Une approche fondée sur la corrélation entre prédicats pour le traitement des réponses pléthoriques

Patrick Bosc*, Allel Hadjali*, Olivier Pivert*, Grégory Smits**

*Irisa ENSSAT - Univ. Rennes 1, Lannion France
{bosc,hadjali,pivert}@enssat.fr
**Irisa IUT Lannion - Univ. Rennes 1, Lannion, France
gregory.smits@univ-rennes1.fr

Résumé. L'interrogation de bases de données, dont les dimensions ne cessent de croître, se heurte fréquemment au problème de la gestion des réponses pléthoriques. Une des approches envisageables pour réduire l'ensemble des résultats retournés et le rendre exploitable est de contraindre la requête initiale par l'ajout de nouvelles conditions. L'approche présentée dans cet article s'appuie sur l'identification de liens de corrélation entre prédicats associés aux attributs de la relation concernée. La requête initiale peut ainsi être intensifiée automatiquement ou par validation de l'utilisateur à travers l'ajout de prédicats proches sémantiquement de ceux spécifiés.

1 Introduction

Depuis le début des années 90, il a été reconnu que l'enrichissement des systèmes d'accès à l'information par des approches coopératives était devenu un besoin pragmatique et indéniable. Gaasterland (1992) souligne que le principal enjeu des approches coopératives d'accès à l'information est de fournir des résultats corrects, fiables et utiles en réponse aux requêtes utilisateurs au lieu de résultats littéraux. Les réponses coopératives ont également pour objectif de fournir des résultats répondant aux besoins et attentes des utilisateurs et ne nécessitant aucun effort et traitement supplémentaires pour être examinés et exploités.

En ce sens, deux problèmes distincts sont fréquemment soulevés. Le premier concerne la gestion des requêtes retournant un ensemble de résultats vide. Lorsque aucun enregistrement ne répond pleinement à la requête soumise par l'utilisateur, l'enjeu des approches coopératives est alors de fournir des réponses alternatives. Le second problème concerne le cas des requêtes fournissant des résultats pléthoriques. Ces requêtes retournent un nombre de réponses trop important pour être exploitées efficacement et facilement par l'utilisateur. Cette situation problématique est d'autant plus actuelle que la présence et l'accès à des bases de données de plus en plus volumineuses se démocratisent et se généralisent, notamment via Internet. Parmi l'ensemble volumineux des résultats retournés, l'utilisateur n'est généralement intéressé que par un sous-ensemble de ces résultats qu'il juge plus pertinent. Pour extraire ce sous-ensemble, l'utilisateur doit parcourir la totalité des résultats retournés, ce qui constitue une tâche fastidieuse et malaisée. Cet article vise à résoudre ce dernier problème en proposant une méthode

permettant d'extraire un tel sous-ensemble des résultats initiaux qui soit pertinent pour l'utilisateur et de taille raisonnable.

Notons que ce problème de réponses pléthoriques a déjà été largement abordé par la communauté des bases de données selon deux axes. Le premier axe, guidé par les données, consiste à considérer l'ensemble initial des résultats retournés comme point de départ et à lui appliquer des procédures de classement pour ensuite retourner les k meilleurs. Cette approche se heurte fréquemment à la difficulté de différencier deux à deux des résultats qui répondent tous pleinement à la requête initiale. Une autre approche de cet axe consiste à résumer l'ensemble des résultats pour en fournir une vue synthétique (voir par exemple Ughetto et al. (2008)).

Le second axe, guidé par la requête, agit sur la requête utilisateur pour la rendre plus restrictive ou fournir des informations *ad hoc* permettant de réduire l'ensemble des résultats retournés ou en privilégier certains. Au sein de cet axe, nous pouvons distinguer différentes approches. La première consiste en une intensification des prédicats spécifiés pour contraindre la requête initiale (par exemple, le prédicat $A \in [a_1, a_2]$ devient $A \in [a_1 + \delta, a_2 - \delta]$). Pour certains types de requêtes cette intensification n'est pas souhaitable dans la mesure où elle peut conduire à une modification importante de la sémantique de la requête initiale. De plus, les requêtes atomiques comportant une contrainte sur une valeur unique ne peuvent être intensifiées.

En s'appuyant à la fois sur une extension du langage d'interrogation des bases de données et sur une modélisation des préférences, une seconde approche consiste à solliciter l'utilisateur pour qu'il spécifie, en complément des contraintes de sélection, des préférences sur d'autres attributs. Ces connaissances peuvent ainsi être exploitées pour privilégier parmi un ensemble de résultats ceux respectant le mieux les préférences spécifiées.

Une troisième approche consiste à compléter la requête initiale par des contraintes supplémentaires afin de la rendre plus restrictive. C'est dans cette dernière approche que s'inscrivent les travaux présentés dans cet article. Le principe de l'approche proposée est basé sur l'identification de liens de corrélation entre prédicats applicables sur les attributs de la relation concernée, afin de contraindre la requête initiale par l'ajout de nouveaux prédicats. Ces prédicats sont choisis car ils partagent un lien sémantique avec ceux spécifiés par l'utilisateur. Cette démarche vise à réduire la cardinalité de l'ensemble des résultats retournés à l'utilisateur tout en respectant le sens des souhaits véhiculés dans la requête.

Le processus d'intensification présenté dans ce document peut être déployé de manière automatique ou semi-automatique en demandant à l'utilisateur de choisir le prédicat à intégrer à sa requête initiale parmi ceux identifiés comme les plus corrélés.

Après une présentation des principaux travaux existants dans la section 2, la section 3 décrit les fondements de notre approche et plus particulièrement l'identification des relations de corrélation entre prédicats. La section 4 montre comment ces relations de corrélation sont exploitées pour contraindre des requêtes atomiques ainsi que des requêtes conjonctives. Un exemple complet et concret illustre cette approche dans la section 5.

2 Travaux existants

Réduire ou organiser un ensemble trop volumineux de résultats pour faciliter son analyse par un utilisateur est une problématique récurrente qui dépasse le cadre de l'interrogation des bases de données. Dans le contexte de la recherche d'information, les documents retournés sont associés à un score représentant des mesures de similarité, de pertinence et de popularité

en accord avec les mots clés spécifiés dans une requête utilisateur. Différentes stratégies ont été envisagées pour classer les documents retournés, allant des modèles vectoriels introduits par Salton et al. (1975) aux modèles de langues statistiques (cf. Croft et Lafferty (2003)), en passant par les systèmes de recherche d'information probabilistes dont un exemple est donné dans Jones et al. (2000). Ces stratégies restent cependant focalisées sur les attributs textuels et sont difficilement transposables aux attributs numériques et catégoriels.

Différents travaux dans le domaine de l'interrogation de bases de données relationnelles ont été proposés pour fournir des méthodes de classement des résultats retournés. Rui et al. (1997) et Wu et al. (2000) exploitent notamment les retours ou exemples donnés par l'utilisateur pour identifier et classer les objets multimédia les plus pertinents par rapport à une requête initiale. Les approches probabilistes en recherche d'information ont également été plus directement reprises par Chaudhuri et al. (2004) pour proposer un modèle de classement prenant en compte les dépendances et corrélations entre attributs spécifiés et non spécifiés dans la requête afin d'associer un score à chaque tuple retourné. Outre le fait que cette dernière approche ne traite que les requêtes comportant des contraintes de type $attribut = valeur$, ces différents travaux nécessitent de disposer d'un historique de requêtes ou de retours d'utilisateurs pour construire leurs modèles. Ces connaissances ne sont pas toujours disponibles, surtout pour les nouvelles bases de données. Dans un contexte plus restrictif de bases de données commerciales, Su et al. (2006) proposent de classer les produits retournés par une requête utilisateur en fonction de leur intérêt commercial (présence obligatoire d'un attribut prix).

Les travaux de Kießling (2002) et Chomicki (2002) visent également à fournir des méthodes permettant de classer un ensemble potentiellement trop volumineux de résultats. Ces méthodes s'appuient sur des préférences spécifiées par l'utilisateur au sein de la requête et donc une extension des langages d'interrogation de bases de données relationnelles. Cette approche exige cependant de l'utilisateur qu'il considère des préférences sur des attributs ne concernant pas directement l'objet de sa requête initiale.

Dans le contexte de l'interrogation flexible de bases de données, Bodenhofer et Küng (2003) enrichissent le langage `VQS` à l'aide d'opérateurs et proposent un classement des tuples retournés par une requête initiale. Dans ce même contexte de requêtes flexibles, Bosc et al. (2008) proposent d'utiliser un opérateur d'érosion sur un des prédicats flous de la requête initiale pour contraindre cette dernière et ainsi réduire l'ensemble des résultats retournés. Pour certains prédicats, cette dernière approche entraîne cependant une modification importante et pas forcément souhaitée par l'utilisateur de la portée et du sens de la requête initiale.

L'étude proposée par Ozawa et Yamada (1994) pour le traitement des requêtes à réponses pléthoriques est celle qui se rapproche le plus de nos travaux. Dans Ozawa et Yamada (1994), les données de la base concernée sont classées dans différents ensembles flous caractérisés par une étiquette linguistique associée à une fonction d'appartenance. Ces ensembles flous constituent une macro-expression des données exprimant leur distribution au sein de la base. En s'appuyant sur ces connaissances établies *a priori*, un degré de dispersion des résultats retournés par la requête initiale est calculé pour chaque attribut. L'approche définie par Ozawa et Yamada (1994) propose ensuite à l'utilisateur d'enrichir sa requête initiale à l'aide d'une nouvelle contrainte portant sur l'attribut sur lequel les résultats de la requête initiale sont le plus dispersés. La limite de cette approche réside dans le fait que l'attribut identifié comme celui proposant une dispersion maximale peut correspondre à une caractéristique non importante pour l'utilisateur et surtout n'ayant aucun lien avec les contraintes spécifiées dans la requête

initiale.

Pour illustrer cette limite, prenons l'exemple d'une base décrivant un ensemble important de véhicules d'occasion et comportant notamment une relation `R_Véhicules` composée des attributs `{désignation, prix, km, année, puissance, type, indiceSecurité, indiceConfort, vitesseMax}`.

Soit la requête Q : *SELECT * FROM R_Véhicules WHERE km < 30000 AND type = 'break' ;*

L'approche envisagée par Ozawa et Yamada (1994) pourrait proposer à l'utilisateur de rajouter une contrainte sur le prix ou la puissance en observant par exemple que les voitures de moins de 30000 km et de type break sont dispersées sur ces attributs.

L'approche présentée dans cet article adopte une démarche différente, en partant notamment de l'hypothèse que les propriétés les plus importantes pour l'utilisateur sont spécifiées dans sa requête et que cette requête doit être complétée par des contraintes respectant le sens de la requête initiale. Notre approche propose d'intégrer de nouveaux prédicats à la requête initiale, mais ces prédicats ne sont pas choisis par rapport à la dispersion des données qu'ils caractérisent, mais principalement pour leur lien sémantique avec un des prédicats spécifiés.

Pour cet exemple, notre approche proposera en priorité à l'utilisateur les voitures récentes et disposant d'indices de sécurité et de confort élevés, qui apparaissent comme des propriétés corrélées sémantiquement à celles spécifiées dans la requête Q.

3 Corrélation entre prédicats

3.1 Présentation du problème

Soit une relation R contenant n tuples $\{t_1, t_2, ..., t_n\}$ définis sur un ensemble Z de m attributs numériques ou catégoriels $\{Z_1, Z_2, ..., Z_m\}$. Soit Q une requête sur R de la forme : *SELECT * FROM R WHERE* $Z_1 = z_1$ *AND* $Z_2 > z_2$ *AND ... AND* Z_k *IN* $(z_{k1}, z_{k2}, ..., z_{ks})$;, ou plus généralement une requête composée d'une conjonction de prédicats d'égalité, inégalité, d'intervalle, d'appartenance, etc. Chaque Z_i correspond à un attribut de R et les z_i à des valeurs de son domaine de définition.

L'ensemble de tuples $\Sigma_Q = \{t_1, t_2, ..., t_p\}$ désigne les résultats de la requête Q sur la base concernée. Le problème de la gestion des résultats pléthoriques apparaît lorsque la cardinalité de l'ensemble des résultats (i.e. $|\Sigma_Q|$) est trop grand [1] pour être facilement exploité par l'utilisateur. Ainsi, pour réduire l'ensemble Σ_Q des résultats retournés, nous proposons d'intégrer de nouveaux prédicats à la requête initiale Q pour former une requête Q' plus sélective et ainsi retourner à l'utilisateur un ensemble $\Sigma_{Q'}$, tq. $\Sigma_{Q'} \subset \Sigma_Q$. Comme suggéré par Chaudhuri et al. (2004), il peut paraître plus judicieux de privilégier les tuples de Σ_Q qui possèdent des propriétés non spécifiées liées sémantiquement aux propriétés spécifiées dans la requête Q. Nous proposons d'identifier ces liens sémantiques en nous appuyant sur la notion de corrélation entre des prédicats prédéfinis et la requête de l'utilisateur.

3.2 Connaissances disponibles *a priori*

Nous supposons disponibles *a priori* des connaissances renseignant sur la distribution des données de la base. Ces connaissances sur la distribution des données sont matérialisées par des

1. Cette limite est évidemment dépendante du contexte applicatif dans lequel le problème est considéré.

partitions du domaine de définition de chacun des attributs de la relation concernée. Un attribut Z_i est donc associé à une partition $P_i = \{P_{i1}, P_{i2}, ..., P_{is_i}\}$. Ces partitions sont construites lors de la définition du schéma de la base par un expert.

Pour les attributs numériques, ces partitions définissent des plages de données qui correspondent à des prédicats d'intervalle et pouvant décrire un regroupement de tuples possédant une propriété commune sur l'attribut concerné. Par exemple, sur l'attribut `année` de la relation `R_Vehicules`, la plage de valeur $[0, 3]$ peut constituer un intervalle décrivant les voitures très récentes. En ce qui concerne les attributs catégoriels, les différentes valeurs constituent individuellement ou par regroupement des éléments de la partition.

Pour rendre ces partitions plus exploitables et surtout décrire la propriété exprimée par chaque regroupement de valeurs, un ensemble de labels linguistiques est associé à chaque partition (Fig. 1). Ainsi, la partition P_i portant sur l'attribut Z_i est associée à l'ensemble de labels $L_i = \{L_{i1}, L_{i2}, ..., L_{is_i}\}$. Ces labels décrivent d'une manière linguistique et naturelle les propriétés possédées par les éléments des partitions (par. ex : $L_{\text{année}} = \{$"très récente", "récente", "d'âge moyen","ancienne", "très ancienne"$\}$).

Pour chaque élément P_{ij} de la partition P_i d'un attribut Z_i , nous dénotons par $bmin(P_{ij})$ et $bmax(P_{ij})$ les bornes inférieure et supérieure de l'intervalle de valeurs qui le caractérise, et par $card(P_{ij})$ le nombre de tuples de la base qui ont une valeur appartenant à l'intervalle concerné. Cette dernière information soulève évidemment le problème de l'adéquation et de la mise-à-jour de cette donnée en fonction de l'évolution de la base de données. Sans toutefois développer cet aspect récurrent dans les approches basées sur une représentation des distributions de données, une solution consisterait à actualiser ces connaissances de manière incrémentale après une ou plusieurs modifications opérées sur la base.

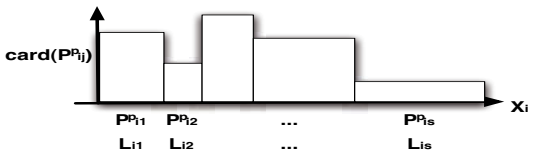

FIG. 1 – *Partition et étiquetage d'un attribut*

Ainsi, chaque élément P_{ij} de la partition P_i d'un attribut Z_i constitue un prédicat d'intervalle décrit par une étiquette L_{ij} et que nous nommons `prédicat prédéfini` (noté P_{ij}^p) par opposition aux `prédicats spécifiés` (notés P_i^s) qui désignent les prédicats présents dans la requête initiale. L'intensification d'une requête repose sur l'intégration de un ou plusieurs prédicats prédéfinis comme contraintes de la forme z_i BETWEEN $bmin(P_{ij})$ AND $bmax(P_{ij})$ si Z_i est un attribut quantitatif et z_i IN $elem(P_{ij})$ [2] si Z_i est un attribut catégoriel.

Ozawa et Yamada (1994) exploitent des connaissances similaires aux partitions présentées précédemment, qui constituent ce qu'ils nomment une macro-expression de la base de données. Cette macro-expression de la base est construite automatiquement en exploitant les résultats d'un processus de classification automatique floue (Fuzzy C-Means) décrit dans Bez-

2. $elem(P_{ij})$ désigne la ou les catégories désignées par P_{ij}.

dek (1984). Les classes floues qui correspondent aux éléments de nos partitions avec des limites graduelles sont ensuite associées à des étiquettes linguistiques définies manuellement et *a priori*, ceci également afin d'améliorer la compréhension des classes construites.

Les partitions du domaine de définition des attributs pourraient également être obtenues en utilisant les méthodes de construction d'histogrammes (Ioannidis (2003)). Cependant, telle que nous l'utilisons, la partition du domaine de définition d'un attribut n'a pas pour objectif principal de capturer une distribution figée des données de la base sur cet attribut, mais bien d'associer un sens ou plutôt une propriété à un intervalle de valeurs par l'intermédiaire d'une étiquette linguistique. C'est pourquoi, nous considérons que ces partitions sont établies lors de la création de la base par un expert et reflètent des regroupements de valeurs cohérents et de « bon sens », apportant une connaissance supplémentaire d'ordre sémantique sur les attributs.

3.3 Corrélation entre prédicats et requête

L'idée sous-jacente de notre approche est de considérer qu'un prédicat prédéfini est corrélé à une requête utilisateur s'il caractérise un ensemble d'éléments similaire à celui retourné par la requête. Par exemple, nous considérons qu'un prédicat désignant les voitures récentes (vétusté inférieure à 3 ans) est corrélé à une requête visant à sélectionner les véhicules de faible kilométrage, dans la mesure où les véhicules récents ont souvent un kilométrage faible. Pour évaluer cette corrélation, nous comparons l'ensemble des résultats retournés par la requête et les tuples décrits par le prédicat prédéfini. Pour quantifier ce lien de corrélation, nous utilisons une mesure définie sur l'intervalle unité $[0, 1]$ exprimant l'égalité graduelle entre deux ensembles. Ainsi, plus le degré d'égalité entre ces deux ensembles de tuples est élevé, plus la requête et le prédicat prédéfini sont corrélés.

Soit une requête Q et Σ_Q l'ensemble des tuples retournés. Soit P_{ij}^p le j^{eme} prédicat prédéfini sur l'attribut Z_i, attribut non spécifié dans Q, et B l'ensemble des tuples caractérisés par ce prédicat. Le degré de corrélation entre P_{ij}^p et Q, noté $cor(P_{ij}^p, Q)$, repose donc sur le calcul du degré d'égalité entre Σ_Q et B :

$$cor(P_{ij}^p, Q) = \frac{card(\Sigma_Q \cap B)}{card(\Sigma_Q \cup B)}$$

Ce degré de corrélation qui correspond à l'indice de Jaccard vérifie les propriétés de réflexivité ($cor(P_{ij}^p, P_{ij}^p) = 1$) et de symétrie ($cor(Q, P_{ij}^p) = cor(P_{ij}^p, Q)$).

Capacité de réduction d'un prédicat

Le choix du prédicat prédéfini pour l'intensification de la requête initiale ne peut reposer uniquement sur la maximisation du degré de corrélation avec les prédicats spécifiés. En effet, cette stratégie pourrait conduire à sélectionner un prédicat prédéfini entièrement ou très fortement corrélé à un prédicat spécifié (degré de corrélation égal ou proche 1) et l'ajout de cette propriété ne réduirait pas suffisamment l'ensemble initial des résultats retournés. Pour pallier ce problème, nous appliquons sur le degré de corrélation une fonction d'appartenance triangulaire sur l'intervalle $[0, 1]$ permettant de favoriser les prédicats offrant un compromis entre corrélation et capacité de réduction. Cette fonction, notée $\mu_{corMod} : [0, 1] \rightarrow [0, 1]$, est définie par un seul point γ caractérisant le degré de corrélation de référence comme l'illustre la

figure 2. Le résultat de l'application de cette fonction sur un degré de corrélation $cor(P_{ij}^p, Q)$ est appelé degré de corrélation modifié noté $corMod(P_{ij}^p, Q)$.

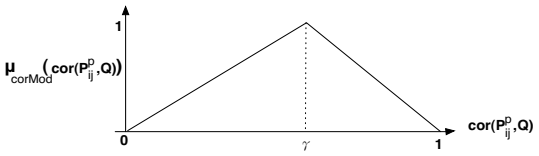

FIG. 2 – *Fonction de calcul du degré de corrélation modifié*

Le point γ permet d'influer sur le comportement du processus d'intensification. En favorisant les degrés de corrélation élevés, les prédicats prédéfinis les plus corrélés seront privilégiés au détriment de leur capacité de réduction. Les degrés les plus faibles permettront une réduction plus rapide de l'ensemble initial des résultats mais entraîneront l'ajout des prédicats les moins corrélés (Algo. 1).

Actuellement, le point γ décrivant cette fonction est un paramètre technique possédant une valeur par défaut ($\gamma = 0.6$). Nous envisageons cependant de développer une méthode permettant d'ajuster automatiquement la valeur de ce paramètre en fonction des spécificités de la requête. L'idée sous-jacente est d'exploiter la clause `TOP N` de la requête, si celle-ci est spécifiée, ou bien de demander à l'utilisateur le nombre N de résultats attendu. À partir de ces informations et d'une estimation du nombre de résultats obtenus suite à l'intégration de chacun des prédicats pré-définis les plus corrélés à la requête initiale, il sera possible d'identifier la valeur γ garantissant le meilleur compromis entre corrélation et capacité de réduction.

4 Intensification des requêtes

4.1 Tables de corrélation entre prédicats

Il serait inefficace de calculer à chaque fois que l'on est confronté à une requête à réponses pléthoriques les degrés de corrélation entre la requête et tous les prédicats prédéfinis sur tous les attributs de la relation. Nous proposons donc d'établir et de maintenir une table de corrélation dans laquelle sont stockées ces connaissances. Cette table contient autant de lignes et de colonnes qu'il y a de prédicats prédéfinis et renseigne pour chaque couple de prédicats prédéfinis $(P_{ij}^p, P_{kl}^p), k \neq i$ le degré de corrélation $cor(P_{ij}^p, Q)$ entre le prédicat P_{ij}^p et une requête $Q = P_{kl}^p$. À partir de ces informations, nous pouvons stocker pour chaque prédicat prédéfini P_{ij}^p, la liste des autres prédicats prédéfinis classés par ordre décroissant de leur degré de corrélation vis-à-vis d'une requête $Q = P_{kl}^p$. Nous estimons qu'il n'est pas nécessaire de conserver la liste ordonnée de tous les prédicats les plus corrélés à chaque prédicat prédéfini dans la mesure où l'intégration de plus de 3 ou 4 prédicats supplémentaires conduirait à une modification trop importante de la visée de la requête. Nous ne conservons et n'exploitons ainsi que les cinq prédicats les plus corrélés.

Afin d'exploiter les connaissances stockées dans cette table, il est nécessaire d'identifier pour chaque prédicat spécifié son prédicat prédéfini le plus proche dans le domaine de définition de l'attribut concerné. Pour les attributs numériques, nous utilisons une mesure de distance entre deux prédicats P_i^s et P_{ij}^p inspirée de la distance de Hausdorff notée $dist(P_i^s, P_{ij}^p)$:

$$dist(P_i^s, P_{ij}^p) = max(|bmin(P_i^s) - bmin(P_{ij}^p)|, |bmax(P_i^s) - bmax(P_{ij}^p)|).$$

Dans le cas où les prédicats P_i^s et P_{ij}^p portent sur des attributs catégoriels, cette mesure est remplacée par le nombre de catégories désignées par P_i^s ou P_{ij}^p moins celles en commun entre P_i^s et P_{ij}^p (i.e. $|elem(P_i^s) \cup elem(P_{ij}^p)| - |elem(P_i^s) \cap elem(P_{ij}^p)|$). Le prédicat le plus proche $P_i'^s$ d'un prédicat P_i^s défini sur l'attribut Z_i, attribut associé à la partition $P_i = \{P_{i1}^p, P_{i2}^p, ..., P_{ih}^p\}$, est celui qui minimise cette distance :

$$P_i'^s = P_{ij}^p \text{ tel que } dist(P_{ij}^p, P_i^s) = inf_{j=1,h} dist(P_{ij}^p, P_i^s)$$

4.2 Processus d'intensification

4.2.1 Requêtes atomiques

L'intensification d'une requête atomique $Q = P_i^s$ peut être automatisée en appliquant l'algorithme 1.

Algorithme 1 Intensification d'une requête $Q = P_i^s$

Entrée : $Q = P_i^s$ //requête initiale
γ //degré de corrélation de référence
$\Sigma_Q = resultats(Q)$

/*Recherche du prédicat prédéfini $P_i'^s$ le plus proche de P_i^s*/
$P_i'^s = P_{ij}^p$ tel que $dist(P_{ij}^p, P_i^s) = inf_{j=1,h} dist(P_{ij}^p, P_i^s)$
/*Récupération des cinq prédicats prédéfinis les plus corrélés à $P_i'^s$*/
$C = correles(5, P_i'^s)$
/*Récupération des cinq prédicats possédant les plus forts degrés de corrélation modifiés avec $P_i'^s$ */
$I = modificationDegres(\gamma, C)$

$k = 1$
tantque est_plethorique(Σ_Q) **faire**
  /*intégration du k^{eme} prédicat prédéfini maximisant le degré de corrélation modifié avec Q ou du prédicat choisi par l'utilisateur si l'intensification n'est pas automatique*/
  $Q = Q \wedge I[k]$
  $\Sigma_Q = execute(Q)$
  $k = k + 1$
fin tantque
Sortie : Σ_Q

En exploitant la table de corrélation, l'intensification d'une requête atomique devient un processus efficace reposant sur l'application directe de l'algorithme 1. L'identification du prédicat prédéfini $P_i'^s$ le plus proche du prédicat spécifié P_i^s dans la requête Q nécessite cependant une comparaison avec l'ensemble des prédicats prédéfinis sur l'attribut Z_i. La récupération des

prédicats prédéfinis les plus corrélés au prédicat prédéfini le plus proche du prédicat spécifié est directe étant donné que ces informations sont accessibles dans la table de corrélation. De même, le degré de corrélation modifié est calculé uniquement sur les cinq prédicats prédéfinis les plus corrélés. L'exécution de la requête intensifiée est évidemment effectuée sur l'ensemble des résultats de la requête initiale et non sur la totalité de la base si tous les attributs nécessaires font partie de la projection initiale.

4.2.2 Requêtes conjonctives

Dans le cas d'une requête conjonctive $Q = P_1^s \wedge P_2^s \wedge \ ... \ \wedge P_r^s$ conduisant à des résultats pléthoriques, la méthode de sélection des prédicats prédéfinis à intégrer diffère quelque peu.

Pour chaque prédicat spécifié $P_i^s, i = 1..r$, nous conservons toujours les 5 prédicats prédéfinis maximisant le degré de corrélation vis-à-vis de chaque requête $Q' = P_i'^s, i = 1..r$, où nous rappelons que $P_i'^s$ est le prédicat prédéfini le plus proche de P_i^s. Après calcul du degré de corrélation modifié, nous reclassons ces prédicats par ordre décroissant de leur degré de corrélation modifié moyen.

Soit P_{ij}^p un des prédicats prédéfinis conservés. Le degré de corrélation modifié de ce prédicat par rapport à la requête conjonctive Q et à la fonction de modification du degré de corrélation μ_{corMod} est recalculé de la façon suivante :

$$corMod(P_{ij}^p, Q) = \frac{1}{r}.\sum_{k=1}^{r} \mu_{corMod}(cor(P_{ij}^p, P_k'^s))$$

En tenant compte de ce nouveau classement partiel, l'algorithme 1 est ensuite appliqué pour l'intensification des requêtes conjonctives.

5 Exemple

Pour illustrer cette approche, reprenons l'exemple de la relation de test `R_Vehicules` introduit en section 2 et la requête :

Q = *SELECT * FROM R_Vehicules WHERE km BETWEEN 10000 AND 30000 AND type = 'break' ;*

Pour illustrer notre approche, nous considérons l'ensemble des résultats Σ_Q pléthorique avec notamment $|\Sigma_Q| = 108$.

La figure 3 illustre des partitions possibles des attributs `année` et `type` concernés par la requête Q. On remarque que les prédicats prédéfinis les plus proches des prédicats spécifiés P_1^s = km BETWEEN 10000 AND 30000et P_2^s = type = 'break' sont respectivement P_{km1}^p (L_{km1} = 'très faible') et P_{type5}^p ((L_{type5} = familial')), d'où $P_1'^s = P_{km1}^p$ et $P_2'^s = P_{type5}^p$.

La table 5 propose des extraits de la table de corrélation et de réduction disponible, permettant notamment d'appréhender la distribution des tuples appartenant à Σ_Q.

Pour cet exemple, nous considérons que $|P_1'^s| = 234$ et $|P_2'^s| = 172$. De même, la cardinalité de chaque partition est spécifiée après chaque label linguistique entre parenthèses. Pour chaque couple de prédicats prédéfinis $(P_k'^s, P_{ij}^p)$, nous donnons leur degré de corrélation $cor(P_k'^s, Q = P_{ij}^p)$ et pour information nous avons ajouté entre parenthèses le cardinal de l'intersection des ensembles de tuples retournés par ces deux prédicats $P_k'^s \wedge P_{ij}^p$.

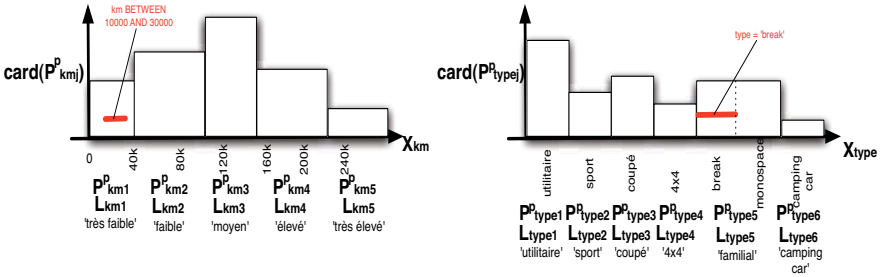

FIG. 3 – *Partitions des attributs* `km` *et* `type`

	Attribut puissance		
$L_{puissancei}$	'faible'(275)	moyenne'(502)	'élevée'(435)
$P_1'^s$	0,067 (32)	0,059 (41)	0,055 (35)
$P_2'^s$	0,067 (28)	0,039 (25)	0,1 (55)

	Attribut prix			
L_{prixi}	'bas'(161)	'moyen'(342)	'élevé'(417)	'très élevé'(262)
$P_1'^s$	0,008 (3)	0,016 (9)	0,085 (51)	0,102 (46)
$P_2'^s$	0,034 (11)	0,078 (37)	0,069 (38)	0,053 (22)

	Attribut année				
L_{anneei}	'très récent'(90)	'récent'(302)	'moyen'(339)	'ancien'(310)	'très ancien'(171)
$P_1'^s$	0,286 (72)	0,045 (23)	0,016 (9)	0,006 (3)	0,02 (1)
$P_2'^s$	0,087 (21)	0,049 (22)	0,045 (22)	0,057 (26)	0,052 (17)

	Attribut indiceConfort			
$L_{indiceConforti}$	'faible'(349)	'moyen'(435)	'élevé'(368)	'très élevé'(60)
$P_1'^s$	0,021 (12)	0,032 (21)	0,058 (33)	0,167 (42)
$P_2'^s$	0,012 (6)	0,032 (19)	0,084 (42)	0,215 (41)

	Attribut indiceSecurite			
$L_{indiceSecuritei}$	'bas'(107)	'moyen'(442)	'élevé'(399)	'très élevé'(264)
$P_1'^s$	0,018 (6)	0,034 (22)	0,082 (48)	0,117 (52)
$P_2'^s$	0,004 (1)	0,022 (13)	0,071 (38)	0,147 (56)

	Attribut vitesseMax		
$L_{vitesseMaxi}$	'faible'(276)	'moyen'(542)	élevée'(394)
$P_1'^s$	0,065 (31)	0,057 (42)	0,059 (35)
$P_2'^s$	0,067 (28)	0,08 (53)	0,05 (27)

TAB. 1 – *Extraits de la table de connaissances pour* $P_1'^s$, $P_2'^s$

Sur cet exemple, nous utilisons une fonction de modification du degré de corrélation définie par $\gamma = 0,6$. Ainsi, on constate que les cinq prédicats maximisant le degré de corrélation modifié avec les prédicats $P_1'^s$ et $P_2'^s$ sont respectivement : {année 'très récent'/0,477; indiceConfort 'très élevé'/0,278; indiceSécurité 'très élevé'/0,195; prix 'très élevé'/0,17; prix 'élevé'/0,142} et {indiceConfort 'très élevé'/0,358; indiceSécurité 'très élevé'/0,245; puissance 'élevée'/0,167; année 'très récent'/0,145; indiceConfort 'élevé'/0,14}. Ces prédicats prédéfinis apparaissent bien comme porteurs de propriétés sémantiquement proches de celles spécifiées dans la requête initiale.

Nous fusionnons ensuite ces classements pour établir un classement final des prédicats selon leur degré de corrélation modifié moyen. On obtient alors : {indiceConfort 'très élevé'/0,318; année 'très récent'/0,311; indiceSécurité 'très élevé'/0,22; prix 'très élevé'/0,085; puissance 'élevée'/0,083; prix 'élevé'/0,071; indiceConfort 'élevé'/0,07}.

Ce classement final est soit exploité pour intensifier automatiquement la requête initiale par application de l'algorithme 1, soit suggéré à l'utilisateur pour qu'il choisisse le (ou les) prédicat(s) prédéfini(s) à intégrer à sa requête initiale. Si dans notre exemple, l'utilisateur choisit le prédicat (année 'très récent') pour intensifier sa requête initiale Q, la requête *Q' = SELECT * FROM R_Vehicule WHERE km BETWEEN 10000 AND 30000 AND type = 'break' AND année BETWEEN bmin(P^p_{annee4}) AND bmax(P^p_{annee4});* [3] sera construite et exécutée pour générer un ensemble de résultats $\Sigma_{Q'}$, tq. $\Sigma_{Q'} \subset \Sigma_Q$. $\Sigma_{Q'}$ est retourné à l'utilisateur ainsi, le cas échéant, que le classement calculé précédemment des prédicats prédéfinis les plus corrélés à Q afin de procéder si nécessaire à une nouvelle intensification.

6 Conclusion et perspectives

L'approche proposée pour le traitement des requêtes à réponses pléthoriques est basée sur le principe d'intensification de requêtes au moyen de prédicats prédéfinis. Le choix et l'ordre des prédicats à intégrer sont déterminés en utilisant une mesure de corrélation entre les prédicats prédéfinis associés aux attributs de la relation concernée. Un des avantages majeurs de l'approche est qu'elle ne conduit pas à une dégradation importante de la requête initiale. Ceci est clairement illustré dans l'exemple considéré. La complexité de l'approche est maintenue acceptable grâce à la table de connaissances qui renseigne sur les corrélations entre les prédicats prédéfinis. Ces connaissances sont établies *a priori* puis maintenues à jour.

Après une validation du fondement de cette approche, nous envisageons de mettre en place une évaluation qualitative de l'intérêt de ce processus d'intensification par retour d'utilisateurs dans un contexte applicatif donné (interrogation de la base de données cinématographique IMDB http://www.imdb.com). En plus de présenter à l'utilisateur les prédicats les plus corrélés à sa requête initiale, nous envisageons également d'exploiter les connaissances disponibles à travers les partitions des domaines de valeurs des attributs pour approximer le nombre de tuples qui lui seront retournés suite à l'intégration de chacun de ces prédicats.

3. Rappel : P^p_{annee4} correspond au quatrième élément de la partition définie sur l'attribut année.

Références

Bezdek, J. (1984). Fcm : The fuzzy c-means clustering algorithm. *Computers and Geosciences*.

Bodenhofer, U. et J. Küng (2003). Fuzzy orderings in flexible query answering systems. *Soft Computing 8*, 512–522.

Bosc, P., A. Hadjali, et O. Pivert (2008). Empty versus overabundant answers to flexible relational queries. *Fuzzy sets and systems 159*((12)), 1450–1467.

Chaudhuri, S., V. H. G. Das, et G. Weikum (2004). Probabilistic ranking of databases query. In *Proc. of Int. Conf on Very Large Databases*, pp. 888–899.

Chomicki, J. (2002). Querying with intrinsic preferences. *EDBT*, 34–52.

Croft, W. et J. Lafferty (2003). *Language Modeling for Information Retrieval*. Kluwer.

Gaasterland, T. (1992). Relaxation as a platform for cooperative answering. *Journal of Intelligent Information Systems 1(3-4)*, 296–321.

Ioannidis, Y. (2003). The history of histograms (abridged). In *Proc. of the 29th Int. Conf. on Very Large DataBases*.

Jones, K. S., S. Walker, et S. Robertson (2000). A probabilistic model of information retrieval : development and comparative experiments. *Inf. Process Management* (Part 1), 809–840.

Kießling, W. (2002). Foundations of preferences in database systems. In *Proc. of the 28th Int. Conf. on Very Large DataBases*, pp. 311–322.

Ozawa, J. et K. Yamada (1994). Cooperative answering with macro expression of a database. In *Proc. of the IPMU conf.*, pp. 17–22.

Rui, Y., T. Huang, et S. Merhotra (1997). Content-based image retrieval with relevance feedback in mars. In *Proc. IEEE Int. Conf. on Image Processing*, pp. 815–818.

Salton, G., A. Wong, et C. Yang (1975). A vector space model for automatic indexing. *Communications of ACM : Information Retrieval and Language Processing 8*(11).

Su, W., J. Wang, Q. Huang, et F. Lochovsky (2006). Query result ranking over e-commerce databases. In *Proc. of the CIKM*.

Ughetto, L., W. Voglozin, et N. Mouaddib (2008). Database querying with personalized vocabulary using data summaries. *Fuzzy Sets and Systems 159*, 2030–2046.

Wu, L., C. Faloutsos, K. Sycara, et T. Payne (2000). Falcon : Feedback adaptive loop for content-based retrieval. In *Proc. of the 26th Int. Conf. on Very Large DataBases*, pp. 297–306.

Summary

Retrieving desired data from large-scale databases often leads to an overabundant answers problem. A possible approach to reduce the set of retrieved items and to make it more manageable is to constrain the initial query with additional predicates. The approach presented in this paper relies on the identification of correlation links between predicates related to attributes of the relation of interest. Thus, the inital query is intensified by additional predicates that are semantically close with the user-specified ones.