Neural-Symbolic Machine-Learning for Knowledge Discovery and Adaptive Information Retrieval

Hager Kammoun, Jean Charles Lamirel, and Mohamed Ben Ahmed

***Abstract*—In this paper, a model for an information retrieval system is proposed which takes into account that knowledge about documents and information need of users are dynamic. Two methods are combined, one qualitative or symbolic and the other quantitative or numeric, which are deemed suitable for many clustering contexts, data analysis, concept exploring and knowledge discovery. These two methods may be classified as inductive learning techniques. In this model, they are introduced to build "long term" knowledge about past queries and concepts in a collection of documents. The "long term" knowledge can guide and assist the user to formulate an initial query and can be exploited in the process of retrieving relevant information. The different kinds of knowledge are organized in different points of view. This may be considered an enrichment of the exploration level which is coherent with the concept of document/query structure.

***Keywords*—Information Retrieval Systems, machine learning, classification, Galois lattices, Self Organizing Map.

I. Introduction

INFORMATION Retrieval Systems (IRS) are essential tools in view of the large amount of information available to the user. An IRS is considered an "intelligent interface" between a user wishing to retrieve relevant documents, with respect to his information need, and a database. In general, the information need is expressed by a query and the database is presented as a collection of documents. The query may be complex and the documents of the database can be of various and evolving nature.

For a conventional IRS, an internal model of the user and the query is build. Documents are then analysed and indexed to build an internal representation of documents. Finally, the query representation is scored or matched against a single document representation to produce a ranked document list. The main examples in this context are the **boolean model**, the **vector based model** with the system SMART using *tf×idf* approach [19] [20] and the **probabilistic model** using probabilistic indexing [4][17].

To extend IRS capabilities techniques such us query-modification, human-interaction and other techniques which are related to the retrieving process, are introduced.

The query-modification is one solution to the issue of translating an information need into a query [7]. The query-modification can be made automatically or interactively to expand or to refine the query using various knowledge sources. Such sources include the relevance feedback provided by the user [10], the top ranked documents retrieved by the system [25] and the general purpose thesauri [23]. Combination of these different methods of query-modification can lead to more effective results [13].

Moreover, the query-modification approach attempts to resolve the problem of query complexity, albeit, partially. By this mechanism, it is possible for the system to acquire knowledge about the user. The latter may have different kinds of needs: precise, exploratory, thematic and connotative [3]. Thus, the IRS has to present information in several forms e.g.; lists, graphs, trees, lattices etc.

In addition to the techniques above, others are related to how to analyse and represent the content of the collection by exploiting relations existing between documents or concepts in the same document. Examples of these are: latent semantic indexing [6], generalized vector based model [12][15], inference networks with the system INQUERY [22] and 2-Poisson model with the system OKAPI [24]. Another technique is the hierarchical cluster-based ranking in which the query is not ranked against individual documents but against a hierarchical grouped set of documents clusters [16]. These techniques are suitable only for a static context of the retrieval, which is a limitation to treat dynamic information.

To perform a relevant retrieval, different kinds of related knowledge have to be considered and different operations have to be performed in a context that has to be maintained, updated and adapted to demands related to users and to the collection.

The problem that we are concerned with is to propose a model for an IRS that takes into account the complexity of the user query, by means of decomposing it in different *points of view* (keywords, authors, terms from the full text, citations, etc.). These points of view provide complementary means for accessing the collection. Also, we propose to consider documents from different angles and various structures. This approach can be advantageous to the user in the process of retrieving relevant information.

To take into account the evolving character of knowledge (about user and collection), we choose mainly to combine, a numeric and a symbolic methods which are suitable for different contexts of clustering and information synthesis. Both methods chosen are suitable for the process of machine learning which is necessary to introduce in an IRS, if we need to build a dynamic system. Conventional methods cited are without the ability of learning. Although, the probabilistic model introduces the learning, it is not in a profound way.

The machine learning process is able to build a "long term" knowledge which allows following the evolution of users' interest and guides them.

The contribution of the model we propose is that it provides the user the means to make a preliminary search in past queries. The objective is to start the research by an initial query or by exploring the past queries organized in a

Hager Kammoun is with RIADI-GDL, Ecole nationale des Sciences de l'Informatique, Campus Universitaire La Manouba, Tunis, TUNISIE (e-mail : hager.kammoun@isd.rnu.tn).

Jean Charles Lamirel is with LORIA, BP 239, 54506 Vandoeuvre CEDEX, Nancy, France (e-mail : lamirel@loria.fr).

Mohamed Ben Ahmed RIADI-GDL, Ecole Nationale des Sciences de l'Informatique, Campus Universitaire La Manouba, Tunis, TUNISIE (e-mail : Mohmed.Benahmed@riadi.rnu.tn).

user friendly way. This is similar to Case Based Reasoning [5] or by analogy.

This step is meant to make an automatic or interactive query-modification. Contrary to the conventional query-modification which deals with documents, here we deal with queries. By analogy to *document feedback,* we call this step *query feedback*. This step is a **surface analysis** in which the system knowledge is memorized in the form of concepts related to queries by user(s) and/or by point of view. The second step is an **in-depth analysis**, invoked only if the user is not satisfied with the response obtained in the first step. In fact, it concerns the research in the collection. An intelligent synthesis of the collection has to be made to reduce the space of exploration and to exploit the relations between the different concepts treated in the collection.

The first section gives details about the model proposed. The second section is devoted to some experimental results related to the surface analysis step that illustrate the choice of Galois lattices rather than the Self Organizing Map for this step.

II. The Proposed Model

The two principal components of an IRS are the documents and the user with his information need. Different type of knowledge may be elaborated about these two components: knowledge about users, information need, documents and domain concepts. Knowledge about the user may be related to a step in a research session, a complete session or several sessions. It is possible to classify this knowledge into:

--"Short term" knowledge associated to a step or different steps of one session. It is elaborated by synthesizing the user need and by updating it in an incremental manner.

--"Middle term" knowledge is elaborated by taking into account the user queries and his relevance feedback about documents. This form of knowledge is not usually used in IRS, it is generally pre-defined.

--"Long term" knowledge is related to a user centre of interest, clustering documents or incremental correction of indexing. This allows to build established knowledge about content of the collection.

The "short term" knowledge contributes to build the "middle" and the "long term" knowledge.

In addition to clustering documents, our objective is to introduce clustering queries to build this knowledge. Our objective is to improve the conventional model by proposing to the user, as alternatives, surface and in-depth analysis. Moreover, clustering queries/documents reduces considerably the space of search.

The surface analysis allows the user to explore validated queries which may have been previously modified. The objective is to help the user in the process of formulating his query. In surface analysis query-modification/feedback is expressed according to the following equation (1) adopted from the Rocchio equation [18]:

$$Q^{new} = \alpha Q^{old} + \beta \frac{1}{|relquer|} \sum_{relquer} w_{t_i} - \gamma \frac{1}{|nonrelquer|} \sum_{nonrelquer} w_{t_i} \qquad (1)$$

relquer corresponds to relevant queries.
nonrelquer corresponds to irrelevant queries.
α, β and γ are parameters.

By exploring past queries in the form of lattices (query profiles), the user may gain useful information about similar subjects of interest. He, then, can select the appropriate terms and themes.

At this stage, we use the Galois lattices algorithm (symbolic machine learning) applied to queries. Here, the query is assumed to be a set of terms.

The theory of concept lattices was introduced to support user interface design [8][9] and to cluster documents in a context of information retrieval. Carpineto [2] named this technique concept lattice-based ranking. The advantages of this method are its theoretical assumption since it is based on mathematical foundation and its operational implementation.

The in-depth analysis relates to the exploration of documents by using clustering approaches. Our objective is to extend the exploration in case of surface analysis failure. Similar to surface analysis, the user can start with a query or by browsing information in the form of lattices (document profiles or user profiles). At this stage, we use Kohonen Self Organizing Map (SOM) (numeric machine learning) applied to documents. The main advantages of SOM are its robustness and its graphical interface for displaying a collection of documents. It was applied successfully for several classification tasks and exploited as a tool for knowledge discovery [11]. It was also applied to IR and to the context of classification documents [12][14].

In this work, the following assumptions are made:

--An information need is expressed under different points of view (keywords, authors, citations, abstracts, etc.) and it can be of different natures (precise, exploratory, connotative).

--A document profile is a result of clustering documents without taking into account the feedback of the user. Profiles of documents cover all documents of the collection.

--A user profile is a result of clustering documents, taking into account the feedback of the user. This knowledge covers parts of the collection.

--A query profile is a result of clustering past queries. For each query we associate a set of relevant documents.

These profiles are evolving knowledge learned by the system on "short" or "long" terms. They may be considered as a memory associated to different sessions covering centres of interest of a user or shared centres of interest of a set of users (user profiles and query profiles).

The architecture of the proposed model is shown in Figure 1.

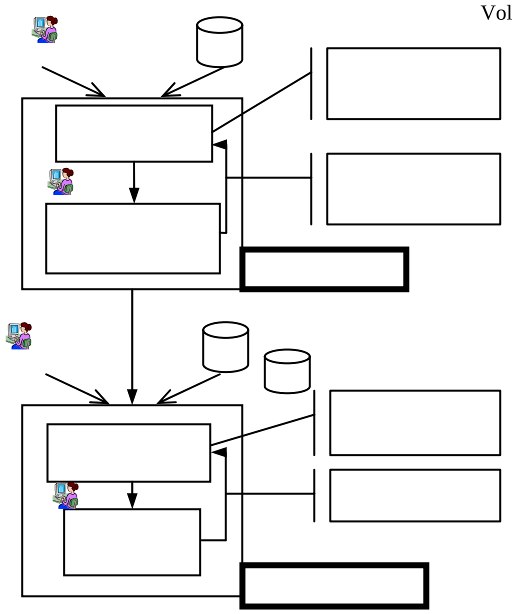

Fig. 1Architecture of the proposed model

III. Experiments about Surface Analysis

Our objective in this experiment is to test the advantages of introducing a surface analysis in the IR process. To do this, we choose the Cranfield collection (CRAN: ftp://ftp.cs.cornell.edu/pub/smart/cran) which is a reference collection for evaluating IRS and consists of 225 queries and 1400 documents. The queries can be partitioned to form learning and a validating/testing base. The description vocabulary of queries is about 585 terms and that of documents is about 3763 terms. This situation is in favour of the use of a search oriented query due to the lower space dimension of query description.

The procedure of generating the lattice and classifying test queries is the same for all experiments made.

In the following, we describe the machine learning process (lattice generation) and the classification using the Galois lattice algorithm.

A. Lattice Generation and Classification

The Galois lattice algorithm consists in two steps: the first one allows us to build a lattice of concepts representing the relations between queries. This step can be considered as machine learning since this algorithm is incremental and can update the lattice representing the knowledge. The second step allows us to classify a given query.

On the one hand, the lattice can be specific to one user. In this case, it consists of concepts that identify different interests. On the other hand, the lattice can be specific to several users. In this case, it consists of concepts that regroup them.

We limit our experiments to the data proposed in the CRAN collection in which a query is defined as a set of terms which corresponds to one point of view and one user.

In the context of queries, each concept in the lattice is identified by two main information:

--An intent is described by a set of terms that are shared by a set of queries. They will serve to identify the relevant concept(s) according to a given test query which is classified in the lattice.

--An extent is described by a set of queries that share the terms of the intent part. It will serve to calculate the Recall and the Precision for a given test query after its classification.

The classification step consists in providing for a given test queries the relevant concept(s) in the lattice. To determine the relevant concept(s) we proceed by using the cosine measure (the normalized version) calculated between the vector of the test query formed by terms and the one corresponding to the intent part of a given concept according to the following equation (2):

$$Cosine\quad (Q_i, IntC_j) = \frac{\sum_{k=1}^{N} q_{ik} IntC_{jk}}{\sqrt{\sum_{k=1}^{N} q_{ik}^{2}}\sqrt{\sum_{k=1}^{N} IntC_{jk}^{2}}} \qquad (2)$$

Q_i represents the vector corresponding to the test query i, $Q_i=(q_1,q_2,\ldots,q_N)$.

$IntC_j$ represents the vector associated to the intent part of the concept j, $InC_j=(c_1,c_2,\ldots,c_N)$.

N represents the size of the description vocabulary of test queries.

The vectors are binary, in reality the frequency of terms per query is either 0 or 1 or 2.

A threshold according to the cosine value or a break value related to the number of relevant concepts can be used.

Recall and the Precision are the two major criteria to evaluate the effectiveness of an IRS and of classification. To calculate Recall and Precision, we consider on the one hand, the test query with the associated set of the relevant documents (this information is given by the CRAN collection). On the other hand, we consider the set of relevant documents associated to the set of queries corresponding to the extent part of the relevant concept. We proceed us described by equations (3) and (4).

The F-measure [16] is a global measure that combines Recall and Precision, as described in equation (5).

$$R(Q_i, ExtC_j) = \frac{\left|DocPert_{Q_i} \cap DocPert_{ExtC_i}\right|}{\left|DocPert_{Q_i}\right|} \qquad (3)$$

$$P(Q_i, ExtC_j) = \frac{\left|DocPert_{Q_i} \cap DocPert_{ExtC_j}\right|}{\left|DocPert_{ExtC_j}\right|} \qquad (4)$$

$DocPert_{Q_i}$ *Corresponds to relevant documents according to a given test query.*

$DocPert_{ExtC_i}$ *Corresponds to relevant documents associated to queries that constitute the extent part of the concept C_j.*

$$F = \frac{2(TR \times TP)}{(TR + TP)} \qquad (5)$$

By separating queries of the CRAN collection into two sets; test queries and learning queries, the vocabulary of each one can be altered with respect to the total vocabulary. Then, some adjustments have to be made.

B. Experimental Results

We have used the implemented version CLOSE [1] and the average of the F-measure to make comparisons between the results obtained.

Experiment 1

In the first experiment, we have partitioned the base in test queries and learning queries by reserving for the learning step queries having more terms. This is done, by assuming that they cover the majority of the vocabulary of

TABLE I
RESULTS OF EXPERIENCE 1

Number of concepts in the lattice	725
Number of test queries	106
Percentage of classified queries (F-measure≠0)	**27%**
Average of F-measure	**0.3**

the queries. We have chosen the threshold 9 of the number of terms: queries with number of terms ≥ 9 are used for the learning and queries with number of terms < 9 are used for the test. Results are presented in the Table I.

Experiment 2

In this experiment we have reversed our choice by reserving queries with the fewer number of terms for the learning and the larger number of terms for the test (queries with number of terms ≤ 9 are used for the learning and those with number of terms > 9 for the test). We obtain the results shown in the Table II.

TABLE II
RESULTS OF EXPERIENCE 2

Number of concepts in the lattice	461
Number of test queries	84
Percentage of classified queries (F-measure≠0)	**41%**
Average of F-measure	**0.4**

Experiment 3

In this experiment we have proceeded randomly. The 150 first queries of the base are used for learning and the remaining 75 queries for the test. Results are presented in Table III.

From these experiments, it is observed that the classification results in the second experiment (see Table II) are better than those obtained in the first one (see Table I). This can be justified by the sparse character of the learning queries (with more terms) which served for the first experiment. Furthermore, the vocabulary that was used for the learning does not cover the vocabulary associated to the test queries. In the third experiment, we obtained intermediary results for the average of the F-measure and better percentage of classified queries (see Table III). In fact, we have mixed the queries, with different thresholds, that were used for the learning.

We carried out other experiments. In some, we have modified the threshold for choosing the queries used in the learning by reducing it. It was observed that the quality of results was degraded. This observation may be explained by the fact that the number of learning queries decreases, whereas the number of test queries increases. Consequently, the description vocabulary used for the learning step does not cover the one used for the classification. In other experiments, we have considered more than one relevant concept; we have observed an increase of the Recall and a decrease of the Precision and the F-measure.

Experiment 4

In this experiment, we have fixed the number of terms for the test queries, the rest is used for the learning. We have selected the threshold value of 6. Results are presented in Table IV.

Experiment 5

In this experiment, we have used the same set of test queries but we have separated the learning ones into two sets. The results obtained are shown in Table V.

We observe that the quality of the results presented in Table IV are better than those in Table V, in terms of percentage of classified test queries and the average of F-measure. This can be explained by the fact that in experiment 4 the learning set corresponds to the union of the learning sets of experiment 5. Moreover, we observe that the quality of classification depends on the proximity between number of terms of the test queries and those of the learning ones (see Table V).

TABLE IV
RESULTS OF EXPERIENCE 4

Test corpus: number of terms per query=6	Number of learning queries	205
	Number of test queries	20
	Number of concepts	1100
	Percentage of classified queries (F-measure≠0)	**50%**
	Average of F-measure	**0.5**

TABLE V
RESULTS OF EXPERIENCE 5

Learning Corpus: number of terms per query≤8	Number of learning queries	86
	Number of test queries	20
	Number of concepts	261
	Percentage of classified queries (F-measure≠0)	**45%**
	Average of F-measure	**0.4**
Learning query: number of terms per query≥9	Number of learning queries	119
	Number of test queries	20
	Number of concepts	725
	Percentage of classified queries (F-measure≠0)	**25%**
	Average of F-measure	**0.44**

TABLE III
RESULTS OF EXPERIENCE 3

Number of concepts in the lattice	658
Number of test queries	75
Percentage of classified queries (F-measure≠0)	**44%**
Average of F-measure	**0.34**

C. *Comparison with SOM*

Our objective, here, is to check the advantage of using the Galois lattices for the classification applied to queries in the surface analysis instead of the Kohonen SOM.

The SOM is classified among competitive learning methods belonging to a class recognized as “Winner takes most”. The idea behind this method is to map (in a non linear process) an input data of n-dimensions onto two or three dimensional (2D, 3D) map of reference vectors. The algorithm uses a distance function to calculate the best match between the input data and the reference vector node and to update the node and it’s neighbourhood to reassemble the input data. At the end of the learning step, the reference vector of each class or node in the map plays the role of a representative of a class of individuals. The topographical properties of the Kohonen map make possible the reprojection of an individual on the map.

A large size of the map gives more details and a small one gives more generalities, about classes generated.

Experiment 6

In this experiment, we have considered the same conditions of experiment 2. The learning step consists in generating the map. In our case, we have chosen to generate a 2D map (XxY: X represents x axis and Y represents y axis). The test step consists in projecting test queries over the map, which determines the relevant class or classes.

The generation of the map is made by tools of the platform MultiSOM [12]. Recall and Precision are calculated by considering the test query and the set of queries corresponding to the relevant class (the nearest one).

TABLE VI
RESULTS OF EXPERIENCE 6

Projection of test queries over the map	Map 7x7	Map 12x12	Map 20x20
Recall	0.21	0.13	0.15
Precision	0.03	0.08	0.14
Average of F-measure	**0.06**	**0.1**	**0.14**

Table VI shows that the results obtained using Galois lattice are of a better quality (see Table II) than those obtained using SOM.

IV. Conclusion

We have demonstrated that to obtain good results for classification of test queries, the system must learn enough through sufficient number of queries having different number of terms and are correlated. Moreover, the distribution of queries sizes (test queries and learning ones) must be homogenous.

The added value of a surface analysis in an IRS is that it can be a way to assist the user in the query formulation and to expand the query automatically or interactively. It can also be a good alternative to the in-depth analysis

The choice of Galois lattice was motivated by their symbolic character which is in coherence with the query representation and by their incremental aspect which responds to the evolving character of knowledge.

We have demonstrated that the Galois lattices give better quality results for queries classification (learning and testing) than the SOM.

Our future focus will be oriented to test the SOM applied to documents clustering by using tools of the MultiSOM platform.

References

[1] Y. Bastide. Data mining: algorithmes par niveau, techniques d'implantation et applications. Theses of University Doctor, University Blaise Pascal, 2000.

[2] C. Carpinetto & G. Romano. Order-theoretical ranking. Journal of the American Society for Information Science, 51(7):587-601, John Wiley and Sons Incorporation, 2000.

[3] M. Cluzeau-Ciry. Typologie des utilisateurs et des utilisations d'une banque d'images. Le documentaliste, 25(3), 155-120, 1988.

[4] W.B. Croft & D.J. Harper. Using probabilistic models of document retrieval without relevance information. Documentation, 35, 285-295, 1979.

[5] P.J. Daniels & E.L. Rissland. A Case Based Approach to Intelligent Information Retrieval. In Proceedings of Conference on Research and Development in Information Retrieval (ACM SIGIR'95), pp. 238-245, Seattle WA, USA, 1995.

[6] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landuer, & R. Harshman. Indexing by Latent Semantic Analysis. Journal of the American Society for Information Science, 41(6):391-407, 1990.

[7] G.W. Furnas, T.K. Landauer, L.M. Gomez & S.T. Dumais. The vocabulary problem in human-system communication. Communications of the ACM, 30, 964-971, 1987.

[8] R. Godin, J. Gecsi & C. Pichet. Design of browsing interface for information retrieval. In Proceedings of the 12th International Conference on Research and Development in Information Retrieval (ACM SIGIR'89), pp. 32-39, Cambridge, MA: ACM, 1989.

[9] R. Godin R. Missaoui & A april. Experimental comparison of navigation in a Galois lattice with conventional retrieval methods. International Journal of Man Machine Studies, 38, 747-767, 1993.

[10] D. Harman. Relevance feedback and other query modification techniques. In W.B. Frakes & Baeza-Yates (Eds). Information retrieval data structures and algorithms, pp. 241-263, Englewood cliffs, NJ:Prentice Hall, 1992.

[11] T. Kohonen. Self-organization and associative memory. Springer-Verlag, third edition, 1989.

[12] J. C. Lamirel. Application d'une approche symbolico-connexioniste pour la conception d'un système documentaire hautement interactif, le prototype NOMAD. Theses of University Doctor, University Henri-Poincaré, Nancy I, Nancy France, 1995.

[13] J. H. Lee. Combining the evidence of different relevance feedback methods for information retrieval. Information Processing Management, 34(6):681-691, 1998.

[14] X. Lin, D. Soergel & G. Marchionini. A self organizing semantic map from information retrieval. In proceedings of 4th international SIGFIR conference on R&D in information retrieval, pp. 262-269, 1991

[15] J-Y. Nie. An outline of general model for information retrieval systems. In Proceedings of International Conference on Research and Development in Information Retrieval (ACM SIGIR'88), pp. 495-506, 1988.

[16] V. Rijsbergen. Information retrieval. London: Butterworths, 1979.

[17] S.E. Roberston & K. Sparck-Jones. Relevance weighting of search terms. Journal of The American Society for Information Science, 27:129-146, 1976.

[18] J. J. Rocchio. Relevance feedback in information retrieval. Technical report ISR-9, Computational Sciences Department, University of Cornell, Ithaca, N.Y., 1965. Reprinted in The Smart Retrieval System, Edition G. Salton, 1971.

[19] G. Salton. The SMART retrieval system. Prentice-Hall, Englewood Cliffs, N. J., 1971.

[20] G. Salton & C. Buckley. Term weighting approaches in automatic text retrieval. Information Processing and Management, 24, 513-523, 1988.

[21] P. Thompson & W. B. Croft. Support for browsing in intelligent text retrieval system. International Journal in Man Machine Studies. 30, 639-668, 1989.

[22] H. Turtle & W. B. Croft. Evaluation of an inference network-bases retrieval model. ACM Transactions on Information Systems, 9(3):187-222, 1991.

[23] E. M. Voorhees & D. Harman. Overview of the sixth text retrieval conference (TREC-6). NIST special edition, 1993.

[24] S. Walker, S.E. Roberston. M. Boughanem, G. J. F. Jones & K. Sparck-Jones. OKAPI at TREC-6. In Proceedings of the sixth Text Retrieval Conference (TREC-6), NIST Special Publication, 1997.

[25] J. XU & B. croft Query expansion using local and global document analysis. In proceedings of the 19th international conference on research and development in information retrieval (ACM SIGIR'96), pp. 4-11, Zurich:ACM.

[26] H. Zhang, S. W. Smoliar & J. H. Wu. Content-based video browsing tools. Multimedia computing and networking, vol. 2417-35. IS&T-SPIE, 1995.