ISSN 1870-4069

Learning-to-Rank for Hybrid User Profiles

Houssem Safi, Maher Jaoua, Lamia Belguith Hadrich

ANLP Research Group, MIRACL Laboratory
Faculty of Economics and management of Sfax,
Tunisia

safi.houssem@gmail.com, {Maher.Jaaoua, la.beguith}@fsegs.rnu.tn

Abstract. In the context of the Personalized Information Retrieval method applied to the Arabic language, this work consists in presenting a personalized ranking method based on a model of supervised learning and its implementation. This method consists of four steps, namely, the user's modeling, the document / query / profile matching, the learning to rank and the result classification. Thus, we proposed a hybrid approach of the user's modeling that relies on both multidimensional and conceptual representations by exploiting Arabic semantic resources. Therefore, to determine the similarity between the document and the profile, we used a learning model that exploits the users' explicit pertinence judgments. In this context, we have proposed learning semantic features related to the user's profile (represented by hierarchies of concepts). The predicted model will then be used in the ordering phase to classify other documents that result from a new query submitted by the user. In this context, we have proposed a novel multi-objective function to order the documents (based on the classic Retrieval Status Value function and the predictive personalized Retrieval Status Value function). Finally, we have explained the evaluation results of the predictive model and the ranking method. These evaluations, which were made based on a training corpus and a test corpus, led to some interesting results. Indeed, the proposed semantic learning criteria connected to the user profile have a significant impact on the performance of our personalized document ranking system.

Keywords: document ranking, learning to rank, hybrid profile, personalized retrieval status value.

1 Introduction

Personalized Information Retrieval (PIR) is one of the best sources of information for acquiring user-based information more precisely and efficiently [1]. PIR is a novel technique where many techniques have been developed and tested; however, many issues and challenges are still to be explored. The most common encountered difficulties, when searching for information, are [2]:

- Problems with the data themselves,

- Problems faced by the users who try to retrieve the data they want,
- Problems in understanding the context of the search queries and
- Problems with identifying the changes in the user's information need.

Moreover, many PIR methods have been discussed in literature [3]. The problems with the existing methods explained in the following observations [3] are the user's protection and the unnecessary disclosure of his profile.

Therefore, the major aim of the researchers who are going to work on this issue will be to completely protect the users and introduce new techniques to prevent unnecessary disclosure of their profiles. We need an innovative approach to create a dynamic user profile based on a submitted query. Furthermore, to our knowledge, very little research has been devoted to personalized information for the Arabic language.

For this reason, the work presented in this paper aims at developing a system for PIR which can be adapted to the Arabic language and provide personalized results based on the user's preferences and interests. This system is dubbed SPIRAL (System for Personalized Information Retrieval applied to Arabic Language). The SPIRAL system uses the reformulated queries (the method adopted the reformulation is proposed by [6]) to reorder the documents retrieved by a search engine while taking into account the user profile. Thus, the implementation and evaluation of personalized learning to rank method and the integration of a hybrid user profile are the subject of this work.

The language targeted by this system at the query and returned documents is the Arabic language. The choice of this language is motivated by the fact that Arabic has not received the same interest as other languages, such as French or English. Similarly, in recent years, we have noticed the emergence of Arabic language resources in the field of automatic language processing. Therefore, the integration of these resources into operational systems dealing with the Arabic language is an additional motivation.

In the second section, we will present a brief overview of the Personalized Information Retrieval (PIR). More precisely, we will briefly explain the learning to rank approaches of documents, then we will present a state of the art of the IR applied to Arabic. In the third section, we will deal in detail with the integration of the user's profile in the proposed method of ranking. In the last section, we will provide a description of the learning to rank system as well as an evaluation of our own corpus.

2 Personalized Information Retrieval

The PIR is a general category of search techniques aiming at providing better research results. The solutions for the PIR can generally be categorized into two types, namely *profile-based* [5] and *click-log-based* [5] methods. The profile-based methods improve the search experience with complicated user-interest models generated from the user's profiling techniques. In the click-log based methods, the authors simply impose a preference to clicked pages in the user's query history. One limitation that reduces its applicability is that it can only work on repeated queries from the same user.

It is emphasized that this work is in the context of the combination of the profile based and click-log-based methods. Thus, the personalization system needs to use all the information about the user (profile, main interests, preferences, information needs) and his research environment [3]. There are mainly tree types of representations of the user profile: Semantics, Multidimensional and Set. The adaptation to the changes in the interest centers, which describe the users, means the upgrading of the user profile. There are two types of user's needs: long-term and short-term profile.

In what follows, we will give a brief review of the learning to rank approaches and a comparison between the models. In addition, we will describe the IR systems applied to Arabic. Finally, we will identify some limitations of these systems.

2.1 Brief Overview of the Learning Approaches to Document Ranking

During the last decade, many algorithms have been proposed to optimize the re-ranking of the search results. These algorithms are generally divided into three categories: pointwise [6], pairwise [7] and listwise [8]. These approaches differ according, first, to their way of considering the input data of the learning system, second, to the type of the variable or judgment of relevance to predict and, third, to the mathematical modeling of the learning problem.

In the pointwise model, each document xi is considered a separate input of the learning model. The judgment of relevance can be an integer or a real score, an unordered class of relevance (not relevant, relevant) or an ordered class of relevance (level 1 relevance <level 2 relevance <...). The judgment of relevance here is a variable that predicts the value which ranks the documents. When the judgment of relevance is an integer or a real score, the learning problem is generally regarded as a linear regression problem. The relationship between the quantitative variable to be explained and the explanatory variables is assumed to be linear.

In the pairwise model, the pairs of documents (x_i, x_j) are considered as an input to the learning step. Each pair of documents is associated with a judgment of preference $y_{i,j}$ with value $-1, 1.$ If $y_{\ i,j} = 1$, then document x_i, which is favorite to document x_j: should be ranked above x_j in the result list. Preference is denoted $x_i \succ x_j$. On the other hand, if $y_{\ i,j} = -1$, then document x_j is preferred to x_i document and notes $x_j \succ x_i$. The learning problem here is a classification problem, in the particular case of pairs of instances. Therefore, most of the algorithms of this model use adaptations of existing classifiers.

In the listwise model, a complete and ordained list of documents is considered as an input of the learning step. The algorithms provide as output the ordered list of documents or a list of their relevance scores ([8, 9, 10, 11, 12, 13]). The algorithms are divided into two subcategories within this model: those minimizing an error function defined from an IR measurement as MAP (MAP is the average of the average precision of all the queries [16]) or NDCG (Normalized Discounted

Cumulative Gain is defined from the Discounted Cumulative Gain (DCG) [16]) and those minimizing a loss function not related to the IR measurement.

Historically, the Pointwise and Pairwise models have been the first to be proposed (around the early 2000s) while the first studies treating the Listwise model have appeared only recently. Some other research studies have been proposed to compare the learning approaches for the above ranking. The conclusions drawn show that the model list shows more interesting results than the models in pairs or points [14] and [15]. It should be noted that these results were obtained following the analysis of large number of algorithms and large data sets (3.0 for the collection Letor [14, 15]). In addition, the Listwise model is generally regarded as easier to implement. Therefore, we chose to use the list approach in our learning model.

2.2 Information Retrieval Applied to Arabic

Faced with the IR, the Arabic language has recently been addressed by conventional search engines, but it is absent in the semantic search engines. It is within this context that this work proposes to develop a personalized information retrieval system for the Arabic language. This system illustrates the implementation of the PIR method that we have proposed and which distinguishes three stages, namely the user's modeling, reformulation (specifically expansion) query and scheduling results.

The attention paid to the Arabic language is explained by the fact that this language does not receive the same degree of attention as the other languages such as French or English. Moreover, the Arabic language resources are emerging in the search field of automatic processing of language which gives extra motivation to integrate these resources into operational system processing of the Arabic language.

In the implementation of our PIRS, we will try to incorporate language resources developed for Arabic. This consists in integrating a chain of linguistic analysis which, besides helping resolve the language ambiguities, enriches the concepts of the users' queries and profiles.

To solve the morphological and lexical ambiguity, a lemmatizer is suggested to place a light lemmatization. The use of semantic resources for the enrichment (expansion) of the user's query can be a solution to solve the problem of semantic variations and disambiguate the query terms. Indeed, the semantic resources provide resources in the form of semantic relationships. They can extend the search field of a query, which improves the research results.

The use of semantic resources in an IRS may be considered at several levels:

- Before being sent, the user's query can be enriched by the near judged concepts in semantic resource through the use of relationships, such as generalization / specialization, synonyms ...
- The indexing of documents is made using the concepts of the semantic resource and not the keywords.
- Filtering of documents in a particular field to the user profiles ([17, 18, 19]).

It should be noted that the query expansion is a double-edged sword so that improving the research in this event may be accompanied by an information overload problem.

Indeed, the query reformulation or expanding may generate a significant number of terms when using multiple relationships in a semantic resource.

To address this problem, we propose a second alternative based on the user profile concept to reduce the enriched elements during the expansion, in order to remove the ambiguity of some terms and filter the returned documents. Similarly, we propose a third alternative to improve the accuracy of the IR entitled "personalized learning to rank". This alternative, which is based on a hybrid user profile (multidimensional and conceptual), makes it possible for the user to put the classified documents, which are "relevant" according to his profile, at the top of the list.

To our knowledge, there are no PIR systems for Arabic. Most of the developed research studies in the field of IR in Arabic have been particularly interested in the query reformulation step. These studies use the thesauri dictionary and the language resources to substitute and / or disambiguate the query terms. In the following part of this section, we will quote the main research studies in the context of an IR in Arabic, then, we can group them according to two axes. The first axis includes the work using morphological stemming of the query words, while the second includes the studies that exploit the thesaurus dictionary.

In the first axis, Xu and al. evaluated two research strategies of Arabic documents using the ArabTREC corpus as a test corpus. The authors developed a strategy that uses first indexation based on the roots. This method resulted in a slight improvement of the research results. Likewise, these authors showed that the second strategy that is the use of a thesaurus dictionary, dramatically improves the performance of an Arabic IRS [20].

On the other hand, Bessou and al. adopted the scheme notion as a base to substitute the query words with their lemmas at the level of indexing and search steps [21].

In the second axis, we can mention the work of Hammo and al. that used the Koran as thesaurus for the query reformulation [22]. For their part, [23] used the Arabic WordNet as thesaurus to supply the ontology designed for the legal field.

The work of [24] proposed to assist the user with the reformulation of his query by adding nearby morphological forms of the initial query word forms. This addition is based on a similarity calculation of n-grams between the words of the original query and those saved in a lexicon. To index and search for operations, [24] used the services of the Google search engine.

The work of [25] can be summarized in the use of an external resource (Arabic WordNet or AWN) and a morphological analyzer to be reformulated by expanding the user's query that can improve the recall but not the precision of the IRS. As an extension of this work, [26] used a reformulation based on two external resources, namely ADS (Arabic Dictionary of Synonymy) and AWN.

It should be emphasized that the already mentioned research studies have some limitations. Indeed, some studies ([23] and [22]) used semantic resource or ontology for a specific field. Besides, there is non-use of conceptual relationships ontology in some studies. Finally, there is a lack of studies ([23] and [26]) about the contribution of each semantic relationship used in some Arabic query expansion systems.

According to the conducted overview, we can conclude that the enrichment of queries based on external resources is an interesting path the exploitation of which can improve the results of the IR. In addition, we noticed that the personalization side is absent in the above studies, which is an additional motivation for this work knowing the performance improvements recorded in other languages. On the other hand, we can emphasize that learning to rank is a technique totally unaffordable by PIR systems for the Arabic language which is another motivation for this work. Indeed, semantic learning features from the user profile and those contained in the semantic resources constitute an original and a promising path that can give good performances in the context of IR in Arabic.

To our knowledge, there is no personalized learning to rank systems dedicated to the Arabic language (that is to say there are not works that integrate the user profile). Likewise, it is worth noting that our contributions of this work in the field of PIR revolve around the following points:

- Modeling of a hybrid user profile that relies on both conceptual and multidimensional representations by exploiting Arabic semantic resources.
- Proposing semantic learning criteria connected to the user profile (represented by concept hierarchies). These criteria have a positive impact on the performance of our PIR system.

3 Proposed Method

The objective of the personalized ranking method is to provide the user with an ordered list of documents in response to a query issued by him. The document ranking is a major theme in the IR. Indeed, several studies have been made to establish the appropriate metrics that help determine the optimal order governing the documents returned by a search engine. The many features that were proposed to develop these ranking metrics are the similarity of documents in relation to the query, their importance and their links [15, 27], etc.

Since the proposed method is based on the user profile, it is quite apparent to integrate the profile in the calculation of its similarity with the documents returned by the search engine. It should be noted that the used queries are reformulated and, therefore, they integrate concepts from the profile. It follows that personalization is given a leading role in the result ordering.

To determine the similarity between the document and the profile, we used a learning model that exploits the users' explicit judgment pertinence. This consists in asking the user to assign a relevance class to document that reflects its significance in relation to his needs. In a second phase, we project these judgments on features related to the documents, the queries and the profile.

This projection helps build a predictive model that discerns the relevant documents meeting the user's profile and query. The predicted model will then be used in the ranking phase to classify other document results of a new query submitted by the user.

In the following part of this section, we will introduce the ranking document method that distinguishes four steps, namely, (1) the user's modeling, (2) the document/query/profile matching, (3) the learning to rank, (4) and the result classification as shown in figure 1.

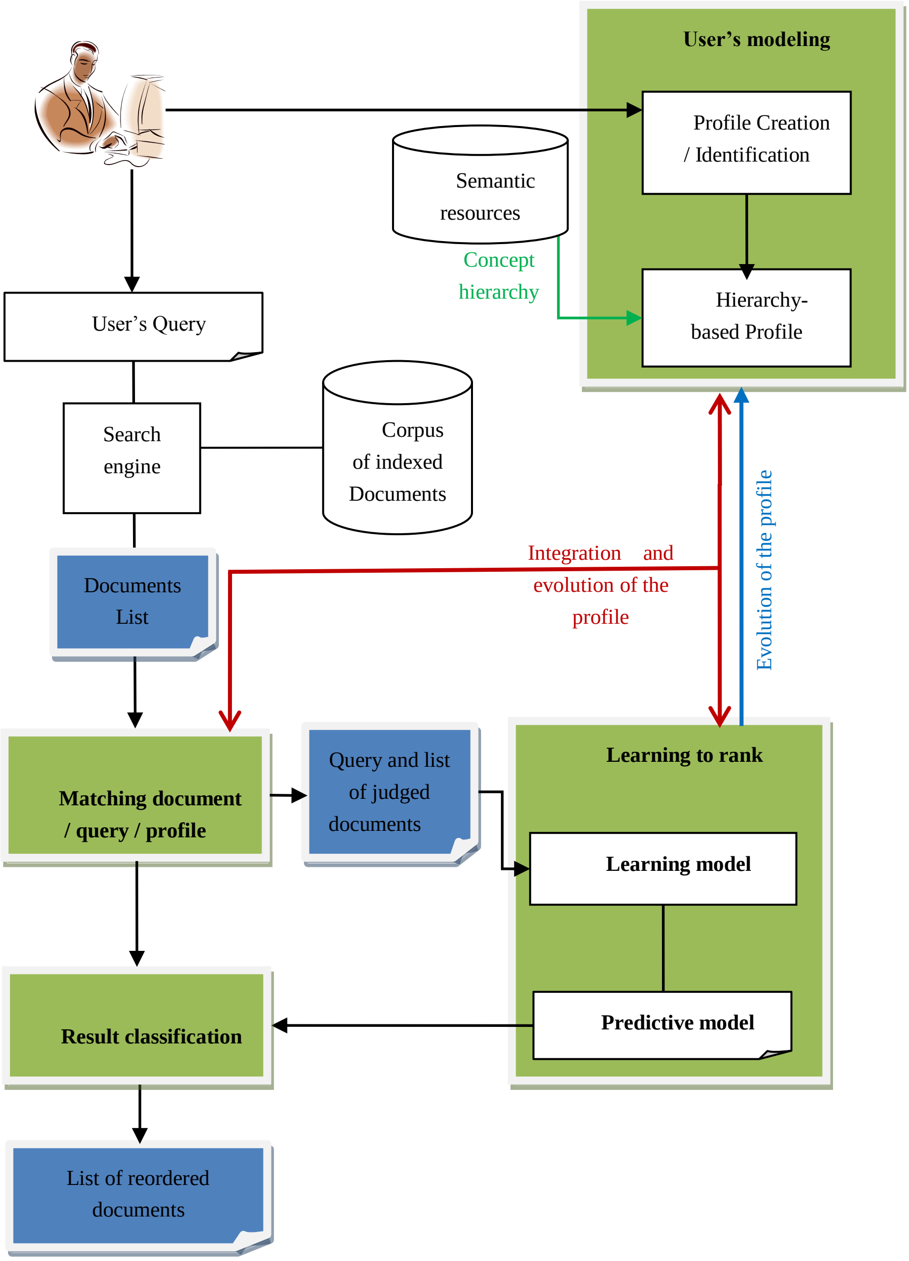

Fig. 1. Personalized learning to rank method

It should be emphasized that, in our ranking document method, we have included the method of document/query/profile matching that was used in [4]. For this reason, step (2) will be presented in brief while steps (1), (3) and (4) will be described in detail.

3.1 Suggested User's Modeling

In the framework of the proposed ranking method, a user's modeling based on a hybrid representation and built on the user profile is proposed. In this approach, an algorithm which automatically builds a hierarchical user profile is introduced to represent the user's implicit personal interests and domain. It is to represent the domain and the interests with a conceptual network of nodes linked together. This network is made through relationships respecting the linking topology (synonymy, hyponymy and hyperonymy) defined in ontologies (AWN [30] and Amine AWN [31]) and the domain of hierarchies.

It should be noted that our method allows updating the short and long term user profile. The evolution of the user profile in short term is jointly linked to a bounding mechanism of search sessions to examine the change of interest over time. In addition, relevant feedback helps refine the user's preferences and consequently update the short-term profile.

Thus, the capture of changes in the centers of interests is concretized by the addition of the search history (queries and search results that have been appreciated by the user) to the short term profile. Indeed, the proposed method establishes an activation score based on the construction and evolution of a user profile from his judgments of relevance. In this context, each user's query will be added to his profile in the short term. A weight averaging the formula tf * idf, will be assigned to each term derived from the document deemed relevant or very relevant by the user.

Then, the first terms with the largest weight will be inserted in the short-term user profile. The number of these added terms can be determined by an experimentation which achieves the compromise between the size of the centers of interest and its real needs. It should be noted that in this method, an algorithm of the concept score propagation is used to update the weights of the profile concepts.

Indeed, the terms of consulted documents and / or submitted queries are aggregated to the user profile according to a similarity threshold between the document and the user profile. In this phase, we adopt a method which models profile V by R vectors $\mathrm{V_i}$ respectively corresponding to R documents $\mathrm{d_i}$ judged as relevant by the user. For each new selected document $\mathrm{d_{i'}}$, the $\mathrm{V_i}$ dimension, which is the most similar to the profile of document $\mathrm{d_{i'}}$, is updated as follows:

$$\mathrm{V_i} = \mathrm{V_i} + \mathrm{V_{i'}};\ \mathrm{V_i} = \mathrm{argmaxv_i} \in \mathrm{v}\ \mathrm{Sim}(\mathrm{V_i}, \mathrm{V_{i'}}) \text{ with}$$

$$\mathrm{Sim}(\mathrm{V_i}, \mathrm{V_{i'}}) = \frac{\mathrm{V_i}.\mathrm{V_{i'}}}{|\mathrm{V_i}|.|\mathrm{V_{i'}}|}. \tag{1}$$

Only m words $\mathrm{t_v} \in \mathrm{V_{i}}$, which have longer weights, are selected for updating dimension$\mathrm{V_i}$ of profile V.

Thus, the long-term user profile enables (implicitly and / or explicitly) to model persistent or recurrent centers of general interests. The evolution process of the long-term profile is to add or change a context formed by concepts associated with a query sent by the user. Identifying a similar context to the user's profile involves merging them and subsequently updating the long-term profile. A new context is therefore added to the long-term profile if no previously learned context is similar to the context of the query. Likewise, the modification of the long-term profile can be envisaged by enabling the user to explicitly integrate a new domain.

Generally, high levels of hierarchy concepts make it possible to represent the profile in the long term whereas low levels make it possible to represent a high level of specificity of the user profile in the short term.

3.2 Personalized Matching Step

The calculation of the personalized matching score between the document and the profile can be determined by the cosine between both $\overrightarrow{D}$ and $\overrightarrow{U}$ vectors. At this level, we can set a threshold for RSV (D, U) below which document D will not be retained in the list of results for a given query. This threshold may be determined after a series of experiments to select the documents that best satisfy the user's needs [4].

3.3 The Learning-to-rank Step

The ranking step takes as input a list of documents judged by the user and his profile. The latter is based on a concept hierarchy extracted from the semantic resources. Similarly, the list of documents, which is the training corpus, contains learning features labeled by the user. Thus, the learning phase is based on the optimization of a ranking function that leads to a predictive model.

In what follows, we will describe the learning to rank principle then we will spread out the adopted learning features.

Principle of learning to rank. The classic ranking function is used to classify only the documents that take account of the user's queries in a descending order of relevance. In the case of personalized learning, our contribution is to classify the documents that take account of the queries but also the user profile. Given that our goal is to order a list of documents, the most appropriate model to use in the learning step is the listwise model. This model also has the advantage of evaluating the performance of the algorithms on the basis of IR measurements, as it displays more interesting results than the other models.

The learning to rank is based on two concepts: the representation of the document-query-profile triplet in the feature space and the use of a learning model. The learning to rank process is divided into two phases: a training phase and a testing phase. In the learning phase, the datasets are used by algorithms to automatically learn the ranking functions that serve as models for the prediction of relevance judgments (the chosen scale is three classes of relevance: relevant, slightly relevant and irrelevant). In the test phase, these functions are then used to order the documents returned by the IRS when new queries have been submitted.

The data set used in the learning phase, consists of the query/document/profile triplets. Each triplet (q_i, d_j, u_k) is represented in the feature space by the vector $x_{i,j,k} \in \mathbb{R}^d$ such that $x_{i,j,k} \in \mathbb{R}^d$ and $x_{i,j,k} = [x_{i,j,k}^{(1)} . . . x_{i,j,k}^{(d)}]$ and associated with a class of relevance $s_{i,j,k}$.

In the test phase, the learned function is used by the ranking system to predict the relevance scores of new triplet query / document / profile which have not been annotated. The ranking model thus returns the relevance of the class for each query / document / profile.

Learning proposed features. The used learning model operates a set of features that depend on the query, the document and the user profile. In order to measure the impact of personalization using the learning technique, we were led to choose learning criteria related to the user profile (represented by hierarchies of concepts). The adopted features can be classified in four categories.

The first category consists in determining the similarity between the query and the returned documents. The features are used to calculate the term frequency (tf) of the original query in the text, the title, the subtitle, the summary, the category and the index of document.

The second category of features includes similar features between enrichment query words and the document. The features help extract the matching frequency of the terms synonyms, the generalization and the specification in the document.

The third category is related to the similarity between the document and the user profile. The purpose of these features is to verify the presence of the short or long term user profile concepts in the text. This feature is based on the tf representing the degree of similarity between the user profile and the document. More precisely we determine the frequency of the centers of interest concepts, of the short and long term profile with the document.

The fourth category includes other contextual features related to documents and query and their statistical characteristics. We can mention, as an example, the number of query words, the number of words in the text, the text length (short, medium or long) as well as the format features (Word, PDF, PowerPoint, etc.).

It should be noted that the learning to rank features consist of one of our contributions in the field of PIR given that, according to our knowledge, there are no research studies that used this type of features.

Relevance class. In the framework of classical IR, the process of judging the information relevance is based on the degree of similarity between the representation of the query and the content of the document found by the system.

However, personalization involves taking into account the user profile as an information source that participates in the judgment of relevance. Thus, relevance can be defined as the adequacy of a document following a given query and a well-defined profile. This notion is subjective because the user's state of knowledge is dynamic. Indeed, for the same user, relevance changes over time while a document can have different types of pertinence for two users who submitted the same query.

To annotate the relevance class of a document, we can borrow the explicit feedback approach of the user. Under this approach, the user directly delivers his interest judgment by giving a relevance value on a graduated scale from the least to the most relevant. In our method, the class of a document compared to a query for a given user can have one of the following words "irrelevant," "medium relevant" or "relevant".

It is noteworthy that we have initially chosen five evaluation degrees, namely "irrelevant," "a little irrelevant," "moderately relevant", "relevant" and "highly relevant". However, we detected two problems of annotation (overlap between the entries) between the first two points "irrelevant" and "moderately relevant" and between the last two "relevant" and "highly relevant". In fact, we found that the users or even experts find it difficult to judge the documents using five rating levels. For this reason, we were led, in a second stage, to keep only three levels.

3.2 The Results Ranking Step

The final result ranking depends on the relevance of the documents in relation to the query and the user profile. This relevance combines two values namely the classic RSV (D, Q) and the predictive personalized RSV (D, Q, U) where D, Q and U are respectively the document, the query and the user profile.

To measure the classic RSV function, we adopt the most known measures from the quantities called tf and IDF. Our choice is justified by the fact that these measures are very successful and very popular in the IR. The weight of a word in a query or in a document is expressed using the tf.IDF measurement. The tf measure is the number of word occurrences within a document, while the IDF measure shows the importance of a word in the considered corpus, such as:

$$IDF(t) = log \frac{N}{n_t}. \qquad (2)$$

It is noteworthy that the predictive personalized RSV function is a relevance class which can either be "irrelevant," "medium relevant" or "relevant", whereas the classic RSV function is a score calculated by the cosine function which belongs to the interval [0..1]. Due to the incompatibility of both functions, we have adopted a multi-objective function that promotes first class relevance of the documents. In the case where two documents have the same class relevance, the multi-objective function uses the classic RSV function. Therefore, as a first step, we ranked the documents based on their similarity to the profile. As a second step, we classified the documents with the same relevance class according to their similarities with the query.

4 Implementation and Discussion of the Results

The implementation of the proposed PIR method resulted in three versions. The first version is the query expansion system, the second version, which is a system that integrates the personalized matching module but does not contain the ranking module.

The third version of our system is the "SPIRAL" that includes all the steps of the proposed method. . In this section, we will provide a description of the SPIRAL system as well as an evaluation of our own evaluation corpus.

4.1 Arabic Corpora for Learning and Ranking

Since there are no evaluation standards for personalized access to information, especially for short-term personalization, we proposed context-oriented assessment frameworks based on simulation collections of TREC campaign by simulated users' profiles and search sessions. We have exploited these evaluation frameworks to validate our SPIRAL contribution. For this reason, we have created a large Arabic text corpus entitled WCAT (Wikipedia Corpus for Arabic Text) using the search engine “Lucene”[1]. This corpus is segmented into 30550 text article, extracted from Wikipedia. This corpus contains texts dealing with topics related to the “natural sciences” domain. Moreover, each article has one or more categories related to the root category of “natural sciences”. We generated 7200 sub-categories from the “natural sciences” category.

The search engine Lucene is capable of processing large volumes of documents with its power and speed due to indexing. In our system, we used Lucene to index a corpus of documents, analyze the queries, search for the documents and present document results.

In this phase, the indexing step of the corpus consists in stemming words, removing stop words, indexing and extracting key words of each document in the corpus.

We also built our own Arabic Query Corpus entitled “AQC_2”, which is composed of 1000 queries submitted by 50 different users and deals with topics related to the "natural science" domain. An Arabic query corpus consists of 90,507 words or 613,021 characters and 3.47 megabyte size. Thus, the evaluation corpus of our system contains different types of queries suggested by various users.

When working on a learning process, it is appropriate to divide an initial corpus into two sub-corpora:

- The learning corpus serves to extract a model or classification from a sufficient occurrence of information;
- The test corpus is used to check the quality of learning from the learning corpus.

In what follows, we will give some features of the learning corpus and the learning evaluation corpus (table 1).

It is emphasized that in the context of evaluating the ranking system, we tested the SPIRAL system for 50 users; each of whom has submitted 20 queries. This gives us a corpus of 1,000 test queries. Therefore, in our assessment of every query, only the first 10 documents returned by the search engine are taken into account, which gives us a test corpus of 20,000 documents.

[1] https://lucene.apache.org/

Table 1. The learning and the evaluation corpora.

	Size of the corpus	Average size of an item	Number of items	Number of Words	Langu age
Learning corpus	65 mega-octets	4 Kilo-octets	20 000	15 333 028	Arabic
Evaluation corpus	35 mega-octets	4 Kilo-octets	10 000	6 159600	Arabic

In what follows, we will present the evaluation results of the SPIRAL system. We used the Weka learning framework to get to know the personalized ranking function of our system that exploits the user profile so as to reorder the documents returned for a given query.

4.2 The Used Indicators of Performance

The indicators of performance are used to evaluate a prediction model; however, the performance of this model can be significantly influenced by the conditions of its experimentation. In this section, we will first describe the different evaluation indicators of the prediction models, then, the standard performance measures. Finally, we will present the cross-validation method that we used to evaluate our learning model.

Standard measures of performance. To evaluate the learning model, we used assessment measures such as the recall, precision and F-measure. In addition, we used the kappa measure which measures the degree of agreement between prediction (predicted classes) and supervision (real classes) after the agreement by chance is removed.

$$\text{Recall}_{(i)} = \frac{\textit{Number of documents assigned correctly to class i}}{\textit{Number of documents belonging to class i}}, \tag{3}$$

$$\text{Precision}_{(i)} = \frac{\textit{Number of documents correctly assigned to class i}}{\textit{Number of documents assigned to class i}}, \tag{4}$$

$$\text{F-Measure}_{(i)} = \frac{2 \text{ x Recall x Precision}}{(\text{Recall} + \text{Precision})}. \tag{5}$$

Cohen's kappa: this coefficient is a statistics which measures the inter-rater agreement for qualitative (categorical) items. It is generally thought to be a more robust measure than the simple percent agreement calculation, since κ takes into account the agreement occurring by chance. The equation for kappa (K) is:

$$\text{Kappa}_{\mathbf{(i)}} = \frac{\Theta 1 - \Theta 2}{1 - \Theta 2}, \tag{6}$$

where ϴ1 is the relative observed agreement among the raters, and ϴ2 is the hypothetical probability of chance agreement, using the observed data to calculate the probabilities of each observer randomly saying each category. If the raters are in a complete agreement then κ = 1. If there is no agreement between the raters other then what would be expected, then, (as given by Θ2), κ ≤ 0.

It should be noted that the error rate is equal to the difference between the rate of the ideal classification (100%) and the good classification rate:

$$\text{Error Rate} = 100\% - \text{good classification rate}. \tag{7}$$

Cross-validation. Cross-validation, which is sometimes called rotation estimation, is a model validation technique for assessing how the results of a statistical analysis will generalize to an independent data set [32] [28] . It is mainly used in settings where the goal is prediction, and one wants to estimate how accurately a predictive model will perform in practice.

In k-fold cross-validation, the original sample is randomly partitioned into k equal sized sub-samples. Among the k sub-samples, only one is retained as the validation data for testing the model, and the remaining k-1 sub-samples are used as training data. The cross-validation process is then repeated k times (the folds), with each k sub-samples used exactly once as the validation data. The k results from the folds can then be averaged (or otherwise combined) to produce a single estimation. The advantage of this method, over repeated random sub-sampling, is that all the observations are used for both training and validation, and each observation is used for validation exactly once. 10-fold cross-validation is commonly used, [29] but in general, k remains an unfixed parameter.

4.3 Evaluation and Discussion of Learning Model Results

This section focuses on the different experiments carried out for our learning model. Indeed, these experiments are expressed in terms of global accuracy using, on the one hand, the decision trees and, on the other hand, the SVM in addition to the K-NN as techniques to measure the quality of learning.

In our search studies, we distinguish two sets of experiments dedicated mainly to the performance evaluation of the proposed method. The first set is manifested by the manual division data into two subsets; one set for learning (80% of the corpus) and a second a distinct set for the test (20% of the corpus). This set allows presenting the evaluation results of the learning and testing phases. The second experimentation set is automatically carried out, using cross validation that allows presenting the results of the ranking phase (test).

The following section consists in presenting the results obtained from the evaluation of our system. It is composed of two parts: the first part presents the results of the evaluation of learning and the second presents the results of the evaluation of the ranking result documents.

Experimental Set 1: manual division. In this section, we present two types of the obtained results: those obtained after learning and those resulting from the projection of the test corpus on the prediction model. Thus, the used evaluation measures are accuracy, recall, precision, F-measure and kappa.

Learning results. In the context of the evaluation by manual division of the corpus and using decision tree algorithms, SVM and KNN, we obtained the results presented in table 2. By referring to this table, it therefore appears obvious that the results of our learning method are very interesting. Indeed, in the case of the KNN algorithm, the recall is in the order of 74.6% whereas precision is equal to 78.1%, hence, the F-measure is equal to 72.1%. Likewise, we obtained an accuracy of 74.6%. Finally, we have achieved a kappa degree of agreement between prediction and supervision which is equal to 0.56.

Finally, in the case of the algorithm of the decision tree, the recall is of the order of 77% while precision is equal to 77.6%, hence, the F-measure is equal to 76.5%. Likewise, we obtained an accuracy of 77%. Finally, the achieved degree of agreement between prediction and supervision (kappa) is equal to 0.61.

Table 2. Experiment No. 1: Evaluation results of the learning phase by manual division based on the SVM, KNN and the decision tree.

	Accuracy	**Recall**	**Precision**	**F-measure**	**Kappa**
SVM	47.4%	74.4%	54.8 %	42.3 %	0.07
KNN	74.6 %	74.6%	78.1%	72.1 %	0.56
Decision tree	77 %	77%	77%	76.5%	0.61

Ranking result. This phase is to use the prediction model obtained from the learning phase to classify new documents. In the context of the evaluation using manual division of the corpus as well as the following algorithms; the decision tree, the SVM and the KNN, we obtained the results presented in table 3. According to this table, it appears that the results of our ranking method are interesting. Indeed, in the case of the algorithm of the decision tree, the recall is in the order of 66.1 % while precision is equal to 72 %, therefore, the F-measure is equal to 67.3 %. Similarly, the obtained accuracy is 66 %. Finally, the degree of agreement archived between prediction and supervision (kappa) is equal to 0.41.

Table 3. Experiment 1: evaluation results of the ranking phase by manual division of the corpus based on the SVM, KNN and the decision tree.

	Accuracy	**Recall**	**Precision**	**F-measure**	**Kappa**
SVM	51.8%	51.9 %	60.6 %	48.5 %	0.16
KNN	68.7 %	60.2%	54.8%	55.9%	0.17
Decision tree	66 %	66.1%	72%	67.3%	0.41

Experimental Set 2: cross-validation. To classify new documents, the proposed ranking method consists in using the classification model obtained during the learning phase. Therefore, the evaluation of the ranking method is to evaluate the predictive model with new documents. On the other hand, the evaluation measures that we have used are the same as those of the evaluation of the learning model, namely, accuracy, confusion matrix, recall, precision, F-measure and kappa.

In the evaluation context using cross-validation (K-fold) with K = 26, the decision tree, the SVM and the KNN algorithms, we obtained the results presented in table 4. From this table, it appears that the results of our ranking method are interesting. Indeed, in the case of the SVM algorithm, the recall is in the order of 60.6 % whereas precision is equal to 45.6 %, hence, the F-measure is equal to 46.1%, besides, an accuracy of 60.5% is obtained. Finally, we can say that the achieved kappa degree of agreement between prediction and supervision is equal to 0.11. Finally, in the case of the algorithm of the decision tree, the recall is in the order of 61.4 % while precision is equal to 58 %, consequently, the F-measure is equal to 59.2 %. Likewise, we obtained an accuracy of 61.4 %. Finally, it can be noted that we have achieved a degree of agreement between prediction and supervision (kappa), which is equal to 0.24.

Table 4. Experiment No. 2: Evaluation results of the ranking phase using the cross-validation method k-fold based on the SVM, KNN algorithms and the decision tree.

	Accuracy	**Recall**	**Precision**	**F-measure**	**Kappa**
SVM	60.5%	60.6 %	45.6 %	46.1 %	0.11
KNN	60.1 %	60.2%	54.8%	55.9%	0.17
Decision tree	61.4 %	61.4%	58%	69.2%	0.24

The discussion of the learning results, using cross validation shows that the decision tree increases the performance of our learning model. For this reason, in the context of our ranking method, we adopted the algorithm of the decision tree to build the predictive model which is also used to classify new returned documents for a query submitted by the same user.

Similarly, we performed a set of learning experiments with the user profile (which means that we have integrated the learning criteria linked to the user profile in the learning model) and a series of experiments without the user profile (that is to say, we eliminated the user profile-related learning requirements from our learning model).

Table 5. Evaluation of the learning outcomes by integrating the user profile and learning outcomes without the use of the user profile.

	Accuracy	**Recall**	**Precision**	**F-measure**	**Kappa**
Learning with profile	61.4 %	61.4 %	58 %	69.2 %	0.24
Learning without profile	40.4 %	41 %	35 %	37.7 %	0.11

As shown in table 5, we found, in all cases, that learning by means of the profile has given better results than without it. Indeed, the accuracy of learning by means of the profile is equal to 61.4%, while that without it is about 40.4%. This proves the contribution of the hybrid user profile in our ranking system.

4.4 Comparison to Baseline Methods

On the other hand, we have also experimentally compared our "SPIRAL" contribution to the method of the search engine "Lucene" (a baseline method in our case). In fact, Lucene uses a model which is derived from Boolean model. Thus, Lucene method is a method without profile that is to say without personalization of IR.

Table 6. Performance gain of personalized search (precision and MAP Measures).

Precision	baseline method	SPIRAL	MAP	baseline method	SPIRAL
%P10	9	20	**%MAP5**	7	14
%P20	10.1	16.9	**%MAP10**	5	13
%P30	3.2	10	**%MAP15**	6	15
%P50	6	9.9	**%MAP**	6	14

Calculation of Precision Average. The results for the SPIRAL system (with hybrid profile) are better than those of the baseline method (Table 6). Indeed, the precisions P10, P20, P30 and P50 of the SPIRAL are better than the one in the baseline method. As a conclusion, we have demonstrated that personalizing the IR showed better results with a hybrid profile than IR with a base line method.

Calculation of MAP (Mean Average Precision). We notice that the results obtained with the SPIRAL system are better than those obtained with the baseline method (Table 6.). Moreover, the MAP5, MAP10 and MAP15 for SPIRAL are better than those of the baseline method. Indeed, SPIRAL system shows all these performances for the first 15 documents by MAP15 = 15 and its MAP is better than the Lucene system by MAP15 = 6. Similarly, we can see that the IR showed better results with hybrid profile (personalization) than with a baseline method.

4.5 Discussion of Results

In a first set of experiments, we have divided our corpus (20,000 documents) in two corpora, namely a training corpus (16,000 documents) and a test corpus (4,000 documents). The results obtained by exploiting the algorithm of the decision tree, when evaluating the learning phase, are very interesting with an accuracy equal to 77%. Thus, the predictive model obtained from the learning phase is a performing model that has interesting results from our ranking system with accuracy equal to 66%.

In a second set of experiments, we used the algorithm of the decision tree when evaluating the ranking phase. It was found that the obtained results are interesting. Indeed, we have obtained an accuracy of 61.4%. It is emphasized that we had 61.4% as recall and 58 % as precision; hence the F-measure is equal to 69.2%. Similarly, it can be said that we have achieved a degree of agreement between prediction and supervision (kappa) equal to 0.24.

On the one hand, the obtained recall rate is explained by the ability of our learning model to return a large number of relevant documents among all the relevant ones in the corpus. This is explained by the contribution of the hybrid user profile in the process of finding relevant documents.

Furthermore, through our hybrid profile, the ranking system helped to return a large number of relevant documents among all the ones proposed by the system, which explains the precision rate of 58 %.

Nevertheless, the Kappa value of 0.24 indicates that the proposed ranking system allows a relatively medium degree between prediction (predicted class) and supervision (real class).

Also, it is observed that the length of the query has a relatively direct impact on the results of our system. Indeed, it was found that if the number exceeds four terms without expansion and the expansion process adds to each term at least three other concepts from the hybrid user profile, then, we'll get at least 12 terms in the enriched query. This will generate a lot of noise in the document search process and therefore, more irrelevant documents.

Similarly, after a comparison of our “SPIRAL” contribution against a baseline method, we can see that the personalization of IR by %MAP = 14 showed better results with the hybrid profile than IR with a baseline method by %MAP = 6.

Concerning the learning criteria, we emphasize that we first adopted classical criteria (the first category and the fourth category of the criteria) used by the majority of the studies on the IR. Secondly, we decided to add user profile criteria (the third category of criteria) and semantic criteria (the second category of criteria). This enabled us to further improve the results that passed for the P10 precision rate from 9% to 20% and the percentage of the MAP average from 6% to 14%.

As a conclusion, one of the strengths of the proposed method of RIP has five aspects:

- The proposed method is interesting because it is more user-oriented progressively adapts to the evolution of his profile and his knowledge.
- Learning is performed for each user apart from what proves the personalization aspect characterizing the method.
- The contribution of the hybridization of the user profile (the conceptual and multidimensional representation) to the mechanisms of the query expansion and the ordering of the documents restored by a search engine.
- The positive impact of the semantic learning criteria (based on information from semantic resources) and the criteria related to the user profile (represented by hierarchies of concepts) on the performance of our RIP system.
- Integrating the user profile in all the levels of the PIR process.

5 Conclusion and Prospects

In this work, we focused on the method of ranking documents that we proposed as part of a personalized information retrieval system. The proposed personalized learning to rank method is based on the integration of the user profile into the learning criteria and the proposed ranking function. The representation of the user profile (hybrid approach) in our method is based on the extraction of semantic relationships found in ontologies (AWN and Amine AWN) i.e. synonymy, hyperonymy and hyponymy.

To ensure the achievement of the ranking method, we used a learning model that exploits the user' explicit relevance judgments. This consists in asking the user to assign a relevance class to a document which reflects the importance of the document with respect to the user's needs. In a second phase, we projected these judgments on criteria related to a document, a query and a profile. This projection helps build a predictive model that can discern relevant documents meeting the profile at the user's query. The predicted model will then be used in the ranking phase to classify other document results from a new query submitted by the same user.

Similarly, we have devoted a part of this article to describe the implementation of a document ranking system of Arabic entitled "SPIRAL". To evaluate the proposed method, we have used a corpus of 30,550 Arabic texts that covers topics related to the field of «علوم طبيعية» "natural sciences". The results of our evaluation ranking system prove the performance of the latter. In fact, we noticed that the results of our ranking method with the cross-validation model (K-fold with k = 26) are interesting. Indeed, the F-measure is in the order of 59.2%. Similarly, we obtained 61.4% as an accuracy rate. Finally, it can be noted that we have achieved a degree of kappa agreement between prediction and supervision equal to 0.24.

Thus, the accuracy of learning by means of the profile is equal to 61.4%, while that without it is about 40.4%. In addition, we note that the semantic learning criteria related to the user have a positive impact on the performance of SPIRAL system. This justifies our choice of the integration of the hybrid user profile into the learning criteria.

At this stage, we can distinguish several research perspectives. Therefore, in the short term, we can choose evaluating the user profile by studying the impact of the number of relevant documents in building the profile, the ranking parameter results and the depth of the hierarchy of the concept profile in improving the search results. Similarly, we intend to build a profile based on search history and compare it with our hybrid profile.

It is emphasized that the evaluation method of learning to rank was made using our own corpus "WCAT" and according to a simulation scenario of TREC research sessions. In order to validate the effectiveness of our method in a real research environment, our outlook in the medium and long term, is to evaluate this method using data from a log file of a search engine.

References

1. Jenifer, K., Yogesh Prabhu, M., Gunasekaran, N.: A survey on web personalization web approaches for efficient information retrieval on user interests. Journal of Recent Research in Engineering and Technology, 2, 34–39 (2015)
2. Jayanthi, J., Rathi, DR. S.: Personalized web search methods – A complete review. Journal of Theoretical and Applied Information Technology (JATIT). 62, 685–697 (2014)
3. Akhila, G.S., Prasanth, R.S.: A Survey on Personalized Web Search. Journal of Advanced Research Trends in Engineering and Technology (IJARTET). Vol.2 (2015)
4. Safi, H., Jaoua, M., Belguith, L.: PIRAT: a Personalized Information Retrieval system in Arabic Texts based on a hybrid representation of a user profile. In: 21st International Conference on Applications of Natural Language to Information Systems (NLDB), pp. 326–334, LN CS, Springer, Royaume-Uni (2016)
5. Shou, L., Bai, H., Chen, K., Chen, G.: Supporting Privacy Protection in Personalized Web Search. IEEE Trans. on Knowledge and Data Engineering. 26, 453–467 (2014)
6. Cossock, D., Zhang, T.: Subset ranking using regression. In: 19th annual conference on learning theory, pp. 605–619. Springer-Verlag, Heidelberg, Berlin (2006)
7. Chapelle, O., Keerthi, S. S.: Efficient algorithms for ranking with svms. Information Retrieval, 13, 201–215 (2010)
8. Xia, F., al.: Listwise approach to learning to rank: theory and algorithm. In: 25th International conference on machine learning, pp. 1192–1199. ACM, New York, NY, USA (2008)
9. Taylor, M.: Softrank: optimizing non-smooth rank metrics. In: International conference on web search and web data mining, pp. 77–86 (2008)
10. Zoeter, O., Taylor, M., Snelson, E., Guiver, J., Craswell, N., Szummer, M.: A decision theoretic framework for ranking using implicit feedback. In: SIGIR 2008 workshop on learning to rank for information retrieval (2008)
11. Xu, J., Li, H.: Adarank: a boosting algorithm for information retrieval, In: 30th annual international ACM SIGIR conference on research and development in information retrieval, pp. 391–398, ACM, New York, NY, USA (2007)
12. Yue, Y., Finley, T. Radlinski, F., Joachims, T.: A support vector method for optimizing average precision. In: 30th annual International ACM SGIR conference on research and development in inform. retrieval, pp. 271–278, ACM, New York, NY, USA (2007)
13. Cao, Z., Qin, T., Liu, T.-Y., Tsai, M.-F. Li, H.: Learning to rank: from pairwise approach to listwise approach. In: 24th International conference on machine learning, pp. 129–135, Technical Report, Corvallis, OR, ACM, MSR-TR-2007-40 (2007)
14. Chapelle, O., Zhang, Y.: A dynamic bayesian network click model for web search ranking, In: 18th International conference on World Wide Web, pp. 1–10, New York, USA, ACM (2009)
15. Liu, T.-Y.: Learning to rank for information retrieval. LNCS, Berlin Heidelberg (2011)
16. Baccini, A., Déjean, S., Lafage, L., Mothe, J.: How many performance measures to evaluate information retrieval systems? Knowledge and Information Systems, 30, 693–713 (2012)
17. Baziz, M. : Indexation conceptuelle guide par ontologie pour la recherché d'information. Ph.D. dissertation, Université de Paul Sabatier, Paris (2005)
18. Mezaour, A. D. : Recherche ciblée de documents sur le web. Ph.D. dissertation, Université Paris-Sud, Paris (2005)
19. Abouenour L., et al.: Improving Q/A using Arabic Wordnet. In: International Arab Conference on Inform. Technology (ACIT), Zarqa Private University, Jordan (2008)
20. Xu, J., Fraser, A., Weischedel, R.: Empirical Studies in Strategies for Arabic Retrieval. In: Ann. ACM Conference on Research and Development in Inform. Retrieval: In: 25th annual

International ACM SIGIR conference on Research and development in inform. Retrieval, 11, 269–274 (2002)

21. Bessou, S., Saadi, A., Touahria, M.: Vers une recherche d'information plus intelligente application à la langue arabe. In: 1st International Conference on Inform. System and Economic Intelligence (SIIE), Hammamet, Tunisia (2008)
22. Hammo, B.: Effectiveness of Query Expansion in Searching the Holy Quran. In: International Conference on Arabic Language Process. (CITALA), pp. 18–19, Morroco (2007)
23. Zaidi, S., Laskri, M.: Expansion de la requête Arabe sur le réseau internet. In: Barmajiat CSLA: Les applications logicielles en arabe: Pas vers l'e-gouvernement, Alger (2007)
24. Farag, A., Andreas, N.: AraSearch: Improving Arabic text retrieval via detection of word form variations. In: International Conference on Information System and Economic Intelligence (SIIE), Hammamet, Tunisia (2008)
25. Abderrahim, M.-A. Abderrahim, M.-E., Chikh, M.-A.: Using Arabic WorldNet for Query Expansion in Information Retrieval System. In: IEEE 3th International Conference on Web and Information Technologies, Marrakech, Morocco (2010)
26. Abderrahim, M-A. : Utilisation des ressources externes pour la reformulation des requêtes dans un système de recherche d'information. The Prague Bulletin of Mathematical Linguistics, PBML, 99, 87–99 (2013)
27. Kleinberg, J.: Authoritative sources in a hyperlinked environment. Journal of the ACM, 5, 604–632 (1999).
28. Kohavi, R., Quinlan, R.: Decision tree discovery. Handbook of Data Mining and Knowledge Discovery, University Press, pp. 267–276 (1999)
29. McLachlan, Do., K.-A., Ambroise., C.: Analyzing Microarray Gene Expression Data. NJ, USA (2004)
30. Arabic Wordnet, http://wwww.globalwordnet.org/AWN/ AWNBrowser.html
31. Amine Arabic Wordnet, http : //amine-platform.sourceforge.net