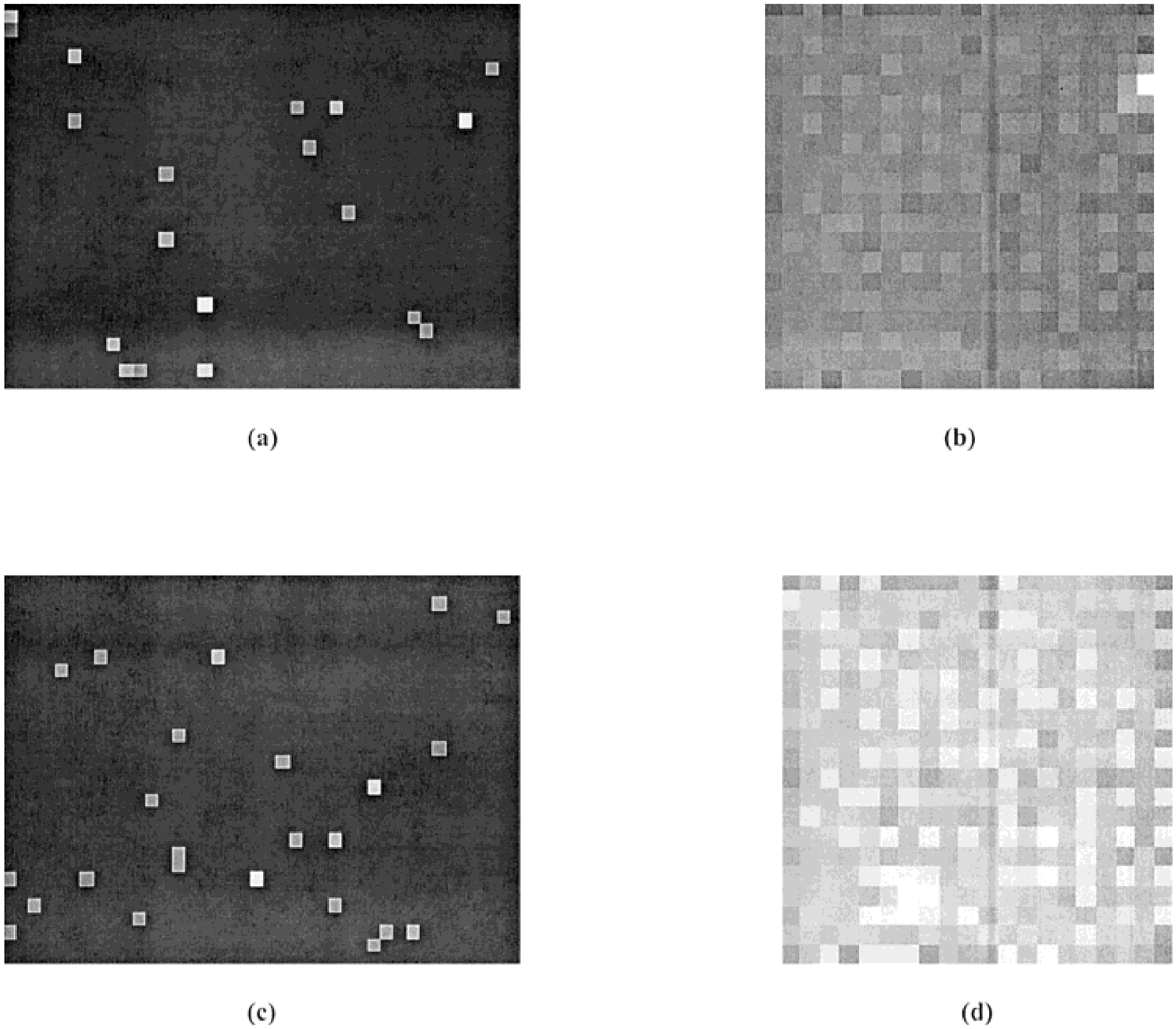

FIG. 3. Grey-scale images corresponding to the vector pairs assigned to three documents (pairs: a/b, c/d), both before (a, c), and after (b, d) performing the transformation. As in Figure 1, each component is represented by a pixel of the image, and the pixel is brighter or dimmer according to the value of the component.

determine the degree of membership of each term to each cluster (or node of the map). This method of fuzzifying the SOM's output has the advantage over other methods described in the literature (Bezdek et al., 1992) of maintaining the topological organization created by the map.

The degrees of membership of each term to each cluster can be used to represent the terms and, by extension as we indicated, the documents. By using a network of smaller dimension than the input, this permits one to obtain a representation of smaller dimension. The said reduction has the following advantages over other existing methods:

(1) *Meaningful components:* while the components that are given by many methods lack meaning, in the present case they represent the degree of membership of each term or document to each cluster. We can consider each component to be labeled by the terms that have the greatest degree of membership to the said cluster.

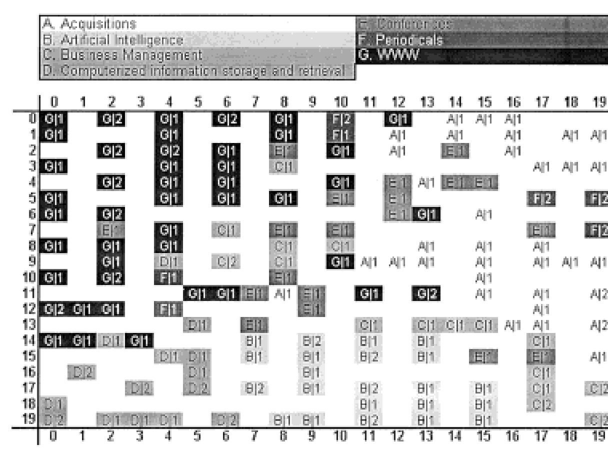

FIG. 4. Document organization generated by a 20 × 20 network with the original document vector. The nodes where the documents were classified have been marked with (redundantly) both a colour and a letter associated to the subject term assigned to the documents, and with the number of documents classified at the said node.

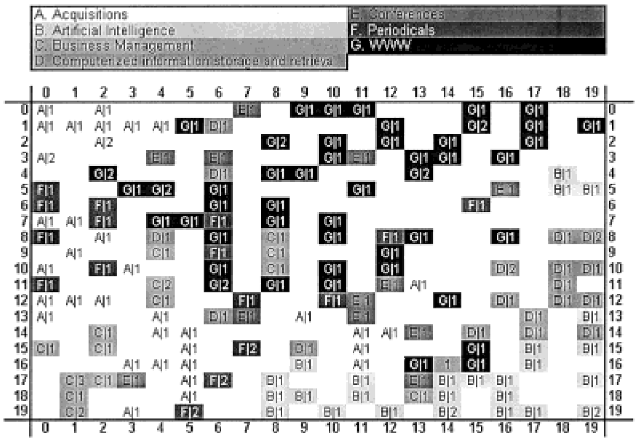

FIG. 5. Document Organization generated by a 20 × 20 network with the reduced document vectors. The nodes where the documents were classified have been marked with (redundantly) both a color and a letter associated to the subject term assigned to the documents, and with the number of documents classified at the said node.

(2) *Representation of terms and documents with the same components:* normally, while the documents are expressed as a function of the terms that they contain, the latter are expressed as a function of the documents in which they appear. In this case, they are all expressed in the same components, and may be processed conjointly, thereby allowing one, for instance, to label clusters, topologies, etc.

(3) *Consideration of all the facets of a term:* as for each term one obtains degrees of membership to each cluster (instead of its assignation to a single cluster), one can state that all of the meaningful facets of each term are taken into consideration.

(4) *Dimension of the output variable:* this will depend on the dimension of the map that classifies the terms, which may be enlarged or reduced. Great reductions in the dimension must be expected to involve major concomitant losses of information.

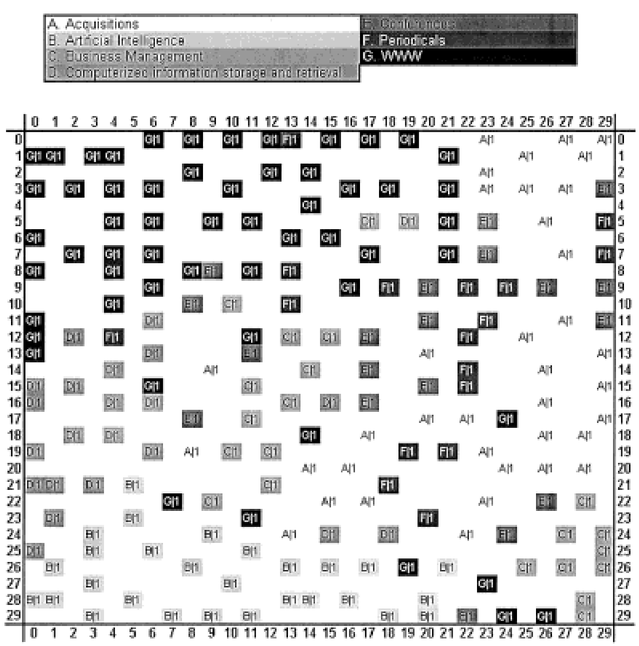

FIG. 6. Document Organization generated by a 30 × 30 network with the original document vectors. The nodes where the documents were classified have been marked with (redundantly) both a color and a letter associated to the subject term assigned to the documents, and with the number of documents classified at the said node.

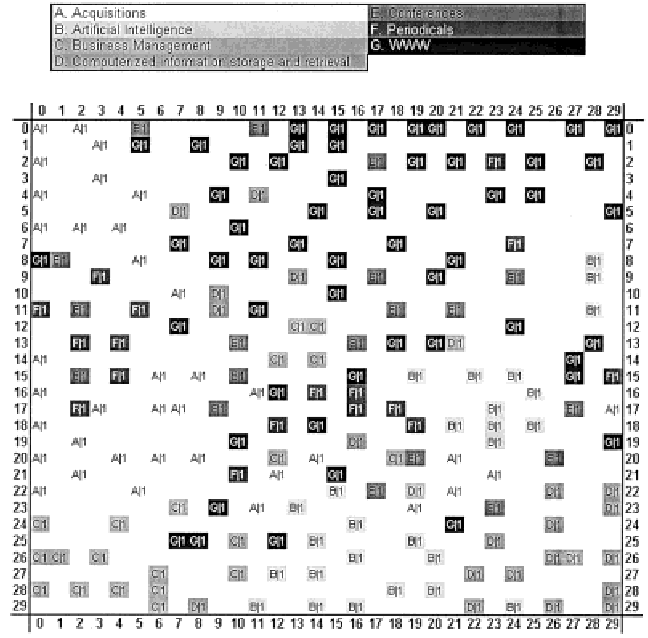

FIG. 7. Document Organization generated by a 30 × 30 network with the reduced document vectors. The nodes where the documents were classified have been marked with (redundantly) both a color and a letter associated to the subject term assigned to the documents, and with the number of documents classified at the said node.

In this present work, as in a previous work (Guerrero Bote et al., 2001), we used a reduced set of selected documents so as to be able to evaluate the result more easily, not because more documents cannot be processed. If, for evaluation, more documents are processed with a network of the same dimension, there will be a greater density of documents. Furthermore, if one does not start with a set of selected documents, the evaluation will become more difficult, because the network will find relationships that were not determined a priori and, that therefore would have to be studied case by case.

The intention of this experiment was to carry out the reduction in dimension that we have described, and to test the goodness of the coordinate transformation. Although we may state that the result of the test is satisfactory, it is not definitive, and further work is needed, especially to optimize the parameters and the fuzzification function, and to study the associated loss and the dependence on the number of components.

Acknowledgments

This work was financed by the Junta de Extremadura-Consejería de Educación Ciencia y Tecnología and the

Fondo Social Europeo, as part of research project IPR99A047.

References

Bezdek, J.C. (1981). Pattern recognition with fuzzy objective function algorithms. New York: Plenum.

Bezdek, J.C.; Chen-Kuo Tsao, E., & Pal, N.R. (1992). Fuzzy Kohonen clustering networks. Proceedings first IEEE conference on fuzzy systems (pp. 1035–1043). San Diego: IEEE.

Chen, H., Houston, A., Sewell, R., & Schatz, B. (1998). Internet browsing and searching: User evaluations of category map and concept space techniques. Journal of the American Society for Information Science, 49, 582–603.

El-Hamdouchi, A., & Willett, P. (1889). An improved algorithm for the calculation of exact term discrimination values. Information Processing and Management, 24, 17–22.

Frakes, W.B. (1992). Stemming algorithms. In W.B. Frakes & R. Baeza-Yates (Ed.), Information retrieval: Data structures & algorithms. Englewood Cliffs, NJ: Prentice Hall.

Freeman J.A., & Skapura, D.M. (1991). Neural networks: Algorithms, applications, and programming techniques. Reading, MA: Addison-Wesley.

Guerrero Bote, V.P. (1997). Redes neuronales aplicadas a las técnicas de recuperación documental. PhD Thesis, Universidad de Granada, Spain.

Guerrero Bote, V.P.; Moya Anegón, F., & Herrero Solana, V. (2001). Document organization using Kohonen's algorithm. Information Processing & Management 38, 79–89.

Honkela, T; Kaski, S.; Lagus, K., & Kohonen, T. (1996). Self-organizing maps of document collections. Alma, 1(2) [On-line]. Available: http://www.diemme.it/luigi/alma.html.

Honkela, T.; Pulkki, V., & Kohonen, T. (1995). Contextual relations of words in Grimm tales, analysed by self-organizing map. In Fogelman-Soulié, F. & Gallinari, P. (Eds.), Proceedings of international conference on artificial neural betworks (ICANN95) (pp. 3–7). Paris: EC2 et Cie.

Kantor, P.B. (1994). Information retrieval techniques. In M.E. Williams (Ed.), Annual review of information science and technology (vol. 29, pp. 53–90). Medford, NJ: Learned Information, Inc.

Kaski, S. (1999). Fast winner search for SOM-based monitoring and retrieval of high-dimensional data. Proceedings of the ninth international conference on artificial neural networks (ICANN99) (pp. 940–945). London: Institution of Electrical Engineers.

Kaski, S.; Honkela, T; Lagus, K., & Kohonen, T. (1996). Creating an order in digital libraries with self-organizing maps. Alma, 1(2) [On-line]. Available: http://www.diemme.it/luigi/alma.html.

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. Biological Cybernetics, 43, 59–69. [Reprint in Anderson, J., & Rosenfeld, E. (Eds.). Neurocomputing. M.I.T. Press, 1988.]

Kohonen, T. (1984). Self-organization and associative memory (3rd ed. 1989). Berlin: Springer Verlag.

Kohonen, T. (1995). Self-organization maps. Berlin: Springer Verlag.

Kohonen, T., Kaski, S., Lagus, K., Salojärvi, J., Honkela, J., Paatero, V., & Saarela, A. (1999). Self organization of a massive text document collection. In E. Oja & S. Kaski (Eds.), Kohonen maps (pp. 171–182). Amsterdam: Elsevier.

Kucera, H., & Francis, N. (1967). Computational analysis of present-day American English. Providence, RD: Brown University Pres.

Lagus, K., Honkela, T., Kaski, S., & Kohonen, T. (1999). WEBSOM for textual data mining. Artificial Intelligence Review, 13, 345–364.

Lagus, K., Kaski, S., Honkela, T., & Kohonen, T. (1996). Self-organizing maps of document collections: A new approach to interactive exploration. In: E. Simoundis, J. Han, & U. Fayyad (Eds.), Proceedings of the second international conference on knowledge discovery and data mining (pp. 238–243). Menlo Park, CA: AAAI Press.

Lagus, K., & Kaski, S. (1999). Keyword selection method for characterizing text document maps. Proceedings of the ninth international conference on artificial neural networks (ICANN99) (pp. 371–376). London: Institution of Electrical Engineers.

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

Moya, F., Herrero, V., & Guerrero, V. (1998). Virtual reality interface for accessing electronic information. Library and Information Research News, 22, 34–39.

Moya Anegon, F. (1994). Sistemas integrados de gestión bibliotecaria. Madrid: ANABAD.

Moya Anegón, F., Moscoso, P., Olmeda, C., Ortiz-Repiso, V., Herrero Solana, V., & Guerrero Bote, V. (1999). NeuroISOC: Un modelo de red neuronal para la representación del conocimiento. In M.J. López Huertas & J.C. Fenández Molina (Eds.), La representación y la organización del conocimiento en sus distintas perspectivas: Su influencia en la recuperación de la información. Actas del IV Congreso ISKO-España (EOCONSID'99) (pp. 151–156). Granada: ISKO-España.

Muñoz García, A. (1994). Redes neuronales para la organización automática de información en bases documentales. PhD Thesis, Universidad de Salamanca, Spain.

Noreault, T.; McGill, M., & Koll, M.B. (1981). A performance evaluation of similarity measures, document term weighting schemes and representations in a Boolean environment. In R.N. Oddy, S.E. Robertson, C.J. Van Rijsbergen, & P.W. Williams (Eds.), Information retrieval research: Papers given at the 1st Joint British Computer Society (BCS) and Association for Computing Machinery (ACM) Symposium: Research and development in information retrieval (pp. 57–76). London: Butterworths.

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

Porter M. F. (1980). An algorithm for suffix stripping. Program 14, 130–137.

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

Van der Besselaar, P., & Leydesdorff, L. (1996). Mapping change in scientific specialties: A scientometric reconstruction of the development of artificial intelligence. Journal of the American Society for Information Science, 47, 415–436.

White, H., Lin, X., & McCain, K. (1998). Two modes of automated domain analysis: Multidimensional scaling vs. Kohonen feature mapping of information science authors. In W. Mustafa el Hadi, J. Maniez, & S. PollitErgon Verlag (Eds.), Structures and relations in knowledge organization: Proceedings of the fifth international ISKO conference, 25–29 August 1998, Lille, France. Proceedings of the 5th international ISKO conference. Würzberg: Ergon Verlag.