Toward a Context-Aware Multilingual Personalized Search

Mohamed Seghir Hadj Ameur(✉), Youcef Moulahoum(✉), Lamia Berkani(✉), and Ahmed Guessoum(✉)

Laboratory for Research in Artificial Intelligence (LRIA), NLP, Machine Learning and Applications (TALAA) Group, Department of Computer Science, University of Science and Technology Houari Boumediene (USTHB), Bab-Ezzouar, Algiers, Algeria
{mhadjameur,lberkani,aguessoum}@usthb.dz, moulahoum.youcef@gmail.com

Abstract. In recent years, personalized search has widely been used in Information Retrieval Systems (IRS) to provide the end user with more sophisticated and accurate search results. A basic element that plays an important role in personalized search is the user context which contains several aspects such as the user preferences, navigation history, habits, etc. A user may express his information needs in various languages. This requires the IRS to be able to consider all the contextual information provided in these languages. In this work, we present M-CAIRS, a Multilingual Context-aware Information Retrieval System that takes into account multilingual user contexts to better model the user search interests. Experimental results show a strong correlation between the user's relevance judgment and the automatic results obtained by our system, which proves the consistency and adequacy of our proposal.

Keywords: Information retrieval · Multilingual information retrieval
Reference ontology · Document indexing · User context · User profile
Relevance judgment

1 Introduction

In recent years, the amount of information available on the web has seen an exponential growth. According to the Internet live stats website[1], there were at least 1.2 billion websites on the indexed web as of May 2017, and in every second, approximately more than 60,000 Google queries are launched. This explosion in both the amount of data and the launched queries has made it hard for Information Retrieval Systems (IRS) to accurately find and identify relevant information that can address the users needs and preferences in a precise manner.

Search engines are one of the most popular tools to find information on the web. Classical search engines return the same results to different users (one size

[1] Internet live stats is a website that provides live statistics regarding the Internet http://www.internetlivestats.com.

A. Lachkar et al. (Eds.): ICALP 2017, CCIS 782, pp. 175–190, 2018.
https://doi.org/10.1007/978-3-319-73500-9_13

fits all) even though each one of them has a distinct context and a specific goal when searching for information. This generally ends up providing the users with some irrelevant results that fail to address their specific information needs. The problem of query ambiguity is also one of the main reasons for search quality decay [1]. The query ambiguity is due to several reasons including polysemy (a single word may have multiple meanings), and synonymy (different words may have the same meaning). For example, a user who searches for the word "apple" may be interested in either a fruit or a company. Therefore, in order to solve these problems, the information retrieval community has made a huge focus on personalization. Personalized Search aims to reduce the queries ambiguity and return the most probable results that are more likely to be of interest to the user based on specific user modeling techniques.

User Modeling aims to build an adequate representation that models the user's interests either individually [2] or as part of a community [3] that shares similar preferences. When user modeling techniques are incorporated, the search process is generally called Personalized Search; which has been widely used in several domains such as Information Retrieval [2,4–7], Recommender Systems [8–10] and many applications such as e-learning [11,12] to provide the end user with more relevant personalized services. One of the main elements that play an essential role in personalized search is the user context. When an IRS takes into account the user context it is called Context-aware Information Retrieval System (CIRS). Two factors are important in contextual information retrieval: (1) the user's short-term context, which includes his requests and various aspects such as spatio-temporal information, and (2) the user's long-term context which includes his interests, preferences, knowledge, habits, expertise, etc. An important issue that needs to be addressed when attempting to model the user's context is the problem of incorporating all the languages he uses in his queries. Indeed, if a user queries the web in various languages, for example English, Arabic and French, the CIRS should be able to effectively model and maintain this user's preferences and interests in each one of these languages.

In this work, we build M-CAIRS a complete context-aware information retrieval framework that effectively models the user's long term interests. Our proposal is based on the work of Sieg et al. [13] and Gupta et al. [14] and proposes improvements in profile updating and results re-ranking. Furthermore, one of the main contributions of this work is the proposition of a method to effectively incorporate multiple languages including Arabic in the ontological user profile modeling framework.

The remainder of this paper is organized as follows: the next section presents the background and some relevant related works that have been done in this area and motivates our contribution. The details of our proposal are then described in Sect. 3. In Sect. 4, we present and discuss the tests we have done and the results we have obtained. In the last Section, we conclude our work and highlight some possible future improvements.

2 Background and Related Work

In this section we define some important concepts that will be used in the remainder of this paper, then we shed light on some of the relevant works that have been made in this research area.

2.1 Background

The notions of context and profile may have several definitions depending on the application at stake. Here we give our own definitions of these concepts along with other relevant concepts that we will be using in this paper.

User Context: We define the user context as all the information about the user that can be used to improve the personalized retrieval process. Two types of user contexts are distinguished: (1) the short-term context, which includes the user's requests and various aspects such as spatio-temporal information (i.e. geographical position, time, etc.), and (2) the user's long-term context, widely known as the user profile, which contains useful information about the user such as his interests, preferences, knowledge, habits, expertise, search history, etc.

User Profile: We define the user profile as a source of knowledge that holds the user's long-term context. A certain structural representation is generally used to store, maintain and update a user profile according to the changes that occur in his interests and preferences.

The Open Directory Project (ODP)[2]: It is one of the largest existing web directories, developed and maintained by a vast community of editors (over 90,000 publishers). It contains about 4 million websites distributed into a hierarchy of over 1 million manually created categories (concepts)[3], where each concept contains a set of manually associated websites. The first level of this hierarchy contains generic concepts such as: Arts, Computers, Sport, etc. These concepts become more specific as one goes deeper and deeper into the hierarchy.

The Reference Ontology [15]**:** It is an instance of a preexisting hierarchy such as the ODP. This ontology is generally used to represent the user profile as a hierarchy of concepts in which each concept is associated with an interest score which indicates the degree to which the user is interested in this concept. This user profile representation is very useful for personalization to keep track of the user's interests and update them according to the user's daily activities.

[2] http://dmoztools.net/.

[3] The nodes of the ontology (hierarchy) are generally called concepts.

2.2 Related Work

The task of personalizing the search results is a very complicated process that includes several important steps, the major steps are: collecting the users' information, building and updating their profiles and re-ranking the search results according to the profiles being built. This section explain these steps and presents some of the most relevant works which attempt to address each one of them.

The first step is to collect the user's information which can be done either explicitly or implicitly. The gathering of explicit user data generally asks the users to provide their areas of interests, their preferences and/or their (positive/negative) feedbacks regarding the returned search results they are provided with. An example of such a system is the work of Syskill and Webert [16] in which the user is explicitly asked to rate web pages, and based on his feedback, a software agent learns to decide which page might be of interest to a specific user. Implicit data collection [17–20] on the other hand is done automatically, where the user context (clicks, bookmarks, search history, desktop information, etc.) is collected without any user intervention.

The second step is to build and maintain the user profiles. User profiles are generally represented based on keywords also known as "Keyword profiles" or concepts known as "Concept profiles". Keyword profiles [21] are created by extracting keywords from a set of documents, web pages and/or bookmarks visited by the user. They are stored as part of his browsing history. The most important keywords on a web page are identified using some specific weighting methods, and only the most highly weighted terms are kept. An example of such a system is the one presented by Moukas and Alexandros [22] in which extracted keywords were weighted using the $TF–IDF$ measure [23] and a vector representation was used to represent both the user profile and the retrieved documents. Concept profiles [20,21] are represented such that each concept represents an abstract domain. These concepts are usually driven from an existing hierarchy such as the ODP. A numerical value called "interest score" is generally associated with each concept to indicate the degree to which the user is interested in it. In terms of concept profiles Sieg et al. [13] constructed the user profile as an instance of a predefined ODP hierarchy. When a user interacts with the system by selecting or viewing new documents, the scores of each concept will be updated on the basis of its similarity with the viewed documents. A specific propagation algorithm is also used to allow activation weights to spread throughout the entire ontological profile.

The last step is to make use of the built profile to reorder the search results so as to better suit the user's interests. To that end, Sieg et al. [13] presented a re-ranking method that reorders the Search Engine results according to the user profile interest scores. Gupta et al. [14] proposed a similar approach with the incorporation of the original Search Engine ordering of the returned pages as a feature along side the user profile.

The amount of focus on personalized retrieval in regard to the Arabic language is very limited as stated in [24]. Some of the efforts in this area includes, for instance, the work of Houssem et al. [25] in which a query enhancement system is proposed to extend the users' search queries on the basis of their profiles and the Arabic Wordnet (AWN) [26][4], and the work of Safi et al. [24] in which a hybrid profile construction method is introduced to incorporate both implicit and explicit users' information using the AWN as a reference hierarchy. A special method is then used to update and maintain the conceptual interest scores in the built profiles.

Even with the amount of research work done on personalized information retrieval, no single strategy has seemed to yield ideal results, thus the continuing efforts to improve them. This work aims to bring further improvements to several aspects of this area such as multilingual retrieval, user profile building and maintenance, as well as results re-ranking.

3 Context-Based Multilingual Personalized Search

This section presents M-CAIRS, our proposed system architecture, and describes in depth the functioning mechanism of each of its components.

Figure 1 illustrates the different components of our proposed system, along with the interactions between them. There are two main tasks: the first one attempts to gather the user daily browsing activities and use them to build and maintain his ontological user profile; the second one re-ranks the search results corresponding to the user's query reflecting the learned user profile.

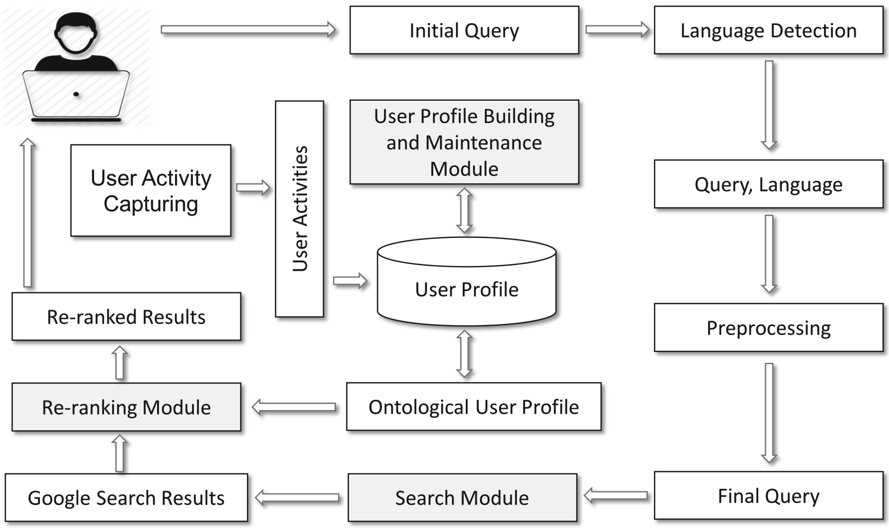


Fig. 1. The architecture of our proposed multilingual context-aware IRS (M-CAIRS)

[4] http://globalwordnet.org/arabic-wordnet/.

3.1 Building and Maintenance

The goal of this model is to gather the users' daily browsing activities and use them to build and maintain their user profiles.

First, we define the user profile as an instance of the ODP hierarchical concept database. The depth of the ODP hierarchy can reach up to 11 levels, which makes the concepts at the bottom very specific. We have used only the first two levels of the ODP hierarchy, which prevents the concepts from being too specific and keeps them relatively general. This, we believe, is more suitable for holding their long-term interests as shown in Fig. 2. We have considered only three languages: English, Arabic and French, which are the languages mostly used by the users we have investigated[5]. Since the ODP is principally an English-based hierarchy, all the other languages such as Arabic and French are found in the first level under the concept "World". We have tweaked the structure of the hierarchy to place the three languages at the top level as shown in Fig. 2. Figure 2 shows the reference ontology used in M-CAIRS. It includes a total of 548 concepts. We assume that our choice will keep the profile a little general yet suitable to hold the users long-term interests in each one of the three considered languages.

For each language, all the documents found under the same concept will be merged together to form a single super-document. These super-documents will then go through a preprocessing step which includes stemming, normalization and empty words removal according to each specific language. A vector space representation is then used to represent each concept by a vector of terms of length n where n is the vocabulary size in the considered language. The $TF\text{–}IDF$ weighting [23] is then used to produce a vector of weights for each concept.

$$TF\text{–}IDF_{i,j} = \frac{TF(t_i, d_j)}{max(TF(t, d_j))} * log(\frac{n}{n_i}) \tag{1}$$

$TF(t_i, d_j)$ is the frequency of the i^{th} term in the j^{th} document, $max(TF(t, d_j))$ is the highest term frequency in document j and $log(\frac{n}{n_i})$ is the inverse document frequency of term i in the collection where n is the total number of terms and n_i is the frequency of term i.

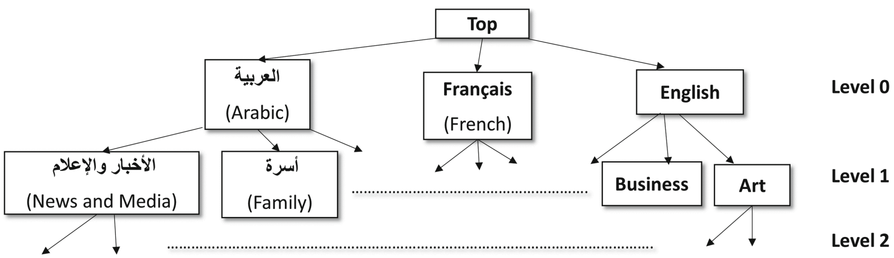


Fig. 2. The ODP reference ontology used in M-CAIRS

[5] We note that other languages can be integrated exactly in the same way.

When a user visualizes a document $d_j = \{t_1, t_2, ..., t_m\}$ containing m terms, this document will be similarly preprocessed and represented in a vector space representation as a vector of weights $w_1, w_2, ..., w_n$. Having this vector representation for both the concepts and the visualized documents allows us to easily estimate the similarity between them, which is very useful to update the user profile accordingly.

As mentioned above, the user profile is going to be an instance of the reference ontology shown in Fig. 2 with the addition of a specific weight called interest score $IS(c_i)$ which is associated to each concept c_i to indicate the user's level of interest in it. The following Algorithm 1 is proposed to update and maintain the user's interests scores:

Algorithm 1. Pseudo algorithm for updating the interest scores in the ontological user profile

```
Input  : C = c_1, ..., c_n; D = d_1, ..., d_n; T = t_1, ..., t_n.
Output: Returns the updated concepts C = c'_1, ..., c'_n.
Function UpdateScores(C, I):
    Initialize the priorityQueue;
    Initialize all the activation scores;
    foreach d_i ⊆ I do
        L_i = language identification of d_i;
        foreach c_j ⊆ C found under L_i do
            if firstLevel(c_j) and sim(d_i, c_j) > 0 then
                c_j.activation = log(t_i / size(d_i)) * IS(c_i) * sim(d_i, c_j)
                priorityQueue.add(c_j)
            end
        end
        while priorityQueue.size > 0 do
            Sort the priorityQueue;
            c_s = priorityQueue[0];
            priorityQueue.dequeue(c_s);
            foreach concept c_k linked with c_j do
                c_k.activation+ = c_s.activation * c_k.weight;
                priorityQueue.enqueue(c_k)
            end
        end
    end
```

Algorithm 1 is based on the spreading algorithm proposed by Sieg et al. [13] with only some differences: we have included the time spent and the size of the visualized documents for the estimation of interest scores as suggested by Gupta et al. [14], and we have also extended it to take into account multiple languages.

Given a set of documents $D = d_1, ..., d_n$ visualized by the user for a given amount of time $T = t_1, ..., t_n$, the algorithm attempts to update the interest scores of the user concepts $C = c_1, ..., c_n$ based on their similarity with the visualized documents.

For each visualized document d_i, we first start by identifying its language L_i[6], then we estimate its degree of similarity with each first level concept c_j in the user profile found under language L_i. Then we associate an activation score to c_j on the basis of its similarity with d_i along with the size and the time spent on it. This will give an activation score to all the concepts found in the first level (for the considered language)[7]. For the rest of the concepts we use the spreading mechanism presented by Sieg et al. [13] to spread the activation from each concept to its children, and so on. The weight of each relation w_{is} between a parent concept i and one of its children s determines how much activation this parent should spread to each one of its children. This weight is calculated using the following formula (proposed in [13]):

$$w_{is} = \frac{\overrightarrow{n_i} * \overrightarrow{n_s}}{\overrightarrow{n_i} * \overrightarrow{n_i}} \tag{2}$$

where $\overrightarrow{n_i}$ is the terms vector of concept i and $\overrightarrow{n_s}$ is the terms vector of its child s. This formula allow a parent to spread more weights to its children that are similar to it.

With this algorithm we make sure that the user profile will always be up to date as the user visualizes new documents on the browser, and expresses new interests in different languages.

3.2 Re-ranking the Search Results

The re-ranking module is used to improve the initial order presented by the Google Search Engine, in response to a certain user query using his built ontological profile.

First, we identify the language in which the user query has been issued to be preprocessed accordingly, then the Google Search API will be used to obtain the corresponding results.

Algorithm 2 is similar to the re-ranking algorithm proposed by Sieg et al. [13], the difference is that we discard the similarity between the document and the query and we instead leave it to be handled by the search engine. We also take account of the original Google ranking when estimating the new result order as suggested by Gupta et al. [14].

Given that the user profile is defined as a set of concepts $C = c_1, ..., c_n$ each with its interest score, and a set of documents visualized by the user $R(q) = d_1, ..., d_n$ in response to his query q in their original order decided by the Google Search Engine, we start by identifying for each document d_i its most similar concept *best_concept*. Then we estimate the user interest level in that document

[6] We have used the Google Language Detection API [27], to identify the document language.

[7] If a certain document contain textual information in several languages, the text identification process will chose the most dominant one among them.

Algorithm 2. Pseudo algorithm for search results re-ranking

Input : $C = c_1, ..., c_n$, $R(q) = d_1, ..., d_n$.
Output: Returns the new order for the documents of $R'(q)$.

Function `Re-ranking`(C, I):
  foreach $d_i \subseteq R(q)$ **do**
    L_i = language identification of d_i;
    $best\_concept = C[0]$;
    $best\_score = 0$;
    foreach $c_j \subseteq C$ *found under* L_i **do**
      if $sim(d_i, c_j) > best\_score$ **then**
        $best\_concept = c_j$
        $best\_score = sim(d_i, c_j)$;
      end
    end
    $userIntrest(d_i) = IS(best\_concept) * sim(q, best\_concept)$
    $final_{rank}(d_i) = \alpha * userIntrest_{d_i} + (1 - \alpha) * Google_{rank}(d_i)$
  end

as the product of the user interest score in d_i with the similarity between the query and the *best_concept*. The final rank of each document d_i is then estimated as a linear combination of the original Google rank and the user interest score in d_i.

Our intuition is that including the original Google ranking will be important since it uses the Google Page Rank (PR) algorithm [28][8] which assigns higher ranking for more frequently referenced web pages/sites. In the same time, we include our estimated profile-based interest scores to hopefully maintain a certain balance between the importance of a given page and the user's degree of interest in it.

4 Experiments

This section presents the details of our experiments and gives an in-depth discussion of the incorporated tests.

Our experiments examine two important aspects: first, we want to make sure that the interest scores in the user ontological profile stabilize after a finite number of updates, which implies that the long-term user interests are successfully learned. The second aspect aims to check if our proposed framework manages to bring an improvement to the ordering of the standard search results order making it more relevant to the user.

We first explain the process of data collection and preparation. We then present the evaluation metrics we have used. Finally, we address the two aforementioned key experiments.

[8] The page rank algorithm works by estimating the rank/quality of a web page based on the number and the importance of the web pages that reference it.

4.1 Users Data

Evaluating personalized systems is a very problematic task, since it requires direct user intervention in order to provide his judgment of relevance which poses a huge problem of confidentiality as pointed out by Gauch et al. [15].

In most of the personalized information retrieval research projects the authors tend to collect user data along with their relevance judgment from their own students or team members since random web users usually don't welcome the idea of sharing their personal search information and it is even more problematic to get them to provide their judgment of relevance since it will cost them a considerable amount of time and effort [14,29].

In order to automate and facilitate the capture of users' activities, we have developed a browser extension that gets installed on the user's web browser and instantly sends his browsing activities (the URLs of the visited websites, time spent on each web site, etc.) to our web server. We have provided this extension to 24 users from our university and automatically gathered their daily browsing activities for about two months period starting from March 2015. Only the top five profiles that have the maximum number of visited URLs have been considered in our evaluations. Table 1 shows the number of visited URLs for each one of these five selected profiles.

Table 1. Statistics about the number of URLs visited by each one of the 5 selected users

User profiles	URLs
1	233079
2	61553
3	49704
4	36694
5	33426

4.2 Evaluation Metrics

To investigate the effectiveness of our personalized re-ranking system we incorporate two measures: the Top-n Recall and the Top-n Precision. First, we define the standard recall and precision metrics then, based on these, we formulate the Top-n Recall and the Top-n Precision.

Recall. The recall is the ratio between the set of relevant documents retrieved by the system and the total number of relevant documents.

$$Recall = \frac{relevant\ documents\ retrieved\ by\ the\ system}{total\ number\ of\ relevant\ documents} \tag{3}$$

Precision. The precision is the ratio between the set of relevant documents retrieved by the system and the number of retrieved documents.

$$Precision = \frac{relevant\ documents\ retrieved\ by\ the\ system}{number\ of\ documents\ retrieved\ by\ the\ system} \tag{4}$$

F_measure. The F-measure metric combines both precision and recall to give a better relevance judgment. We use the F-measure defined as follows:

$$F\_measure = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{5}$$

Top-n Recall [30]**:** The Top-n Recall is the ratio between the number of relevant documents found by the system in the first n returned documents, and the total number of relevant documents that exist in the first n documents.

$$Top\_n\ Recall = \frac{number\ of\ relevant\ documents\ in\ the\ first\ n\ returned\ documents}{total\ number\ of\ relevant\ documents\ in\ the\ first\ n\ documents} \tag{6}$$

Top-n Precision [30]**:** The Top-n Precision is the proportion of relevant documents found by the system in the first n returned documents.

$$Top\_n\ Precision = \frac{number\ of\ relevant\ documents\ in\ the\ first\ n\ returned\ documents}{n} \tag{7}$$

4.3 Convergence of the Users' Profiles

To ensure that the conceptual interest scores of the user profile will stabilize after a certain number of updates, we have investigated the average rate of their incremental increase.

Figure 3 shows the average increase rate in the conceptual user profile over incremental updates. We can see that initially there is a considerable change rate of the interest scores of the user profile. This changing rate starts decreasing with the number of processed documents, and then reaches a stability level (convergence) when about 600 documents (updates) have been processed, which indicates that the user profile is such that the system has managed to learn the long-term user interests.

4.4 Re-ranking Evaluation

This evaluation aims to investigate in a practical way if the reordered search results better suit the user. To this end we have built 5 ontological user profiles

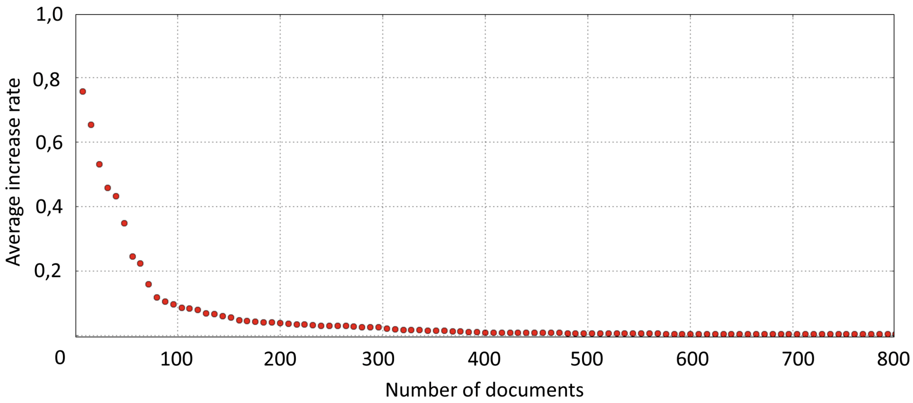


Fig. 3. Change in average increase rate when incrementally updating the user profile

Table 2. Manually prepared queries for re-ranking evaluation

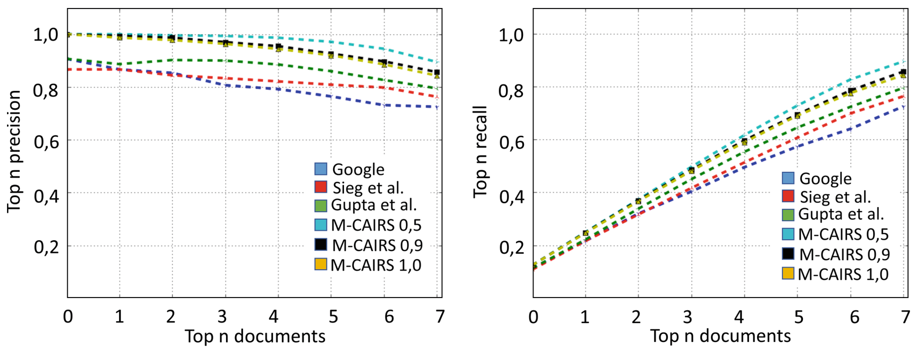
	Arabic	English	French
Query 1	إعراب القرآن وبيانه	Command and conquer the game	Le concours Algeria Game Challenge
Query 2	شهادة إرجاء من الخدمة الوطنية	Find element vector C++	Venir au Canada
Query 3	تحميل المصحف المرتل للقارئ ماهر المعيقلي	Serious Game	Emploi et recrutement
Query 4	أحمد بن علي العجمي	Installing OpenCV 2.3.1 in Windows	Don du Sang Dz
Query 5	خمسات لبيع وشراء الخدمات المصغرة	Isometric character sprite	GTA V

for 5 selected users, and prepared for each one of them a set of 15 queries, 5 for each language (Arabic, English and French). These queries were manually selected according to the profile of each user. Each user was asked to provide his optimal results order for each one of his queries to be considered as references.

Table 2 shows five queries for each of the three languages for one of the five investigated users. We have executed these queries using our IRS with different α priorities (α decides the priority between Google ranking and the user-profile-based ranking). We have tested these queries with the reordering systems proposed by Gupta et al. [14] and Sieg et al. [13] for comparison purposes[9].

The graphs presented in Fig. 4 illustrate the average Top-n Recall and Top-n Precision concerning five users, reported for the original results returned by Google's Page Rank and the algorithms proposed by Sieg et al. [13] and

[9] We note that we have implemented ourselves all the systems we have compared; thus conclusions should be taken with some caution.



(a) Average top n precision (b) Average top n recall

Fig. 4. Comparing the average Top-n Accuracy and Top-n Recall for the different re-ranking methodologies

Table 3. Comparing the different re-ranking methods based on the F-measure metric for each user

	User 1	User 2	User 3	User 4	User 5
Google	0,70	0,78	0,74	0,72	0,67
Sieg et al.	0,72	0,76	0,75	0,77	0,80
Gupta et al.	0,81	0,83	0,79	0,79	0,74
M-CAIRS 0.5	**0,87**	**0,92**	**0,90**	0,85	**0,90**
M-CAIRS 0.9	**0,87**	0,85	0,84	0,85	0,84
M-CAIRS 1.0	0,85	0,84	0,84	**0,87**	0,80

Gupta et al. [14], as well as our proposed re-ranking system for $\alpha = 0.5, 0.9, 1$. We can see that our proposed re-ranking approach yields more accurate results, and that when the value of $\alpha = 0.5$, the best re-ranking results are achieved.

Table 3 shows the F-measure scores for the aforementioned systems as reported for each individual user. The results show that our re-ranking approach for $\alpha = 0.5$ yields an improvement between 15% and 23% compared to those of the Google system and between 5% and 15% compared to the other re-ranking methods. This result confirms again the effectiveness of our combined linear re-ranking. As a matter of fact, we note that the reported F-measure of our proposed system for $\alpha = 0.5$ is always around 0.9 which is very close to the optimal possible re-ranking provided by the users.

The results we have obtained suggest that giving the same importance to the personalized profile-based re-ranking and the Google-based ordering produces more relevant re-ranking results. This also confirms our original intuition that encourages the inclusion of Google ranking of the returned pages along side the ontological user profile interest score.

5 Conclusion

In this work we have presented M-CAIRS, a multilingual Context-aware IRS in which the user's long-term interests are automatically learned and represented using an ontological user profile. The constructed profile is then used to re-rank the search results in a way that better fits the user's needs and preferences.

Our system can easily be deployed on a web server and accessed using any web browser. We believe that the contributions of this work are as follows:

– This work addresses the case of multilingual personalized retrieval with the inclusion of the Arabic language.
– An effective re-ranking method is proposed to better meet the users' information needs.
– A comparative study has been done between different IR systems.

This work can be developed in various directions. These include the incorporation of more contextual short-term features such as the user's geographical position, time, etc. Also other languages beside English, Arabic and French can be incorporated in the user profile.

References

1. Stokoe, C., Oakes, M.P., Tait, J.: Word sense disambiguation in information retrieval revisited. In: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 159–166. ACM (2003)
2. Sugiyama, K., Hatano, K., Yoshikawa, M.: Adaptive web search based on user profile constructed without any effort from users. In: Proceedings of the 13th International Conference on World Wide Web, pp. 675–684. ACM (2004)
3. Teevan, J., Morris, M.R., Bush, S.: Discovering and using groups to improve personalized search. In: Proceedings of the Second ACM International Conference on Web Search and Data Mining, pp. 15–24. ACM (2009)
4. Teevan, J., Dumais, S.T., Horvitz, E.: Personalizing search via automated analysis of interests and activities. In: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 449–456. ACM (2005)
5. Speretta, M., Gauch, S.: Personalized search based on user search histories. In: Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence, pp. 622–628. IEEE (2005)
6. Qiu, F., Cho, J.: Automatic identification of user interest for personalized search. In: Proceedings of the 15th International Conference on World Wide Web, pp. 727–736. ACM (2006)
7. White, R.W., Chu, W., Hassan, A., He, X., Song, Y., Wang, H.: Enhancing personalized search by mining and modeling task behavior. In: Proceedings of the 22nd International Conference on World Wide Web, pp. 1411–1420. ACM (2013)
8. Schafer, J.B., Frankowski, D., Herlocker, J., Sen, S.: Collaborative filtering recommender systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web. LNCS, vol. 4321, pp. 291–324. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72079-9_9

9. Berkovsky, S., Kuflik, T., Ricci, F.: Mediation of user models for enhanced personalization in recommender systems. User Model. User-Adapted Interact. **18**(3), 245–286 (2008)
10. Adomavicius, G., Tuzhilin, A.: Context-aware recommender systems. In: Ricci, F., Rokach, L., Shapira, B. (eds.) Recommender Systems Handbook, pp. 191–226. Springer, Boston (2015). https://doi.org/10.1007/978-1-4899-7637-6_6
11. Henze, N., Dolog, P., Nejdl, W., et al.: Reasoning and ontologies for personalized e-learning in the semantic web. Educ. Tech. Soc. **7**(4), 82–97 (2004)
12. Chen, C.M., Lee, H.M., Chen, Y.H.: Personalized e-learning system using item response theory. Comput. Educ. **44**(3), 237–255 (2005)
13. Sieg, A., Mobasher, B., Burke, R.: Web search personalization with ontological user profiles. In: Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management, pp. 525–534. ACM (2007)
14. Gupta, K., Arora, A.: Web search personalization using ontological user profiles. In: Babu, B.V., Nagar, A., Deep, K., Pant, M., Bansal, J.C., Ray, K., Gupta, U. (eds.) SocProS 2012. AISC, vol. 236, pp. 849–855. Springer, New Delhi (2014). https://doi.org/10.1007/978-81-322-1602-5_90
15. Gauch, S., Speretta, M., Pretschner, A.: Ontology-based user profiles for personalized search. In: Sharman, R., Kishore, R., Ramesh, R. (eds.) Ontologies. ISIS, vol. 14, pp. 665–694. Springer, Boston (2007). https://doi.org/10.1007/978-0-387-37022-4_24
16. Pazzani, M.J., Muramatsu, J., Billsus, D., et al.: Syskill & webert: identifying interesting web sites. In: AAAI/IAAI, vol. 1, pp. 54–61 (1996)
17. Joachims, T.: Optimizing search engines using click through data. In: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 133–142. ACM (2002)
18. Jay, P., Shah, P., Makvana, K., Shah, P.: An approach to identify user interest by reranking personalize web. In: Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies, p. 64. ACM (2016)
19. Lovaraju, D., Devi, G.L.: An ontology like model for gathering personalized web information. Int. J. Adv. Res. Comput. Sci. **8**(3), 401–403 (2017)
20. Vicente-López, E., de Campos, L.M., Fernández-Luna, J.M., Huete, J.F.: Use of textual and conceptual profiles for personalized retrieval of political documents. Knowl.-Based Syst. **112**, 127–141 (2016)
21. Gauch, S., Speretta, M., Chandramouli, A., Micarelli, A.: User profiles for personalized information access. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web. LNCS, vol. 4321, pp. 54–89. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72079-9_2
22. Moukas, A.: Amalthaea information discovery and filtering using a multiagent evolving ecosystem. Appl. Artif. Intell. **11**(5), 437–457 (1997)
23. Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Commun. ACM **18**(11), 613–620 (1975)
24. Safi, H., Jaoua, M., Belguith, L.H.: PIRAT: a personalized information retrieval system in arabic texts based on a hybrid representation of a user profile. In: Métais, E., Meziane, F., Saraee, M., Sugumaran, V., Vadera, S. (eds.) NLDB 2016. LNCS, vol. 9612, pp. 326–334. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-41754-7_31

25. Houssem, S., Maher, J., Lamia, B.H.: Axon: a personalized retrieval information system in Arabic texts based on linguistic features. In: 2015 6th International Conference on Information Systems and Economic Intelligence (SIIE), pp. 165–172. IEEE (2015)
26. Black, W., Elkateb, S., Rodriguez, H., Alkhalifa, M., Vossen, P., Pease, A., Fellbaum, C.: Introducing the Arabic wordnet project. In: Proceedings of the Third International WordNet Conference, pp. 295–300 (2006)
27. Google Corporation: Google language detection API. https://cloud.google.com/translate/docs/detecting-language. Accessed 2017
28. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: bringing order to the web (1999)
29. Daoud, M., Tamine, L., Boughanem, M.: A personalized graph-based document ranking model using a semantic user profile. In: De Bra, P., Kobsa, A., Chin, D. (eds.) UMAP 2010. LNCS, vol. 6075, pp. 171–182. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13470-8_17
30. Hawking, D., Craswell, N., Bailey, P., Griffihs, K.: Measuring search engine quality. Inf. Retriev. **4**(1), 33–59 (2001)