COMPARISON BETWEEN AD-HOC RETRIEVAL AND FILTERING RETRIEVAL USING ARABIC DOCUMENTS

GHASSAN KANAAN
Yarmouk University
Irbid-Jordan
ghassank@yu.edu.jo

RIYAD AL-SHALABI
Yarmouk University
Irbid-Jordan
shalabi@yu.edu.jo

ABSTRACT

We have selected 242 Arabic abstracts used by (Hmeidi and Kanaan, 1997); all of which involve computer science and information system. We have also designed and built a new system to compare two different retrieval tasks: Ad-hoc retrieval and filtering retrieval. However, we have defined Ad-hoc and filtering retrieval systems and illustrated the development strategy for each system. We have compared both tasks using recall/precision evaluation, system usability, searching domain, ranking, construction complexity, and methodology.. From this experiment, we conclude that Ad-hoc retrieval is better than filtering retrieving. We, also, take in our account the advantages of using filtering services in information retrieval process.

The objective of this research is to automate the process of examining documents by computing comparisons between the representation of the information need (the Queries) and the representation of the documents. Also, we will automate the process of representing information-needs as user-profiles by computing the comparison with the representation of documents.

The automated process is considered successful when it produces results similar to those produced by human comparison of the documents themselves with actual information need. However, as a result, we will compare ad-hoc retrieval and filtering retrieval tasks and conclude the differences between them in term of information retrieval process.

INTRODUCTION

Information retrieval (IR) deals with the representation, storage, organization of, and access to information items. The representation and organization of the information items should provide the user with easy access to the information in which he is interested. Unfortunately, characterization of the user information-need is not a simple problem [3]. However, user-needs might be represented as a query to the repository of information available. In addition, information-items might be delivered to the user in an intelligent way, depending on a pre-defined user profile.

In a conventional information retrieval system, the documents in the collection remain relatively static while new queries are submitted to the system. This operational mode has been turned into ad-hoc retrieval in recent years, and become the most common for user-tasks [3]. In other words, the ad-hoc retrieval system is based on querying the information-items (documents) directly by the user and getting the relevant documents as a result to specified query. The most popular example of ad-hoc retrieval is the internet search engines.

Ad-hoc is assumed to be dealing with the problem of helping a user to find information related to a current and specific problem. It attempts to represent current, rather than long term information needs [7].

A similar but distinct task is one in which the queries remain relatively static while a new document comes into the system (and leaves); stock-market and news wiring services are good examples of this task. This operational mode has been turned into filtering [3]; formerly called SDI (selective dissemination of information) or current awareness [7]. However, users of this type of systems retrieve the relevant documents depending on a pre-defined information about user-favorites.

Filtering was one of the earliest application areas of mechanized IR. It works by having users construct long-term information need representations, which are periodically compared to new information objects. Filtering is a long-term modeling of user preferences, including search result characteristics and typical information problems. It can be accomplished through direct elicitation and observation of user behavior [7].

The filtering task simply indicates the documents which might be of interest to the user. The task of determining which one is really relevant is fully reserved to the user. However, ranking of the filtered documents is not even provided in this task. A variation

of this procedure is to rank the filtered documents and show this ranking to the user. The motivation is that the user can examine smaller number of documents if he assumes that the ones at the top of this ranking are more likely to be relevant. This variation of filtering is called "Routing", but it is not popular [3].

Information Retrieval system must be concerned with both short-term and long-term characteristics of the actors in the system. This implies either two kinds of representations, or highly flexible representation types [7].

The promise of the information age entails making information available to people any time, any place, and in any form. Realizing such a promise depends on innovations in areas that impact the creation of information services and their communication infrastructures. However, this realization can easily become a mixed blessing without methods to filter and control the potentially unlimited flux of information from sources to their receiving end-users [5].

The basic symmetry/duality argument arises if we assume that in some sense, documents and queries are similar kinds of objects, or that they are at some level interchangeable. Given such an assumption, any statement we make about documents and queries has a dual statement in which the roles of documents and queries are interchanged [6].

Some statements are self-dual: that is, the interchange will leave the sense of the statement unchanged. Others have duals meaning quite different things – indeed the dual may be incompatible with or contradicting to the original. By 'statement' here we mean to include theories or models, empirical observations, system or function descriptions, etc.

In the present context, the following pair of dual statements indicates the relationship between ad-hoc retrieval and filtering [6]:

- Maintain a collection of documents. When a new query comes along, we search the collection, and identify appropriate documents for this query.
- Maintain a collection of queries. When a new document comes along, we search the collection, and identify appropriate queries for this document.

Over the course of many instances of comparison, ad-hoc retrieval (or Retrospective retrieval) is increasingly becoming more accurate than filtering retrieval. The information need is assumed to be one-time rather than long term. Ado-hoc retrieval concerned with a single information seeking and a single query is compared to static document collection rather than a single document compared with static queries collection. [7].

LITERATURE REVIEW

Because it is the most commonly used, many researches and projects were implemented previously about ad-hoc retrieval and its related developments. However, internet search engines are the most popular examples of such systems. AltaVista is one of thousands search engines created in the last decade; it is based on ad-hoc retrieval like all other search engines available on the web. In this engine, ranking documents is based on vector space model.

Franz, Scott and Roukos [1] specify an implementation strategy for developing an ad-hoc retrieval system. They identify how to create a multi-lingual ad-hoc retrieval system and compare the performance between the traditional ad-hoc system and one with multi-lingual property.

Christian Michel [2] distinguishes three types of ad-hoc system evaluation: adequacy evaluation, diagnostic evaluation and performance evaluation. However, he also identifies some problems arise when performing these evaluations and proposes some solutions for it.

Stephen Robertson [6] identifies in a theoretical way a comparison between ad-hoc and filtering retrieval systems. However, his comparison is based on precision evaluation, ranking output and a system based on phrases statements.

Nick [7] creates a comparison between ad-hoc retrieval and filtering retrieval. However, he found that ad-hoc retrieval systems can retrieve more accurate results compared with filtering retrieval. Nick used the task properties in his comparison such as: information need (short vs. long term), query types, and document collection types.

The demand for information filtering technology is not new, on the other hand, this technology is not limited to new information services. Over a decade ago, Peter denning's ACM President's Letter on "Electronic Junk" (Common. ACM, March 1982, 163-165) focused on the implications of automatic document preparation systems and electronic mail, and on the quantity of information being received by end users.

Peter denning pointed out that "The visibility of personal computers, individual workstations, and local area networks has focused most of the attention on generating information--the process of producing documents and disseminating them. It is now time to focus more attention on receiving information--the process of controlling and filtering information that reaches the persons who must use it " [5].

In November 1991, Bellcore hosted a Workshop on High Performance Information Filtering in

Morristown, N.J. Organized and sponsored by Bellcore in cooperation with ACM SIGOIS. This workshop was the first of its kind. The event brought together over one hundred researchers from major universities and industrial research labs, who share a strong interest in the creation of large-scale personalized information delivery systems [5] .

In "Using Collaborative Filtering to Weave an Information Tapestry," Goldberg, Nichols, Oki, and Terry describe an experimental system that manages an in-coming stream of electronic documents, including email, newswire stories and Netnews articles. The system implements a novel mechanism for collaborative filtering in which users annotate documents before the documents are filtered. Because annotations are not available at the time a new document arrives, the system supports continuous queries that examine the entire database of documents and take into account newly introduced annotations during the filtering process [5].

DESCRIPTION

In the meantime, we will implement the two retrieval tasks so that for each query the ad-hoc system will retrieve automatically the relevant documents to this query. In addition, for each profile, the system will automatically assign the relevant document to a specified user-profile.

For ad-hoc system, we use a natural language queries written in Arabic. As an example of them " التعليم بواسطة الحاسوب" (Learning using computer), which querying for all documents related to the education using computers. On the other hand, we create users-profiles consist of all important information related to each user. Such information must represent a way for the system to know users' favorites in the document collection. The following is an example of user-profile:

User Name: A. Nour	اسم المستخدم: علي أحمد نور
Place of birth: Jordan	مكان الولادة: الأردن
Residence place: KSA	مكان الاقامة: السعودية
Job Title: Sys. Analyst	مسمى العمل: محلل نظم
Educational Major: CS	التخصص العلمي: علوم حاسوب
Knowledge fields:	حقول المعرفة: ١) نظم خبيرة
1. Expert Systems	٢) مترجمات لغات البرمجة
2. Compilers	
3. Database	٣) قواعد البيانات
4. Arabic Language	٤) اللغة العربية

Words which are too frequent among the documents in the collection are not good discriminators; in fact, a word which occurs in 80% of the documents in the collection is useless for purposes of retrieving. Such words are frequently referred to as stop-words and are normally removed [4]. In our system, we remove all stop-words from all documents to enhance the retrieval process.

Stems are thought to be useful for improving retrieval performance because they reduce variants of the same root word to a common concept. For example, the root "حسب" (Compute) is a common concept for the following words "حاسـوب" (Computer), "حاسـبات" (Computers), "حوسـبة" (Computing) and "حـساب" (Computations). In our system, we convert the documents into stemming form.

At this point, we need to index the document collection into a form that facilitates calculations and comparisons. We choose an inverted file indexing mechanism, which is a word-oriented mechanism for indexing a text collection in order to speed up the searching task [4]. However, the inverted file structure will consist of the following fields:

Root	Doc NO	Freq _ijR	Max_Freq _ijR	NumOf Docs	ni	wijR
حسب	**1**	**1**	**10**	**242**	**4**	**?**

Automatic retrieval is based on a specific model that indicates the similarity between each document and the corresponding query or user-profile. However, vector space model is one of the most efficient models for computing similarities and, therefore, ranking the retrieved documents.

In ad-hoc systems, query and document represented as vectors in a vector space, and a comparison technique based on the assumption that documents whose representations are similar to the query will be likely to satisfy the associated information-need used. The angle between two vectors (query and document) has been found to be a useful measure of content similarity [4].

However, in filtering retrieval system the same process will be performed, but the distinction between them will be the representation of the profile vector. Profile vector consists of a collection of all profile properties rather than a user query. The similarity will be calculated for each property separately [4].

One common schema, known as "Term-Frequency … inverse document frequency" weighting, assigns term (**i**) in document (**j**) a value computed as [4]:

$$W_{ij} = \frac{\text{Frequency}_{ij}}{\text{Max}_l \text{Frequency}_{lj}} * \log_2 \left(\frac{\text{Number of Docs}}{\text{Number of Docs with term}_i} \right)$$

This formula is used to compute the weight for each document and queries.

$$W_{iq} = 0.5 + 0.5 * \left[\frac{\text{Frequency}}{\text{Max}_l \text{Frequency}_{li}} \right] * \log_2 \left[\frac{\text{Number of Docs}}{\text{Number of docs with term}_i} \right]$$

Based on the previous formulas, we will create a file containing each term in the documents' collection corresponding to its related weight. Also, another file will be created for the terms in the queries.

At this point, we will apply every query on the system and calculate the similarity between the query and each document in the collection (rank the documents according to their degree of similarity to the query); a document might be retrieved even if it only matches the query partially [3]. On the other hand, we will compare every coming document to the system with user-profiles and calculate the similarity between them. However, similarity will be calculated based on the following cosine-similarity formula:

$$\text{Similarity}_{ij} = \frac{\sum (W_{ij} * W_{iq})}{\text{Sqr}(\sum (W_{ij})^2) * \text{Sqr}(\sum (W_{iq})^2)}$$

(please revise the statement and rephrase if needed)

It is noted here that ranking of documents in the ad-hoc retrieval system is based on the similarity values while in filtering retrieval system there is no documents' ranking. Recardo Baeza mentions in his book that filtering routing task in which the retrieval algorithm has only to decide whether a new incoming document is relevant or not, does not require ranking of the documents-taken (page 22 line 13 and page 90 line12) [3]. In other words, in filtering system, while the documents are coming to the system and being processed one by one; the system will calculate their similarities to the profile and judge whether they are acceptable or not. So, we can't rank documents based on their similarity. For example, in the e-mail system when a relevant document is delivered, the e-mail system adds it to the in-box folder, and filters it according to the time of entering the system.

DESCRIPTION STEPS:

1. Removing stop-words: we expose all documents to an efficiently-working algorithm that removes the Arabic stop-words.
2. Extracting Stems for each term in the documents: we expose all terms in the documents to a stemming algorithm that translates any Arabic word into its correspondent root.
3. Creating the inverted file: we use an Access database as a data repository for inverted file information.
4. Calculating weight for each term and storing it in the inverted file: we calculate the weight for each term in the inverted file using vector space model equations and store it in a separate column in the database.
5. Calculating the degree of similarity between each document and the corresponding query or profile using cosine similarity equation.

Finally, we will run the system and retrieve the relevant documents to each query and profile automatically. From this result, we will compare the manual retrieval with automatic retrieval using recall-precision evaluation measurements. However, based on recall-precision results, a comparison will be made between the efficiency of ad-hoc and filtering retrieval tasks.

The formulas used for computing recall and precision measurements are:

$$\text{Recall} = \frac{\text{(Number of Relevant Docs Retrieved)}}{\text{(Total \# of Relevant Docs)}}$$

$$\text{Precision} = \frac{\text{(Number of Relevant Docs Retrieved)}}{\text{(Total \# of retrieved Docs)}}$$

We average the precision figures at each recall level as follows:

$$P(r) = \frac{\sum \text{Pi}(r)}{Nq}$$

$$\text{Where, } P(r_j) = \text{Max } r_j <= r <= r_{j+1} \; P(r)$$

TESTING

To compare the two user-tasks, ad-hoc and filtering, we have used the manual results to apply each one on an available collection of documents. However, the system will be tested using a set of 242 Arabic abstracts from the proceedings of the Saudi Arabian National Computer Conferences. On the other hand, we create 60 Arabic queries for ad-hoc retrieval and 20 users-profiles for filtering retrieval.

Because part of the comparison is based on the efficiency of retrieval process, we assign to each query

the corresponding relevant documents manually. More over, we perform the same assignment to the user-profiles by specifying the relevant documents to each profile. For example Q1 is relevant to D1, D2, D3, and D4.

After running the two systems, we compared the automatic results with manual ones. These comparisons are based-on recall/precision evaluation measurements. However, for each query and profile we calculate recall and precision values. In addition, we calculate the average recall/precision for each system.

Additional runs have been made to choose the best thresh-hold. Using a good thresh-hold will eliminate junk documents, which are retrieved but not relevant to a specific query or profile. We found that 0.03 thresh-hold is the best selection for the evaluation retrieval. In the system implementation, we make the thresh-hold choice optional to the user by allowing him to choose any thresh-hold value.

The two systems are also compared in terms of response time. Response time is the time needed for searching, calculating, and retrieving resulted documents. However, because the two systems are running on the same computer, we can compare the average response time for each of them.

RESULTS

After we have computed the recall and precision for each query, we can show the graph of recall/precision for each query. However to show the performance of information retrieval system, an average recall/precision for several distinct queries should be evaluated.

We use this formula to average the precision for each of the 11 recall levels:

$$P(r) = \frac{\sum P(i)}{Nq}$$

Where p(r) is the average precision at the recall level r, N_q is the number of queries used, and p (r) is the precision at recall level r for each of the 60 queries. The average recall/precision for ad-hoc retrieval system is shown in figure 1.

The figure shows for an example that we have 68 % precision at recall level of 20 %, and a 60 % precision at recall level of 50 %.

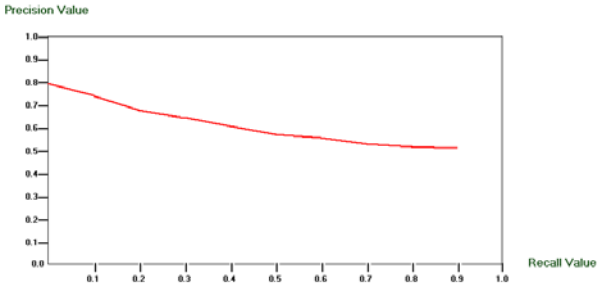

Figure 1 Average Recall / Precision Ad-hoc

In the filtering retrieval system we draw the graph for each user profile, however; to show the performance of information retrieval system, an average recall/precision for several distinct user profiles should be evaluated. As mentioned before, we use the profiles of the 20 users to compute the average recall and precision, figure 3 show the average recall/precision of the filtering system.

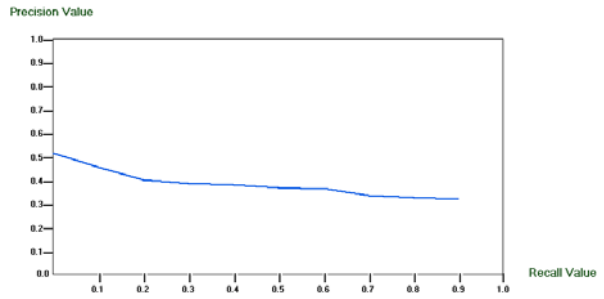

Figure 2 Average Recalls / Precision Filtering

Average recall precision is used to compare with the retrieval system. Here we use recall/precision to compare the ad-hoc retrieval system with filtering retrieval system. Figure 3 illustrates the average recall/precision for the distinct system. From this figure we can see that filtering retrieval system has a higher precision at lower recall levels, while the ad-hoc retrieval system has a higher precision at higher recall levels

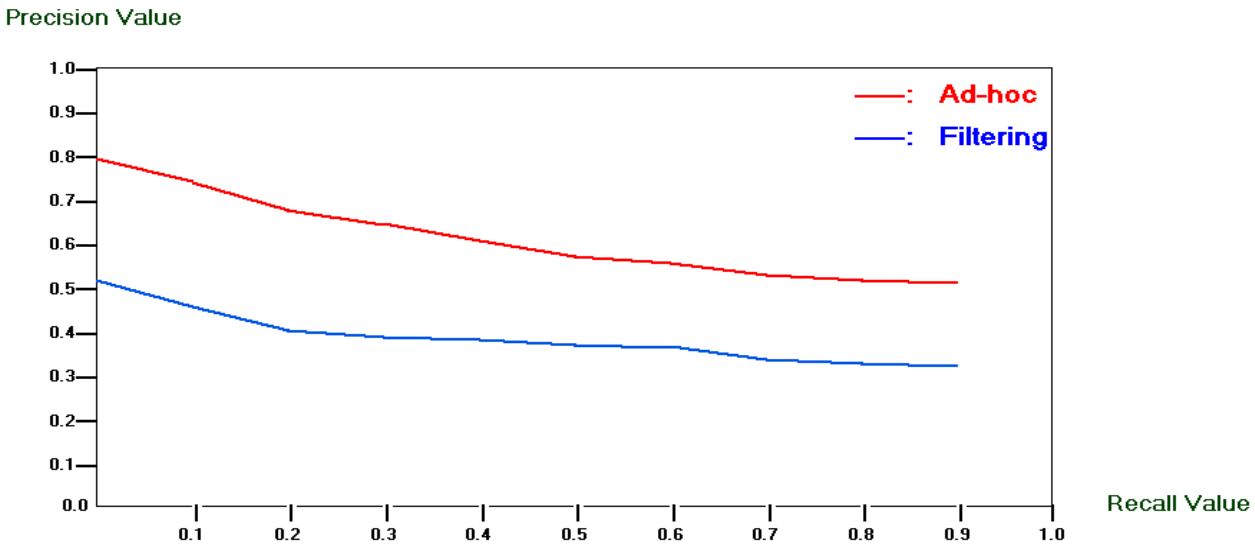

Figure 3 Average Recall / Precision Ad-hoc, Filtering

CONCLUSION

We can compare ad-hoc and filtering retrieval tasks from many different angles. However, system performance and retrieval efficiency is not the only measurements to compare these different tasks. In addition, there are some properties for each task that makes it more preferable than the other in certain situations.

Based-on the previous experiment, we can conclude that ad-hoc retrieval is better than filtering retrieval in term of recall/precision evaluation. However, in the average recall/precision graph for the two systems, it is obvious that there is a gap between the two curves. This gap generated between the two curves in figure (3) comes from the following reasons:

1. The sample documents collection belongs to the same domain, Computer Science. This type of collections makes it difficult to distinguish the power of each task.
2. Synonyms and roots enhance the retrieval of the systems especially filtering retrieval system, which depends heavily on using such tools.
3. No ranking in filtering retrieval system. This decreases precision values at the beginning intervals.

The question arising here is whether recall/precision is the only measurement to compare these retrieval tasks or not? Absolutely the answer is not. Ad-hoc retrieval systems have many advantages over filtering retrieval; these advantages are:

1. Filtering retrieval is domain specific; the retrieval depends only on what the user specifies in his profile. There is no profile structure that allows user to type anything he wants. However, the idea of creating profiles is to reduce the domain of searching, so such systems usually provide choices to the users to select between them. On the other hand, ad-hoc retrieval provides an open domain to the user; the user specifies what he needs as query without any restriction [3].
2. In ad-hoc retrieval systems, retrieved documents are ranked based-on some similarity measurements (like cosine similarity) while there is no ranking in filtering retrieval systems. However, from the definition of filtering retrieval, the documents come to the system one by one; the system, then, computes their similarity and decides whether they are relevant or not. No ranking algorithm can be applied to judge which document is more relevant than the other [3].
3. Ad-hoc retrieval systems have the advantage of high usability than filtering retrieval systems because it is most commonly used. This result documented by nick [7] comparison.
4. The construction of filtering retrieval systems is more complex than the construction of an ad-hoc one. This is because filtering retrieval systems are built based-on a huge profiles database rather than simple text-query in ad-hoc systems. We derived this fact from the implementation of the two systems.

Despite of these advantages of ad-hoc retrieval, there are many advantages of filtering strategy. However, delivering users relevant documents without querying it is a high intelligent mechanism for publishing such documents. In addition, filtering is not only used for seeking related documents or information. Many useful and important systems depend on this type of systems. However, e-mail and news-wiring services; for example, have become very important in our life; they have changed the concept of communication services in the world. People can send letters to recipients in any country in the world in few seconds.

Another distinction between these retrieval tasks is the development methodology. In ad-hoc system, we compare the coming queries with the documents collection available. On the other hand, filtering retrieval is based-on comparing the incoming documents with those queries specified in each user profile. High-number of user-profiles in the system database decreases the speed of searching, and consequently, the retrieval process in terms of real delivery time.

REFERENCES:

[1] Martin Franz, J.Soot McCarley, Salim Roukos.1999, Ad hoc and Multilingual Information Retrieval at IBM, IBM T.J Watson Research Center

[2] Christine MICHEL, Diagnostic Evaluation of a personalized filtering information retrieval system. Methodology and experimental results.

[3] Ricardo Baeza-Yates & Berthier Ribeiro-Neto, 1999, "Modern Information Retrieval". Addison Wesley.

[4] Douglas W. Oard, 1997, "Information Retrieval". www.ee.umd.edu/medlab/filter/papers/filter/node9.htm.

[5] Shoshana Loch and Doug Terry, 1992, "Information Filtering". http://www.cs.washington.edu/homes/kepart/index.html.

[6] Stephen Robertson, 1999, "Salton Award Lecture: On theoretical argument in information retrieval". Microsoft Research, St George House, 1 Guildhall Street, Cambridge CB2 3NH.

[7] Nick, 1997, "User Modeling in Information Retrieval". Available at: http://www.scils.rutgers.edu/~belkin/um97oh/ .

[8] Hmeidi, I., Ghassan K., Evans, M. 1997. "Design and Implementation of Automatic Indexing for information Retrieval with Arabic Documents. Journal of the American Society of information science. 48 (10): 867-881.

APPENDIX A

The following are the queries used to run this system:

١. استخدام الحاسب الالي
٢. استرجاع المعلومات
٣. الادارة و التخطيط
٤. التدريب و التعليم
٥. الترميز و التشفير
٦. التعليم بمساعدة الحاسب
٧. التعليم بواسطة الحاسب
٨. الحاسب الالي
٩. الحاسبات الصغيرة
١٠. الحاسبات المتناهية الصغر
١١. الحاسوب و التعليم
١٢. الحج و العمرة
١٣. الحرف العربي
١٤. الخطة الوطنية للمعلوماتية
١٥. الخليج العربي
١٦. الدوائر المتكاملة
١٧. الذكاء الاصطناعي
١٨. الذكاء الالي
١٩. العالم العربي
٢٠. القران الكريم
٢١. الكلمات العربية
٢٢. اللغات الطبيعية
٢٣. اللغة العربية
٢٤. المدرسة الالكترونية
٢٥. المملكة العربية السعودية
٢٦. الموارد البشرية
٢٧. النص العربي
٢٨. امن المعلومات
٢٩. انظمة الحاسبات الالية
٣٠. برامج الحاسب الالي
٣١. برمجة الحاسبات الالية
٣٢. بنوك المعلومات
٣٣. تدريس مواد الحاسب
٣٤. تركيب الجملة العربية
٣٥. تطبيقات الكومبيوتر
٣٦. تعريب البرامج
٣٧. تعريب الحاسبات الالية
٣٨. تعريب الحاسوب
٣٩. تعليم الكومبيوتر
٤٠. تقنية المعلومات
٤١. تمييز الاشكال بواسطة الحاسب
٤٢. جامعة الملك سعود
٤٣. جامعة الملك عبدالعزيز
٤٤. جمعية الحاسبات السعودية
٤٥. شبكات الحاسب الالي
٤٦. شبكة اتصالات الحاسبات
٤٧. شبكة الاتصالات
٤٨. علوم الحاسب و المعلومات
٤٩. قواعد البيانات
٥٠. قواعد المعلومات
٥١. لغة برمجة عربية
٥٢. مجتمع المعلوماتية
٥٣. محاكاة الحاسب الالي
٥٤. مهارات استخدام الكومبيوتر
٥٥. نظم خبيرة
٥٦. نظم المعلومات
٥٧. هندسة البرامج
٥٨. هندسة الحاسب الالي
٥٩. هندسة الحاسوب

APPENDIX B

The following are sample documents from the document collection used for building this system:

ببرقم ١
صنف الحاسبات الآلية - لغات
نوع مؤتمر
عنو نظرية للاشتقاق الآلي في النص العربي غير المشكول
مؤل الفداغي ، صباح ، ياسين ، مصطفى
جهه قسم الهندسة الكهربائية والكمبيوتر ، جامعة الكويت ، الكويت ، قسم الرياضيات ، جامعة الكويت ، الكويت
عنم التخطيط لمجتمع المعلوماتية ، المؤتمر والمعرض الوطني الثاني عشر للحاسب الآلي ، جامعة الملك سعود ، الرياض
مجل ١
صفح ٦٦٠ - ٦٧٤
نشر ١٤١١ هـ
لغه العربية
ملخ تقدم هذه الورقة تحليلا لعملية الاشتقاق الآلي في النص العربي غير المشكول ، وتبين الشروط الواجب توفرها في المحلل الاشتقاقي ، ومن اهمها رفض المفردات القياسية التي لها جذر ووزن عربي ولكنها مفردات غير عربية مثل كلمة حشرون ، وكذلك بناء المحلل على اسس رياضية بحيث يمكن تنفيذ ذلك البناء بواسطة الاخرين . تقوم النظرية على اساس بناء جدول ارتباط بين الجذور من ناحية القواعد اصغر مفرده ذات معنى ، والقمم اكبر مفرده ذات معنى من ناحية اخرى ، ومن ثم توليد المفردات العربية الواقعة بين القواعد والقمم . تمت تجربة القانون التوالدي على مفردات القرآن الكريم والتي بلغ عددها ١٢٨٥٣ مفرده وقد تم استخلاص ٢٥٦ قاعدة و ١٢٠٨ قمم وادى القانون التوالدي الى انتاج ٣٧٦٧٧ مفرده وجد ان ٩٦ في المئه منها هي مفردات عربية صحيحة . ان هذه الطريقة تعني ان تخزين القواعد والقمم وارتباط تلك القواعد والقمم بالجذور يعتبر كافيا من الناحية العملية للتعرف على المفردات الواقع بينها .

رقم ٢
صنف البرمجة
نوع مؤتمر
عنو تنفيذ لغة برمجة عربية على حاسب آلي مصغر
مؤل خياط ، محمد غزالي
جهه جامعة البترول والمعادن ، الظهران
ممو معهد الادارة العامة ، الرياض
عنم المؤتمر والمعرض الوطني السابع للحاسبات الالكترونية ١٨ - ٢٢ ربيع الثاني ١٤٠٤ هـ : سجل البحوث
صفح ٩٤ - ١٠٣
عدم ٢٠
نشر ١٤٠٤ هـ
ناش اللجنة الفرعية للبحوث والبرامج ، معهد الادارة العامة ، الرياض
لغه العربية
ملخ ان استخدامات الحاسبات الآلية الحالي باللغة العربية تعتمد اساسا على برامج مكتوبة بلغة برمجة اجنبية . وعلى هذا فان ذلك يمثل تقدما سطحيا في مجال تعريب استخدامات الحاسبات الآلية . ولذا فانه يتوجب تطوير لغة برمجة عربية لتمكين المستخدم والمبرمج من استخدام الحاسبات الآلية باللغة العربية دون اللجوء الى استعمال لغة اجنبية . نقدم في هذا البحث تصميم وتنفيذ مترجم للغة برمجة عربية . لغة الضاد تم تطويرها بدعم من المركز الوطني للعلوم والتكنولوجيا (مشروع ات ٤٥٥) ويقوم المترجم بترجمة برامج مكتوبة بلغة البرمجة المفتوحة الى سلسلة من التعليمات التي يمكن ان يقوم الحاسب الآلي بتنفيذها . ويجب ان
تتوافر في المترجم عدة صفات من اهمها الوضوح وسهولة وانتظام التركيب وامكاني استخدامه على حاسبات آلية اخرى باقل قدر من التغييرات والتعديلات .