An Intelligent Tool for Building E-Learning Contend-Material Using Natural Language in Digital Libraries

A. Drigas, J. Vrettaros
NCSR Demokritos
DTE/YE, Department of technological application, NetMedia Lab
Ag. Paraskevi – 15310, Athens, Greece

Abstract: - In this paper is developed an intelligent searching tool using the Self-Organizing Map (SOM) algorithm, as a prototype e-content retrieval tool. The proposed searching tool has the ability to adjust and scale into any e-learning platform that requires concept-based queries. The SOM algorithm has been used successfully for the document organization as well as for document retrieval. In the proposed methodology, maps are used for the automatic replacement of the unstructured, the half structured and the multidimensional data of text in such a way that similar entries in the map are represented near between them. The performance and the functionality of the document organization and retrieval tool employing the SOM architecture, is presented. Furthermore, experiments performed to test the time performance of a learning algorithm used for the direct creation of teams of terms and texts enabling efficient searching and retrieval of the documents.

Key-Words: - Leave one blank line after the Abstract and write your Key-Words (6 - 10 words)

1 Introduction

With the advent of new technologies and the World Wide Web (WWW), enormous quantities of informative material are nowadays available on-line. Computers are increasingly changing from computing systems to portals, which permit to access big volumes of information. In parallel, due to the interest in reducing costs of education and stimulating people to never stop learning, e-learning applications have recently developed in educational, industrial and research institutions. However, e-learning platforms present some difficulties concerning the instructor and the student interaction. Effective mining and retrieval of the e-content is the major bottleneck of e-learning application. The lack of metadata and classification of the e-content, force the development of powerful search engines.

The basic approaches for information retrieval and data mining in textual documents collections are: (1) searching using keywords or key documents, (2) exploration of the collection referring to some organization or categorization of the documents, and (3) filtering of interesting documents from the incoming document stream. Keyword search systems can be automated rather easily whereas the organization of document collections has traditionally been carried out by hand. In manual organization carried out for example, in libraries, classification schemes are defined and each document is positioned into one or several classes by a librarian. Similarly, in the current hypertext systems the links between related documents are most often added by hand.

One of the traditional methods of searching for texts that match a query is to index all the words (terms) that have appeared in the document collection. The query itself, typically a list of appropriate keywords, is compared with the term list of each document to find documents that match the query.

In the existed applications, the educational content is multilingual and heterogeneous. Therefore, simple keyword queries are not capable for efficient mining of the available information.

In order to bypass the aforementioned bottlenecks, we propose the use of Artificial Neural Networks (ANN). Due to their wide range of applications, ANNs have been an active research for the past decades [2]. A large variety of learning algorithms have been evolved and being employed in ANNS. A further categorization divides the network architectures into three distinct categories: feedforward, feedbackward and competitive [2].

The self-organizing maps or Kohonen's feature maps are feedforward, competitive ANN that employ a layer of input neurons and a single computational layer [7]. The neurons on the computational layer are fully connected to the input layer and are arranged on an N-dimensional lattice. In this paper, we shall use the SOM algorithm to cluster contextually similar documents into classes.

The ability of the SOM algorithm to produce spatially organized representations of the input space can be utilized in document organization, where organization refers to the representation and storage of the available data.

An architecture based on the SOM algorithm that is capable of clustering documents according to their semantic similarities is the so-called WEBSOM architecture [4,5,6,7]. The WEBSOM consists of two distinct layers where the SOM algorithm is applied. The first layer is used to cluster the words found in the available training documents into semantically related collections. The second layer, which is activated after the completion of the first layer, clusters the available documents into classes that high probability contains relevant documents with respect to their semantic content. Due to that, the WEBSOM architecture regarded as a prominent candidate for document organization and retrieval.

The structure of the paper is as follows. In the first section is presented the basic structure of the SOM architecture. The system architecture is presented in the second section. The software component constructed to illustrate the applicability of the proposed architecture is shown in the third section. Finally the performance of the training algorithm is illustrated in the final section

2 SOM Algorithm

The basic Self-organizing Map (SOM) can be visualized as a sheet-like-neural-network array (figure), the cells (or nodes) of which becomes specifically tuned to various input signal patterns or classes of patterns or classes of patterns in an orderly fashion. The learning process is competitive and supervised, meaning no teacher is needed to define the correct output (or actually the cell into which the input is mapped) for an input. In the basic version, only one map node (winner) at a time is activated corresponding to each input. The locations of the responses in the array tend to become ordered in the learning process as if some meaningful nonlinear coordinate system for the different input features were being created over the network.

Let us denote by X the set of vector-valued observations,

$$\mathrm{X} = \{x_j \in R^{N_w} \mid x_j = (x_{1j}, x_{2j}, ..., x_{N_w j})^{T,}\ j = 1,2,...N\}$$

Where N_w corresponds to the dimensionality of the vectors that encode the N available observations. Let also W denote the set of reference vectors of the neurons, that is,

$$W = \{w_i(k) \in R^{N_w}, l = 1,2,...,L\},$$

Where the parameter k denotes discrete time and L is the number of neurons of the lattice. Finally, let w(0) be located on a regular lattice that lies on the hyperplane which is determined by the two eigenvectors that correspond to the largest eigenvalues of the covariance matrix of $x_j \in X$ (linear initialization [4]).

Due to its competitive nature, the SOM algorithm identifies the best-matching, winning reference vector w(k) (or winner for short), to a specific feature vector xj with respect to a certain distance metric. The index s of the winning reference vector is given by:

$$s = \arg\min_{l=1}^{L} \| x_j - w_l(k) \|,$$

where ||.|| denotes the Euclidean distance.

The reference vector of the winner as well as the reference vectors of the neurons in its neighbourhood are modified towards xj using:

$$w_i(k+1) = \begin{cases} w_i(k) + a(k)[x_j - w_i(k)] & \forall i \in N_s, \\ w_i(k) & \forall i \notin N_s \end{cases}$$

where a(k) is the learning rate and Ns denotes the neighbourhood of the winner. A neighbourhood updating, especially in the early iterations, is performed in order to achieve a global ordering of the input space onto the lattice, which is crucial for the good resolution of the map [4]. The term basic SOM will henceforth denote the on-line algorithm proposed by kohonen [4] without any modification of speed-up techniques. The previous equation can be rewritten as follows:

$$w_i(k+1) = w_i(k) + a(k)c_{ij}(k)[x_j - w_i(k)],$$

where $c_{ij}(k) = 1$ if the jth feature vector assigned to the ith neuron during the kth iteration, otherwise $c_{ij}(k) = 0$.

2.1 Salton's Vector Space Model

The vector space model [8] has been widely used in the traditional IR field. Most search engines also use similarity measures based on this model to rank web documents. The model creates a space in which both documents and queries are represented by vectors. For a fixed collection of documents an m-dimensional vector is generated for each document and each query from sets of terms with associated weights, where m is the number of unique terms in the document collection. Then a vector similarity function, such as the inner product, can be used to compute the similarity between a document and a query.

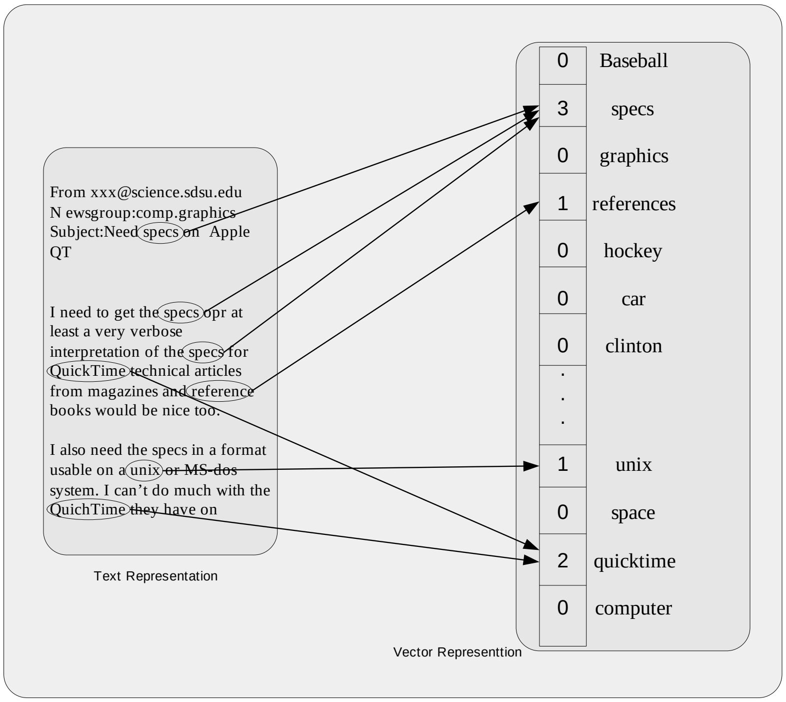

From xxx@science.sdsu.edu
N ewsgroup:comp.graphics
Subject:Need specs on Apple QT
I need to get the specs opr at least a very verbose interpretation of the specs for QuickTime technical articles from magazines and reference books would be nice too.
I also need the specs in a format usable on a unix or MS-dos system. I can't do much with the QuichTime they have on
Text Representation
0 Baseball
3 specs
0 graphics
1 references
0 hockey
0 car
0 clinton
1 unix
0 space
2 quicktime
0 computer
Vector Representtion

Fig. 1 *The VSM. The text on the left is represented as the vector on the right. Each line of the vector is a different word of the text. Each line's record is the frequency of the word in the text*

In VWSM, weights associated with terms are calculated based on the following two numbers:

Term frequency f_{ij}, the number of occurrence of the term y_i in document x_i.

Inverse document frequency, $g_j = \log(N/d_j)$, where N is the total number of documents containing the term y_i

The similarity $sim_{vs}(q,x_i)$, between a query q and a document x_i can be defined as the inner product of the query vector Q and the document vector X_i:

$$sim_{vs}(q,x_i) = Q \cdot X_i = \frac{\sum_{j=1}^{m} u_j \cdot w_{ij}}{\sqrt{\sum_{j=1}^{m}(u_j)^2 \cdot \sum_{j=1}^{m}(w_{i,j})^2}}$$

Where m is the number of unique terms in the document collection. Document weight w_{ij} and query weight u_j are:

$$w_{ij} = f_{ij} w_{ij} = f_{ij} \bullet \log(N/d_j) \qquad \textit{and}$$

$$u_j = \begin{cases} \log(N/d_j & y_i \textit{ is a term in q} \\ 0 & \textit{otherwise} \end{cases}$$

the main problem of the vector space model is the large vocabulary in any sizable collection of free text documents, which results in a vast dimensionality of the document vectors.

In the following, some methods for reducing the dimensionality will be discussed. These are applicable to all cases where the documents are encoded using the vector space model, i.e. as the document-by-word matrix

2.2 Latent Semantic Indexing

Latent Semantic Indexing (LSI) is one alternative to the original vector space model. LSI tries to make account to the co-occurrence of terms in documents when encoding the documents. One way of interpreting the LSI is that it represents the jth document by the vector

$$n'_j = \sum_k n_{jk} x'_k,$$

where n_{jk} denotes again the number of times the word k occurs in the jth document. The x'_k is the code that the LSI forms of the kth word by investigating the co-occurrence of the words within the documents. The term-by-document matrix, a matrix where each column is the word histogram corresponding to one document, is decomposed into a set of factors (eigenvectors) using the singular-value decomposition (SVD). The factors that have the least influence on the matrix are then discarded. The motivation behind omitting the smallest factors is that they most likely consist of noise. The vector x'_i can then be formed by using only the remaining factors, whereby the dimensionality is reduced.

2.3 Random projection

A low-dimensionality representation for documents can be obtained as a random projection of the high-dimensional representation vector into a much lower-dimensional space [3]. The benefit compared with alternative methods such as the LSI is extremely fast computation. The accuracy of the results is still comparable.

2.4 Word Clustering

Clustering methods can be used to reducing the number of data by grouping similar items together [3]. If similar words can be clustered together, documents can be represented as histograms of words clusters rather than of individual words. Various early approaches for categorizing words have been described in [3]. In languages with rigid word order, such as English, the distribution of words in the immediate context of a word contains considerable amounts of information regarding the syntactic and semantic properties of the word. The SOM has been used to cluster words based on the distributions of words in their immediate contexts [3].

2.5 SOM computation complexity

The computational complexity of the SOM algorithm is only linear in the number of data samples. However, the complexity depends quadratically on the number of map units.

For document maps intended for browsing the document collection, the resolution (number of map units per number of documents) should be good, since browsing is easier if there are representations of only a few, say, ten documents in a map unit on the average. Hence, for such resolution, the number of map units has to be proportional to the number of documents. For very large document collections the resulting computational complexity might become problematic.

3 System description

We determine the characteristics of the intelligent search based on the nature of the educational material (e-content) that interest the users (word docs, htms pages, plain text). The system has the capability of document retrieval from databases aiming at the preparation and presentation of an e-learning course. The system is capable of retrieving certain educational texts by the users according to their “physical” questions. In the following paragraph we summarize the basic elements of the system.

We export the descriptors from the text of the multimedia material and transforming of these descriptors into compound search descriptors, in suitable vectors of characteristics. We concretize of the non supervised learning algorithm SOM for the successful information retrieval. The concrete methodology was peered against the method of simple equation of keywords as well as the one that makes use of the simple metric resemblance in the representation space of the texts (e.g. calculation of cosine between vectors and retrieval of those nearest in the vector that represents a question) because it provides better retrieval performance and releases the user from the need of creation of complicated educational components. The big advantage of the non supervised search models is that content managers are not obliged to create huge learning material (examples of questions with the connected answers). Taking into consideration that the user cannot as an expert in neural networks training, the software search module is supposed to supply him/her with the capability to search the database intelligently, via the combination of the automatic exported characteristics and his/her own keywords

The proposed methodology for the creation of the intelligent search system is based on the SOM algorithm, described in the previous section. In the concrete application, the SOM maps are used fro the automatic placements of the unstructured or half structured and multidimensional data of text in such a way that similar entries in the map are represented near between them. Via a learning process, the t the performance is illustrated in the following section, that final map allows the direct creation of teams of terms and teams of texts so that the distances between the different data can be directly used during the search and retrieval duration.

An example of a previous successful application of the SOM networks in information retrieval is the web application WebSOM [2]. This application is based on the export of descriptors of texts from different SOMs, which replace the department of pretreatment, and representation of texts, in combination with a self-organized of the retrieved texts. It also provides the capability of a two dimension depiction of texts relative between them, so that the user has in his disposal a visual representation of the material in relevant categories. As recent researchers have shown that the functionalism of such visualization considering the help that it provides in the final user is arguable, in the concrete work, we do not use the depiction of map. On the contrary, we provide the capability of information retrieval from the database according to the content of texts and thus present the results in form of a list in a declining line of resemblance so that we decrease the difficulties faced by users.

3.1 Content retrieval using the SOM algorithm

The SOM algorithm has been used to retrieve educational material from a database. The methodology of this operation is as follows. We export the descriptors from the text of the educational material. For this task responsible are the designers of the database system. An automate method for transformation of the descriptors of the material as well as the compound search descriptors, in suitable vectors of characteristics is used. To perform this automatic operation we have used the VSM algorithm described previously. The VSM provides efficiency of the search results. The next step is to learn the SOM with the vectors of text characteristics. The result is the clustering of the used terms and texts in teams of relevant content. Following this step we must search for relevant texts with the use of questions. After the creation of the vector of characteristics of the question it is supplied in the entry of trained SOM network. The result is the calculation of the nearest Euclidean distance of teams of texts towards the question based on the activated neurons of the map. This team will contain texts with terms of approximate weight and consequently will present the highest affinity of content with the question. Search of relevant texts in neighbor teams is the following step. The attribute of the self-organization allows the user to search different relevant texts found in teams of neighbor neurons of map.

A question that is placed to the system, formulated as word or as a combination of words, activates the processes of retrieval of texts relative with the question. The system seeks in the map of teams of terms thus neurons that correspond in the terms or in the combination of terms that exists in the question. Those texts represent in concrete neurons of the map of the teams of texts activated by the terms, are selected and thus are presented in the user. Additionally, it is possible to present texts by a concrete method of using metric resemblance between texts.

3.2 Training

After the creation of the vector maps, we perform training. The feature vectors are presented iteratively an adequate number of times to the neural network which perform clustering in an effort to build word classes containing semantically related words. This is based on empirical and theoretical observations that semantically related words have more or less the same preceding and succeeding words.

The above process yields the so-called word categories map (WCM) [1]. After the computation of the document vectors the SOM method is used to cluster them. It is expected that the constructed documents classes contained contextually similar documents.

4 Experimental Results

The performance of the SOM algorithms in the proposed case study is illustrated in this section. The performance is measured using the Mean Square Error (MSE) between the reference vectors and the document vectors assigned to each neuron in the training phase. Furthermore the recall-precision performance is measured using query documents from a test set during the recall phase is used as an indirect measure of the quality of the document organization provided by the SOM algorithm. Fig. 2 depicts the MSE curves during the formation of WCM using the SOM architecture. Similar MSE curves are plotted in Fig. 2 that correspond to the training phase. During the formation of the WCM, the number of training iterations needed by the SOM so that the MSE drops to the e^{-1} of its initial value was nearly 800.

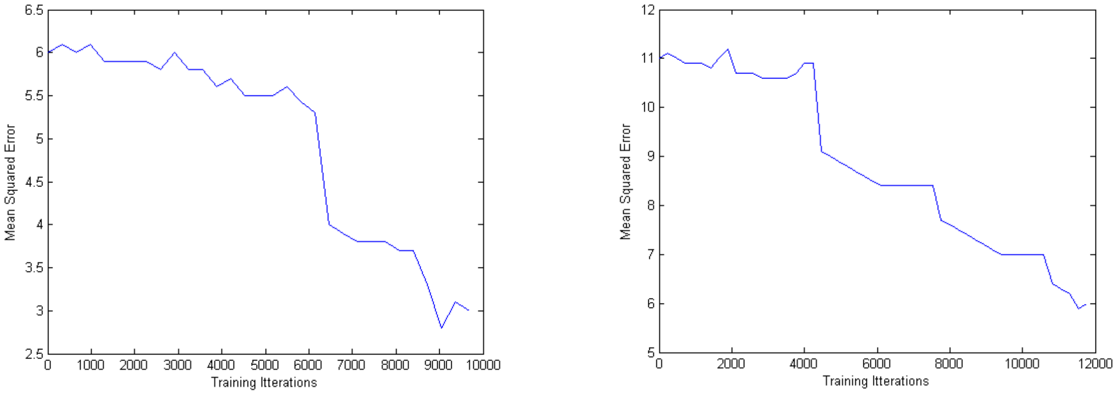


Fig. 2 MSE curves during the formation of WCM using the SOM architecture

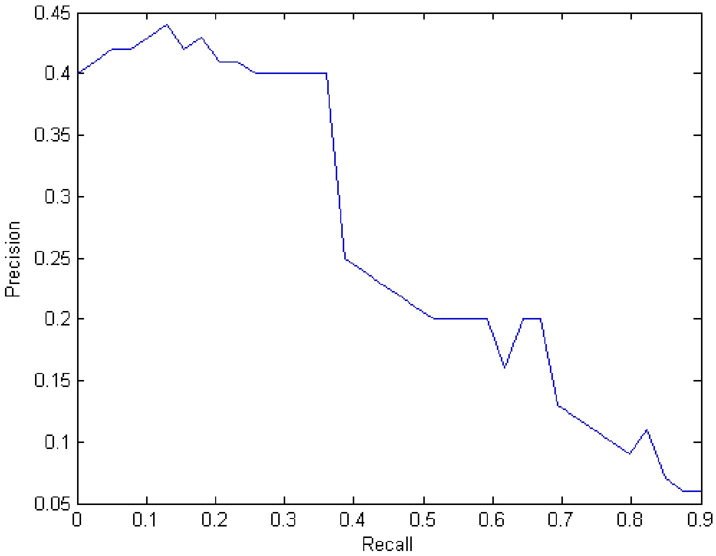


Fig.3 The average recall-precision curve of the SOM

To measure the effectiveness of a retrieval system, two widely used ratios are employed: the precision and the recall. Precision is defined as the proportion of retrieved documents that are relevant:

$$P = \frac{r}{n_2}$$

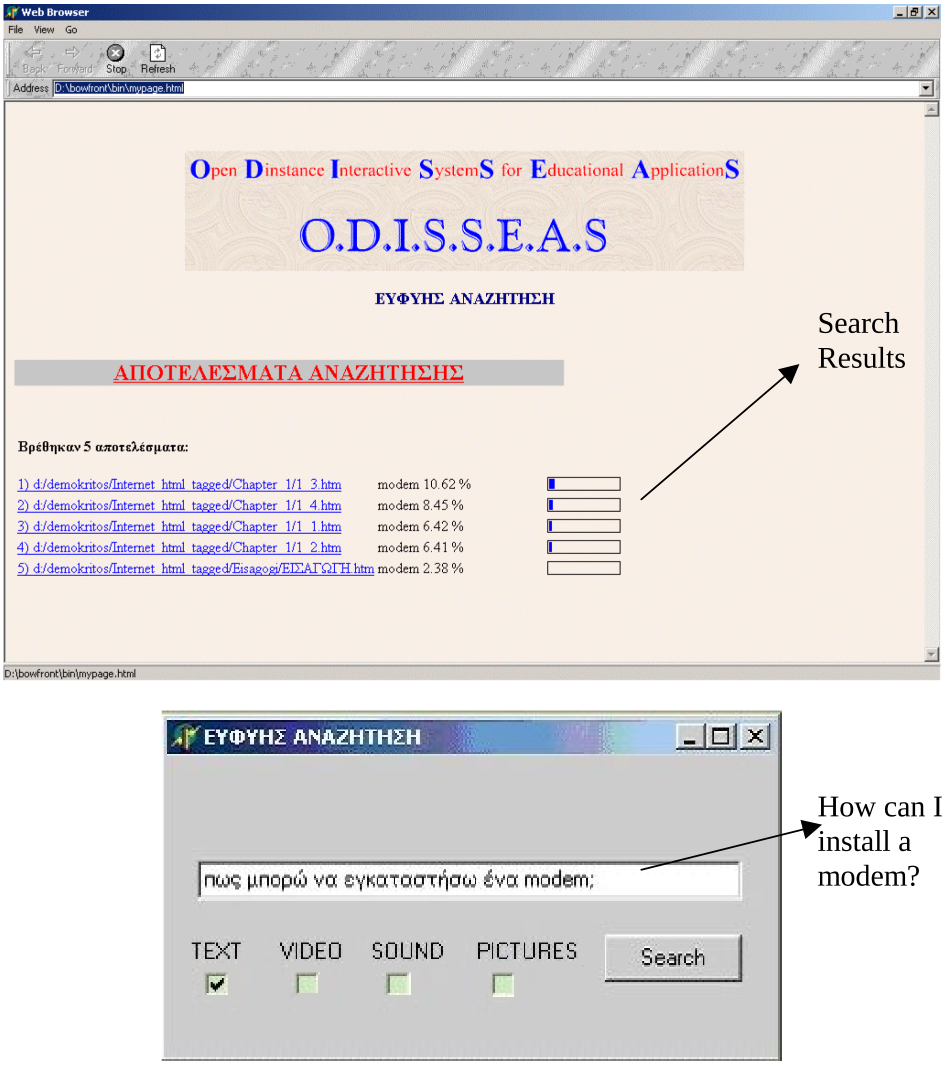
Recall is the proportion of relevant documents that are retrieved:

$$R = \frac{r}{n}$$

as the volume of the retrieved documents increases the above ratios are expected to change. The sequence of recall-precision pairs obtained yields the so-called recall-precision curve. Each query-document in the test set produces one recall-precision curve. An average over all the curves corresponding to query documents of the same topic obtained from the test set produces the average recall-precision curve. If the recall level does not equal to one we proceed with the second best winner neuron and repeat the same procedure and so on. The comparison of the effectiveness between the retrieved documents utilizes that above-mentioned curve. Fig. 3 depicts the average recall-precision curve of the SOM.

4.1 Software component of intelligent search

The core of the system of the SOM algorithm and the VSM has been developed in ANSI C to ensure portability and compatibility in platforms of different type (Windows, UNIX, Linux, etc). the user interface can be designed overall round the basic and autonomous departments of the system in order to activate, through a GUI, the system


Web Browser
File View Go
Back Forward Stop Refresh
Address D:\bowfront\bin\mypage.html
D:\bowfront\bin\mypage.html

Open Dinstance Interactive SystemS for Educational ApplicationS

O.D.I.S.S.E.A.S

ΕΥΦΥΗΣ ΑΝΑΖΗΤΗΣΗ

ΑΠΟΤΕΛΕΣΜΑΤΑ ΑΝΑΖΗΤΗΣΗΣ

Βρέθηκαν 5 αποτελέσματα:

1) d:/demokritos/Internet_html_tagged/Chapter_1/1_3.htm modem 10.62 %
2) d:/demokritos/Internet_html_tagged/Chapter_1/1_4.htm modem 8.45 %
3) d:/demokritos/Internet_html_tagged/Chapter_1/1_1.htm modem 6.42 %
4) d:/demokritos/Internet_html_tagged/Chapter_1/1_2.htm modem 6.41 %
5) d:/demokritos/Internet_html_tagged/Eisagogi/ΕΙΣΑΓΩΓΗ.htm modem 2.38 %

Search Results

ΕΥΦΥΗΣ ΑΝΑΖΗΤΗΣΗ

πως μπορώ να εγκαταστήσω ένα modem;

TEXT ☑ VIDEO □ SOUND □ PICTURES □

How can I install a modem?

Fig. 5 *Screen shots of the software component O.D.I.S.S.E.A.S*

operations and presentation of search results (standalone applications) via Visual Basic/C++, Delphi or Web based enabled as CGI scripting, ASP, JSP, PHP, etc.

A basic issue of the final interface is the confidence estimation. The user tends to need a score of the results of its query in order to ameliorate his queering style. The following figure present an embodiment example of the search system into the e-learning platform O.D.I.S.S.E.A.S. [1] (Open Distance Interactive SyStem for Educational Applications) with the JSP (Java Server Pages) technology.

In order to prove the functionality of the proposed system, we analysed a collection of text and multimedia documents. Although the SOM algorithm can be applied only to text documents, in our collection we include multimedia documents. Multimedia documents are analysed through the annotation that is created for each picture. Video and sound. In this way we have all the collection in raw text format. They differ from each other in their topic and in the document average size.

The test collection consists of 157 documents, 65 are multimedia and 92 are textual. The average size of the documents is 4.55 Kb and the biggest document size is 13 Kb.

Table 1 Summary of the results obtained from the analysis of the test collections.

Collection	Size (Kb)	Number of	Average Docume	Average Number	Number of	Number of Terms

		Document	nt size	of size	Lexical Form	
Textual	370	92	4.55	124	11,789	32,456
Audio	98	25	7.88	30	5,890	15,678
Image Video	140	40	3.44	51	3,540	7,654

4.2 Discussion

The proposed system has been evaluated by means of its usability by the e-learning users. The e-learning users are the teachers who upload new material in the database and the students who download the teaching material. In order to measure the applicability of the system, we have set two questions to the users:

Question 1: how relevant (in percentage) are the retrieved documents to the query compared to ordinary search engine?

Question 2: are you satisfied with the degree of correlation of the system?

The results of the questions are depicted in Fig. 6.
The only disadvantage of the proposed system is that every time a new document is uploaded in the database, the learning process must run for the new document. This process is time consuming and costly. In the future work we are planning to improve the applicability of the system concerning the automatic learning process.

5 Summary

In this paper is presented the use of the SOM algorithm along with a training algorithm, for document retrieval. The applicability of the algorithm is illustrated in an e-learning case study. A software component has been constructed to perform intelligent search in the educational documents. The performance of the training algorithm using the MSE measure has been presented.
One of the basic issues concerning the Intelligent Systems is its ability to adjust and be installed into any platform that requires methods of intelligent retrieval. The system was awarded after its application in various tests. Its pedagogical advantages were underlined not only by the students but also by the instructors handling the educational material. The instructors gained valuable time during their course as they could retrieve information using simple queries while students found the intelligent system necessary at their self-paced learning. Moreover, in a future expansion, the system is expected to provide reasons of the confidence estimation accompanied by the retrieved texts. That means that the user will be supplied with reasons of the certain search results and the scope of its query.

References:

[1] A. Drigas, J. Vrettaros and S. Kouremenos. "Open Distance Interactive System for Educational Applications O.D.S.S.E.A.S". *In Proceedingss of the general conference of Samos* "Technology and learning in Higher Education" (2001).

[2] J. Honkela, K. Lagus and S. Kaski. "Self-organizing maps of large document collections." *In Deboeck, G. and Kohonen, T., editors, Visual Explorations in Finance with self-organizing Maps*, pp. 168-178. Springer, London (1998)

[3] K. Lagus and S. Kaski. "Keyword selection method for characterizing text document maps." *In Proceedings of ICANN99, 9th international Conference on Artificial Neural Networks*, vol. 1, pp. 371-376, IEE, London (1999).

[4] T. Kohonen, S. Kaski, K. Lagus, J. Honkela, V. Paatero and A. Saarela."Self organization of a massive text document collection." *In Oja, E. and Kaski, S., editors, Kohonen* Maps pages 171-182, Elsevier, Amsterdam (1999).

[5] K. Lagus, "Text retrieval using self-organized document maps". *Technical Report A61*, Helsinki University of Technology, Laboratory of computer and information science (2000).

[6] T. Kohonen, Self-organization and Associative Memory. Third edition, *Spriger-Verlag*. Berlin Heidelberg (1989).

[7] T. Kohonen. Self-Organization Maps. Springer-Verlag erlin Heidelberg

[8] A. Salton. Automatic Text processing. *Addition-Wesley Publishing Company*, Inc, Reading, MA (1995).

[9] Kohonen. "Self-organization of very large documents collections. State of the art." *In Niklasson, L., Boden, M., and Ziemke, T., editors, Proceedings of ICANN98* , 8th international Conference on Artificial Neural Networks, vol. 1, pp. 65-74, IEE, London (1998).

[10] G. Deerwester, T. Dumais, W. Furnas, K. Landauer, R. Harshman."Indexing By Latent Semantic Analysis". *Journal of the American Society of Information Scieence* (1990).

[11] Honkela, S. Kaski, T. Kohonen and K. Lagus."Self-organizing maps of very large document collections: Justification for the WEBSOM method." *In Balderjahn, I., Mathar, R., and Schader, M., Editors, Classification, Data Analysis, and Data Highways*, pp.245-252. Springer, Berlin (1998)

[12] R. Perfetti, G. Costantini, "Associative memories on BBS neural networks: a hardware-oriented learning algorithm" in *WSEAS NNA-FSFS-EC 2003*, pp 454-109

[13] I. Triantafyllou, G. Carayannis, "Architectures and Techniques for Monolingual and Multilingual Information Retrieval Systems in a SOM Framework", in *WSEAS NNA-FSFS-EC 2003 pp 454-205*

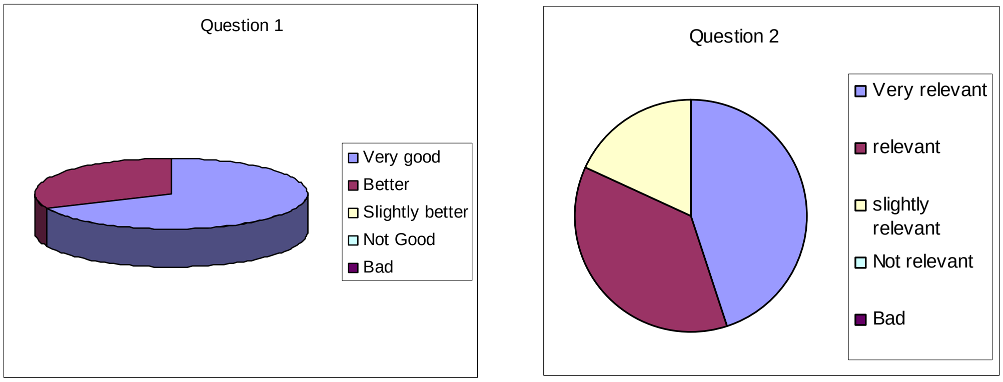


Fig. 6 *The results of the two questions*