Profiles re-usability for personalized information access: application to users contexts determination

Max Chevalier[*,**], Chantal Soulé-Dupuy[*], Pascaline Laure Tchienehom[*]

[*]IRIT, 118 route de Narbonne, 31062 Toulouse cedex 4, France
Max.Chevalier@irit.fr, Chantal.Soulé-Dupuy@irit.fr,
Pascaline.Tchienehom@irit.fr
[**]LGC, 115 avenue de Rangueil, 31077 Toulouse cedex, France

Abstract. The notion of context is important when we are dealing with personalization. Users often change their needs in time and the determination of contexts can help to identify those changes. In this article, we associate to each information or user a profile which describes it. We show that the description of a profile can be done by re-using existing profiles and we propose a method to determine various types of users contexts based on structure, contents and re-usability of profiles. We discuss the interest of the suggested method through qualitative experiments carried out with the collection "Los Angeles Times 94" of the CLEF evaluation campaign.
Key Words: Information Access, profile, personalization, context.

1 Introduction

Relevance in information access techniques has led to the consideration of the users contexts for restituting personalized information to each user or users groups. The notion of context can help to build a stronger approach for information access. In this article, we propose a profile generic model for the description of information and users in information access techniques. This model allows the re-use of profiles to describe others profiles. This concept of re-usability, which is materialized by the profile composition, will enable us to define a method that allows the building of various users (research) contexts or interest centers contexts related to a particular task or request. A user context can be: of short or long term, positive and/or negative. The aim is to respond to users waitings. Qualitative experiments carried on the collection "Los Angeles Times 94" (LaTimes 94) allow us to analyse the proposed method.

2 Literature review

Information access techniques allow an individual to obtain information that meets his needs. We can gather them in two main groups: the pull technique, which needs an explicit request of an individual and the push technique, which does not need an explicit demand to return information to users.

Information Retrieval (IR), which is a pull technique, rests on need expression of an individual through a query formulated in a more or less structured free language [1]. However, in Information Retrieval, the real intention of the user is not always obvious in his manner of formulating his query and that can generate ambiguities on the sense of words that it contains. Many solutions exist in order to refine the sense of a query through query reformulation based on: relevance feedback [4], research context [13].

Information Filtering (IF), which is a push technique, is a relatively passive task because the user does not explicitly formulate his needs through a query, as it is the case in IR. In Information Filtering, we rather use a representation of the user called user profile to send information to him. There are several methods of filtering [10] based on: users interests centers [11]; users judgements [6]; users demographic data (age, profession, etc.) [9] or hybrid methods [12].

There is a multitude of information access methods. They are based on a description of information and users (needs) handled by processes of retrieval and filtering that are called profile. The profile of an object is a whole of characteristics which allows to identify and to represent it. The profiles used in information access techniques are of varied nature: user profile, document profile, etc. In this article, we propose a profile generic model and a method for profiles exploitation allowing to determine various users contexts of short term or long term that can be positive and/or negative for individuals or individuals groups. There are existing approaches which determine user context of short term (user profile built on a short period of time) [3] and/or of long term (user profile built on a relative important period of time) [8], [2], [14] or a negative context (user profile describing what the user dislikes) [7] or positive user context (user profile describing what the user likes). The specificity of our method is that it is based on profiles structure and contents and also on profiles re-usability in the description of others profiles. Re-usability of profiles is the fact that a profile structure can be described by other existing profiles. We use this re-usability to deduce various users contexts with an aim to improve quality of information access results. Experiments were carried out on a collection of the CLEF campaign in order to judge the interest of our method.

3 Re-usability of profiles

In this section, we present a generic profile model for the description of any type of profiles for information access. This model allows re-usability of profiles through the concept of composition. This will enable us to define thereafter a transformation method for profiles made up of others profiles and containing several identical attributes, to deduce a simpler profile (profile obtain after transformation) describing users contexts.

3.1 Generic profile model

In order to be able to define profiles which are re-usable and evolutionary, we define a generic profile model.

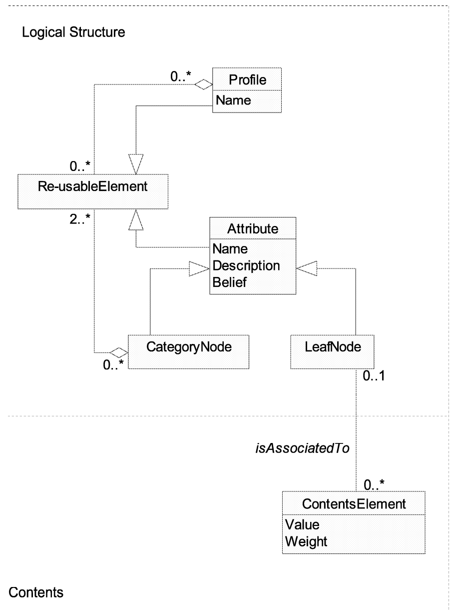

Fig. 1. Profile generic model

The profile generic model of figure 1 presents the general structure of a profile. This structure is in the form of a hierarchy of elements that can be either a profile or an attribute. Those elements characterize a given profile. In this hierarchy an attribute can be either a set or class of nodes (*CategoryNode*) or a leaf (*LeafNode*) that is simply an attribute to which one can affects values.

Profiles derived from this generic model can have the following characteristics:

– *re-usable profiles*: in fact a sub-tree of a profile can have the structure of another existing profile. For example, a long term user profile can be composed of its various usage profiles (or short term profiles);
– *multi-facets profiles*: profiles can be analysed under various aspects (attributes, sub-profiles). Thus, each profile or attribute or combination of those last can constitute a facet of a user;
– *adaptive and evolutionary profiles*: our profiles can be modified and can evolve in time. For example, the user profile can evolve if many of his short term profiles are different from his long term profile.

On the other hand, the organization of the various attributes by categories makes it possible to gather similar attributes in the same class. From the profile generic model, we can derive the structure of various profiles by applying decomposition rules. Figures 2 and 3 present, respectively, examples of profiles structures for: a user profile, an information (document, document parts, thesis, etc.) profile, a users group profile.

More particularly, figure 2 instantiates examples of profiles derived from figure 1 by highlighting: the logical structure (or taxonomy) and the contents of profiles.

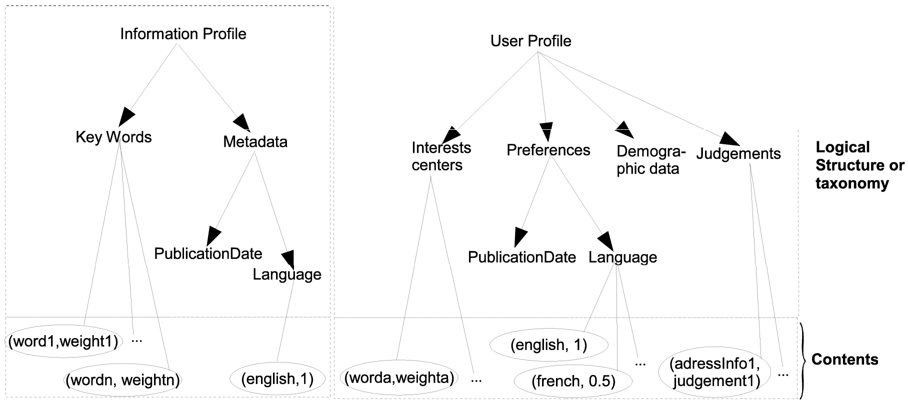

Fig. 2. Information and user profiles examples: structure and contents

However, figure 3 instantiate a profile example also derived from figure 1 which is, in this case, composed of others profiles. This profile illustrates the re-usability of profiles for the description of others profiles. Let us note that various profiles can belong to the same depth of a given tree structure.

The interest of using a generic profile to define any type of profile is that the basic structure it proposes can be used by any application in order to define any type of profiles [5]. The profile generic model of the figure 1, will enable us to derive the structure of various profiles in the form of profiles and attributes hierarchy describing the taxonomy of these profiles (*cf.* figure 2 and 3).

However, profiles composition (or re-usability) can create confrontation of several taxonomies where identical attributes could appear several times in the same tree structure. We use this redundance in the definition of a method allowing to transform these profiles to a structure containing only one profile where each attribute appears once. This method makes it possible to deduce various users contexts with an aim to improve quality of information access results.

3.2 Profiles transformation and contexts determination

The profiles transformation method which we propose is based mainly on existing profiles for the building of others profiles. In this section, we define how to build a simple profile (*cf.* figure 2) by using a description of this profile which is composed of several other profiles. The aim is to transform a structure like the one of figure 3 to a structure like the one of figure 2. This approach allows the exploitation of profiles *re-usability.*

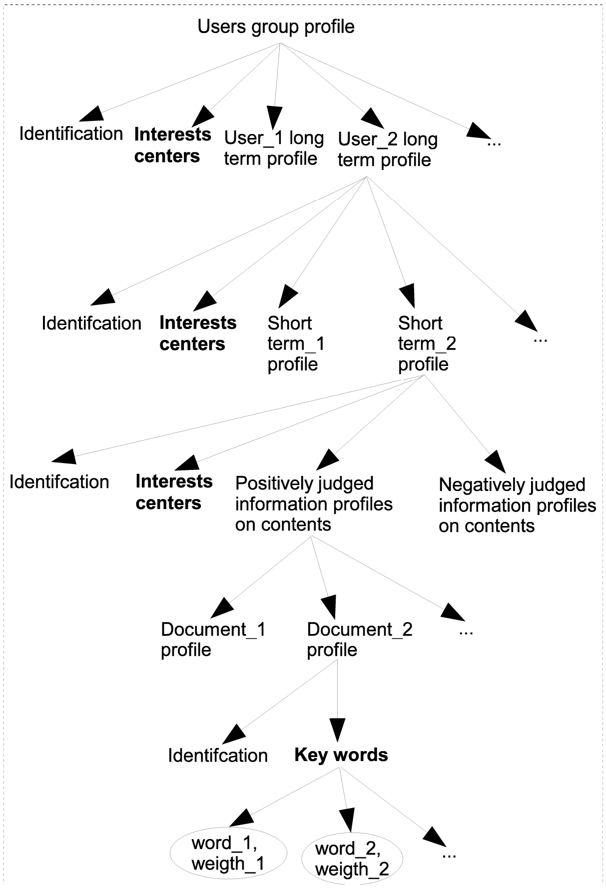

Fig. 3. Illustration of profiles composition for description of users (users group profile)

The proposed method will directly analyse existing profiles rather than analysing rough information. The characteristic of our approach is that it is based on exploitation of profiles structure and contents. Before describing this approach, we have defined a certain number of concepts related to profiles tree structures that we are going to use.

Thus, given A a tree structure with $N = \{\eta_1, \ldots, \eta_n\}$ being the set of A nodes, $C = \{\varsigma_1, \ldots, \varsigma_n\}$ the set of A *leaf* type nodes and $P = \{\rho_1, \ldots, \rho_n\}$ the set of A *profile* type nodes, these concepts are the following:

1. $\forall \varsigma_i \in C$, $\varsigma_i \in N$;
2. $\forall \rho_i \in P$, $\rho_i \in N$;
3. $A[\eta]$ is the sub-tree of A where node η is the root. If $\eta \in C$ then $A[\eta]$ contains only η;
4. $children[\eta]$ is the children list of node η;

5. $parent[\eta]$ is a node representing the parent node of η. If η is the root then $parent[\eta]$ does not exist;
6. $name[\eta]$ is the name associated to node η;
7. $anc[\eta, A]$ is the set of ancestors of node η in tree A, η not included;
8. $path[\eta, A]$ is the list of nodes which composed the path from the root of A to node η, η included. Hence, $path[\eta, A] = \eta + anc[\eta, A]$;
9. $val[\varsigma, A]$ is the value of leaf ς in tree A;
10. $nodeType[\eta]$ is the node type of η which can be: *CategoryNode, LeafNode or Profile*;
11. $profileType[\rho]$ is the profile type of ρ which can be: information profile (document, collection of documents, parts of document, etc.), negative profile, positive profile, short term profile, long term profile, individual profile, group profile, etc.;
12. $depth[\eta, A]$ is the depth of node η in tree A.

The transformation of a profile ρ composed by others profiles into a tree structure only made up of categories and leaves is useful, because of the simplicity of the resulting structure which facilitates their use by information access processes. Note that, *we consider the postulate according to which, two nodes or leaves are semantically identical if they have the same name and the same ancestors of type "CategoryNode"*. Let us note that, however, we regard the criteria *key words* and *interest centers* as synonyms.

In addition, this profiles transformation method will also make it possible to define various contexts (of an individual or group) of short or long term which are positive and/or negative. Indeed, our method makes it possible to deduce a synthesis profile from a profile which is composed of others profiles. Thus, in the case of a user profile composed of profiles describing its various research sessions (documents judged, consulted, etc.), one will be able to deduce the various contexts (description of user needs) of this user.

To transform the structure and modify the contents of profiles composed by others profiles or to determine various users contexts, we defined an algorithm which we present in the following section.

3.2.1 Transformation method for profile composed of others profiles

The transformation algorithm of profiles composed by different others profiles is defined in table 1. This algorithm analyses each leaf or criterion of the tree. The transformation is made in an ascending way. We first seek the maximum profiles depth (noted m) of the tree A to be transformed. Then, for each profile of this depth, we check its type and the one of his profile parent and we apply the corresponding case. For a given case and for each leaf, the stages *paths search* and *paths comparisons* are done first of all:

- *paths search*: we seek the path of leaf ς_i in $A[\rho_i]$ noted $path(\varsigma_i, A[\rho_i])$, as well as the path of this leaf in the tree structure $A[parent[\rho_i]]$ noted $path(\varsigma_i, A[parent[\rho_i]])$. The aim is to obtain the paths leading to leaf ς_i in both tree structures: $A[parent[\rho_i]]$ and $A[\rho_i]$. This also allows to be sure that leaf ς_i, in the two tree structures, has the same semantic;

– *paths comparison*: we compare the paths of ς_i in $A[parent[\rho_i]])$ with that in $A[\rho_i]$ as follow:

$$(path(\varsigma_i, A[\rho_i]) - \rho_i) = (path(\varsigma_i, A[parent[\rho_i]]) - parent[\rho_i])$$

If there is no path filling this constraint then we proceed to the *creation of the path* $(path(\varsigma_i, A[\rho_i]) - \rho_i)$, from node $parent[\rho_i]$.

Then, specific treatments are applied for each profile leaf (key words, language, etc.) of the tree structure. If for a given leaf the principle to follow is not detailed, the leaf disappears in the resulting structure.

In this algorithm (*cf.* table 1), we identified various particular cases of profiles to treat specifically:

Case 1. *profile of any type composed of information profiles*: we define P_{info} a set of information profiles of same depth m for example. Then, specific treatments are carried out for each profiles leaf ς_i of each P_{info} profiles ρ_i:

If $name[\varsigma_i]$ = *key words*: the criterion "key words" being generally multi-valued, we first of all *modify each key word weight value* of $A[\rho_i]$. Given $w_{m_{t,\rho_i}}$ this new weight, it is obtained by dividing the weight w_{t,ρ_i}, of the key word t in the profile ρ_i, by the ratio between the size of profile $A[\rho_i]$ (noted $length_{\rho_i}$) and the average size of ρ_i same depth profiles (noted $length_{moy_{\rho_i}}$). The goal is to obtain a key word weight which is inversely proportional to the normalized size of the element it describes in order to not favour weights of key words resulting from profiles of big size compared to those resulting from profiles of small size. Thus, we have:

$$w_{m_{t,\rho_i}} = \frac{w_{t,\rho_i}}{length_{\rho_i}/length_{moy_{\rho_i}}}$$

Then, we proceed to the insertion of the various $A[\rho_i]$ key words in $A[parent[\rho_i]]$ key words. The principle of insertion for each $A[\rho_i]$ key word, noted t, is the following:

– if the key word t does not exist in leaf "key words" of $A[parent[\rho_i]]$ then we add t with its new weight;
– if the key word t exists in leaf "key words" of $A[parent[\rho_i]]$ then the weight of the key word t in $A[parent[\rho_i]]$ is replaced by the sum between the weight of t in $A[parent[\rho_i]]$ and the modified weight of t in $A[\rho_i]$, noted $w_{m_{t,\rho_i}}$. Thus, the new weight $w_{n_{t,parent[\rho_i]}}$ of the key word t in $A[parent[\rho_i]]$ is calculated as follow:

$$w_{n_{t,parent[\rho_i]}} = w_{t,parent[\rho_i]} + w_{m_{t,\rho_i}} = w_{t,parent[\rho_i]} + \frac{w_{t,\rho_i}}{\frac{length_{\rho_i}}{length_{moy_{\rho_i}}}} \quad (1)$$

Where:
$length_{\rho_i}$ is the size of profile ρ_i;
$length_{moy_{\rho_i}}$ is the average size of ρ_i same depth profiles;

w_{t,ρ_i} is the weight of key word t in profile ρ_i;
$w_{t,parent[\rho_i]}$ is the weight of key word t in profile $parent[\rho_i]$.

The weight w_{t,ρ_i} is divided by the ratio $\frac{length_{\rho_i}}{length_{moy_{\rho_i}}}$, and not by $length_{\rho_i}$, in order to avoid a great difference between the initial weight of term t and its new weight. The ratio $\frac{length_{\rho_i}}{length_{moy_{\rho_i}}}$ allows to normalize the profiles sizes, of a given depth, in interval $[\frac{length_{min_{\rho_i}}}{length_{moy_{\rho_i}}}, \frac{length_{max_{\rho_i}}}{length_{moy_{\rho_i}}}]$ where $length_{min_{\rho_i}}$ is the minimal size of ρ_i same depth profiles and $length_{max_{\rho_i}}$ is the maximum size of those profiles. Hence, $w_{m_{t,\rho_i}}$ is calculated with the *normalized size* of same depth information profiles of the given tree structure.

Case 2. *user profile of any type composed of positive or negative profiles*: specific treatments are carried out for each criterion in this case. Thus, for the criterion *key words (or centers of interests)* for example, we consider the key words weights of negative profiles as being negative and those of positive profiles as being positive and we insert them in the parent profile. The insertion principle is the following:

- if the key word does not exist in the parent profile, we insert it with its positive or negative weight;
- if the key word already exists in the parent profile, we add the positive or negative key word weight in profile ρ_i to the weight of the same key word in ρ_i parent profile. The different weights of key word t in ρ_i and $parent[\rho_i]$ can be modulated by parameters α and β that show the importance of each weight. This operation allows to increase or to decrease the key word weight considered in the parent profile. α and β are determined by experimentations.

Case 3. *long term profile composed of short term profiles*: we define a short term profiles set P_{ct} which compose the long term profile. In order to be able to apply the transformation, it should be checked that the number of short term profiles composing the long term profile is higher or equal to an integer value n fixed by the application. If it is the case, we can define a subset of P_{ct} noted P'_{ct} which contains the n last short term profiles of the user. Then, specific treatments are carried out for each profiles leaf ς_i of P'_{ct} profiles ρ_i.

Thus, for the leaf *"key words" (or interest centers)* for example, *we determine the key words of all P'_{ct} short term profiles.* For each key word of this set, we calculate a weight w_{m_t} which describes at which point a term t is recurring in profiles set P'_{ct}. This weight can be calculated as follow:

$$w_{m_t} = \frac{\sum_{i=1}^{cardinal(P'_{ct})} w_{t,\rho_i}}{cardinal(P'_{ct})}$$

If the absolute value of this weight w_{m_t} is higher than a given threshold then the key word is inserted in the long term profile (parent profile). The insertion principle is the following:

– if the key word does not exist in the parent profile, we insert it in that parent profile with its weight w_{m_t} (which can be positive or negative);
– if the key word already exists in the parent profile, we add the weight w_{m_t} to the weight of the same key word t in the parent profile. The new weight w_{n_t} of the key word t in the long term profile, is calculated as follow:

$$w_{n_{t,parent[\rho_i]}} = w_{t,parent[\rho_i]} + \frac{\sum_{i=1}^{cardinal(P'_{ct})} w_{t,\rho_i}}{cardinal(P'_{ct})} \tag{2}$$

Where:
$cardinal(P'_{ct})$ is the profiles number of set P'_{ct};
w_{t,ρ_i} is the key word t weight in profile ρ_i;
$w_{t,parent[\rho_i]}$ is the key word t weight in profile $parent[\rho_i]$.

Let us note that a non-weighted key word is regarded as having a weight of 1. Moreover, the selection threshold of a key word, in order to integrate it into the long term profile, can be determined through experiments by analysing the P'_{ct} modified key words weights (w_{m_t}) distribution.

Case 4. *users group profile composed of individual users profiles*: the principle is analogue to the previous one with the nuance that we consider all the individual profiles that compose the group profile.

Let us note that at the end of the various transformations, we proceed to the suppression of some *key words* or *interest centers* of A root, whose weight is lower than a given threshold θ. θ is determined through experiments, by analysing the weights distribution of the "key words" or "interest centers" resulting from the various transformations. The goal of this operation is to eliminate the terms which have a very low weight in comparison to others;

In addition, the proposed method of profile transformation also allow the *evolutionary* of user profiles contents in time. This is made through modification of a user long term profile using his different short term profiles.

We describe, in the following section, the experiments which we carried out on the *profiles transformation algorithm and users contexts determination* with the collection "LaTimes 94" of the CLEF evaluation campaign.

3.2.2 Preliminary experimentations The experiments consisted in building three types of user contexts (or profiles) with our algorithm: a positive short term context, a negative short term context and a positive and negative short term context (combination of a positive and a negative context). This was done on the basis of the first five requests of CLEF 2001 which defined our research tasks. The various users contexts are constructed with documents judged (positively or negatively) by a user for a given request. The positive and negative profiles were composed of information (documents judged) profiles and the positive and negative profile was composed by the previous positive profiles and negative profiles (*cf.* figure 3). Table 2 shows an example of the algorithm results for the fifth query of CLEF 2001.

Table 1. Transformation algorithm for a profile composed of others profiles

1. Determination of set P of the tree A to be transformed
2. Ascending construction
    2.1 Research of the maximum depth of A profiles: noted m
    2.2 From profiles of depth m to profiles of depth $m-1$
    Repeat
        For each profile ρ_i of depth m belonging to P do
            Case ρ_i
            2.2.1 Case 1
                $profileType[\rho_i]$=*information profile:*
                    For each leaf ς_i of $A[\rho_i]$ do
                        Particular treatments for each leaf
                    End for
            2.2.2 Case 2
                $profileType[\rho_i]$=*negative profile:*
                    For each leaf ς_i of $A[\rho_i]$ do
                        Particular treatments for each leaf
                    End for
            2.2.3 Case 3
                $profileType[\rho_i]$=*short term profile*
                AND $profileType[parent[\rho_i]]$=*long term profile*:
                    Determination of set P_{ct} composed of profiles ρ where:
                    $(depth[\rho, A] = depth[\rho_i, A]$ *AND* $parent[\rho] = parent[\rho_i]$
                    AND $profileType[\rho]$=*short term profile*)
                    If $card[P_{ct}] > n$ then
                        For each leaf ς_i of $A[\rho_i]$ do
                            Particular treatments for each leaf
                        End for
                    End If
                    Suppression of sub-trees $A[\rho]$ of A, where: $\rho \in P_{ct}$
                    AND suppression of P_{ct} profiles in set P
            2.2.4 Case 4
                $profileType[\rho_i]$=*individual profile*
                AND $profileType[parent[\rho_i]]$=*group profile:*
                    For each leaf ς_i of $A[\rho_i]$ do
                        Particular treatments for each leaf
                    End for
                    Suppression of sub-trees $A[\rho]$ of A, where:
                    $(depth[\rho, A] = depth[\rho_i, A]$ *AND* $parent[\rho] = parent[\rho_i]$
                    AND $profileType[\rho]$=*individual profile*)
                    AND suppression of these profiles ρ in set P
            End case
            Suppression of $A[\rho_i]$ in tree A
        End for
        $m \leftarrow m - 1$
    Until $(m = 0)$

Table 2. Illustration of the algorithm results

initial query	positive context	negative context	negative and positive context
Peace 3	Peace 95.33	peace 52.76	Peace 42.57
Treaty 4	Treaty 103.92		Treaty 103.92
Israel 2	Israel 118.17	Israel 47.67	Israel 70.50
Jordan 2	Jordan 69.40	Jordan 35.78	Jordan 33.62
	Israeli 50.33		Israeli 50.33
	Jordanian 28.21		Jordanian 28.21
		PERSPECTIVE 48.67	PERSPECTIVE -48.67
		HEBRON 35.78	HEBRON -35.78

We selected words morphologically closed to the words of the initial query using the *Levenstein distance* and also words with an absolute weight value above a given threshold. We then evaluated the similarity measure between the initial requests, here the first five requests of the collection CLEF 2001, and the various users contexts (or profiles) obtained from a user judgements of relevance or non-relevance on documents relative to these requests with our proposed method. This similarity measure was calculated with the *cosine* formula and it allows to have an idea on the resemblance between the initial query and the various users contexts. In a general way, one can say that the negative context remains generally very different from the initial request (average similarity is equal to 0.046). On the contrary, the positive context is the most similar to the initial request (here his average similarity is equal to 0.466) while positive and negative context has an intermediate similarity measure which is between that of the positive and that of the negative context (average similarity is equal to 0.352).

A qualitative study shows that synonymous terms and morphologically close terms of some terms of the initial query appear in positive or positive and negative profiles (for example: baby and child, storm and rainfall, Israel and Israeli, Jordan and Jordanian, etc.). Moreover, one notices in those profiles an important weight increase of words of the initial request. This will help to make the discrimination between documents.

4 Conclusion

In this article, we present a profile generic model which enables us to describe various types of profiles. We use the property of profiles re-usability in this generic model to define a profiles transformation method which enables us to build various users contexts: positive, negative, positive and negative. A qualitative analysis of those contexts is done compare to an initial user query. It however remains to measure the impact of these users contexts in terms of average recall and precision on research results during several research tasks performs by several users. For that, those different contexts should be used for query reformulation.

References

1. Baeza-Yates R. and Ribeiro-Neto B.: Modern Information Retrieval. First edition, Addison Wesley, ISBN 0-201-39829-X, (1999).
2. Bauer T. and Leake D. B.: Exploiting Information access Patterns for Context-Based Retrieval. Communications of the ACM, (2002), ISBN 1-58113-459-2/02/0001.
3. Bauer T. and Leake D. B.: Real Time User Context modeling for Information Retrieval Agents. Conference on Information and Knowledge Management (CIKM'01), (2001).
4. Boughanem M., Chrisment C. and Soulé-Dupuy C.: Query modification based on relevance backpropagation in adhoc environment. Information Processing & Management Journal, Elsevier Science, (1999) vol. 35, pages 121-139.
5. Chevalier M., Soulé-Dupuy C. and Tchienehom P. L.: A profile-based architecture for a flexible and personalized information access. IADIS International Conference, (2004), vol 2, p. 1017-1022, IADIS - ISBN 972-99353-0-0.
6. Goldberg D., Nichols D., Oki B. M., Terry D.: Using Collaborative Filtering to weave an Information Tapestry. Communications of the ACM, Information Filtering, (1992), vol. 35, n° 12, pages 61-70.
7. Hoashi K., Kazunori M., Naomi I. and Hashimoto K.: Document filtering Method using non-relevant information profile. In proceedings of the 23rd Annual International ACM-SIGIR Conference on research and development in information Retrieval, (2000), pages 176-183.
8. Kim H. R. and Chan P. K.: Learning implicit interest hierarchy for context in personalization. In Proceedings of the 8th international conference on Intelligent User Interfaces, (2003), pages 101-108.
9. Krulwich B.: LifeStyle Finder : Intelligent User Profiling Using Large-Scale Demographic Data. AI Magazine, (1997), vol.18, n° 2, pages 37-45.
10. Montaner M., Lopez B. and Rosa J. L. D. L.: A Taxonomy of Recommender Agents on the Internet. Artificial Intelligence Review, (2003), vol. 19, pages 285-330, Kluwer Academic Publishers.
11. Pazzani M., Muramatsu J. and Billsus D.: Syskill & Webert : Identifying interesting web sites, In Proceedings of the Thirteenth National Conference on Artificial Intelligence, (1996), Pages 54-61.
12. Pazzani M.: A Framework for Collaborative, Content-Based and Demographic Filtering. Artificial Intelligence Review, (1999).
13. Pitkow J., Schütze N., Cass T., Cooley R., Turnbull D., Edmonds A., Adar E. and Breuel T.: Personalized Search : A contextual computing approach may prove a breakthrough in personalized search efficiency. Communications of the ACM, (2002), vol. 45, No. 9, Pages 50-55.
14. Widyantoro D. H., Ioerger T. R. and Yen J.: An Adaptative Algorithm for Learning Changes in User Interests. In Proceedings of the Eighth International Conference on Information and Knowledge Management (CIKM'99), (1999), Pages 405-412, New York, ACM Press.