Soft Comput (2010) 14:857–867
DOI 10.1007/s00500-009-0468-3

FOCUS

Fuzzy optimized self-organizing maps and their application to document clustering

Francisco P. Romero · Arturo Peralta ·
Andres Soto · Jose A. Olivas · Jesus Serrano-Guerrero

Published online: 25 July 2009

Abstract In this paper, an approach using fuzzy logic techniques and self-organizing maps (SOM) is presented in order to manage conceptual aspects in document clusters and to reduce the training time. In order to measure the presence degree of a concept in a document, a concept frequency formula is introduced. This formula is based on new fuzzy formulas to calculate the polysemy degree of terms and the synonymy degree between terms. In this approach, new fuzzy improvements such as automatic choice of the topology, heuristic map initialization, a fuzzy similarity measure and a keywords extraction process are used. Some experiments have been carried out in order to compare the proposed system with classic SOM approaches by means of Reuters collection. The system performance has been measured in terms of *F*-measure and training time. The experimental results show that the proposed approach generates good results with less training time compared to classic SOM techniques.

Keywords Self-organizing maps · Document clustering · Fuzzy logic

F. P. Romero (✉) · A. Peralta · J. A. Olivas ·
J. Serrano-Guerrero
Department of Information Technologies and Systems,
University of Castilla La Mancha,
Paseo de la Universidad, 4, 13071 Ciudad Real, Spain
e-mail: FranciscoP.Romero@uclm.es

A. Soto
Department of Computer Science, Universidad Autònoma del
Carmen, CP 24160 Ciudad del Carmen, Campeche, Mexico

1 Introduction

Self-organizing maps (SOMs), introduced by Kohonen (1982), are neural networks that represent the input space using the topological structure of a grid to store neighborhood relations. In contrast to other neural networks methods, its main feature is the usage of unsupervised learning for clustering tasks. This neural network has proved to be very useful in a wide range of problems, and it is considered especially suitable for clustering large high dimensional data sets such as images, documents or financial data (Cottrell and Verleysen 2006).

In document retrieval, text clustering using SOMs is widely used, mainly to arrange documents by means of a similarity measure. In Lin et al. (1991), this technique was used to classify a huge volume of technical documents based on words extracted from the title. Unfortunately, in other text clustering applications using just title terms is not adequate.

On the other hand, in order to improve the usability of the document structures obtained using SOM, the application of a hierarchical feature map is proposed (Merkl 1998). This hierarchical map establishes a conceptual document taxonomy making easier the exploration task for the user of the document collection.

Self-organizing maps can be used to organize a web document repository (e-mail, newsgroups, web pages, etc.). The most popular application in this scope is WebSOM (Lagus et al. 1999). This project is based on an automatic document organization tool in order to provide an interactive exploration of document collections.

In WebSOM, the document collection is analyzed by means of the SOM leading to a word category map (words clustered in an unsupervised manner based on the distributions of words in their contexts) (Ritter and Kohonen 1989). Besides, a document map based on document similarity is obtained. The synergy of these two components produces an efficient structure for dealing with new incoming documents.

Self-organizing maps major drawbacks are the amount of training time and resources required for training of the document map and the need to repeat the process if the number of document increases. Thus, these drawbacks can make difficult the application of SOM to document organization of large documents collections. Azcarraga and Yap (2001) proposed a system to reduce the training time of SOM using Random Mapping (Kaski 1998) for dimensionality reduction. On the other hand, the application of fuzzy logic techniques provides a mechanism which allows to improve the neural network performance. There are successful approaches such as the Fuzzy Self-organizing Maps proposed in Bezdek et al. (1992).

Another shortcoming is that the map topology has to be defined by the user. In Nürnberger and Detyniecki (2006), an approach that tries to adapt the map to the distribution of the underlying data by a growing process is developed.

The quality of the document maps depends on the document representation methods; if the semantic similarity of the documents is clearly expressed by the similarity of the document vectors, then best document maps are generated. Traditionally, a document is considered as a bag of words, once the stop words have been removed. Term frequency is a statistical measure often used in IR to evaluate the relative importance of a word in a document. According to Salton and McGill (1986), each document can be represented by a vector of term frequencies where these values could be calculated by the number of times the word appears in a document divided by the total number of words in it. The main problem of these approaches is that they only consider lexicographic aspects and do not consider the semantic relations between words (Baeza-Yates and Ribeiro-Neto 1999).

FIS-CRM (Fuzzy Interrelations and Synonymy Conceptual Representation Model) (Olivas et al. 2003) is a methodology oriented toward processing the concepts contained in any kind of document, which can be considered an extension of the vector space model (VSM), that uses the information stored in a fuzzy synonymy dictionary and fuzzy thematic ontologies (Garcés et al. 2006).

In this work, FIS-CRM approach is kept but a new version of the formulas is introduced in order to calculate the degree of synonymy between terms. Therefore, in order to evaluate conceptual aspects of documents, fuzzy measures of synonymy and polysemy are introduced, based on WordNet synsets (Miller et al. 1990). Then, concept frequency is calculated as a statistical measure of the relative importance of a concept or meaning in a document.

The paper is structured in the following way. In Sect. 2 a description of the architecture and the learning process of SOM is provided. In Sect. 3, the proposed improvements including the fuzzy re-presentation method, the heuristic initialization process and the fuzzy similarity function are described. In Sect. 4, the test document collection used for the experiments is introduced and the results from the clustering process are presented. A software implementation of this approach for monitoring the training process is also presented. In Sect. 5 conclusions and future works are pointed out.

2 Self-organizing maps

Self-organizing maps are one of the most popular neural network models. They are especially suitable for high-dimensional data visualization and modeling. The idea is to create a neural network with the purpose of representing the high-dimensional input space using a low-dimensional topologic structure. The data proximity relationships are preserved in this structure (Kohonen 1998).

The network structure has two layers of neurons (see Fig. 1). The neurons of the input layer correspond to the dimension of the input vector. The output layer contains as many neurons as clusters are needed. These neurons are constrained to a regular grid that usually is two-dimensional. All neurons in the input layer are connected to all neurons in the output layer. The weights (w_{ij}) of the connection between the input and output layer of the neural network encode the different positions in the high-dimensional data space.

This neural network is self-organizing. The maps are trained in an unsupervised way using a competitive learning scheme. In the output layer, the neurons compete among themselves to determine which one will be activated according to the corresponding input. In contrast with most of the other neuronal networks, this method needs no classification information for any sample vector.

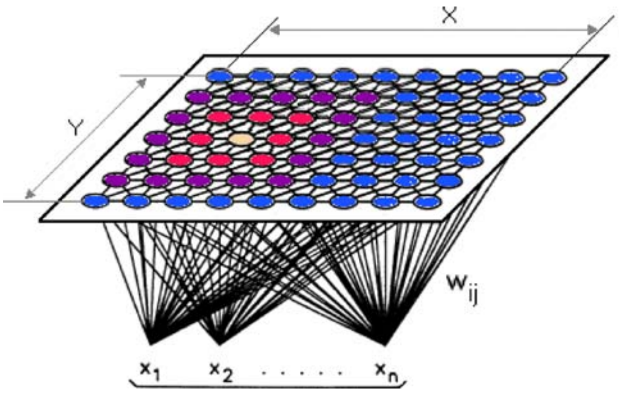


Fig. 1 Structure of a rectangular self-organizing map

2.1 Architecture of SOM

Self-organizing maps architecture has the following elements and parameters:

- *Neuron layers* Self-organizing maps consist of an input layer with the same dimension than the input vector. The output or competition layer contains series of neurons disposed in a two-dimensional lattice. Each neuron from the competition layer is connected to all neurons from the input layer through the synaptic weight vector.
- *Similarity measure* It is a mechanism which permits to calculate the matching degree between each neuron in the competition layer and an input vector. For this purpose similarity as well as distance measurements are employed, as for example, the Euclidean distance.
- *Winner neuron choice* It is a mechanism based on the similarity measure which permits to choose the winner neuron, the one which is the most similar to the input vector.
- *Neighborhood function* In order to preserve the neighborhood relations, the neurons that are closer to the winner neuron are modified to a larger extent than those which are farther away. Therefore, the neighbor neurons will offer similar answers because they will have vectors presenting smooth changes. The neighborhood function is generally taken as a decreasing function of the distance between the nodes in the map grid, usually a Gaussian function.
- *Learning rate* In each training step, it is necessary to update the synaptic weight vector according to the winner neuron and its neighbors. The learning rate reflects the speed of change. The learning rate must decrease monotonically with time. Therefore, this rate should have a smaller value at the end than at the beginning of the process in order to avoid that the map becomes unstable.

2.2 Learning process

The goal of the learning process algorithm is to adapt iteratively the neuron weights in order to obtain the final neurons that represent clusters of data vectors. The basic SOM learning algorithm is organized in the following steps:

1. *Initialization* Initially, it is necessary to establish the features of the regular low-dimensional grid (2D lattice). Then, the synaptic weight vectors of the SOM are initialized with random values.
2. *Training* The training process is iterative. On each step a single element v_i of the training set is considered.
3. *Choosing the winner neuron* The distance between the input vector and the synaptic reference vectors is calculated and the winner neuron is chosen.
   (a) *Update parameters* The synaptic reference vector of the winner neuron (w) and its neighbors are updated according to Eq. 1. This formula take into account the learning rate (l) and the radius of the neighborhood function (r_{nf}).

$$w_k(t+1) = w_k(t) + l r_{nf}(w, w_k)(\|v_i - w_k(t)\|) \tag{1}$$

   (b) *Iteration* Back to the training step for a fixed number of iterations or until the solutions can be regarded as steady.

2.3 Fuzzy self-organizing maps

Neuro-fuzzy systems combine the theory of neural networks and fuzzy sets. There are several approaches that introduce Fuzzy Logic in SOM with the aim of improving the algorithm and solving its drawbacks. These systems are named fuzzy self-organizing maps (Vuorimaa 1994).

Most of the known projects include a combination of SOM with the fuzzy c-means algorithm (Bezdek et al. 1992). In the work of Huntsberger and Ajjimarangsee (1992) it is proposed that the learning rate of SOM is treated as a fuzzy membership value of the current input sample in the neuron's output class. In this case, the fuzzy membership values have been computed using the fuzzy c-means algorithm.

Another approach is proposed in Kong and Kosko (1992), where the input and output spaces are first divided into overlapping fuzzy sets. After that, the best relations between the input and output fuzzy sets, expressed as fuzzy rules, are calculated according to certain predefined learning laws. This method has a draw-back. The accuracy of the fuzzy model is limited by the initial partitioning of the fuzzy sets.

In Pascual-Marqui et al. (2001) a different version of SOM, not based on the Kohonen approach, is proposed. This approach consists of a modified version of the fuzzy c-means clustering algorithm, where the code vectors are distributed on a regular low-dimensional grid. The learning method is based on the optimization of well-defined cost functions.

There exist other approaches less intrusive. In Mitra and Pal (1994), where the input is modified in order to accept linguistic representations of the original crisp input values. The output of this approach will provide fuzzy descriptions by membership functions. Therefore, the system's information is in a human understandable form and the user can easily verify and even modify the information, if it is necessary. In contrast to previous works, the learning process is not modified.

3 Fuzzy optimization of SOM to document clustering

In our approach, some fuzzy logic techniques are used in order to solve the drawbacks presented by the classic SOM algorithm. The aim has been, at first, to reduce the training time (iterations and epochs) and at last, to improve the quality of the results. Therefore, several steps of the learning process have been modified.

- Using a process for the semantic document indexing in order to build the input vectors according to the document contents.
- Choosing the number of neurons and the 2D lattice of the output layer according to the collection features.
- Heuristic initialization of the output layer with the purpose of obtaining meaningful prototype clusters to reduce the learning stages.
- Some improvements in the learning process: the use of a fuzzy similarity measure and changes in the neighborhood functions and learning rate.
- Cluster representation. An advanced representation of each neuron using keywords and fuzzy sets is used.

3.1 Semantic document indexing

In order to apply SOM to document clustering, it is necessary to represent the document using the VSM. In this model each document is represented by a numerical feature vector where the components are determined by the weight of a term of the document collection. The weights reflect the importance of a term in a specific document of the considered collection. In the classic TF-IDF approach the weight is obtained through the term frequency. This approach cannot reflect the semantic content of a document (Han 2005) precisely because the semantic relationships among words are not considered.

In order to solve this problem, in this work, a fuzzy semantic indexing model based on WordNet synsets (Fellbaum 1998) is used. In WordNet, nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each one expressing a different concept. Synsets are interlinked by means of conceptual-semantic and lexical relations. Every synset contains a group of synonymous words or collocations (a collocation is a sequence of words that go together to form a specific meaning, such as "car pool"); different senses of a word are in different synsets. Gonzalo et al. (1998) had shown how the use of WordNet synsets as input vectors can improve the performance of the information retrieval tasks. The use of these synsets allows relating different words with the same meaning. Therefore, a content logical representation using the fuzzy synonymy relationship between words is obtained. This fuzzy synonymy will be understood as a gradual fuzzy relation between terms as in Fernandez et al. (2002), which is closer to its behavior in dictionaries, where it is possible to find cases in which equivalence relation does not hold. For example, *auto* and *automobile* share a common meaning: *"a motor vehicle with four wheels; usually propelled by an internal combustion engine"* (Fellbaum 1998). But *automobile* has another meaning: as a verb, it means *"to travel in an automobile"*. Therefore, *auto* and *automobile* are not identical or equivalent, but similar.

The degree of synonymy between words is closely related to their polysemy level. Polysemy means the ambiguity of a word that can be used (in different contexts) to express two or more different meanings. Words with only one meaning are considered strong as long as words with several meanings are considered weak. For example, *auto* is a strong word, while *automobile* is weaker than auto, and *car* is even weaker because it has five meanings according to Fellbaum (1998).

Due to the occurrence of weak terms it is important to identify the exact meaning of each weak word. Term meaning disambiguation is achieved through term context and its WordNet semantic relationships.

In this paper, new formulas (Soto et al. 2008) are used to deal with synonymy and polysemy, which are based on those ones developed in FIS-CRM. With these new fuzzy formulas, the whole process of concept matching is simplified. As in FIS-CRM, although a certain term does not appear in a document, its presence could be estimated from its degree of synonymy based on terms that do appear in the document. To measure the degree of presence of a concept in a document, a concept frequency formula is used. Using the concept frequency coefficient, it is possible to measure how similar are two or more documents depending on their use of some concepts.

Let V be a terms set which belong to a particular dictionary, and M the set of meanings (i.e., synsets) associated to the terms in V. Therefore, each term in V has one or more meanings in M. According to WordNet, *auto* has only one meaning, while *automobile* has two. On the other hand, each meaning in M has one or more terms associated in V.

Let us define a fuzzy relation S between two terms t_1, t_2 in V such that $S(t_1, t_2)$ expresses the degree of synonymy between the two terms. $M(t)$ denotes the set of different meanings associated with a certain term t.

$$S(t_1, t_2) = \frac{|M(t_1) \cap M(t_2)|}{|M(t_1)|} \tag{2}$$

In this way, the synonymy degree between *auto* and *automobile* is 1, which means that the concept *auto* totally corresponds with the concept *automobile*. But, in the other way, the degree of synonymy between *automobile* and *auto* is just 0.5 because *automobile* corresponds with *auto* just in half of the meanings.

In order to measure the weakness of a term, an index $I_p(t)$ is defined to represent the polysemy degree of the term t where $I_p(\mathrm{auto}) = 0, I_p(\mathrm{automobile}) = 0.5$ and $I_p(\mathrm{car}) = 0.87$.

$$I_p : V \to [0, 1] \quad \text{where } I_p = 1 - \frac{1}{|M(t)|} \tag{3}$$

Furthermore, based on this degree of polysemy, concept frequency measure Cf of the meaning m ($Cf(m)$) is introduced:

$$Cf(m) = \sum_{t_i \in T(m)} \left(n_{ij}\left(1 - I_p(t_i)\right)\right) \tag{4}$$

where n_{ij} is the number of occurrences of term t_i in the document d_j and $T(m)$ the set of different terms associated with the meaning m. The final formula is shown in the Eq. 5:

$$Cf(m) = \sum_{t_i \in T(m)} \left(\frac{n_{ij}}{|M(t_i)|}\right) \tag{5}$$

In this way, it is easy to compare the relative importance of different meanings in a document d_j. This measure is equivalent to term frequency which is one of the most referenced in IR, but for meanings. It is not just lexical but includes some semantic interpretation of the terms. The relative concept frequency measure Cfr of a meaning m with respect to a document will be calculated by Eq. 6. This formula will be used to estimate the similarity between documents after the disambiguation process.

$$Cfr(m) = \frac{Cf(m)}{\max_{m \in M}(Cf(m))} \tag{6}$$

Before synset identification it is necessary to apply some document preprocessing methods such as stop words filtering and stemming. The aim is to reduce the vector dimensions and to eliminate meaningless words. When a word is found in a text, WordNet is asked for the meanings. The answer for this question is a list of all the word synsets. Each synset has its corresponding entry in the vector that represents the document. The relevance of each entry is calculated using the concept frequency formula shown in Eq. 4. If another word with the same meaning appears in the document, the weights associated with the meaning would be increased. This fact is known as meaning promotion.

An example of the process of meaning promotion is shown in Table 1. The nouns *cry*, *yell* and *shout* co-occur in a document and share the meanings *07022785*, *07023418*. In this example n_{ij} is assumed to be 1 for all i.

The performance will be worse if all possible senses for every word form are considered. Therefore, it is necessary to execute a process in order to identify the relevant meanings that appear in each document. The automatic word sense disambiguation process has particular importance because the number of elements in a vector can be too large if all the meanings for each word are considered. Not all the meanings in the document are relevant, therefore, useless meanings should not be considered. The disambiguation process can reduce drastically the features of document vectors and can also improve the relevance of the clustering results.

The automatic word sense disambiguation process is based on WordNet. The process consists of removing the useless meanings using the following criteria:

- *First criterion* two words in the same paragraph are related using the main semantic WordNet relations: synonym, hypernymy, meronymy. The not related meanings of these words will be removed.
- *Second criterion* two words in the same paragraph are also related if they share a WordNet meaning (i.e., a synset) that is promoted. In this case, the not shared meanings are removed.

For example, in Table 2, two documents (d_1, d_2) with two words each are considered: ({*machine, car*}; {*car, gondola*}). One of the words, car, is shared between both documents.

The meanings associated to each word are indicated in Table 2. The number of meanings is considered in order to build the document vectors. Each document vector has 12 positions (Table 2).

The common word (*car*) is related with the others. In Table 3 the useless meanings has been removed using the previously explained second criterion. Each vector has only two positions and each meaning has been promoted to a frequency value equal to 2.

Therefore, useless meanings have been rejected and the weight of the selected meanings have been increased reducing the impact of its degree of polysemy.

At last, in order to obtain a good clustering performance, it is necessary to reduce the number of synsets in the vector description. A simple method based on the number of synsets is applied to achieve this aim. Thus, any synset that occurs in less than 2% of the documents or in more than 33% is eliminated. Finally, the vectors are normalized using Eq. 6.

Table 1 Example of meaning promotion

	07022785	07023418	07054564	07281641
cry	X	X	X	X
yell	X	X		
shout	X			
Cf	1.75	0.75	0.25	0.25
	0.25 + 0.5 + 1	0.25 + 0.5		

Table 2 Example of meaning distribution and promotion

Meanings	Words			Docs.	
	machine	car	gondola	d_1	d_2
03659195	X			0.16	0.00
08151167	X			0.16	0.00
10125307	X			0.16	0.00
02929975	X	X		0.36	0.20
08150991	X			0.16	0.00
03660165	X			0.16	0.00
02932115		X	X	0.20	0.53
02931966		X		0.20	0.20
02931574		X		0.20	0.20
02906118		X		0.20	0.20
03410940			X	0.00	0.33
03410794			X	0.00	0.33

Table 3 Meanings distribution and promotion via word sense disambiguation process

Meanings	Words			Docs.	
	machine	car	gondola	d_1	d_2
02929975	X	X		2	0
02932115		X	X	0	2

3.2 Building the document map

After the document collection has been processed as described above, a synset vector for every document in the collection is generated. These document vectors are then clustered and arranged into a SOM through the learning process.

The application of SOM in document clustering has one shortcoming: the clustering result is highly dependent on the user-defined parameters, i.e., the number of clusters, the map shape and the initial centroid seeds. An improper value of these parameters leads to a poor clustering. Besides, the old ineffective principle of random model initialization does not guarantee optimum results (Kohonen 1998).

In order to obtain these values, in first place a fuzzy hierarchical clustering over a document subset chosen in a random manner will be done. The employed algorithm is a combination of the techniques employed in Romero et al. (2006) and the ones expressed in Wallace et al. (2003).

On the other hand, the topology definition is based on the calculation of two parameters: the number of clusters and the shape of the map. In the most general case, a 2D map can have rectangular or polygonal shape. The map shape will be chosen using the number of clusters obtained in the previous process. At first, it is necessary to take into consideration the overlap degree amongst the different clusters obtained. This degree is calculated employing the fuzzy Jaccard coefficient (Bordogna et al. 2006).

Three levels of overlapping are defined: low, medium and high. If the overlapping level is low, a linear topology will be chosen. On the other hand, if the level of overlapping is medium the topology will be rectangular, if the overlapping level is high and the cluster number allows it, the topology will be hexagonal.

3.3 Improving the learning process

During the training process, for each step, the similarity between a document and each neuron has to be evaluated in order to choose the winning neuron (n_{win}). For each document the winner neuron n_{win} is the one with the maximum similarity value.

In the VSM, the cosine similarity is one of the most commonly used methods to estimate document similarity. This similarity is based on the angle between two vectors that represent the documents (Baeza-Yates and Ribeiro-Neto 1999).

In this work, a fuzzy similarity function between documents and neurons has been defined taking into account the conceptual interpretation of the vectors and the Jaccard coefficient generalized to the fuzzy sets context (Miyamoto 1990) (see Eq. 7):

$$\text{sim}(d, n_j) = \frac{\sum_{m\in d} Cfr(m, n_j) \otimes Cfr_d(m)}{\sum_{m\in d} Cfr(m, n_j) \oplus Cfr_d(m)} \tag{7}$$

where $Cfr(m, n_j)$ is the weight of the meaning M in the neuron j, $Cfr_d(m)$ is the relative conceptual frequency value of meaning M in document d, and $\otimes$ and $\oplus$ denote the fuzzy conjunction (t-norm) and disjunction operators (t-conorm), respectively. The use of the algebraic product as t-norm and the algebraic sum as t-conorm allows us to obtain better results than applying classical functions.

When the winner neuron is selected, this neuron and its neighborhood change their values. After this modification, these neurons move nearer toward the input sample. The most widely used neighborhood function is the Gaussian, which is symmetric about the maximum point defined by the winner neuron and decreases monotonically to zero. The spatial width of adaptation is guided by means of the time-varying parameter σ:

$$h_{c(x),i} = \alpha(t)\exp\left(\frac{d_{ji}^2}{2\sigma^2}\right) \tag{8}$$

where $h_{c(x),i}$ denotes the neighborhood of the winning neuron, the subscript $c(x)$ represents the condition to choice the winner neuron i (i.e., the neuron that matches best with the sample), $\alpha(t)$ is the learning rate ($0 < \alpha(t) < 1$) that decreases monotonically with the regression steps; d_{ji} is the distance between the neuron j and the winner neuron i in the display grid, and σ corresponds to the width of the neighborhood function, which also decreases monotonically with the regression steps.

The proper choice of the learning rate is somewhat critical. Too small values lead to unnecessarily many iteration steps, while too large values can lead to divergence. The decrease of the learning rate and the neighborhood range are not the same as the setting of a standard SOM implementation whereas the difference lies in the efficient construction of the feature space. If the SOM is approximately organized at the beginning, it is necessary to start with a narrower neighborhood function and smaller learning-rate factor.

Different speeds of change of the learning rate are shown in Fig. 2. The vertical axis represents the learning rate and the horizontal axis represents the percentage of stimulus. The behavior number 1 presents an equal sized proportional reduction with step size equal to 10%. The behavior number 2 presents a faster reduction during the initial part of the learning process than the behavior 1. Behavior number 3 presents a faster drop in the first third of the graph than the behavior 2; but after that it converges to 0.1. We have obtained the best experimental results using the third approach. Thus, it was selected for our approach. The use of this behavior of the learning rate produces better experimental results and a reduction of the network training time, because the main modifications just occur in a reduced percent of the documents.

3.4 Labeling method

In information retrieval, a document may be related to more than one cluster with different membership degree. Thus, the resulting clusters can be considered as fuzzy sets, where each document has a membership degree (obtained from an average similarity degree) with respect to each one of these clusters. Therefore, in order to improve the quality of the results, a specific fuzzification method is used. The algorithm described in Wallace et al. (2003) is used, with several changes to fit the method for finding features. In this way clusters' cardinalities will be modified using a fuzzy classifier based on the main document features.

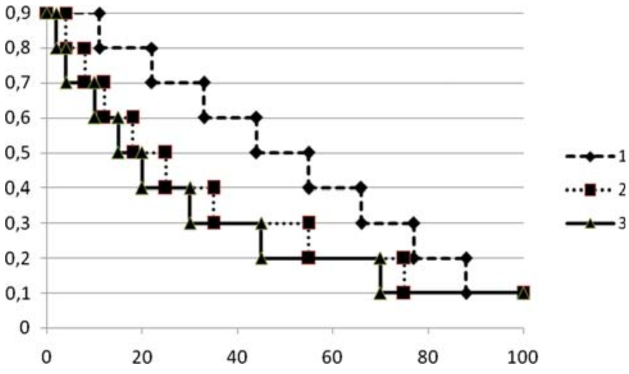


Fig. 2 Learning rate evolution

One of the most important shortcomings of SOM is the absence of a labeling method for the obtained clusters. In this approach, the map is labeled with automatically identified descriptive words that represent the concepts within the document clusters.

A fuzzy relevance degree of each meaning within a cluster is calculated using the following formula:

$$w_i^C = \sum_{j \in C} w_{ij} \times \left(1 + \frac{\mathrm{docs}(i, C)}{|C|}\right) \times Ln\left(\frac{|cl|}{cl(i)} + 1\right) \qquad (9)$$

where

- $\mathrm{docs}(i, C)$: number of documents in the cluster C where the meaning has a membership degree greater than 0.
- $|cl|$: total number of clusters.
- $cl(i)$: number of clusters in which the meaning i has a membership degree greater than 0.
- w_{ij}: the membership degree of the meaning i in the document j of the cluster C.

Using this relevance coefficient, the more relevant concepts are selected as neuron's label. Therefore, after the map training, each grid cell is labeled by several keywords that describe the majority of the documents associated with this cell.

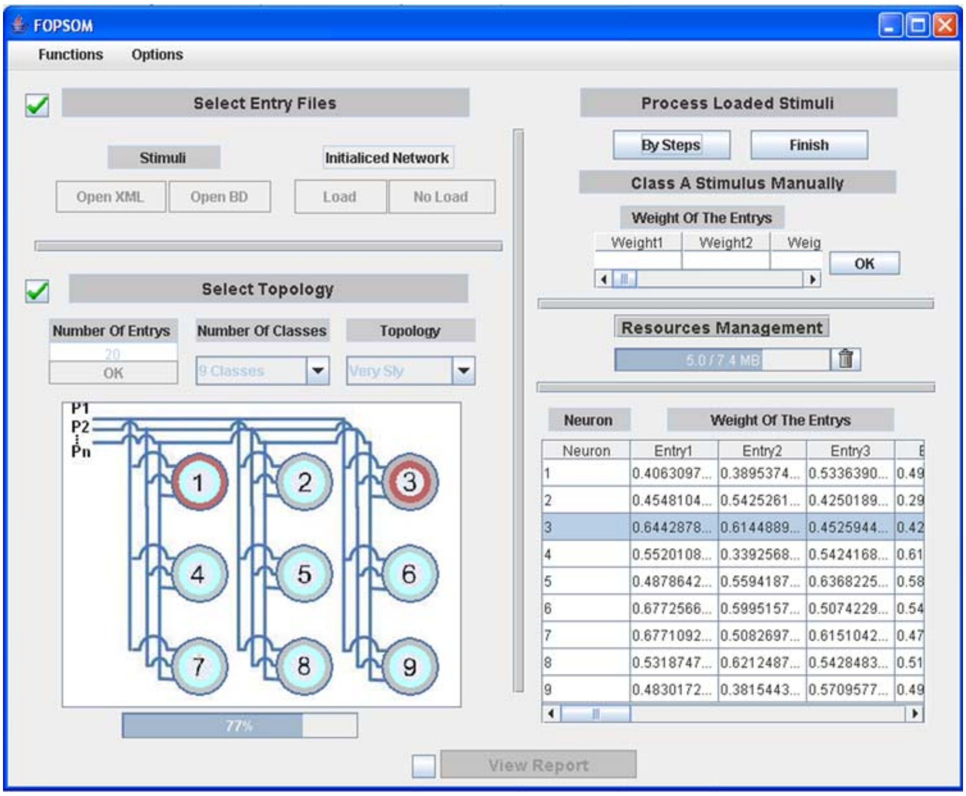
4 Experiments

In this section, the performance of the fuzzy optimized SOM is evaluated in terms of the accuracy of clustering.

To assess the usability of this approach a software prototype was developed in Java. The tool provides an interface in order to monitor the learning process. Once preprocessed and learned, the representations of documents and maps are stored in XML files. Furthermore, the document map can be used for the visualization of the conceptual structure. A screenshot of the software tool and its components is shown in Fig. 3. The components are the following:

- The topology (left bottom).
- The neuron values (right bottom).
- The user interaction interface (top right and left)

Fig. 3 Screenshot of the software tool



Number Of Entrys: 20 OK

Number Of Classes: 9 Classes

Topology: Very Sly

Neuron	Entry1	Entry2	Entry3	E
1	0.4063097...	0.3895374...	0.5336390...	0.49
2	0.4548104...	0.5425261...	0.4250189...	0.29
3	0.6442878...	0.6144889...	0.4525944...	0.42
4	0.5520108...	0.3392568...	0.5424168...	0.61
5	0.4878642...	0.5594187...	0.6368225...	0.58
6	0.6772566...	0.5995157...	0.5074229...	0.54
7	0.6771092...	0.5082697...	0.6151042...	0.47
8	0.5318747...	0.6212487...	0.5428483...	0.51
9	0.4830172...	0.3815443...	0.5709577...	0.49

4.1 Test collection

A pre-classified set of documents is necessary in order to evaluate the effectiveness of the fuzzy optimized self-organized maps in document clustering problems.

In this paper, the Reuters-21578 text categorization test collection Distribution 1.0[1] is used. This test data set consists of 21,578 articles from the Reuters news service in the year 1987. Only 12,902 articles had been assigned to 118 categories and each story is assigned to one or several categories. Several categories are not relevant. For instance, one category ("earn") consists of about 4,000 documents while many other categories have less than ten documents. Therefore, the nine most relevant defined categories have been selected, which consist of about 75% of documents.

Our training set has been obtained from the "Mod-Apte" (Apté et al. 1994) split criterion. 6,015 articles without errors have been used to train and evaluate the self-organized map. In Table 4 the distribution of these articles into the nine overlapped categories is shown.

4.2 Measures

In the context of document clustering, the ability of an algorithm to classify a document into one or several categories is very interesting for the user. In classical information retrieval this ability is usually measured by comparing the clusters produced by the algorithm to known categories (Steinbach et al. 2000).

In order to obtain the precision and recall values of the cluster i respect the category j (Van Rijsbergen 1979), the following parameters have been considered:

- n_i: number of documents of the cluster i.
- n_j: number of documents of the category j.
- n_{ij}: number of documents of the cluster i in the category j.

Then, precision and recall can be defined by Eqs. 10 and 11, respectively:

$$p_{ij} = \frac{n_{ij}}{n_i} \tag{10}$$

$$r_{ij} = \frac{n_{ij}}{n_j} \tag{11}$$

The precision and recall measures are related in an inverse way, the higher the level of precision, the lower the level of recall and vice versa. Ideally, a system should achieve a high precision at high recall levels, but naturally

[1] http://www.daviddlewis.com/resources/testcollections.

Table 4 Reuters documents distribution

Category	Training docs
earn	2,709
acq	1,488
money-fx	460
grain	394
trade	337
crude	349
interest	289
ship	191
coffee	110

there exists a trade-off between both: as the recall rises, precision tends to get lower. F-measure (Van Rijsbergen 1979) has been used to solve this problem. This measure is a harmonic mean of the precision and recall values.

The F-measure represents the accuracy of clustering. A higher value of the F-measure means high quality level.

$$F(i,j) = \frac{2 \times r_{ij} \times p_{ij}}{r_{ij} + p_{ij}} \tag{12}$$

Then, the F-measure of the whole clustering results is defined as the weighted average among the different categories:

$$F = \sum_i \left(\frac{n_i}{N} \max_j (F(i,j)) \right) \tag{13}$$

where n_i are the number of documents of the cluster i, N is the total number of documents and $F(i,j)$ is the value of F-measure of the category j in the cluster i.

4.3 Results

The maximum precision, maximum recall and maximum F measure obtained for each cluster after the experiment and the corresponding aggregated metrics are shown in Table 5. The second column (%Els.) represents the percentage of elements that belongs to each neuron.

Resulting values are good enough to remark the system performance. All the improvements already included in this version of the SOM algorithm were oriented to improve the recall values of the results. This way, better quality results, with higher values of F-measure, have been obtained.

By comparing with other SOM approximations, it could be observed that the proposed modifications improved the quality of the results. Table 6 and Fig. 4 show the results obtained after applying different variations of SOM algorithm:

- *Basic SOM* based on TF-IDF.
- *SOM+* Basic SOM applied to the vectors obtained by the proposed model.

Table 5 Reuters experimental results

Neuron	%Els.	Max. prec.	Max. recall	Max. F
1	13.47	0.46	0.91	0.61
2	0.09	0.50	0.01	0.01
3	12.31	0.45	0.93	0.61
4	0.56	0.62	0.06	0.11
5	0.00	0.00	0.00	0.00
6	3.63	1.00	0.08	0.15
7	29.07	0.84	0.89	0.87
8	0.47	0.91	0.01	0.02
9	40.40	0.90	0.82	0.86
Total	100.00	0.77	0.83	0.76

- *SOM++* SOM+ with the improvements already mentioned (learning rate, similarity function, etc.)

Following, our SOM approach is compared with the two modified clustering algorithms:

- *BBK* (Bisecting k-means using background knowledge) (Hotho et al. 2003): This algorithm uses the VSM and enhances the text representation by adding synonyms and up to five levels of hypernyms for each noun based on the document context using WordNet as ontology.
- *CFWMS* (Clustering based on Frequent Word Meaning Sequences) (Li et al. 2008): This algorithm uses

Table 6 Reuters experimental results

Algorithm	F-measure
Basic SOM	0.53
SOM+	0.46
CFWS	0.57
BBK	0.46
SOM++	0.76

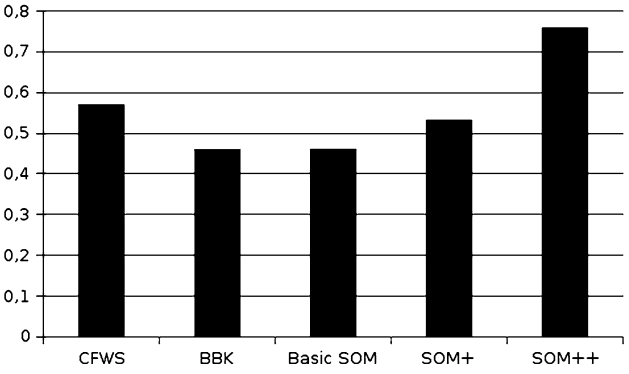


Fig. 4 Comparison between different clustering algorithms

frequent word meaning sequences to measure the similarity between documents.

The performances of our SOM algorithm, CFWMS and BBK are compared in Table 6 and Fig. 4. Our algorithm applied to the Reuters data-set has a better performance than those two algorithms.

5 Conclusions and future work

Self-organizing maps are a valuable tool for document retrieval purposes on the web. In this article an optimization of SOM for document clustering has been presented. The proposed fuzzy optimized SOM for document clustering showed to be an effective and fast alternative to traditional SOM applied to document organization.

The proposed method takes into account word meanings in spite of clustering by term frequency. A new model to calculate the degree of synonymy between terms in order to cope with their meanings has been applied. Several new fuzzy formulas have been introduced to calculate the polysemy degree between words in order to deal with their meanings. By these formulas, it is possible not only to measure the frequency of use of terms in documents, but to measure the frequency of use of meanings or concepts in documents. A coefficient, denoted concept frequency coefficient, was introduced in order to quantify the presence of a concept in a document. With the concept frequency coefficient, it is possible to measure how similar are two or more documents depending on their use of same concepts.

Usually the standard model has to be retrained several times until an appropriate size is found. In contrast to the standard model, a heuristic initialization of the neural network is proposed. This way, the system automatically selects the map topology and initializes the nodes. Therefore, less training time and a better result quality is achieved. The proposed method is based on a fuzzy formula of similarity between documents based on the concepts that appear in those documents. Some improvements of the learning rate and the neighborhood function are also used. The post-processing of the SOM's results provides a true document organization based on concepts, and a hierarchy that allows the user to easily inspect the document collection.

The experiments demonstrate that the quality and effectiveness of clustering using this method is better than the other approaches using TF-IDF representation, classic similarity or neighborhood functions. Besides, the approach is computationally efficient allowing a fast training.

The present approach has also some shortcomings. There is a lack of accuracy processing short documents. In these documents, there is not enough information to obtain an appropriate treatment. Further work should be focused on the integration of information filtering techniques, user profiles and user feedback in order to solve this problem. The use of advanced visualization techniques could also improve the usability of the obtained results.

Another problem is the evolution of the document map when some new documents come into the repository. It is necessary to define a method to update incrementally the map (Bouchachia and Mittermeir 2006). The use of growing SOM (Nürnberger and Detyniecki 2006) could be an interesting approach.

Future work could be also focused on the use of techniques to manage multilingual document collections. A possible approach could be to build a multilingual index based on meanings using EuroWordNet. In this approach, a component called InterLingual Index (ILI) (Ellman 2003) could be used to get the equivalent meanings between different languages in the step of conceptual document representation. Another approach to represent semantically a document could be the use of the Universal Networking Language (UNL). The UNL method is a proposed multilingual semantic representation for sentences (Uchida et al. 1995).

The disambiguation process could be addressed by using other techniques. The exploitation of other relations (specialization, context, instrument, part, patient, location and agent) and the study of their influences in the sense disambiguation model could improve the quality of results in some aspects.

Future investigations and experiments should consider the applications of this clustering method on semi-structured documents such as XML documents.

It would be also interesting to take into account multimedia documents (e.g., images, videos and sounds) and user profiles (Lazzerini and Marcelloni 2007).

Acknowledgments This research has been partially supported by TIN2007-67494 F-META project, MEC-FEDER, (Spain) and PEIC09-0196-3018 SCAIWEB-2 excellence project of Autonomous Government of Castilla-La Mancha (Spain).

References

Apté C, Damerau F, Weiss SM (1994) Automated learning of decision rules for text categorization. ACM Trans Inf Syst 12(3):233–251

Azcarraga AP, Yap TN (2001) Som-based methodology for building large text archives. In: Proceedings of the 7th international conference on database systems for advanced applications, pp 66–73. IEEE Computer Society, Washington

Baeza-Yates R, Ribeiro-Neto B (1999) Modern information retrieval. ACM Press, New York

Bezdek JC, Tsao EC, Pal NR (1992) Fuzzy kohonen clustering networks. In: Proceedings of the IEEE international conference on fuzzy systems, pp 1035–1043

Bordogna G, Pagani M, Pasi G (2006) A dynamical hierarchical fuzzy clustering algorithm for document filtering. Stud Fuzziness Soft Comput 197:1–23

Bouchachia A, Mittermeir R (2006) Towards incremental fuzzy classifiers. Soft Comput 11(2):193–207

Cottrell M, Verleysen M (2006) Advances in self-organizing maps. Neural Netw 19(6):721–722

Ellman J (2003) Eurowordnet: a multilingual database with lexical semantic networks. Nat Lang Eng 9(4):427–430

Fellbaum C (1998) WordNet: an electronic lexical database. MIT Press, Cambridge

Fernandez S, Grana J, Sobrino A (2002) A spanish e-dictionary of synonyms as a fuzzy tool for information retrieval. In: Actas de las I Jornadas de Tratamiento y Recuperacion de Informacion

Garcés P, Olivas J, Romero F (2006) Concept-matching IR systems versus word-matching information retrieval systems: considering fuzzy interrelations for indexing web pages. J Am Soc Inf Sci Technol 57(4):564–576

Gonzalo J, Verdejo F, Chugur I (1998) Indexing with wordnet synsets can improve text retrieval. In: Proceedings of the COLING/ACL'98 Workshop on Usage of WordNet for NLP, pp 38–44

Han J (2005) Data Mining: concepts and techniques. Morgan Kaufmann, San Francisco

Hotho A, Staab S, Stumme G (2003) Ontologies improve text document clustering. In: Proceedings of the third IEEE international conference on data mining, pp 541–544. IEEE Press, Washington DC

Huntsberger T, Ajjimarangsee P (1992) Parallel self-organizing feature maps for unsupervised pattern recognition. In: Fuzzy Models for Pattern Recognition, pp 483–495

Kaski S (1998) Dimensionality reduction by random mapping: fast similarity computation for clustering. In: Proceedings international join conference on neural networks, vol 1, pp 413–418

Kohonen T (1982) Self-organized formation of topologically correct feature maps. Biol Cybern 43:59–69

Kohonen T (1998) The self-organizing map. Neurocomputing 21:1–6

Kong S, Kosko B (1992) Adaptive fuzzy system for backing up a truck-and-trailer. IEEE Trans Neural Netw 3:211–223

Lagus K, Honkela T, Kaski S, Kohonen T (1999) Websom for textual data mining. Artif Intell Rev 13(5–6):345–364

Lazzerini B, Marcelloni F (2007) A hierarchical fuzzy clustering-based system to create user profiles. Soft Comput 11:157–168

Li Y, Chung SM, Holt JD (2008) Text document clustering based on frequent word meaning sequences. Data Knowl Eng 64(1): 381–404

Lin X, Soergel D, Marchionini G (1991) A self-organizing semantic map for information retrieval. In: Proceedings of the 14th annual international ACM SIGIR conference, pp 262–269. ACM, New York

Merkl D (1998) Text classification with self-organizing maps: some lessons learned. Neurocomputing 21:68–77

Miller GA, Beckwith R, Fellbaum C et al (1990) Introduction to wordnet: an on-line lexical database. Int J Lexicogr 3(4): 235–244

Mitra S, Pal SK (1994) Self-organizing neural network as a fuzzy classifier. IEEE Trans Syst Man Cybern 24(3):385–399

Miyamoto S (1990) Fuzzy sets in information retrieval and cluster analysis. Kluwer, Dordrecht

Nürnberger A, Detyniecki M (2006) Externally growing self-organizing maps and its application to e-mail database visualization and exploration. Appl Soft Comput 6(4):357–371

Olivas JA, Garcés PJ, Romero FP (2003) An application of the fis-crm model to the fiss metasearcher: using fuzzy synonymy and fuzzy generality for representing concepts in documents. Int J Approx Reason 34:201–209

Pascual-Marqui RD, Pascual-Montano AD, Kochi K, Carazo JM (2001) Smoothly distributed fuzzy c-means: a new self-organizing map. Pattern Recognit 34:2395–2402

Ritter H, Kohonen T (1989) Self-organizing semantic maps. Biol Cybern 61:241–254

Romero FP, Olivas JA, Garcés PJ (2006) A soft approach to hybrid models for document clustering. Proc Inform Process Manag Uncertain Knowl Based Syst 1:1040–1045

Salton G, McGill MJ (1986) Introduction to modern information retrieval. McGraw-Hill, New York

Soto A, Olivas JA, Prieto M (2008) Fuzzy approach of synonymy and polysemy for information retrieval. Stud Fuzziness Soft Comput 224:179–198

Steinbach M, Karypis G, Kumara V (2000) A comparison of document clustering techniques. In: Proceedings of the knowledge discovery on databases, pp 3–7

Uchida H, Zhu M, Della ST (1995) UNL: a gift for a millennium. The United Nations University, Tokyo

Van Rijsbergen C (1979) Information retrieval. Butterworth, London

Vuorimaa P (1994) Fuzzy self-organizing map. Fuzzy Sets Syst 66:223–231

Wallace M, Akrivas G, Stamou G (2003) Automatic thematic categorization of documents using a fuzzy taxonomy and fuzzy hierarchical clustering. In: Proceedings of the 12th IEEE international conference on fuzzy systems, vol 2, pp 1446–1451