

# Une architecture à base des profils pour la reformulation contextuelle des requêtes utilisateur

Abdelkrim Bouramoul\*, Khireddine Kholadi \*\*, Bich-Liên Doan\*\*\*

\* Département Informatique, Université 8 mai 1945-Guelma,  
BP 401 Guelma 24000 - Algérie  
a.bouramoul@yahoo.fr

\*\* Département Informatique, Université de Constantine,  
B.P. 325, Constantine 25017 - Algérie  
kholladi@yahoo.fr

\*\*\*SUPÉLEC, Département Informatique  
Plateau de Moulon, 3 rue Joliot-Curie, 91192 Gif sur yvette, France  
bich-lien.doan@supelec.fr

**Résumé.** Cet article s’inscrit dans le domaine de la recherche d’information sur le web et propose une architecture basée sur les profils utilisateurs pour la prise en compte du contexte dans la reformulation des requêtes. Il s’agit de capitaliser l’ensemble d’informations caractérisant chaque utilisateur sous forme d’éléments contextuels afin de les utiliser par la suite lors de la reformulation de la requête. Après un bilan sur les approches classiques pour la reformulation des requêtes, nous présentons la notion du contexte et celle du profil utilisateur. Nous décrivons par la suite l’architecture proposée toute en montrant comment les contextes statique et dynamique sont capturés et la manière selon laquelle ces deux types de contexte sont utilisés dans notre proposition. Nous présentons enfin le prototype développé et nous concluons.

## 1 Introduction

La Recherche d’Information (RI) peut être définie comme une activité dont la finalité est de localiser et délivrer un ensemble de documents à un utilisateur en fonction de son besoin (Hernandez, 2006). Afin d’améliorer la performance de ce type de systèmes, le domaine de la RI contextuelle est apparu récemment comme une priorité (Allan 2003), son objectif est de replacer l’utilisateur au cœur des modèles en rendant explicites certains éléments du contexte qui peuvent influencer les performances des systèmes.

D’autre part, les utilisateurs d’un système de recherche d’information, ne sont pas des professionnels de la documentation (Lin et Wang, 2006), ils ne savent pas choisir les bons termes qui expriment le mieux leurs besoins d’information, une reformulation de requête s’impose alors, cette reformulation est motivée par le fait que la requête initiale retourne rarement un résultat qui satisfait ce dernier. Il s’agit en particulier de modifier la requête initiale de l’utilisateur en lui rajoutant des termes significatifs afin de retourner un résultat plus pertinent.

Dans cet article nous proposons un système de reformulation contextuelle des requêtes utilitaires, cette reformulation est qualifiée de contextuelle car elle prend en compte la notion du contexte via les profils utilisateurs pour modifier leur requête initiale. L’ajout de la notion

Une architecture à base des profils pour la reformulation contextuelle des requêtes utilisateur.

du contexte lors de la reformulation vise à augmenter l'efficacité des SRI en améliorant leur pertinence et en permettant ainsi de prendre en compte les caractéristiques personnelles de l'utilisateur, ses intérêts, ses préférences et l'historique de ses interactions avec le système de recherche. Ces éléments sont capitalisés dans notre système sous forme du contexte statique et dynamique pour être utilisés par la suite dans la reformulation contextuelle.

L'organisation retenue pour cet article s'articule autour de quatre sections. La première sera consacrée à la présentation des différentes approches pour la reformulation de requêtes. Dans la section suivante nous présentons la notion du contexte et celle du profil ainsi que leur apport en recherche d'information. Les deux dernières sections présentent notre système de reformulation contextuelle de requêtes, elles décrivent respectivement l'architecture proposée et le prototype développé. Enfin nous concluons.

## 2 Approches classiques pour la reformulation des requêtes

L'utilisateur est souvent incapable de formuler son besoin exact en information. Par conséquent, parmi les documents qui lui sont retournés, certains l'intéressent moins que d'autres. Compte tenu des volumes croissants des bases d'information, retrouver celles qui sont pertinentes en utilisant seulement la requête initiale de l'utilisateur est une tâche quasi impossible.

La reformulation de la requête consiste donc à modifier la requête de l'utilisateur par ajout de termes significatifs, cette idée d'affinement de requêtes n'est pas nouvelle. Plusieurs approches utilisent différentes techniques pour sélectionner les termes à ajouter à une requête. Nous distinguons trois types d'approches pour la reformulation de requête, la différence entre ces approches réside, soit dans la source des termes utilisés dans la reformulation et qui peuvent provenir des résultats de recherches précédentes (réinjection de pertinence) ou d'une ressource terminologique (réseau sémantique, thesaurus, ontologie), soit dans le mécanisme qui permet de sélectionner les termes à ajouter à la requête initiale (probabiliste, lien sémantique).

Un premier type d'approches repose sur l'analyse globale de la collection de documents considérée, la plus répandue d'entre elles est basée sur des analyses statistiques de corpus de documents (Cui et al, 2002). L'objectif est de relever la fréquence des termes apparaissant conjointement sur un même document et de sélectionner les termes avec le plus grand coefficient. Les informations ainsi extraites sont généralement utilisées pour reformuler automatiquement une requête par ajout des termes liés aux termes initialement présents dans la requête. Les termes ainsi ajoutés sont issus des documents et permettent donc une meilleure adéquation entre le besoin d'information et la collection.

Un deuxième type d'approches basé sur le principe de réinjection de pertinence vise également à reformuler une requête initiale pour qu'elle corresponde mieux au contenu de la collection. Le principe est le suivant : l'utilisateur soumet sa requête initiale, le système restitue un premier ensemble de documents que l'utilisateur doit juger (pertinent, non pertinent). La connaissance de la pertinence des documents initialement restitués est utilisée pour sélectionner des termes à ajouter à la requête initiale. Nous citons dans cette catégorie les travaux de (Lin et Wang, 2006) dans lesquels le système propose, suite à une requête, un ensemble de documents et suivant ceux visualisés par l'utilisateur, le système met à jour son index de termes concordant par des méthodes d'apprentissage automatique.

Le dernier type d'approches figurant dans la littérature utilise des ressources terminologiques telles que des ontologies ou des thésaurus contenant le vocabulaire servant à l'enrichissement des requêtes. Les approches de ce type utilisent des ontologies avec des relations d'équivalence et de subsomption (Navigli et Velardi, 2003) afin d'extraire les termes à rajouter à la requête initiale.

L'approche de reformulation que nous proposons dans cet article est basée sur la prise en compte du contexte utilisateur via son profil, elle offre un double avantage par rapport aux approches présentées précédemment, d'une part, et contrairement aux deux premières classes d'approches elle est utilisable directement sans phase d'analyse ou d'apprentissage, d'une autre part elle n'est pas contrainte du problème présent dans la troisième classe d'approche, ces dernières n'utilisent que les relations d'équivalence et de subsomption et n'exploitent pas toutes les relations sémantiques offertes par une ontologie. Dans la section suivante nous présentons la notion du contexte et celle du profil utilisateur, nous donnons par la suite une classification des profils et de leur utilisation dans les systèmes de recherche d'information.

### 3 Aspects du profil utiles pour la capture du contexte

#### 3.1 Définition du contexte

Le contexte n'est pas un concept nouveau en informatique : dès les années soixante, systèmes d'exploitation, théorie des langages et intelligence artificielle exploitent déjà cette notion. Avec l'émergence des systèmes de recherche d'information, le terme est redécouvert et placé au cœur des débats sans pour autant faire l'objet d'une définition consensuelle claire et définitive (Gaëtan, 2006). Toutefois, l'analyse des définitions présentes dans la littérature conduit à ce double constat :

- « Il n'y a pas de contexte sans contexte » (Brézillon, 2006). Autrement dit, le contexte n'existe pas en tant que tel. Il émerge, ou se définit, pour une finalité ou une utilité précise.
- « Le contexte est un ensemble d'informations. Cet ensemble est structuré, il est partagé, il évolue et sert l'interprétation » (Winograd, 2001). La nature des informations, de même, l'interprétation qui en est faite, dépendent de la finalité.

En recherche d'information, le contexte est défini comme « l'ensemble des facteurs cognitifs et sociaux ainsi que les buts et intentions de l'utilisateur au cours d'une session de recherche », (Calabretto et Egyd-Zsigmond, 2006). D'une manière générale, le contexte regroupe des éléments de natures divers qui délimitent la compréhension, le champ d'application ou les choix possibles. Les éléments les plus couramment invoqués concernent des données spatiotemporelles (lieu, heure, jour.) ou des connaissances spécifiques en relation avec le domaine étudié. Plus rarement nous observons l'utilisation des éléments concernant les émotions, des états d'esprit, des données culturelles (Brézillon, 2006). Ainsi certains éléments du contexte peuvent être difficile à cerner car nous les utilisons inconsciemment, d'autres se trouvent hors d'atteinte des périphériques d'entrée des machines et donc difficile à mettre en œuvre dans des systèmes de recherche d'information.

#### 3.2 Définition du profil

Un profil utilisateur est défini comme « une source de connaissance qui contient des acquisitions sur tous les aspects de l'utilisateur qui peuvent être utiles pour le comportement du

Une architecture à base des profils pour la reformulation contextuelle des requêtes utilisateur.

système » (Wahlster et Kobsa, 1986). Cette proposition, bien que générale, correspond à nos orientations, elle met en évidence trois aspects du profil qui s'exploitent ainsi :

- *Source de connaissance* : le profil utilisateur peut regrouper des informations très diverses selon la tâche à accomplir, en recherche d'information le contenu d'un profil utilisateur se résume en : ses caractéristiques personnelles, ses intérêts et ses préférences, ses compétences, son but courant et enfin l'historique des ses interactions avec le système (Belkin et al, 2004). Nous signalons que la notion du contexte, présenté précédemment, est une extension du profil utilisateur. Le contexte contient des informations complémentaires permettant une meilleure adaptation du profil.
- *Acquisitions* : le contenu du profil utilisateur est une connaissance à récupérer, selon le degré d'adaptation du système, les données du profil utilisateur peuvent être soit, renseignées par l'utilisateur lui-même, soit récupérées par sélection d'un profil pré-existant créé par des experts du domaine, ou encore capturées par le système de recherche d'information au cours de l'utilisation.
- *Utile pour le comportement du système* : en recherche d'information l'apport du profil utilisateur est de permettre une personnalisation ou une adaptation des services pour améliorer les performances du système, ou encore pour filtrer les résultats retournés par un moteur de recherche.

### 3.3 Classification des profils et de leurs utilisations en RI

Nous présentons dans cette section les différents types de profils abordés dans la littérature et qui sont en lien avec la tâche de recherche d'information, à cet effet nous avons défini quatre classes de profils en se basant sur des critères de regroupement, ces critères s'articulent autour du degré de l'implication de l'utilisateur, du moment de l'utilisation du profil, du contenu de profil en information et enfin de la complexité des informations capitalisées par le profil.

#### 3.3.1 Selon l'implication de l'utilisateur

Il s'agit de mesurer le degré de l'implication de l'utilisateur dans le processus de capture de son profil, le travail de (Benammar et al, 2002), distingue deux types de gestion de profils :

- *Indirecte* : c'est le cas où la gestion des profils est transparente à l'utilisateur, autrement dit l'utilisateur n'intervient pas dans la gestion de ses profils.
- *Directe* : à l'opposé, dans la gestion directe des profils, l'utilisateur doit intervenir dans toutes les étapes du processus de recherche pour gérer ses profils.

#### 3.3.2 Selon le moment de la reformulation

Nous nous intéressons ici au moment de l'utilisation du profil dans un système de recherche d'information, deux possibilités sont à retenir :

- *Pré-recherche* : un profil peut être utilisé dans une étape de pré-recherche pour aider l'utilisateur à formuler ou reformuler son besoin. Il peut s'agir par exemple d'affiner l'expression d'une requête proposée par l'utilisateur en fonction de son profil.
- *Post-recherche* : un profil peut également être utilisé dans une étape post-recherche pour filtrer les résultats d'une recherche.

### 3.3.3 Selon la complexité

Dans cette catégorie le focus est mis sur le degré de la complexité des informations présentes dans le profil, différents formats de ce type de profils ont été étudiés dans (Korfhage, 1997), les plus répandus d'entre eux sont :

- *Simple* : un profil simple se présente sous la forme d'un ensemble de mots-clés et éventuellement des poids associés, un poids traduit l'importance de chaque terme dans le profil.
- *Étendu* : un profil étendu inclut, en plus des mots-clés et de leur poids, une série d'informations qui décrivent le contexte de la recherche.

### 3.3.4 Selon la nature d'information

Enfin, la dernière classe distingue les profils utilisateurs en se basant sur la nature d'information qu'ils contiennent, les travaux de (Benammar et al, 2002) exploitent les profils suivants :

- *Profil d'identification* : Cette première composante du profil sert à identifier un utilisateur à travers une série d'informations. Il est défini à la première connexion au système des profils et est mis à jour par incrémentation à chaque création d'un profil d'interrogation.
- *Profil d'interrogation* : Il peut être assimilé à une requête. Il traduit le besoin en information de l'utilisateur et il facilite l'association de la recherche faite par un utilisateur à son contexte.

Les différentes classes de profils que nous avons identifiées peuvent être exploitées simultanément dans un même système et l'utilisation d'un type de profils n'implique pas l'isolement des autres, néanmoins un système à base de profils doit définir les caractéristiques de chaque type de profils utilisé ainsi que les liens les reliant. Dans la suite de cet article nous présentons notre système et nous justifions le choix de ses paramètres en terme du profil, nous décrivons également l'architecture proposée et le prototype développé.

## 4 Une architecture à base des profils pour la reformulation contextuelle des requêtes

### 4.1 Choix des paramètres du système en termes de l'utilisation du profil

Dans les sections précédentes nous avons dressé une étude comparative des éléments nécessaires à la définition de notre système, à savoir, la reformulation, le contexte et l'utilisation des profils dans les systèmes de recherche d'information, cette étude nous a permis de catégoriser séparément les caractéristiques de ces éléments et de cerner les limites de chaque catégorie. Nous présentons dans cette section les différents paramètres caractérisant notre proposition et nous éclairons nos choix par rapport aux approches étudiés précédemment.

Notre choix s'est fixé sur l'utilisation du contexte pour la reformulation de la requête utilisateur. Nous avons présenté dans la section 3 les différentes classes du profil et les particularités relatives à chaque classe. Le tableau 1 présente les paramètres caractérisant notre système en termes du profil utilisateur.

Une architecture à base des profils pour la reformulation contextuelle des requêtes utilisateur.

Selon l'implication de l'utilisateur	Directe	Indirecte
	+	
Selon le moment de la reformulation	Pré-recherche	Post-recherche
	+	
Selon le degré de complexité	Simple	Étendu
		+
Selon la nature d'informations	Profil d'identification	Profil d'interrogation
	+	+

TAB. 1 – *Choix des paramètres du système en terme de l'utilisation du profil.*

Ce tableau présente les choix que nous avons adoptés pour les profils utilisateurs, et donc la manière selon laquelle le contexte est utilisé pour aider à la reformulation contextuelle des requêtes, ces choix s'interprètent comme suit :

- *Implication de l'utilisateur* : l'utilisateur intervient en partie dans la définition de son profil, l'implication est donc directe, le système récupère automatiquement des informations qu'on suppose pertinentes pour enrichir le contexte de l'utilisateur pour des éventuelles prochaines recherches, et les propose à l'utilisateur qui valide par la suite celles qu'il juge réellement pertinentes parmi l'ensemble de propositions.
- *Moment de la reformulation* : dans notre cas il s'agit d'un profil pré-recherche, le système reformule le besoin de l'utilisateur en affinant l'expression de sa requête en fonction de son contexte.
- *Degré de complexité* : le profil est étendu, il inclut, en plus des mots-clés, une série d'information qui décrivent le contexte de la recherche, ces information sont stockées dans une table sous la forme de couples attribut-valeur où chaque couple représente une propriété du profil.
- *Nature d'informations* : nous utilisons à la fois et d'une façon complémentaire, un profil d'identification et un profil d'interrogation. Le premier sert à identifier un utilisateur à travers une série d'informations défini à la première connexion au système, le deuxième est issu de l'historique des recherches faite par le même utilisateur dans des sessions ultérieures, donc son contenu se développe à chaque fois que l'utilisateur formule une nouvelle recherche.

**Synthèse** : pour rendre l'utilisation des profils utile pour la reformulation contextuelle des requêtes et utilisable dans un système de recherche d'information, nous regroupons nos choix en terme de l'utilisation du profil pour modéliser le contexte en deux grandes classes :

1. **Contexte statique** : il prend les caractéristiques d'un profil d'identification étendu qui sera capturé dans une étape de pré-recherche et qui se caractérise par une implication directe de l'utilisateur
2. **Contexte dynamique** : Il constitue l'élément principal de notre système, il regroupe les caractéristiques d'un profil d'interrogation étendu qui est utilisé dans une étape pré-recherche et qui nécessite une implication directe de l'utilisateur

## 4.2 Présentation de l'architecture

Notre système s'articule autour de quatre modules afin de permettre la reformulation contextuelle de la requête utilisateur en se basant sur son profil, il s'agit dans un premier temps de capturer les deux types de contexte nécessaires à la catégorisation de l'utilisateur (contexte statique et contexte dynamique), puis de les utiliser par le module de reformulation pour générer une nouvelle requête à partir de la requête initiale, enfin le module de recherche prend en charge la délivrance d'un résultat qui se rapproche le mieux des besoins de l'utilisateur. Nous décrivons dans les sections suivantes chacun de ces modules en donnant ses différents composants et son principe de fonctionnement. Le regroupement de ces quatre modules nous a permis par la suite de définir notre architecture pour la reformulation des requêtes à base de profils.

### 4.2.1 Module pour la capture du contexte statique

Cette première composante du contexte sert à identifier un utilisateur à travers une série d'informations afin de catégoriser l'utilisateur. Le contexte statique est défini à la première connexion au système, à cet effet nous avons défini quatre catégories d'informations relatives au contexte statique, ces informations se résument en :

- *Les paramètres de connexion* : e-mail, mot de passe.
- *Les caractéristiques personnelles* : nom, prénom, pays, langue ...
- *Les intérêts et préférences* : domaine, domaine secondaire, spécialité...
- *Les compétences et niveau de savoir-faire* : profession, niveau d'étude...

La figure 1 présente les éléments du contexte statique, et la manière selon laquelle les informations composant ce type de contexte sont capturées.

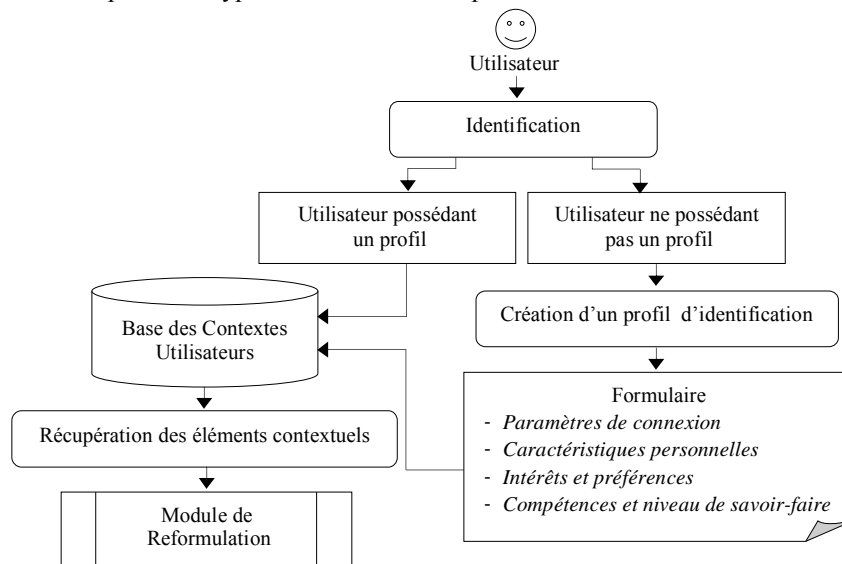


FIG. 1 – Module pour la récupération du contexte statique.

Une architecture à base des profils pour la reformulation contextuelle des requêtes utilisateur.

#### 4.2.2 Module pour la capture du contexte dynamique

Dans le but d'optimiser la réutilisation des profils et faciliter leur compréhension, cette deuxième composante du contexte consiste en l'association de la recherche au contexte de l'utilisateur. A la fin de chaque session de recherche le module de capture du contexte dynamique procède à l'extraction automatique d'un ensemble d'éléments relatifs au contexte de l'utilisateur, il les organise sous forme de compte (attribut, valeur) et les propose à l'utilisateur, ce dernier valide par la suite ceux qu'il juge réellement pertinents. Ces informations seront enfin stockées dans la base des contextes utilisateurs. La figure 2 présente la manière selon laquelle les éléments du contexte dynamique sont capturés.

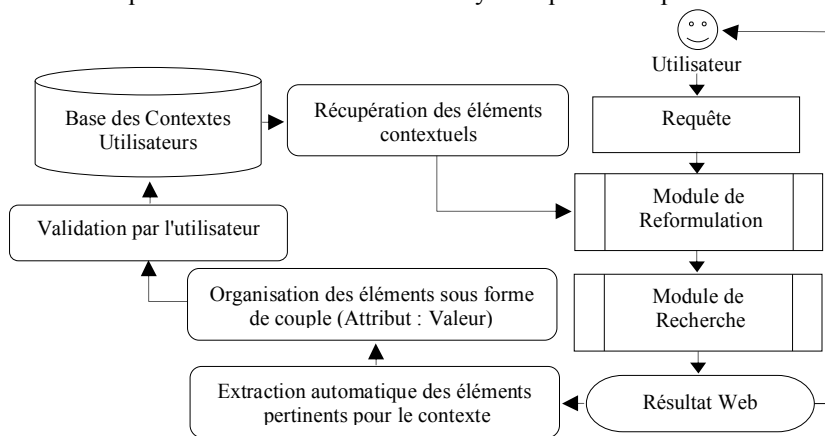


FIG. 2 – Module pour la récupération du contexte dynamique

#### 4.2.3 Module de reformulation

Le module de reformulation a pour objectif de produire une nouvelle requête à partir de la requête initialement formulée par l'utilisateur et cela en rajoutant des termes issus de son contexte de recherche actuelle. Dans un premier temps l'utilisateur formule sa requête en utilisant ses propres termes, par la suite le système procède à l'extraction de l'ensemble des termes à rajouter afin de produire une nouvelle requête, ces termes sont extraits de la base des contextes utilisateurs. Une fois la requête reformulée elle sera envoyée au module de recherche qui prend en charge la délivrance des résultats à l'utilisateur. La figure 3 présente le processus de reformulation de la requête à base du contexte utilisateur.

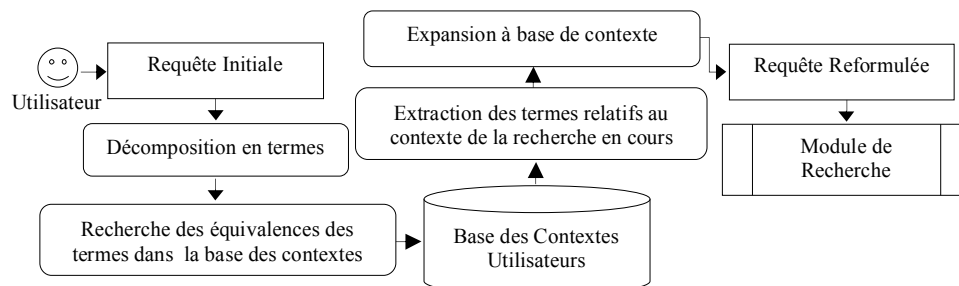


FIG. 3 – Module de reformulation contextuelle des requêtes



#### 4.2.4 Module de recherche

Notre système offre une recherche ouverte sur internet, l'utilisateur exprime son besoin en information sous forme de requête, et le module de reformulation procède par à son expansion en rajoutant des termes issus du contexte utilisateur et renvoie la requête au module de recherche. Ce dernier prend en entrée la requête reformulée et offre à l'utilisateur la possibilité de choisir l'un des trois moteurs de recherche que le système propose (Google, Yahoo, Msn), le résultat obtenu est enfin communiqué à l'utilisateur. La figure 4 montre le principe de fonctionnement du module de recherche.

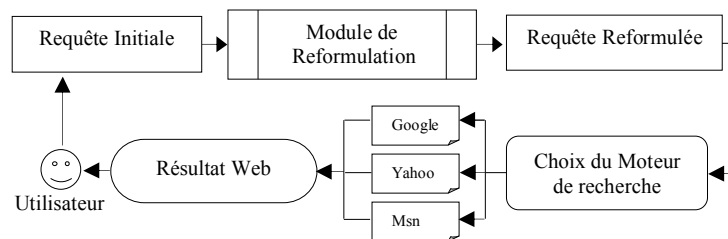


FIG. 4 – *Module de Recherche*

#### 4.2.5 Architecture Générale

La composition des quatre modules décrit précédemment nous a permis de définir l'architecture générale de notre système, nous signalons que le fonctionnement des quatre modules est étroitement lié dans le sens où les sorties de chaque module sont les entrées du module suivant. La figure 5 présente l'architecture générale de notre système pour la reformulation contextuelle des requêtes utilisateur.

Avant de lancer sa requête, l'utilisateur s'identifie dans le système qui procède alors à la récupération de son *Contexte Statique*, il s'agit de ses caractéristiques personnelles pouvant influencer le contexte de recherche. Ces renseignements ont été enregistrés dans la *Base des Contextes Utilisateurs* lors de la première connexion au système. Dans le cas d'un utilisateur qui ne possède pas un profil, le système lui demande de remplir ses préférences et la *Base des Contextes Utilisateurs* sera mise à jour pour une éventuelle utilisation dans des prochaines sessions de recherche.

Une fois le *Contexte Statique* récupéré, l'utilisateur peut alors formuler sa requête et le système procède à la reformulation contextuelle, il se charge de générer la nouvelle requête en sélectionnant les termes relatifs au contexte de la recherche en cours, cette sélection est faite à partir de la *Base des Contextes Utilisateurs*, les deux types de contexte (*Statique* et *Dynamique*), contribuent donc mutuellement à l'opération de reformulation. Par la suite le système lance une recherche ouverte sur internet en utilisant la requête reformulée et en appelant selon le choix de l'utilisateur l'un des trois moteurs de recherche qu'il propose (Google, Yahoo ou Msn). Le résultat de la recherche est retourné enfin à l'utilisateur, il sera stocké également dans l'historique des recherches pour être utilisé par la suite dans la capture du *Contexte Dynamique*.

A la fin de chaque session de recherche et en se basant sur l'historique des recherches faites, le système récupère automatiquement des informations (*Eléments Contextuels*) qu'il suppose pertinentes pour enrichir le *Contexte Dynamique*. Il les propose à l'utilisateur qui valide par la suite celles qu'il juge réellement pertinentes parmi l'ensemble de propositions.

Une architecture à base des profils pour la reformulation contextuelle des requêtes utilisateur.

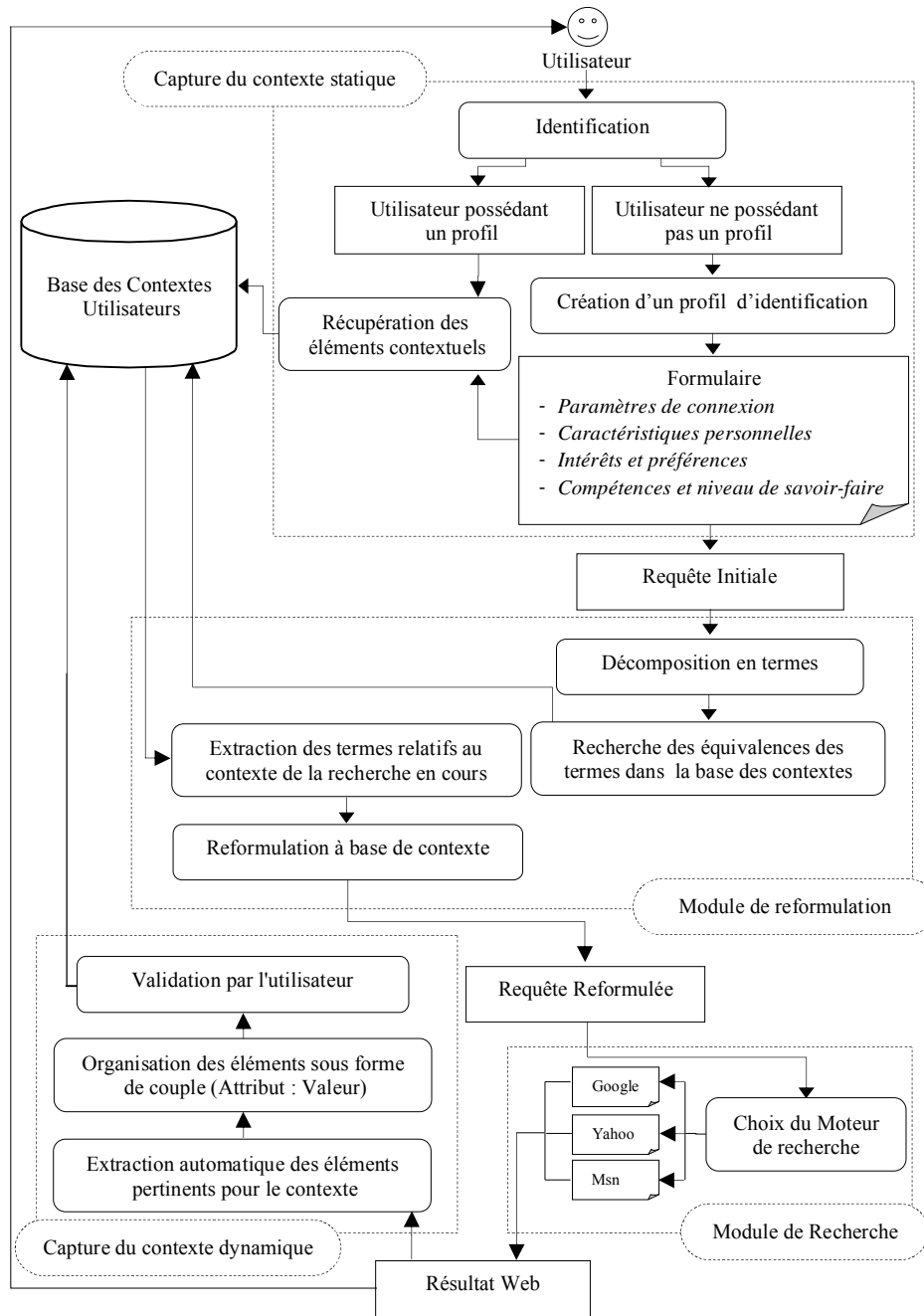


Fig. 5 – Architecture générale d'un système pour la reformulation contextuelle des requêtes utilisateur à base des profils

## 5 Prototype développé

Afin de montrer l'applicabilité de l'architecture proposée, nous avons mis en place un prototype d'un système pour la reformulation contextuelle des requêtes. L'application communique avec une base de données pour stocker les éléments contextuels. Cette base de données contient deux tables, la première sert à garder les préférences de l'utilisateur (contexte statique) ainsi que les éléments contextuels (contexte dynamique), la deuxième table contient les historiques des recherches servant à la capture du contexte dynamique. Les figures 6, 7 et 8 présentent respectivement la fenêtre principale de l'application avec les différentes fonctionnalités offertes par le système, le mécanisme pour la capture du contexte dynamique et enfin la manière selon laquelle la requête initiale est reformulée.

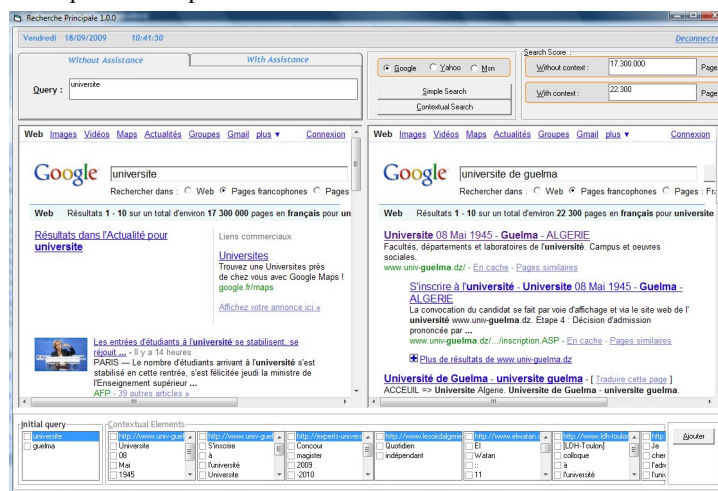


Fig. 6 – Fenêtre principale de l'application.

Cette fenêtre propose à gauche un résultat de recherche en utilisant la requête sans reformulation, et à droite un résultat de recherche de la même requête après reformulation. L'interface offre également la possibilité de choisir le moteur de recherche souhaité et donne le score de pertinence en terme de nombre de page retournée dans les deux cas, recherche avec et sans reformulation. L'analyse du résultat retourné pour la requête 'université de guelma' en utilisant le moteur de recherche 'Google' montre que le nombre de pages retournées sans reformulation était de '5.800.000 pages' tandis que dans le cas de la recherche avec reformulation le nombre de pages est réduit à '143.000 pages', ce dernier score est nettement plus satisfaisant par rapport à l'utilisateur car le résultat retourné ne contient que les pages qui l'intéressent réellement.



Fig. 7 – Mécanisme pour la capture du contexte dynamique

Une architecture à base des profils pour la reformulation contextuelle des requêtes utilisateur.

Pour la récupération du contexte dynamique, le système analyse automatiquement le contenu de la page web résultante. En utilisant les titres de chaque résultat il procède à la segmentation des phrases récupérées puis à l'élimination des mots vident en utilisant un anti-dictionnaire. Le système propose enfin les mots obtenus à l'utilisateur, ce dernier sélectionne ceux qu'il juge réellement pertinents et qui seront donc rajouter à sont contexte dynamique.

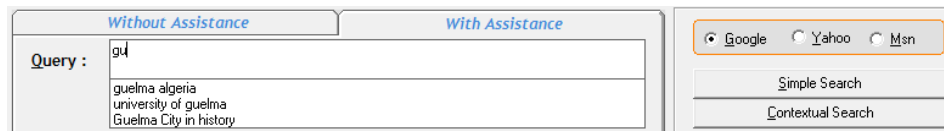


FIG. 8 – Reformulation de la requête initiale

Afin d'assurer la reformulation de la requête initiale, le module de reformulation récupère la saisie de l'utilisateur, il la compare avec le contenu de sa base de contextes et il propose à l'utilisateur des possibilités de reformulation selon le contexte de la recherche en cours. Le système propose également un autre type de reformulation sans intervention de l'utilisateur dans lequel la requête est reformulée après la fin de sa saisie.

## 6 Conclusion

Dans cet article nous avons proposé un système de reformulation des requêtes basé sur le contexte de l'utilisateur. La contribution essentielle de notre travail consiste en la proposition d'une architecture qui s'articule autour de quatre modules afin de permettre la récupération des deux types de contexte (statique et dynamique) permettant la catégorisation de chaque utilisateur, puis leurs utilisations dans la reformulation de la requête initiale. Le prototype que nous avons implémenté démontre l'applicabilité de l'architecture proposée et montre que le résultat obtenu avec une requête reformulée est plus pertinent que celui rendu en utilisant une requête sans reformulation.

Ce travail ouvre la voie vers diverses perspectives qui se situent sur deux plans : un plan d'approfondissement de la recherche réalisée et un plan d'élargissement de son domaine d'application. Pour ce qui est de l'approfondissement, il serait intéressant d'améliorer le prototype développé pour que la capture des éléments contextuels soit faite sans l'intervention de l'utilisateur. En ce qui concerne l'élargissement du domaine d'application, il serait intéressant d'expérimenter l'architecture proposée sur plusieurs moteurs de recherche afin de mesurer l'apport de la reformulation contextuelle sur chacun d'entre eux et focaliser par la suite le prototype sur le moteur de recherche qui s'adapte au mieux à notre proposition.

## Références

- Allan, J. (2003). *Challenges in information retrieval and language modeling*, SIGIR Forum, 37(1), pp 31-47.
- Benammar, A., J. Mothe et G. Hubert (2002). *Automatic profile reformulation using a local document analysis*. European colloquium on IR research, Glasgow. Springer-Verlag , pages 124-134.

- Belkin, N., G. Muresan et Zhang, X. (2004). *Using User's Context for IR Personalization*. Proceedings of the ACM/SIGIR Workshop on Information Retrieval in Context.
- Brézillon, P. (2006). *Expliciter le contexte dans les objets communicants*. C. Kintzig, G. Poulain, G. Privat, P.-N. Favennec (Eds.), Hermès, chapitre 21, 2002, p. 295-303.
- Calabretto, S., E. Egyd-Zsigmond (2006). *Recherche d'Information en contexte*. EARIA'06, France.
- Cui, H., J-R. Wen et Nie, J-Y. *Probabilistic query expansion using query logs*. (2002). 11th international conference on World Wide Web, p. 325-332, Honolulu, Hawaii.
- Gaëtan, R. (2006). *Méthode pour la modélisation du contexte d'interaction*. RSTI - ISI – 11/2006. Adaptation et contexte, pages 141 à 166.
- Hernandez N. (2006). *Ontologie de domaine pour la modélisation du contexte en recherche d'information*, thèse de doctorat en informatique, Université Paul Sabatier.
- Korfhage, R. (1997). *Information storage and retrieval*. Wiley Computer Publishing 0-471-14-338 3.
- Lin, H et L. Wang. (2006). *Query expansion for document retrieval based on fuzzy rules and user relevance feedback techniques*. Expert Systems with Applications, 31(2), 397-405.
- Navigli, R et P. Velardi. (2003). *An analysis of ontology-based query expansion strategies*. Proceeding of the Workshop on Adaptive Text Extraction and Mining, Croatia.
- Wahlster, Wet A. Kobsa, (1986). *Dialogue-based user models*. Proceedings of IEEE, Vol. 74(7), pp. 948-960.
- Winograd, T. (2001). *Architectures for context, Human-Computer Interaction*,, p.402-419.

## Summary

This paper falls under the field of information retrieval on the Web and proposes an architecture based on the user's profiles to take context into account in the queries reformulation. It is to make information of each user in the form of contextual elements to be used later in the process of query reformulation. After an assessment of classic query reformulation approach, we introduce the notion of context and user's profile. Next we describe the proposed architecture and we show how static and dynamic contexts are captured and how these two types of context are used in our proposal. Finally, we present the developed prototype, and we conclude.