MED. INFORM. (2000) VOL. 25, NO. 2, 133–146

Computer assisted information resources navigation

N. N. KARANIKOLAS† and C. SKOURLAS‡

† Areteion University Hospital, 76 Vas. Sofias St., 115 28, Athens, Greece

‡ Technological – Educational Institute (TEI) of Athens, Department of Informatics, Ag. Spyridonos St., 122 10, Egaleo, Greece

Abstract. In this paper, the design and development of Computer Assisted Information Resources Navigation (CAIRN) is discussed. CAIRN system is a medical information retrieval system that allows physicians and students to store full text medical information from any resource, organize and retrieve it. The most important feature of CAIRN is its capability to assist the user, physician, student etc. in selecting documents against a submitted query in Natural Language. The retrieved documents are presented in decreasing order according to their similarity to the submitted query. The nearest neighbour method is used. An alternative similarity measure based on a new calculation of the length of documents is proposed and some experimentation with it is discussed.

Keywords: Medical information system; Information retrieval system; Nearest neighbour method; Similarity.

1. Objectives

The idea that information in any format is a strategic resource and it should be available to the user (e.g. the physician, the student etc.) from the desktop as if it were located in a personal library has gained ground [1]. The user needs easily accessible information and tools to exploit it. In this context, the popularity of the Internet has caused an exponential increase in the number of people who use on-line text [2] but the quality and usefulness of documents varies widely [3] and web search tools are characterized by extremely low precision [4].

Hence, it is more important for users to have information on their desktop/personal computer (PC). Such information should be easily accessible, and digestible. When the users have a question, they should have the means (the software system etc.) to find in their personal file their own previously collected related material (e.g. collected tutorials, papers, personal comments and notes) and support their own answers/decisions/research.

The general objective of an information retrieval (IR) system is to minimize the overhead (the time a user spends) in locating needed information [5]. Word processing software cannot allow the same flexibility, efficiency and speed in searching for relevant information. IR systems use indexing terms (which can be likened to those in the back of a book) to improve retrieval efficiency and provide the fast retrieval speeds required. A search can be carried out using one or more search terms, and the relevance of retrieved documents is automatically calculated.

We now have the ability to provide information support to a new generation of users because the technological level seems to be more mature [5, pp. 2–3]:

Medical Informatics & The Internet in Medicine ISSN 1463-9238 print/ISSN 1464-5238 online © 2000 Taylor and Francis Ltd
http://www.tandf.co.uk/journals

- Almost every PC comes with a CD-ROM, some GBs of hard disk, a modem and a connection to an on-line service (usually a connection to Internet Provider), which has to be thought of on the same level as a word processor.
- IR software is also cheaper and more powerful [5].

The purpose of this work is to create a prototype of a full-text IR tool to assist the users, physicians, medical physicists, technologists, and students in storing, and then accessing, selected information from different resources: web based documents, bibliographies in machine readable form, ASCII (plain) text, text in well known formats (eg. doc files, rtf) etc.

As an example, the users of such a system as the prototype IR tool must be able to download and store into the system full text conference papers from the Internet. They should be able to buy a bibliography in machine-readable form, extract some portion of interest and load it into the system. They should even be able to add some personal comments on the stored information, papers etc. They should be able to store personal notes related to various subjects, and also store their personal research reports etc.

Such an objective is justified by the following considerations:

- There is an increased demand for medical information and medical information systems [1]. Special emphasis is given to information retrieval (IR) systems, which are among the most important of specialized medical information systems. The new information technology offers new opportunities for the development of low cost, effective, user-friendly information retrieval systems [5, pp. 21, 44–45], [6–9].
- A number of IR research software tools are available, for example, the public domain famous SMART system [10, 11], the powerful INQUERY system [12], and public domain general purpose tools for indexing and retrieving information (see [13]).
- A number of commercial tools for enterprise and personal use are emerging, e.g. Folio [14], Sirsi [15], Verity [16]. These tools are mainly focused on information providers' needs but can also cover information consumers' needs. Such tools offer the possibility of downloading, periodically, specific HTML pages of interest and storing the information (new or updated) in the hard disc etc. Special emphasis is given in covering legal, insurance, banking and government publishing etc.
- The massive information explosion, particularly in the areas of science and technology, makes it difficult to create a personal file (a personal 'data base') based on a traditional commercial IR system and 'follow' the resulting publications of interest without a level of maturity in information management and technical knowledge in librarians' tasks such as cataloguing, indexing and abstracting. All these tasks are extremely time and money consuming activities to the user, and also result in less time being spent with the patients, etc. In our point of view, the solution is to use a simple full text retrieval system, such as our **Computer Assisted Information Resources Navigation (CAIRN)** prototype IR tool, for storing, organizing and accessing information.
- The most crucial requirement is to increase the value of the stored information, introducing the possibility that the user can access information using Natural Language queries.

To satisfy these needs we designed and implemented the CAIRN system. CAIRN has been implemented on a personal computer using a general-purpose information retrieval system called EREVNITIS [17, 18] (erevnitis is the transliterated Greek word for the word researcher) which was developed by the first author of this work. There are some differences between the two systems:

- The two systems have different user interfaces.
- At the moment, the possibility of having associated data for a document e.g. BLOBs (Binary Large Objects) is not included in CAIRN system.
- CAIRN can handle document collections and every collection can store over 20000 document texts. We have decided to eliminate this restriction in the future.

CAIRN (and Erevnitis) is based on the vector space model (VSM) [5, 10, 19, 20] for information retrieval. IR systems based on this model and its associated research results have been evolving for over 30 years [5, p. 57].

Section 2 describes the materials and methods used, and the implementation of CAIRN is briefly presented. The results are presented in section 3, with special emphasis on the reactions to CAIRN. The use of an alternative method for calculating the similarity between a document and a query in the CAIRN system is also presented. Finally, section 4 discusses planned enhancements to CAIRN and our conclusions.

2. Materials and methods

The CAIRN system is based on the vector space model (VSM) for representing document collections. More precisely, each document is represented by a vector of index (or search) terms. The 'index term' is usually the stem of a word extracted from a document text. The term VSM is derived from the fact that the whole document collection may be represented as a vector space and each vector represents the assignment of the search terms to one document of the collection.

There are two approaches:

- A document vector is a set of values and each value in this set can be equal to one or zero with one representing the existence of a search term in the document (binary approach).
- Each value in the document vector can be a positive number that calculates the relative importance of the search term in representing the semantics of the document (weighted approach).

A user query can be also represented by a vector in the same manner. Therefore, the document and query vectors can be envisioned as an n-dimensional vector space. Search is accomplished by calculating the distance (or the similarity) between the query vector and the document vectors. A vector matching operation, based on the cosine correlation is normally used to measure the cosine of the angle between vectors. This then can be used to compute the similarity [20, 21].

Vector techniques have very powerful representations and have been shown to be successful for more than 30 years [5, pp. 109–110, 122–123]. Hence, the vector space model was selected for two reasons:

(1) It is well established and has been used for many years.
(2) It is the de facto standard for the commercial IR systems.

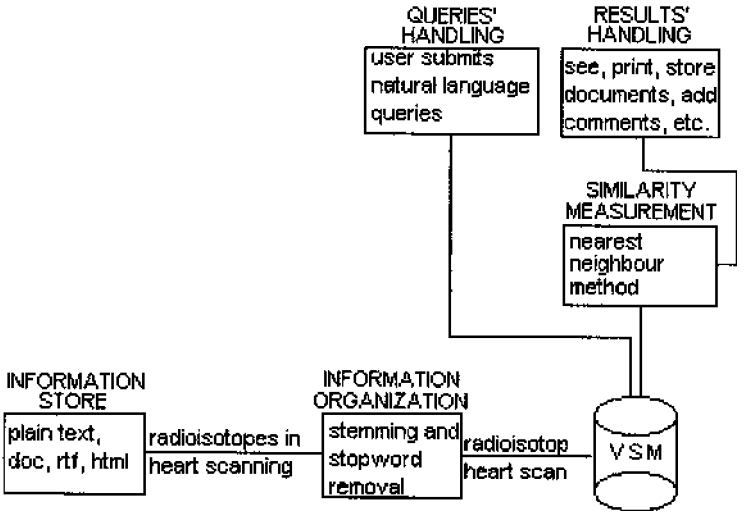

Figure 1. An overview of the CAIRN system and its modules.

The only problem with this model is that it does not have a mechanism to associate correlation factors between search terms. Bayesian techniques (and experimental IR systems based on them) have been proposed as a way to relax some of the constraints inherent in a pure vector approach [5]. The fact that most commercial IR systems use VSM does not mean that the performance of a system based on it is always better. There are tools, mainly experimental, (e.g. the INQUERY system which is based on Bayesian inference methods [12]) with better performance for specific data collections. There are open forums (e.g. TREC conferences) where different approaches and tools are tested against 'huge' test collections (e.g. TREC data [22]).

The CAIRN system is written in C language. The pilot system was developed to run on a standard PC with Pentium processor (90 MHz or faster), 16 MBs of RAM and 2 GBs of hard disk drive. The system will also include a CD-ROM and optionally a connection to the Internet. The operating systems supported are Windows 3·1, 95, 98, NT Workstation and NT Server.

Figure 1 depicts the architecture of the system. There are the following modules:

- Information Store
- Information Organization
- Query handling
- Measurement of Similarity
- Result handling

2.1. *Information store*

The CAIRN system was designed to be autonomous, modular, with data stored from different resources. At the moment, the following formats are supported for importing data into the system:

- ASCII (plain) text
- popular Word Processor formats (doc files, rtf)

- HTML
- structured information including the well known tag fields (e.g. authors' field, keywords' field) of bibliographic records (e.g. MEDLINE records)

The conversion of imported data (based on the above formats) into the internal format of the CAIRN system is automatic. In the future, some other popular formats (TIFF, JPEG, PDF) will be supported for converting imported documents.

The users' queries can also be stored for future use. Users tend to type short queries (one to three words) without giving much thought to query formulation [3]. However, the longer the query prose the more accurate the results returned [5, p. 30]. By storing sophisticated successful queries the users can forward the same query, as and when required. They can also improve their search strategy by enhancing a stored query and they can combine more than one queries to form new ones.

There is also the possibility of organizing and storing the document texts into different thematic collections.

2.2. *Information organization*

This is an important module which uses stems for automatic indexing. As an example, for the text:

> Radioisotopes in heart scanning. mainly used in diagnosis of pericardial effusions. also used to study tumors, heart enlargement, aneurysms and pericardial thickening. technetium, rihsa, radioactive hippurate, cholegraffin are used.

the system recognizes the individual words that make up the text. It then eliminates words such as 'in', 'of', 'to', 'and' from consideration in the subsequent processing. It also reduces the remaining words to word stem form by using suffix removal methods [10, 23]. Hence, the following terms are automatically constructed:

> ANEURYSM, CHOLEGRAFFIN, DIAGNOSI, EFFUS, ENLARG, HEART, HIPPUR, MAINLI, PERICARDI, RADIOACT, RADIO-ISOTOP, RIHSA, SCAN, STUDI, TECHNETIUM, THICKEN, TUMOR

Then the document vector (which is a set of values) is constructed and stored into the system. This is defined as follows.

The word 'radioisotopes', for example, in the document text is represented by the stem 'RADIOISOTOP'. Each value in the document vector representing one of the above stems (e.g. RADIOISOTOP) is equal to one. All the other index terms (stems) extracted from other document texts of the collection are represented by zero value. Therefore, the stem of the word 'hypothermia' which exists in another document and is not included in the document under consideration will be represented by zero value in the document vector.

Hence, all the above terms (stems) are links to the document text and any query containing such an index term can retrieve it. The document texts are stored into the same file (collection) and each document has a unique accession (or identification) number assigned by the system. The index (or search) term vectors are constructed to represent the documents and are separately stored into the system. This approach is called automatic indexing of document texts.

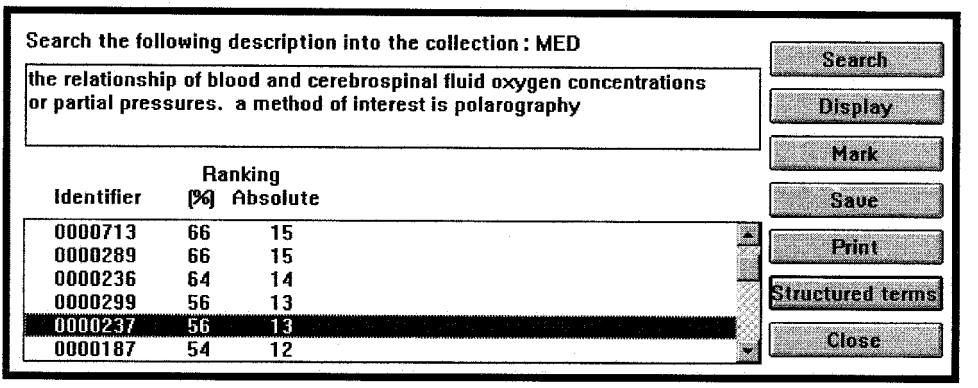

Figure 2. Submission of queries in natural language.

2.3. *Query handling*

Another interesting module that uses similar techniques with the previous one to split the search expression in search terms is the query handling module.

Figure 2 depicts the user interface for the query handling. The users can submit their queries using natural language (NL). NL queries allow the users to enter a prose statement describing the information that they want to find [5].

The users type the query text into the text box of the window shown in figure 2. The query handling module splits the query phrases into search terms [23] and then constructs the query term vector employing the same algorithm used in the previous information organization module. Then, the system tries to search/identify all documents whose word stem vectors are sufficiently similar to the query vector. The matching of the relevant documents is based on the calculation of a similarity measure described at the next section. The system displays the retrieved documents in decreasing order of query document similarity. There is also the possibility of storing the queries in a file and using them in the future.

A query results in the following:

- accession (or identification) number of the document
- comparative and absolute evaluation of the document's relevance.

As an example, the following document which has accession number 237

> Cisternal fluid oxygen tension in man.
> Using a beckman micro-oxygen-electrode we have studied the oxygen tension simultaneously in the cisterna magna, the internal jugular vein and in arterial blood under various conditions. the results suggest that the cisternal oxygen tension to some degree reflects the average oxygen tension of the surrounding brain tissue and besides reflecting the available free oxygen to the brain it registrates changes of short duration in the cerebral blood flow.

was retrieved by the following query

> The relationship of blood and cerebrospinal fluid oxygen concentrations or partial pressures. a method of interest is polarography

More precisely, the document with accession number equal to 237 is retrieved because there are common search terms (stems) between the document text and the query (e.g. blood, fluid oxygen). These terms are represented in the document and query vectors.

The concept of the relevance of a document against a submitted query is clarified below. It is dependent on:

- The number of common terms between the query and the document
- The frequency of each of the common terms in the document.
- The length (size) of document.
- The ability of each of the common terms to characterize/distinguish documents. Whenever the number of documents that include a term is increased, the ability of the term to distinguish documents is decreased.

Figure 2 also presents the result's list for the submitted query. The absolute relevance (depicted as 'Absolute Ranking') of document 237 is 13/100, because our similarity measurement algorithm evaluates the relevance to a base of 100. The comparative relevance (depicted as '% Ranking') of the same document is 56/100 and it means that this document is 44% less similar than the best matching to the query document.

The display and the handling of the retrieved documents will be discussed and described at the relevant 'Results' handling module.

2.4. *The measurement of similarity*

The problem of defining similarity between a document and a query (e.g. the nearest neighbours model [20]) has been the subject of continuing research for many years. Chapters of well-known books (e.g. [5,10,19,21]) are dedicated on this topic.

In particular, the problem of similarity measurement for a document against a submitted query has been one of the most interesting topics in information retrieval. C. J. van Rijsbergen [19] and Salton [10] describe many similarity functions (e.g. Dice and Jaccard coefficients, cosine coefficient, the overlap measure, some asymmetric measures). Bentley *et al*. [24] extensively study best match algorithms (known also as nearest neighbour search algorithms). Smeaton and C. J. van Rijsbergen [25] suggested a search algorithm that eliminates many of the query-document comparisons while it still identifies the most relevant documents. Lucarella [20] presented a straightforward nearest neighbour algorithm and an improved one which optimizes both the number of documents to be evaluated and the number of inverted lists to be inspected. Borlund and Ingwersen [26] introduced the concepts of the relative relevance measure and a new performance indicator of the positional strength of the retrieved and ranked documents.

The following equation/notation presented in Lucarella [20] gives a simple method to measure the similarity of document against query Q:

$$S(D_i, Q) = \frac{\sum_{j=1}^{n} q_j t_{ij}}{\sqrt{\sum_{j=1}^{n} q_j^2 \cdot \sum_{j=1}^{n} t_{ij}^2}} = \frac{\sum_{j=1}^{n} q_j t_{ij}}{L_Q \cdot L_{D_i}} \tag{1}$$

where n is the number of index terms used in the documents collection, t_{ij} is the weight of term j in document D_i and q_j is the weight of term j in the query.

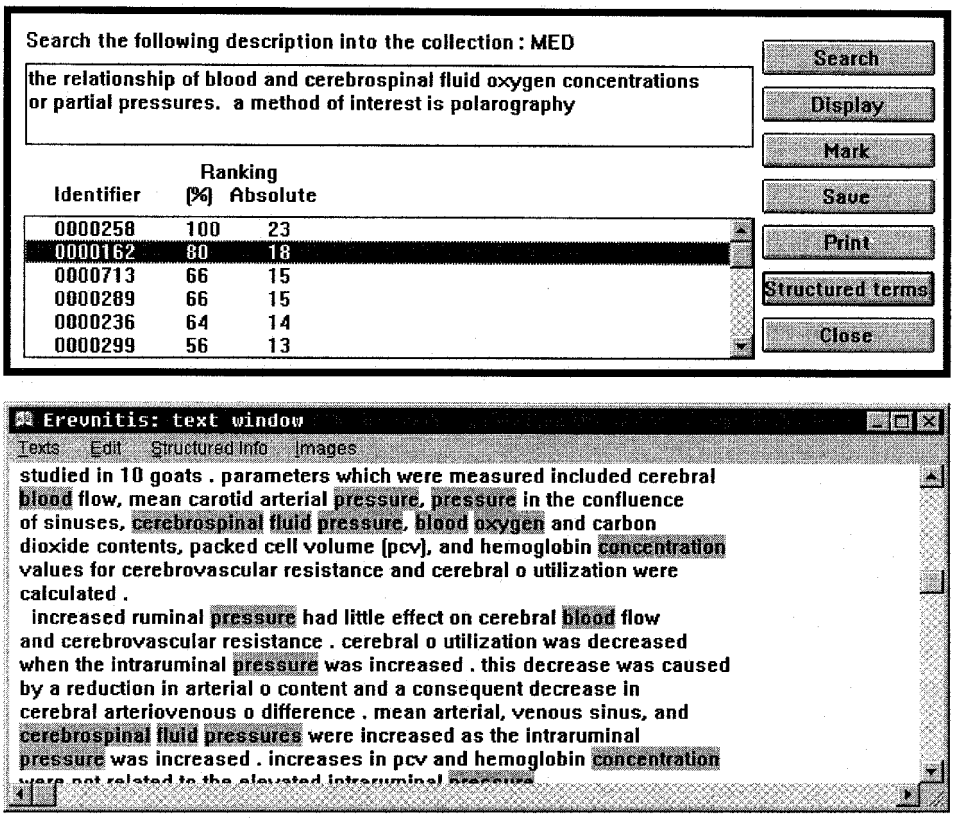

Figure 3. Presenting and handling the results.

The following two equations [20] can be used to measure the terms t_{ij} and q_j:

$$t_{ij} = 0.5 + 0.5 \cdot \frac{F_{ij}}{\max F_i} \tag{2}$$

$$q_j = \log_2 \left(\frac{N}{DOCFREQ_j} \right) \tag{3}$$

where F_{ij} is the frequency of term j in document D_i, $maxF_i$ is the maximum frequency of the terms in document D_i, N is the number of documents in the collection and $DOCFREQ_j$ is the number of documents that include the index term j.

According to (1) the document length is given by:

$$L_{D_i} = \sqrt{\sum_{j=1}^{n} t_{ij}^2} \tag{4}$$

and the query length is given by:

$$L_Q = \sqrt{\sum_{j=1}^{n} q_j^2} \tag{5}$$

In Section 3, another similarity measure is presented and discussed.

2.5. *Result handling*

Figure 3 depicts some possibilities offered by the module. The users can use the Query handling module to retrieve the relevant documents and then they can choose between the various possibilities:

- show document(s)
- print document(s)
- 'cut' phrases from a (some) retrieved document(s) and 'paste' them into the query text and then save and execute more sophisticated new queries. The 'cut' and 'paste' operations are supported by the screen editor, which is incorporated in the system (lower window of figure 3).
- store retrieved document(s)
- add comments/notes into retrieved document(s) and then store it (them). More precisely, when the users display the retrieved document(s) they use the screen editor, which is incorporated in the system. If the user wants they can add/replace text and save it as the same or new document.
- see the search terms/index terms in the retrieved document that matches the given query

3. Results

3.1. *User reaction*

A group of nine persons (one information retrieval specialist, four hospital physicians, two medical physicists and two students) used the pilot system. A prior brief introduction to the CAIRN system was given and a training session was conducted.

3.1.1. Use of a test base – the three phases. People in our team are familiar with the MEDLINE bibliographic database. Hence a portion of the database mainly related to oncology was imported into the system using a CD-ROM. Then a rather restricted number of HTML documents (100 conference papers mainly in oncology and radiotherapy) were also converted and stored into the system. Some comments and notes were added (by the information retrieval specialist) to the corpus of the test base for experimentation purposes.

The main result of this phase was the conclusion that the CAIRN system is easy to learn, and helpful.

We then decided to store a more ambitious test base. The OHSUMED test collection was collected [27]. This test collection was created to assist information retrieval research. It is a clinically oriented MEDLINE subset, consisting of 348566 references from 270 medical journals over a five-year period (1987–1991). The test base is accompanied by 106 queries, generated by physicians. The main task in this second phase was to prove that the proposed system is effective enough to compete with such a 'big' (for personal use) test base of 400 megabytes of size.

There were two main results of this second phase (which is a continuing one):

(1) It appears that the CAIRN system can handle effectively such large volume of stored information. More precisely, Erevnitis can achieve this at the moment because in CAIRN there is a limitation of storing 20000 documents per collection. The response time appears to be reasonable. Dependent on

the number of submitted query words the response time varies, e.g. for 15000 document collections, from one to 15 seconds.

(2) There was an interesting noticed indication/conjecture that the nearest neighbour method (and the similarity measure) used in CAIRN system presents a 'preference' to short documents against longer ones. As an example, analysing the retrieved document texts for specific queries we saw that some short document texts were calculated as more relevant than 2–3 times longer ones in spite of the fact that the longer documents contain more relevant search terms. This is undesirable because the users want to retrieve documents containing more search terms of interest. For example, if users search for documents related to 'hypothermia' and 'heart surgery' they prefer to retrieve documents where all these search terms are contained even if the length of the document text is longer than the length of documents where only the term 'hypothermia' is included (see also the example at section 3.2.1).

Therefore, a third phase was planned and executed. New similarity measures based on a different calculation of the length of documents were used. A well-known small collection was selected as a test base. This collection is accessible through the IDOMENEUS technology transfer server [28, 29], at the University of Glasgow Department of Computing Science. The test collection of articles contains three files:

med.all – The 1033 documents
med.que – The 30 queries
med.rel – The file of the relevance assessments

A new version of the CAIRN system was built up based on the modified similarity measure.

The experimentation with the two versions of the system confirmed our conjecture for the 'preference' of the first similarity measure to short documents against larger ones and that the proposed measure improves the whole situation. There was also an indication that the number of the relevant retrieved documents was increased using this modified similarity measure.

In the next subsection we discuss some details of the third phase.

3.2. *The measurement of similarity in IR systems and the problem of document length*
3.2.1. Problem outline – example. The following query (query number nine of med.que) was submitted to the CAIRN system:

> the use of induced hypothermia in heart surgery, neurosurgery, head injuries and infectious diseases

In the list of relevant documents retrieved by CAIRN system (using the Lucarella measure) documents number 273 (a short one) and number 415 (a longer one) were included. Document 273 has only one term in common with the query, while document 415 has two terms in common with the query. These documents follow:

Document 273: **hypothermia** in management of acute renal failure.

1. prolonged **hypothermia** begun in the period immediately following the infusion of epinephrine into the renal artery appears to give partial protection against renal damage.

Table 1. The relative position of documents 273 and 415 according to two algorithms.

Document number	Position of document according to the calculation of the classic similarity measure	Position of document according to the new calculation of similarity measure
273	12	28
415	17	13

2. shorter periods of **hypothermia** do not appear to be beneficial.
3. prolonged **hypothermia** at 28 to 30 °C has a mortality rate of 50 per cent to 60 per cent.

Document 415: 948.cardiac activity in cranio-cerebral **hypothermia** the onset of **hypothermia** rarely alters the **heart** rate. as it deepens to 35–32, the rate slows, and at the level of 30–29 it usually amounts to only half its original value. at 28 or below, the development of bradycardia is observed. after warming to 32 the normal **heart** rate is restored. during operations on the abdominal organs the **heart** rate is only slightly modified. the appearance of solitary extrasystoles is rare and is usually associated with stimulation of the diaphragm. the most marked changes in the **heart** rate are observed during operations on the heart, especially if it is excluded from the circulation. an idioventricular rhythm may develop before the **heart** stops beating. after removal of the ligatures from the venae cavae the normal rhythm is restored. as the temperature falls, the excitability of the myocardium increases. conduction is more resistant. areflexia continues even during direct stimulation of reflexogenic zones. in the surgical stage of cranio-cerebral **hypothermia** it is clear that no significant degree of energy or hemodynamic insufficiency develops, whether in experimental conditions or during operations on patients. at operation a well-marked stabilization of the contractile power of the myocardium may be observed.

The same query was also submitted to the version of CAIRN with the new calculation of the similarity measure (see also section 3.2.2). Table 1 presents the relative position of these two documents according to both algorithms.

When the system, calculating the classic similarity measure, displays the retrieved documents in decreasing order of query document similarity then the document with accession number 273 is in the twelfth place and is characterized as more relevant than the document 415, which is at the seventeenth one. This problem is corrected using the new similarity calculation.

3.2.2. Attacking the problem. To tackle the problem of 'preference' of short documents against longer ones we decided to decrease L_D for longer documents. After some experimentation we found that for the measurement of the document length it is better to use equation (6) instead of equation (4).

$$L_{D_i} = \ln\left(\sum_{j=1}^{n} t_{ij}^2 + e - 1\right) \tag{6}$$

3.2.3. Comparison. An open discussion of the system was held after all the participants had completed their sessions.

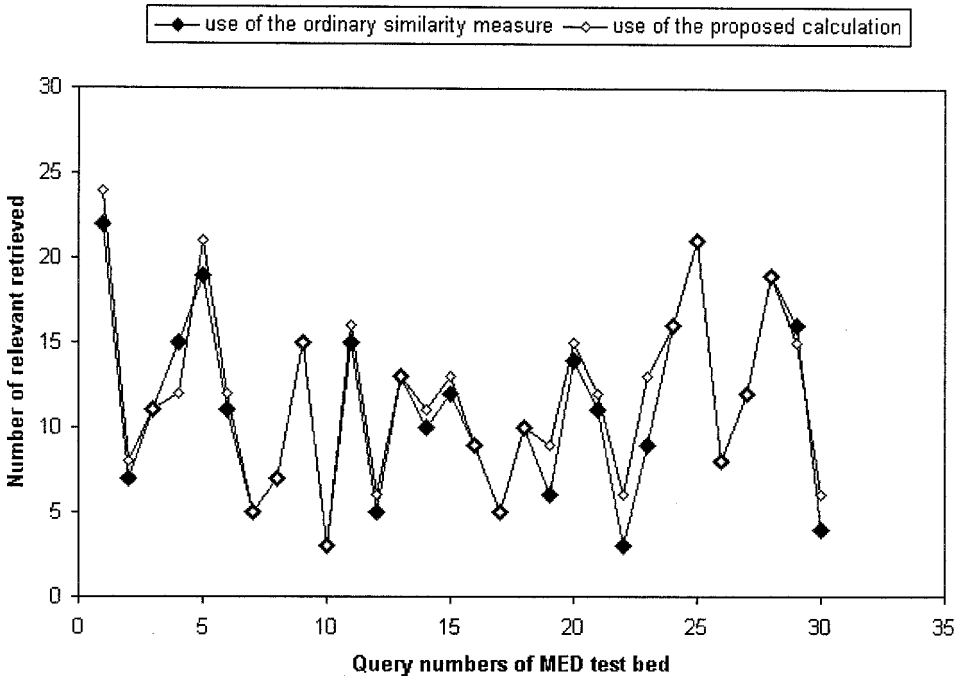

Figure 4. Comparison of the number of relevant retrieved.

Analysing the results of the various queries based on the test collection we confirmed that the similarity measure based on the equations (1), (2), (3), (4) and (5) presents a preference to short documents against longer ones but our test base is rather small.

Figure 4 gives us a comparison of the number of retrieved documents. The two versions of the systems were compared under the same conditions: We focused on retrieved documents having 'comparative relevance' of at least 40/100. As depicted at figure 4, the number of the retrieved documents with the use of the proposed calculation of the similarity measure is slightly greater (almost 10%).

4. Discussion

4.1. *Future work*

CAIRN is a constantly evolving system. Currently we are working to expand access to CAIRN. At present, CAIRN is a standalone, single-user system. We are working on a local area network version of CAIRN and a Netscape Communicator/ Microsoft Internet Explorer front end to the system. This means that we are working towards the development of an integrated environment that combines Internet and information retrieval facilities. Therefore, in this new version of the system, users will access selective resources on the Internet through the navigator, will send e-mails, will download and store the information into their database etc.

We also intend to experiment more with the proposed, new similarity measure. The OHSUMED test collection [29] and full text documents (e.g. Internet documents) will be used, to give us a more accurate evaluation of the new measure. We have also some new ideas in using other similarity measures for improving the

effectiveness of the system. There is also a plan for further investigation into Bayesian techniques.

4.2. *Conclusion*

The reactions of all involved with CAIRN has convinced us that this kind of accessible, easy to use, computerized personal full text information retrieval system can be a great benefit to the physicians, students, etc.

Compared to searching in ordinary library, in Internet resources through search engines, etc., the system is easily expandable, can have more information available, more effectively organized, and presented more conveniently.

Using a modular approach with 'Erevnitis' as the underlying information retrieval system has allowed us a great flexibility in designing the current interface, and gives us many possibilities for future enhancements. Using modular design has also made it possible to implement easily two versions of the system and conduct experimentation with alternative similarity measures. Therefore, CAIRN system can be easily modified to incorporate other similarity measures and take advantage of improvements suggested by users.

References

1. Mon, D., Herbst, M. R., and Nunn, S., 1998, Data resource administration. The road ahead. *Journal of American Health Information Management Association*, **69** (10).
2. Baker, L. D., and McCallum, A. K., 1998, Distributional clustering of words for text classification. *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'98,* edited by W. Bruce Croft, Alistair Moffat, C. J. Van Rijsbergen, Ross Wilkinson and Justin Zobel (Melbourne, Australia: ACM Press), pp. 96–103.
3. Bharat K., and Henzinger M., 1998, Improved algorithms for topic distillation in a Hyperlinked environment, *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'98,* edited by W. Bruce Croft, Alistair Moffat, C. J. Van Rijsbergen, Ross Wilkinson and Justin Zobel (Melbourne, Australia: ACM Press), pp. 104–111.
4. Zamir O., and Etzioni O., 1998, Web document clustering: A feasibility demonstration. *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'98*, edited by W. Bruce Croft, Alistair Moffat, C. J. Van Rijsbergen, Ross Wilkinson and Justin Zobel (Melbourne, Australia: ACM Press), pp. 46–54.
5. Kowalski, G., 1997, *Information Retrieval Systems. Theory and Implementation*, 1st edn (Boston, USA: Kluwer Academic Publishers), ISBN 0–7923–9926–9.
6. Miaoulis, G., Skourlas, C., Christopoulou, A., and Xanthakis, S., 1992, New role of a medical documentation system. *Medical Informatics*, **17**, 165–178.
7. Miaoulis, G., Protopapa, E., Skarpetas, G., Skourlas, C., and Delides, G., 1993, Information retrieval for pathology information systems. *In Vivo*, **7**, 373–378.
8. Karanikolas, N. N., and Mantzaris, S. L., 1992, Innovative directions in information retrieval. *Proceedings of Hellenic Research on Mathematics and Informatics, HERMIS'92*, edited by Elias A. Lipitakis (Athens: Hellenic Mathematical Society), pp. 529–536.
9. Karanikolas, N. N., 1993, Pronominal and anaphor resolution. *Computing and Information Technology* (*CIT*) *Journal*, **1**, 213–224.
10. Salton, G., 1983, *Introduction to Modern Information Retrieval*, 1st edn (USA: McGraw-Hill), ISBN 0–07–054484–0.
11. Buckley, C., 1992, SMART ver. 11.0. ftp://ftp.cs.cornell.edu/pub/smart.
12. Harman, Overview of the fourth text retrieval conference (TREC-4). http://trec.nist.gov/pubs/trec4/overview.ps.gz.
13. MG public domain software for indexing and retrieving text, 1995. ftp://munnari.oz.au/pub/mg, http://www.mds.rmit.edu.au/mg/.
14. NextPage's Folio products, e.g. Folio 4 family of products for CD-ROM, LAN and WAN publishing. http://www.folio.com, or http://www.nextpage.com.
15. http://www.sirsi.com, e.g. The Hyperion Digital Media Archive System: http://www.sirci.com/Prodserv/Dma/dmatoc.htm/.
16. http://www.verity.com/products/index.html, e.g. Topic creator – Document Navigator.
17. Moumouris, N., 1995, The document retrieval system Erevnitis. *CHIP* (*Greek edition*), **11**, March, 60–61.

18. Papathanasiou, S., 1996, Text researcher. *Techniki Eklogi*, (*in Greek*), no 359, November, 100–101.
19. van Rijsbergen, C. J., 1979, *Information Retrieval*, 2nd edn (London, UK: Butterworths).
20. Lucarella, D., 1988, A document retrieval system based on nearest neighbour searching, *Journal of Information Science*, **14**, 25–33.
21. Frakes, W., and Baeza-Yates, R., 1992, *Information Retrieval Data Structures and Algorithms*, 1st edn (USA: Prentice-Hall), ISBN 0–13–463837–9.
22. Thttp://trec.nist.gov.
23. Porter M. F., 1980, An algorithm for suffix stripping. *Program*, **14**, 130–137.
24. Bentley, J. L., Weide, B. W., and Yao, A. C., 1980, Optimal expected time algorithms for closest point problems. *ACM Transactions on Mathematical Software*, **6**, 563–580.
25. Smeaton, A. F., and van Rijsbergen, C. J., 1981, The Nearest Neighbour problem in information retrieval. An algorithm using upperbounds. *ACM SIGIR Forum* 16, pp. 83–87.
26. Borlund and Ingwersen, 1998, Measure of relative relevance and ranked half life: performance indicators for interactive IR. *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'98*, (Melbourne, Australia: ACM Press), pp. 324–331.
27. Hersh, W., Buckley, C., Leone, T., and Hickam, D., 1994, Ohsumed: an interactive retrieval evaluation and new large test collection for research. *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'94*, edited by B. Croft and C. Van Rijsbergen (Dublin, Ireland: ACM Press), pp. 192–200.
28. The med.all collection at the IDOMENEUS server. http://www.dcs.gla.ac.uk/idom/ir_resources/test_collections/.
29. OHSU Health Informatics Program, Division of Medical Informatics and Outcomes Research, School of Medicine, Oregon Health Sciences University. ftp://medir.ohsu.edu/pub/ohsumed.