Modèle d'interclassement de résultats basé sur les profils des utilisateurs dans un SRI-P2P

Rim Mghirbi*,** — Khedija Arour* — Yahya Slimani* — Bruno Defude**

* Département des Sciences de l'informatique
Faculté des sciences de tunis
Tunis, Tunisie
rim.mghirbi@laposte.net
Khedija.arour@issatm.rnu.tn
yahya.slimani@fst.rnu.tn

** Département d'informatique
Institut de Télécom et Management Sud Paris
Paris, France
Bruno.Defude@it-sudparis.eu

RÉSUMÉ. La recherche d'information (RI) s'est basée pendant de longues années sur le modèle client-serveur. Cependant, l'avènement des systèmes Pair-à-Pair (P2P) et leur vive exploitation dans le partage des fichiers multimédia, a motivé la communauté de la RI à exploiter de telles architectures. Nous parlons de recherche d'information P2P (RI-P2P). Interclasser des résultats des différents pairs est un volet important de la RI-P2P vu l'hétérogénéité de ces résultats. Cette principale raison, entre autres, rend difficile d'appliquer des approches classiques pour l'interclassement. Pour cela, il nous a semblé intéressant d'explorer un mécanisme d'interclassement basé sur les profils des utilisateurs deduits de leurs comportements lorsqu'ils interagissent avec le résultat d'une requête.

ABSTRACT. The information retrieval (IR) has for many years based on the client-server model However, the advent of systems Peer-to-Peer (P2P) and their strong operation in sharing media files, has motivated the IR community to exploit such architectures for search: that's what we call P2P information retrieval (P2PIR). Collating the results of different peers using classical approaches is a difficult task of IR-P2P given the huge heterogeneity of these results in P2P systems. For this, we seemed interesting to explore a mechanism for merging based on user profiles deducted from their behavior when interacting with the result of a query.

MOTS-CLÉS : Recherche d'information, Systèmes distribués, Systèmes P2P, Interclassement de résultats, Profils utilisateurs.

KEYWORDS : Information retrieval, Distributed systems, P2P systems, Rank Aggregation, User profiles.

1. Introduction : Les contraintes d'interclassement dans les systèmes de Recherche d'Information Pair-à-Pair (SRI-P2P2)

La recherche d'information (RI) s'est basée pendant de longues années sur le modèle client-serveur. Cependant, l'avènement des systèmes Pair-à-Pair (P2P) et leur vive exploitation dans le partage des fichiers multimédia, a motivé la communauté de la RI à exploiter de telles architectures. Nous parlons de recherche d'information P2P (RI-P2P). Dans un tel processus,Interclasser les résultats provenant de systèmes hétérogènes constitue un vrai problème. En effet, ces différents systèmes peuvent être hétérogènes de point de vue schémas d'indexation, types et tailles de collections utilisées, modèles de recherche[5]. Cette hétérogénéité devient plus contrariante si les SRIs n'acceptent pas de déléguer des informations sur leurs ressources ; c'est ce qu'on appelle autonomie des SRIs.
Notre défi consiste alors à essayer de trouver un consensus de classement entre les différentes listes retournées de façon à créer une liste unique classée. Ce défi devient de plus en plus pesant si on cible les systèmes P2P qui présentent, dans un contexte purement distribué, les containtes suivantes :

– Aucun pair n'a une vision globale du réseau.

– L'hétérogénéité des ressources s'impose de plus en plus avec le fateur d'échelle.

– le problème d'autonomie des pairs devient plus insistant et même si ces derniers acceptent de partager une partie de leurs informations, l'échange et le maintien de statisques globales devient quasilent impossible.

Les modèles classiques d'interclassement (à base de score et à base de rangs[3]) ont de faibles chances pour réussir cette tâche dans un SRI-P2P. En effet, une agrégation simple de scores ou de rangs suppose l'homogénéité des SRIs. Elle s'avère peu efficace vu qu'elle se base simplement sur des efforts de calibrage de rangs et de scores pour interclasser. L'echange des statistiques afin de recalculer un score global constitue à son tour un vrai tabou dans les systèmes à large échelle vu le côut d'échange. Partant de ces contraintes d'interclassement dans les SRI-P2P et des limites des approches classiques, il nous a semblé intéressant d'explorer un mécanisme basé sur les profils des utilisateurs pour interlasser nos résultats. Le reste du papier sera organisé comme suit : la section 2 définit la solution d'interclassement que nous proposons. Une étude empirique de ce modèle sera présentée et discutée dans la section 3. Nous terminerons ce papier par une conclusion et quelques perspectives de travaux futurs.

2. Solution Proposée : Interclassement à base de profils

2.1. Définition de profils

Deux types de profils sont définis afin de résoudre notre problème d'interclassement.

Ils peuvent être vus comme étant des correlations sémantiques entre :

– Les pairs sollicités pour la reponse à un ensemble de requêtes et les termes de ces requêtes : Les profils Pairs-termes PPT.

– Les documents téléchargés, ou consultés lors de la réponse à un ensemble de requêtes et les termes de ces dernières : Les profils Douments-termes PDT

2.1.1. Representation formelle d'un profil

Un profil utilisateur P est représenté dans notre cas comme étant un couple un couple de deux ensembles dits $dom(P)$ et $codom(P)$ où $dom(P)$ regroupe des objets partageant les mêmes propriétés contenues dans le $codom(P)$. Nous utilisons ici, les deux profils $PPT = (P_i, T_j)$ et $PDT = (D_i, T_k)$ tels que, respectivement, $P_i = dom(PPT)$ représente l'ensemble de tous les pairs sollicités dans la réponse aux requêtes de termes T_j. Ces dernières représentent le codomaine de PPT $(codom(PPT))$ et $D_i = dom(PDT)$ représente l'ensemble de tous les documents sollicités dans la réponse aux requêtes de termes T_k qui représentent également le codomaine de PDT $(codom(PPT))$.

2.1.2. La génération de profils

La génération de profils repose sur une technique d'Analyse formelle de concepts basée à son tour sur une analyse du fichier journal capturé au niveau de chaque pair. Selon les besoins, une projection sur certaines informations du fichier journal permet de créer un contexte formel reliant un ensemble d'objets O avec un ensemble de propriétés P et exprimant que chaque objet o de O est en relation avec une propriéte p de P. Suite à cette génération de profils, une étape de sauvegarge de ces derniers dans deux bases de données sera entamée. Une base Base Pairs-Termes (BPT) sera utilisée pour les profils de type PPT, et une deuxième, Base Documents-Termes (BDT), pour les profils de type de PDT. Les étapes détaillées de la génération des profils à partir des fichiers journeaux sont illustrées à travers l'exemple de la figure 1. Un profil est généré hors ligne et de façon décentralisée sur chaque pair. ceci a pour objectif de ne pas contrarier l'utilisateur d'attendre la préparation de profils lors de la soumission de sa requête et permet à notre approche depasser à l'échelle puisque les profils sont constitués localement au niveau de chaque pair.

2.2. Principe du modèle d'interclassement à base de profils

2.2.1. Score d'interclassement à base de profils

Pour notre contexte d'interclassement, estimer un score de similarité requête-document est indispensable pour avoir un référenciel de classement commun pour tous les documents. Pour cela notre score intègre quatre fateurs clés dont deux reposent sur les profils. Nous parlons respectivement de :

a. L'Importance passée du pair P_j $(PPI(P_j)$: ce paramètre représente l'importance du pair par rapport aux requêtes passées. Il est fourni par la procédure SimilartyPeerQuery

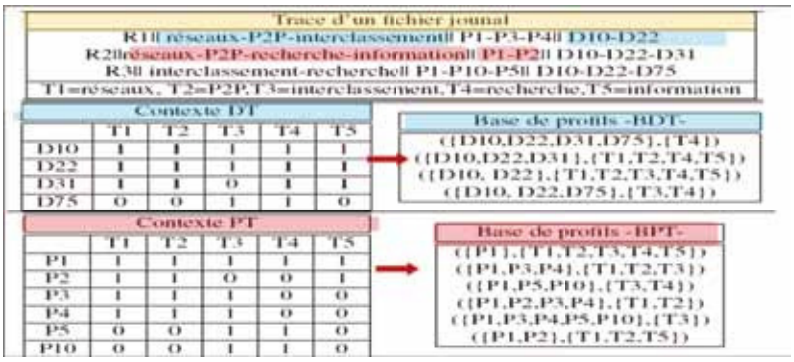


Figure 1. *Exemple de génération de profils*

de l'algorithme 1 Une illustratiobn de calcul de ce paramètre est fournie dans l'étape2 dans la figure 2).

b. L'Importance passée d'un document d_i $(DPI(d_i))$: il définit le poids de présence d'un document dans l'historique des requêtes similaires stockées. (voir l'étape 3 de l'exemple de la figure 2).

c. La Valeur positionnelle du document d_i $(PV(d_i))$: elle est déduite du rang du document dans sa liste locale (voir étape 1 de la figure 2).

d. L'Importance actuelle d'un document (Pd) : représente le taux de présence d'un document dans tous les résultats des différents pairs. Par ce principe la popularité du document $d10$ (figrant dans l'exemple 2).

Etant donné l'ensemble des paramètres décrits, une première formule de score a été définie :

$$Sg(d) = \frac{\frac{PPI}{NbTotalPeers} * \sum_{i=1}^{nbrPairs-d} SP(DPI + PV(d))}{NbTotalPeers} \quad [1]$$

avec :

- $NbTotalPeers$ est le nombre total des pairs répondant à la requête, et

- $nbrPairs - d$ est le nombre total des pairs répondant à la requête et contenant le document d ;

2.2.2. Algorithme d'interlassement

Dans cet algoritme les méthodes $similarityPeerQuery(BPT, Q)$ et $similarityDocQuery(BDT, Q)$ permettent de calculer les poids des poids et documents contenus dans les profils PPI et DPI en se basant sur la notion de distance sémantique proposé par salton [4]. Ainsi pour calculer le poids d'un pair P par rapport à une requête Q donnée, la formule suivante est utilisée :

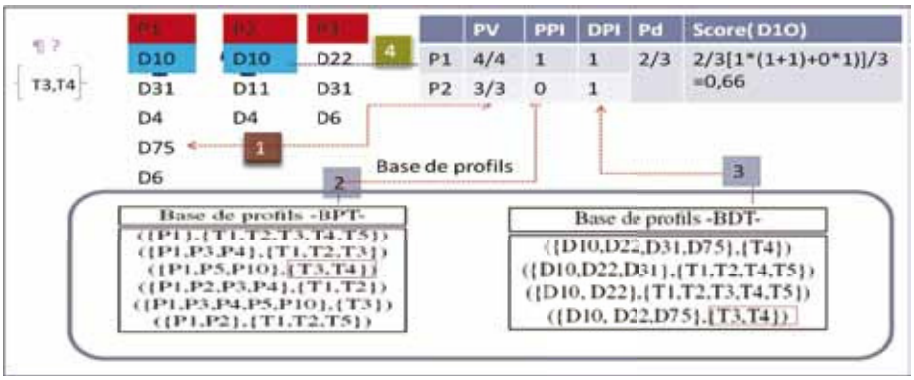


Figure 2. *calcul de score à base de profils*

Algorithme : INTERCLASSEMENT $(B, Q, Lrp,)$
Entrées :
B : Base de connaissances (BPT, BDT).
Q : Requête soumisell Lrp : Liste de résultats par différents pairs.
p_i, d_j : un pair p_i ; un document d_j retourné par p_i.
Sortie :
LF : Liste Finale classée
Traitement
    $similarityPeerQuery(BPT, Q)$;
    $similarityDocQuery(BDT, Q)$;
    tant que $Lrp \neq \emptyset$ **faire**
        $SCP := selecteConcepts(BPT, Lrp)$;
        tant que $Lrp.p_i \neq \emptyset$ **faire**
            $SCD := selecteConcepts(BDT, Lrp.pi)$;
            $card := cardinalityDocPeers(dj, Lrp)$;
        $setGlobalScore(Lrp.pi.dj)$;
    $LF := mergeSort(Lrp)$;
fin

Algorithme 1 : ALGORITHME D'INTERCLASSEMENT DE RÉSULTATS

$$Sim(Q,P) \quad = \quad \frac{|T(Q) \bigcap codom(PPT)|}{|T(Q) \bigcup codom(PPT)|} \qquad [2]$$

De même pour la similarité entre le document et les termes de la requête.

3. Etude Empirique

3.1. Environnement d'expérimentation

Pour tester l'approche proposée dans ce papier, nous avons utilisé un simulateur d'interclassement présentant des fichiers XML de définition de documents et de requêtes contenant leurs identifiants et leurs mots clés respectifs. Pour les requêtes, ces fichiers présentent aussi les réponses de la collection centralisée avec les similarités. Nous avons utilisé également un module de distribution et de réplication de documents et requêtes sur les différents pairs du réseau [6]

La génération des profils indispensables à notre modèle d'interclassement a été faite en utilisant l'algorithme de Godin implémenté dans la plate-forme Galicia V 3 [2]. Comme jeu de de données, nous avons utilisé "BigDataSet"[1], composé de 25 000 documents et de 4999 requêtes répartis sur 500 pairs.

Dans notre système nous avons recouru à deux modèles de distribution de documents et requêtes sur les différents pairs : une distribution aléatoire avec deux pourcentages de réplication de données et une distribution uniforme sans duplication [6]

Afin d'apprendre, nous avons lancé la moitié des requêtes sur chaque nœud en simulation afin de construire un fichier log initial par nœud. Par la suite, nous avons créé nos profils comme indiqué dans 2.1.2.

Pour la phase de test, nous avons lancé le reste des requêtes sur les 100 premiers nœuds. Egalement une partie des requêtes apprises a été lancée pour le test.

3.2. Métrique d'évaluation utilisée

L'une des mesures les plus utilisées pour comparer les modèles de RI est la mesure de Précision pour les k premiers documents ertinents[3]. Elle est définie comme suit : soit P l'ensemble de documents pertinents pour une requête q. soit R, l'ensemble de documents retournés en réponse à la même requête pour une position, k, donnée. La métrique de base que nous avons utilisée, entre autres, est definie comme suit :

$$Prec@k \quad = \quad \frac{|R \bigcap P|@k}{k} \qquad [3]$$

3.3. Résultats expérimentaux

Notre approche a été comparée aux modèles Round Robin (RR)[3] et Borda Count(BC) [3]. Ce sont des modèles à base de rangs dont les bons résultats sont observés quand il s'agit de collections homogènes et d'une distribution aléatoire ou uniforme, ce qui est le cas de nos tests. La Figure 3 présente les résultats obtenus pour la métrique $Prec@k$

en fonction du nombre de requêtes. Les premiers tests presentés dans cette figure sont, à notre avis, très encourageants. La comparaison de notre approche avec celles existantes montre que notre approche est competitive. Pour un cadre défaforable à notre approche (modèle uniforme et aléatoire), la Figure 3 montre que l'interclassement à base de profils donne les meilleurs résultats comparé à RR et BC. En jouant sur les differents modèles de distribution testés, nous remarquons que notre approche donne les meilleurs résultats même sans recourir à la duplication (cas du graphe (c) de la figure 3). Quant à la réplication de données, nous remarquons que si la constante de réplication augmente, le modèle apporte une difference de performance plus grande même avec peu de requêtes (cas du graphe (b) de la figure 3).

4. Conclusion

Nous avons proposé, dans ce papier, une approche d'interclassement de résultats à base de profils utilisateurs dans le contexte des SRI-P2P. Pour valider notre approche, nous avons réalisé un certain nombre d'expérimentations et nous l'avons comparé à des approches existantes. L'étude empirique que nous avons menée nous a permis de donner un avant goût sur les performances de notre solution d'interclassemnt puisque nous l'avons appliquée dans les conditions qui lui sont défavorables par rapport aux approches classiques (des collections homogènes et des distributions uniformes et aléatoires. Aucune importance n'est donnée à certains nœuds du réseau. Tous réagissent de la même façon. Les premiers résultats obtenus sont assez encourageants et nous laissent envisager de nouvelles améliorations ainsi que des expérimentations beaucoup plus poussées. En effet, il est possible de jouer sur le nombre de pairs testés, sur certains paramètres de distribution et de scoring. Il nous semble également nécessaire de qualifier les performances du modèle en jouant d'une part les requêtes apprises (meilleure performance) et d'autres part, les nouvelles requêtes.

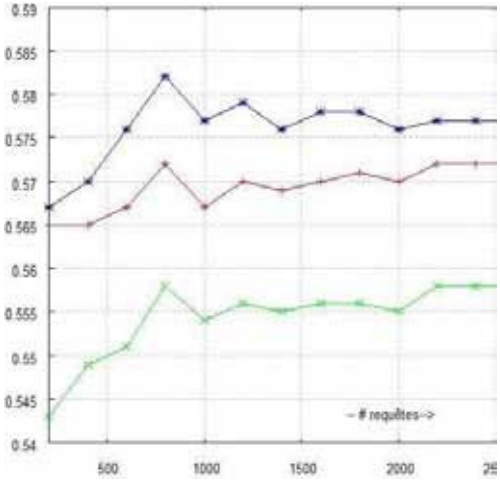
5. Bibliographie

[1] DEFUDE B., « Le projet rare : Routage optimisé par apprentissage de requêtes », http ://www-inf.int-evry.fr/defude/RARE, 2008.

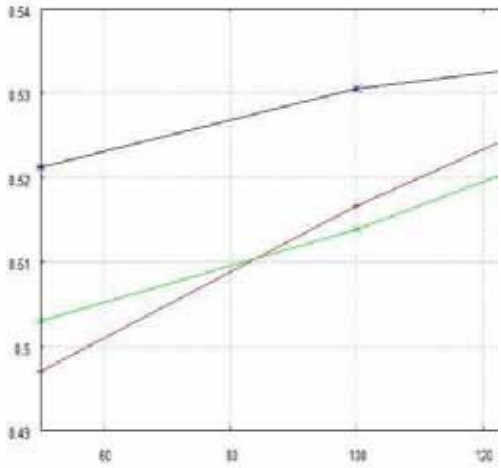
[2] GODIN R., MISSAOUI R., ALAOUI H., « Incremental concept formation algorithms based on galois (concept) lattices »,*J.Computational Intelligence*,246-267, 1995.

[3] RENDA M. E. , STRACCIA U.« Web metasearch : rank vs. score based rank aggregation methods. », *SAC'03 : Proceedings of the 2003 ACM symposium on Applied computing*, pp. 841-846, New York, NY, USA, 2003.

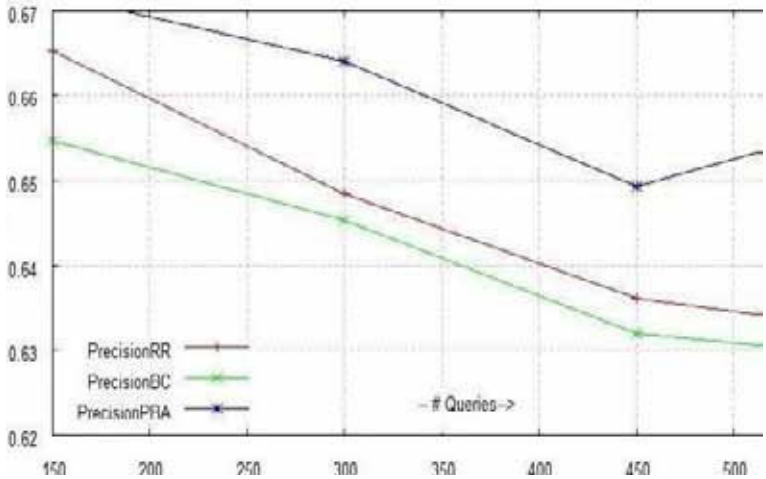
[4] SALTON G. « Automatic text processing : the transformation, analysis, and retrieval of information by computer », *Addison-Wesley Longman Publishing Co., Inc.*, Boston, MA, USA, 1989.



(a) Modèle aléatoire, constante de replication 40



(b) Modèle aléatoire, constante de replication 60



(c) Modèle uniforme, sans duplication

Figure 3. *Variation de la précision moyenne @k en fonction du de requêtes*

[5] SHOKOUHI M., ZOBEL J., , BERNSTEIN Y.,« Distributed text retrieval from overlapping collections. », *ADC'07 : Proceedings of the eig hteenth conference on Australasian database*, pp. 141150, Darlinghurst, Australia, Australia, . Australian Computer Society, Inc, 2007.

[6] ZAMMALI S. , AROUR K., « P2PIRB : benchmarking framework for P2PIR », *Globe 2010, LNCS 6265*, pp, 148–159, Springer, Heidelberg, 2010.