Évaluation de la pertinence dans les moteurs de recherche géoréférencés

Léa Laporte*,*, Laurent Candillier*, Sébastien Déjean**, Josiane Mothe*****

**Nomao SA*
1 Avenue Jean Rieux, F-31500 Toulouse
{laurent,lea}@nomao.fr

*** Institut de Mathématiques de Toulouse*
Université Toulouse 3, 118 Route de Narbonne, F-31062 Toulouse cedex 9
sebastien.dejean@math.univ-toulouse.fr

**** Institut de Recherche en Informatique de Toulouse*
Université Toulouse3, 118 Route de Narbonne, F-31062 Toulouse cedex 9
{laporte,mothe}@irit.fr

RÉSUMÉ. *Optimiser le classement des résultats d'un moteur par un algorithme de learning to rank nécessite de connaître des jugements de pertinence entre requêtes et documents. Nous présentons les résultats d'une étude pilote sur la modélisation de la pertinence dans les moteurs de recherche géoréférencés. La particularité de ces moteurs est de présenter les résultats de recherche sous forme de carte géographique ou de liste de fiches. Ces fiches contiennent les caractéristiques du lieu (nom, adresse, téléphone, etc.) dont la plupart sont cliquables par l'utilisateur. Nous modélisons la pertinence comme la somme pondérée des clics sur le résultat. Nous montrons qu'équipondérer les différents éléments du modèle donne de bons résultats et qu'un ordre d'importance entre type de clics peut être déduit pour déterminer les pondérations optimales.*

ABSTRACT. *Learning to rank documents on a search engine requires relevance judgments. We introduce the results of an innovating study on relevance modeling for local search engines. These search engines present search results on a map or as a list of maps. Each map contains all the attributes of a place (noun, address, phone number, etc). Most of these attributes are links users can click. We model the relevance as the weighted sum of all the clicks on a result. We obtain good results by fixing the same weight for each component of the model. We propose a relative order between clicks to determine the optimal weights.*

MOTS-CLÉS : *collection d'apprentissage, fichiers de logs, modèles de clics, learning to rank, moteurs de recherche géoréférencés.*

KEYWORDS : *training data, logs files, clicks models, learning to rank, local search engine.*

Cet article centré solution a pour thème majeur « Adaptation des systèmes d'information à l'utilisateur », pour sous-thème « Modélisation des connaissances extraites des données collectée » et pour thème mineur « Evaluation des systèmes d'information ».

1. Introduction

De nombreux moteurs de recherche d'information géoréférencés ont émergé au cours des dernières années. Ces moteurs sont dédiés à la recherche de lieux qu'ils recommandent en fonction de leur proximité géographique par rapport à l'utilisateur, de leur e-réputation, ou encore des goûts de l'utilisateur et de son réseau social. Apprendre à classer pertinemment les résultats d'une recherche en tenant compte de l'ensemble de ces éléments est un sujet émergent en Recherche d'Information.

Certains systèmes de RI utilisent des algorithmes pour apprendre à ordonner correctement les documents vis-à-vis de leur pertinence à une requête. Le *learning to rank* est le domaine de recherche qui s'intéresse aux méthodes permettant d'apprendre automatiquement des fonctions d'ordonnancement pour optimiser le classement des résultats de recherche. Ces fonctions d'ordonnancement sont apprises sur des jeux de données d'apprentissage constituées d'un ensemble de paires requête-document pour lesquelles la pertinence est connue. Ces fonctions sont ensuite utilisées pour ordonner les documents lorsque de nouvelles requêtes sont soumises au système. La performance des algorithmes, en termes de temps de calcul et de prédiction de classement, est mesurée sur des collections pour lesquelles la pertinence est évaluée manuellement par des experts, selon la méthodologie Cranfield (Cleverdon, 1991).

Dans le cadre du *learning to rank* pour les moteurs de recherche géoréférencés, l'absence de jeux de données adaptés constitue un verrou technique important pour l'évaluation des algorithmes. En effet, les approches de *learning to rank géoréférencé* sont généralement évaluées sur la collection en langue anglaise proposée par les campagnes GeoCLEF (Mandl *et al.*, 2008). Celle-ci est constituée d'articles issus de la presse anglaise et faisant référence à certaines informations géographiques. Or, les moteurs de recherche géoréférencés manipulent des lieux représentés par des fiches très détaillées et très structurées. Ainsi, la collection GeoCLEF ne semble pas adaptée à l'évaluation des approches de *learning to rank* pour des moteurs de recherche de lieux. Il nous paraît donc nécessaire de proposer des collections plus adaptées.

L'objectif de ce travail est de proposer une méthode efficace permettant la création automatique de collections adaptées à l'évaluation des approches de *learning to rank* dans le cadre de la recherche géoréférencée. Deux solutions peuvent être envisagées. La première consiste à évaluer manuellement la pertinence suffisamment de couples requête-document. Cette méthode est longue, coûteuse et difficile à mettre en place. La seconde solution consiste à inférer la pertinence de chaque couple requête-document à partir des clics réalisés par les utilisateurs sur les résultats de recherche et stockés dans les fichiers de connexion des moteurs. Ces clics sur les résultats de recherche sont considérés comme des jugements implicites de la pertinence et sont utilisés pour étiqueter les jeux de données, i.e. associer à chaque paire requête-document un jugement de pertinence.

De nombreuses modélisations de la pertinence ont été proposées afin de créer des collections d'évaluation à partir des clics des utilisateurs. Malheureusement, ces modélisations ne sont pas adaptées dans le cas des moteurs géoréférencés. En effet, elles considèrent que plusieurs résultats peuvent être consultés au cours d'une session de recherche, mais qu'un résultat n'est lui-même cliqué qu'une seule fois dans la session. Or, il existe des situations dans lesquelles un même résultat peut être cliqué plusieurs fois. Les utilisateurs peuvent revenir consulter un résultat déjà cliqué au cours de la session de recherche. Plus spécifiquement, dans le cadre des moteurs de recherche géoréférencés, plusieurs types d'actions sont possibles sur un même résultat. Chaque lieu est représenté sous la forme d'une fiche présentant plusieurs liens cliquables. Ainsi, sur le moteur de recherche Nomao, les utilisateurs peuvent cliquer sur le nom du lieu, sur son numéro de téléphone ou encore sur son site Internet. On retrouve cette spécificité sur des moteurs de recherche bibliographiques, comme par exemple sur le site « The European Library »[1]qui offre un accès aux collections des bibliothèques nationales européennes. Les résultats y sont également présentés sous forme de fiches sur lesquelles différentes actions sont possibles : clic sur le titre, téléchargement d'une copie du document ou encore ajout dans la liste de favoris. Des clics multiples sont alors possibles sur une même référence. A notre connaissance, les modélisations actuelles ne prennent pas en compte cette spécificité.

Nous proposons une nouvelle modélisation de la pertinence lorsque plusieurs actions sont possibles sur un même résultat. Nous présentons les résultats d'une étude pilote au cours de laquelle nous avons évalué notre approche sur des données issues du moteur de recherche géoréférencé Nomao. Les objectifs de cette étude sont triples : mieux comprendre les différents facteurs influençant la pertinence sur les moteurs de recherche géoréférencés, proposer un modèle de prédiction de la pertinence adapté et efficace, étiqueter de façon automatique de grands jeux de données pour évaluer les méthodes de learning to rank géoréférencé. Il est important de noter que si l'approche proposée est évaluée sur un moteur de recherche géoréférencé, elle ne se limite pas à ce cadre et peut a priori être utilisée sur d'autres moteurs pour lesquels plusieurs clics sont possibles pour les résultats.

Cet article est structuré de la façon suivante. La section 2 présente un état de l'art des algorithmes de *learning to rank* et de la modélisation de la pertinence à partir des actions utilisateurs. La section 3 définit le modèle proposé. La section 4 décrit les caractéristiques des données. La section 5 présente l'algorithme d'ordonnancement et le processus d'apprentissage. La section 6 présente le protocole et les résultats de l'expérience permettant de déterminer les valeurs de paramètres de l'algorithme d'ordonnancement. La section 7 détaille la deuxième expérience, qui met en évidence l'intérêt de notre approche et l'impact des spécificités de notre jeu de données sur les mesures d'évaluation. La section 8 détaille le protocole expérimental et les résultats de l'expérience analysant

[1] www.theeuropeanlibrary.org

l'influence des actions sur la pertinence. Enfin, la section 9 est une discussion des résultats et des perspectives de recherche à court terme.

2. Etat de l'art

Dans cet article, nous proposons une approche pour la modélisation de la pertinence sur les moteurs de recherche géoréférencés. La pertinence prédite permettra d'étiqueter des jeux de données utilisés pour l'apprentissage de fonctions d'ordonnancement. Dans cet état de l'art, nous présentons les méthodes de learning to rank et les problématiques du learning to rank géoréférencé. Puis nous détaillons les travaux existants sur la modélisation de la pertinence à partir des actions utilisateurs.

Learning to rank : Les algorithmes de *learning to rank* sont regroupés en trois grandes approches (*pointwise*, *pairwise*, *listwise*) qui diffèrent sur leur façon de considérer le problème d'apprentissage (Liu, 2011). L'approche *pointwise* considère les documents séparément en entrée du système d'apprentissage. A chaque document est associé un score de pertinence ou un degré de pertinence pour la requête donnée. Le problème d'apprentissage est alors assimilé à un problème de régression (Cossock *et al*., 2006) ou de classification (Nallapati, 2004) respectivement. L'approche *pairwise* considère en entrée du système d'apprentissage des paires de documents auxquels sont associées des jugements de préférence à valeur dans {-1,1}. Pour une requête fixée, associer le jugement de préférence r=1 à la paire de documents (d_1, d_2) signifie que le document d_1 est plus pertinent que le document d_2. On dit qu'il est préféré à d_2 et on note $d_1 \succ d_2$. A l'inverse, si r=-1, alors le document d_2 est préféré au document d_1 et on note $d_1 \prec d_2$. Le problème d'apprentissage est ici ramené à un problème de classification (Joachims, 2002) (Burges *et al*., 2005). Enfin, l'approche *listwise* considère en entrée du système d'apprentissage une liste ordonnée de documents. La fonction d'ordonnancement est apprise par minimisation de la distance entre la liste apprise et la liste de référence (Cao *et al.*, 2007) ou par optimisation d'une mesure de recherche d'information (Yue *et al.*, 2007).

Les travaux en *learning to rank géoréférencé* portent sur la définition et le choix des variables à utiliser dans les algorithmes de *learning to rank*. Martins *et al.* (Martins *et al.*, 2010) ont montré qu'utiliser des similarités textuelles et géographiques permettait d'améliorer les performances des algorithmes de *learning to rank* pour la recherche géoréférencée. D'autres approches proposent de prendre en compte le contexte pour améliorer l'ordonnancement (Kumar, 2011). Les travaux sont généralement évalués sur la collection GeoCLEF (Mandl *et al.*, 2008).

Ces méthodes sont évaluées sur des collections d'apprentissage pour lesquelles des jugements de pertinence sont connus. Or, construire manuellement une collection d'apprentissage est un processus long et coûteux. Les clics, qui traduisent un intérêt de l'utilisateur, constituent des jugements implicites de pertinence. Plusieurs études

ont utilisé les clics pour modéliser la pertinence d'un document pour une requête. Cet aspect est décrit dans la section suivante.

Modélisation de la pertinence : (Joachims, 2002) et (Joachims *et al.*, 2005) ont proposé des stratégies d'extraction de préférences basées uniquement sur les séquences de clics. Ainsi, la stratégie *SkipAbove* considère que le dernier document cliqué est préféré à l'ensemble des documents de rangs supérieurs non cliqués pour la requête et doit être renvoyé après apprentissage au dessus de ces documents. Radlinski *et al.* (Radlinski *et al*., 2005) ont étendu cette règle dans le cas des chaînes de requêtes reformulées. D'autres approches ont été proposées pour prendre en compte le biais de position dans la modélisation de la pertinence. Le biais de position est défini comme la tendance des utilisateurs à cliquer sur les premiers documents proposés, même si ceux-ci ne sont pas pertinents et à délaisser des documents pertinents situés plus bas dans la liste.

Le modèle de position (« position model ») (Craswell *et al.*, 2008) suppose que tous les documents n'ont pas la même probabilité d'être cliqués et que cette probabilité diminue avec le rang du document. Dans ce modèle, l'utilisateur clique sur un seul document qui est alors considéré comme pertinent. La probabilité qu'un utilisateur considère un résultat comme pertinent dépend de la position de celui-ci dans la liste. Le modèle en cascade (Craswell *et al.*, 2008) suppose que l'utilisateur consulte la liste du haut vers le bas et décide soit de cliquer soit de passer au résultat suivant. L'utilisateur clique sur le document qui lui paraît pertinent, par rapport aux autres résultats qu'il a déjà passé. Ces deux modèles considèrent qu'un seul document est pertinent : celui qui a été cliqué en premier. Le modèle de clics dépendants (« Dependent Click Model ») (Guo *et al.*, 2008) généralise le modèle en cascade pour prendre en compte les sessions de recherche avec clics multiples. Les auteurs supposent que l'utilisateur consulte les résultats du haut vers le bas de la liste, qu'il clique sur un résultat qui lui paraît pertinent par rapport aux précédents et qu'une fois le document consulté, l'utilisateur peut ou non revenir sur la liste pour continuer à parcourir les résultats. La pertinence du document est évaluée globalement sur l'ensemble des utilisateurs. Les auteurs montrent que cette pertinence peut s'écrire comme le ratio entre le nombre de clics sur le document pour l'ensemble des sessions et le nombre de sessions pour lesquelles le document était dans la liste de résultat. Le modèle Bayésien dynamique (« Dynamic Bayesian Network Click Model ») (Chapelle *et al.*, 2009) généralise le modèle en cascade aux sessions avec clics multiples, mais introduit de nouvelles notions sur la pertinence. Les auteurs opposent ainsi la pertinence perçue à la pertinence effective. La pertinence perçue traduit le fait que l'utilisateur est attiré par un document avant sa consultation. Elle est définie comme la probabilité que le document soit cliqué (inférée à partir de l'ensemble des sessions utilisateurs). La pertinence effective traduit la satisfaction de l'utilisateur après consultation du document. Si l'utilisateur est satisfait, il stoppe sa recherche, s'il ne l'est pas, il retourne sur la liste de résultats et consulte d'autres documents. La pertinence globale est le produit de la pertinence perçue et de la pertinence effective. Cette pertinence globale est utilisée pour annoter

la collection. Enfin, (Liu *et al.*, 2009) ont proposé un modèle bayésien gérant les sessions à clics multiples et qui prédit la probabilité qu'un document soit préféré à un autre. Cette approche est utile pour annoter des collections sur lesquelles seront évalués des algorithmes de type *pairwise*.

Ces modèles sont évalués par comparaison des jugements prédits aux jugements d'experts (Joachims *et al.*, 2005), par comparaison de l'erreur de prédiction d'un algorithme d'ordonnancement appliqué sur des données étiquetées par l'approche (Chapelle *et al.*, 2009) ou bien par comparaison avec des résultats obtenus lors d'expériences utilisateurs, comme des suivis par oculométrie (Joachims *et al.*, 2005).

3. Modélisation de la pertinence pour les moteurs géoréférencés

L'objectif de cet article est de proposer une modélisation de la pertinence adaptée aux moteurs de recherche géoréférencés. Cette pertinence doit pouvoir être utilisée pour construire rapidement de nouvelles collections d'apprentissage. Nous souhaitons également identifier le lien entre les différents types de clics et la pertinence, afin de modéliser celle-ci au mieux.

Pour définir le score de pertinence, nous nous basons exclusivement sur les clics des utilisateurs. Nous partons de l'hypothèse qu'un clic est un jugement implicite de pertinence. Chaque document peut être cliqué plusieurs fois (sur son titre, son numéro de téléphone, etc). Nous supposons ainsi qu'un document est d'autant plus pertinent qu'il enregistre beaucoup de clics. Les documents peuvent être ordonnés suivant le nombre de clics qui ont été effectués. La pertinence peut donc être approximée par le nombre de clics sur un document.

Cette approche considère que toutes les actions effectuées traduisent un intérêt similaire de l'utilisateur. Autrement dit, que l'utilisateur n'est pas plus intéressé par le document lorsqu'il clique sur le lien vers le numéro de téléphone que lorsqu'il clique sur le titre pour avoir accès à la fiche complète. Or, intuitivement, nous supposons que ces actions traduisent un niveau d'intérêt différent de l'utilisateur, et que l'on peut ordonner les clics suivant leur importance. Ainsi, un clic sur le numéro de téléphone traduirait le fait que l'utilisateur a été suffisamment satisfait pour contacter le propriétaire d'un lieu, tandis qu'un clic sur le titre indiquerait seulement que l'utilisateur est a priori intéressé et souhaite observer les informations supplémentaires disponibles. Le niveau d'intérêt, donc de pertinence, serait alors moindre. Cela nous conduit à considérer la somme pondérée des clics en temps que score de pertinence, afin de prendre en compte cette nuance. Nous définissons alors le score de pertinence de la façon suivante:

$$r_{d,q,u} = \sum_i \alpha_i c_{i,d,q,u}$$

avec $r_{d,q,u}$ la pertinence du document d pour la requête q et l'utilisateur u

$c_{i,d,q,u}$ le nombre de clics de type *i* sur le document *d* pour la requête *q* et l'utilisateur *u*

α_i un coefficient traduisant l'importance des clics de types *i*

Nous proposons d'étudier dans un premier temps le score $\alpha_i = 1 \forall i$ afin de vérifier la cohérence de notre approche. Nous chercherons à montrer que le score est adapté pour la construction d'une collection d'apprentissage. Nous vérifierons qu'utiliser ce score pour définir la pertinence des couples requête-document permet l'apprentissage d'un modèle d'ordonnancement donnant de bons résultats. Dans un second temps, afin d'améliorer la modélisation, nous nous intéresserons à l'influence de chaque type de clics sur l'apprentissage de la fonction d'ordonnancement. Nous souhaitons ainsi définir un ordre d'importance entre types de clics afin d'estimer les pondérations adéquates dans nos futurs travaux.

4. Données

4.1. *Présentation des données du moteur de recherche géoréférencé Nomao*

Dans cette étude, nous disposons de fichiers de connexions issus du moteur de recherche géoréférencé Nomao. Nomao permet aux utilisateurs de rechercher des lieux qui leur sont recommandés suivant leurs goûts et de leur proximité géographique. Les résultats sont présentés sous forme de fiches dont plusieurs éléments sont cliquables. Les résultats sont aussi localisés sur une carte géographique et peuvent être cliqués à partir de celle-ci.

Nous extrayons plusieurs informations du fichier de connexion: texte et identifiant de la requête, identifiant de l'utilisateur, identifiant du document, liste des vingt premiers résultats associés à la requête, nombre et type de clics pour chaque triplet utilisateur-requête-document. Ces informations nous permettent d'une part de créer les couples requête-document qui seront utilisés dans la collection d'apprentissage, et d'autre part, de calculer la pertinence pour chacun de ces couples. Nous avons identifié cinq types de clics sur le moteur de recherche : clics sur le titre (α_{titre}), le bouton « Réserver » ($\alpha_{réservation}$), le numéro de téléphone ($\alpha_{téléphone}$), l'adresse du site internet ($\alpha_{internet}$) et le résultat depuis la carte géographique (α_{carte}).

Nous ne conservons que les requêtes avec au moins deux résultats renvoyés et au moins un résultat cliqué. Pour chaque requête, nous disposons de la liste des vingt premiers documents retournés. Nous avons extrait 14700 triplets utilisateur-requête-document pour lesquels le nombre et le type de clics sont connus. Il y a au total 2014 requêtes distinctes, 1745 utilisateurs différents et 14343 documents.

4.2. *Analyses préliminaires*

Nous remarquons que la proportion de clics n'est pas la même pour chaque type de clics. Les clics sur le titre, le téléphone et la carte sont les plus fréquents et représentent respectivement 44%, 28% et 24% du nombre total de clics. Les clics sur le site Internet et sur le bouton « Réserver » ne représentent respectivement que 3% et 1% du nombre total d'actions. Par ailleurs, nous remarquons que pour certaines requêtes, les utilisateurs ne réalisent qu'un seul type d'action. Ainsi, pour 37% des requêtes, les utilisateurs n'ont cliqué que sur le titre. Les requêtes avec un seul type de clic $\alpha_{téléphone}$, α_{carte}, $\alpha_{internet}$, $\alpha_{réservation}$ représentent respectivement 22%, 12%, 1% et 0,6% du nombre total de requêtes.

Le jeu de données est caractérisé par le fait que le nombre de documents restitués n'est pas le même pour toutes les requêtes (Figure 1). On constate que seules 38% des requêtes retournent vingt documents, tandis que 26% en retournent trois ou moins.

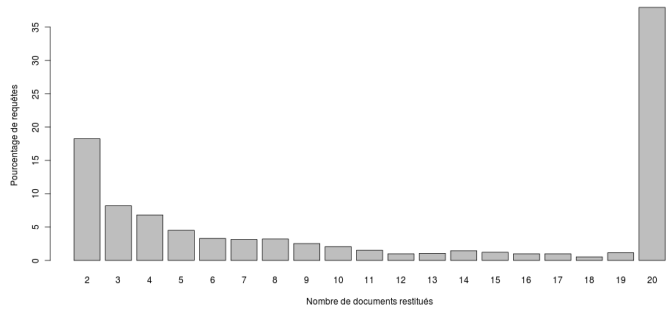

Figure 1. *Répartition du nombre de documents retournés par requêtes*

Cette disparité peut avoir une influence sur les mesures d'évaluation que nous utilisons : précision au rang k (P@k), précision moyenne (AP) et moyenne de la précision moyenne sur l'ensemble des requêtes (MAP). Pour une requête fixée, on note n le nombre de documents restitués et n_{rel} le nombre total de documents pertinents. La précision au rang k et la précision moyenne sont alors définies de la façon suivante :

$$P@k = \frac{\text{Nombre de documents pertinents restitués jusqu'au rang k}}{\text{k}}$$

$$AP = \frac{\sum_{k=1}^{n} P@k.rel(k)}{\text{n}_{\text{rel}}} \text{ où } rel(k) = \begin{cases} 1 \text{ si le document au rang k est pertinent} \\ 0 \text{ sinon} \end{cases}$$

Nous rappelons que nous ne conservons que les requêtes avec au moins deux résultats restitués dont au moins un cliqué. Dans le cas de requêtes avec deux résultats restitués, dont un seul pertinent, la précision moyenne sera ainsi égale à 1 si

$$MAP = \frac{\sum_{q=1}^{Q} AP^{(q)}}{\mathrm{Q}} \text{ où } Q \text{ est le nombre total de requêtes}$$

le document pertinent est retourné en première position et égale à 0,5 s'il est retourné en deuxième position. Les requêtes ne restituant que peu de documents risquent ainsi de biaiser la MAP.

5. Algorithme et processus d'apprentissage

Dans nos expériences, nous évaluons chaque score avec le même algorithme d'apprentissage, dans lequel nous utilisons 145 variables réparties en quatre catégories : similarités requête-document, caractéristiques de la requête, du document et de l'utilisateur. Nous analysons l'influence de chaque score sur la qualité de l'apprentissage.

5.1. Présentation de l'algorithme d'ordonnancement utilisé : RankingSVM

Ranking SVM fait partie, avec RankBoost, des algorithmes de référence pour l'apprentissage des fonctions d'ordonnancement. Ils ont été évalués sur de nombreux jeux de données et présentent des performances globalement très proches. Nous avons choisi d'utiliser RankingSVM car plusieurs implémentations sont mises à disposition librement et maintenues par l'auteur sur sa page personnelle[2], ce qui n'est pas le cas pour RankBoost.

L'algorithme RankingSVM prédit les jugements de pertinence par minimisation de la fonction objectif suivante (Liu, 2011) :

$$\min \frac{1}{2}\|w\|^2 + c\sum_{i=1}^{n} \sum_{u,v:y_{u,v}^{(i)}=1} \xi_{u,v}^{(i)}$$

sous la contrainte $w^T(x_u^{(i)} - x_v^{(i)}) \geq 1 - \xi_{u,v}^{(i)}$ si $y_{u,v}^{(i)} = 1$ et $\xi_{u,v}^{(i)} \geq 0$, $i = 1,...,n$

où n est le nombre de requêtes de l'échantillon d'apprentissage, $(x_u^{(i)} - x_v^{(i)})$ la paire de documents u et v représentés dans l'espace des variables et associés à la requête i, $y_{u,v}^{(i)}$ le jugement de préférence pour la paire de document et $\xi_{u,v}^{(i)}$ les termes d'erreur pour les SVM non linéairement séparables. Le paramètre c permet de contrôler la largeur de la marge et donc le taux d'erreur. Il doit être déterminé par validation croisée.

---

[2] http://svmlight.joachims.org/

Cette étape de validation croisée peut être coûteuse en temps de calcul. Il est possible de réduire ce temps d'exécution en agissant sur certains paramètres de l'algorithme, comme le critère d'arrêt ε (défini pour satisfaire aux conditions d'optimalité, il s'agit de la borne supérieure de la somme des termes d'erreur $\xi_{u,v}^{(i)}$) ou le nombre d'itérations nécessaires pour résoudre chaque sous-problème d'optimisation (Joachims, 1999), noté #.

5.2. *Description des données d'apprentissage*

Pour évaluer la stabilité des approches, nous avons simulé cinq échantillons différents à partir du jeu de données initial. Nous avons effectué un tirage aléatoire sans remise des requêtes pour constituer cinq sous-ensembles disjoints de tailles sensiblement égales. Chaque ensemble constitue un jeu de données de test, tandis que la réunion des quatre échantillons restants correspond au jeu de données d'apprentissage. Les expérimentations sont réalisées sur chaque jeu de données simulé. Cette approche par validation croisée permet d'évaluer la robustesse de l'approche à la composition des échantillons d'apprentissage et de test. Ce protocole nous permet ainsi de mesurer la capacité de généralisation du modèle proposé. Le paramètre *c* est fixé par validation croisée sur la grille {0.001, …,0.009, 0.01, …, 1, 1.1, …, 1.3}. Chaque échantillon d'apprentissage est découpé en dix sous-échantillons utilisés uniquement pour le choix de *c*. Cette étape peut être très coûteuse en temps. Nous avons donc mené une expérience préliminaire pour choisir les critères d'arrêts permettant d'apprendre les modèles avec une erreur d'apprentissage et un temps de prédiction raisonnable.

6. Détermination des critères d'arrêts de Ranking SVM

L'objectif de cette expérience est de déterminer les valeurs des critères d'arrêt permettant un bon compromis entre temps d'exécution et qualité de prédiction. Ces valeurs seront utilisées dans les expériences pour réduire le temps de calcul de l'étape de validation croisée et permettre une évaluation rapide de la méthode proposée.

6.1. *Protocole expérimental*

Nous analysons l'influence sur le temps de calcul de deux critères d'arrêt de RankingSVM : le nombre d'itérations #, et ε. Sur des données annotées avec le score $\alpha_i = 1$ pour tout *i,* nous appliquons l'algorithme RankingSVM en faisant varier le paramètre # dans {5 000, 10 000} et ε dans {0.01, 0.1}, Nous analysons ensuite les temps d'exécution, l'erreur de prédiction et la MAP afin de déterminer le meilleur couple de valeurs des paramètres.

6.2. *Résultats*

Le temps d'exécution est le plus faible pour ε=0.1et 5 000 itérations et est égal à 10 000 secondes. Il augmente de 8,7% avec le nombre d'itérations. Par ailleurs, le temps de calcul augmente lorsque ε diminue, d'un facteur dix pour 5 000 itérations et d'un facteur six pour 10 000 itérations. Pour ε=0.01 et 5 000 itérations, la durée de l'étape de validation croisée a été très disparate suivant les échantillons, avec une moyenne de 18 heures et 20 minutes[3] pour les échantillons 2, 4 et 5, contre une moyenne de 43 heures et 20 minutes pour les échantillons 1 et 3. D'après ces résultats, il paraît judicieux de fixer ε = 0.1 pour réduire les temps d'exécution. Nous devons néanmoins vérifier que les erreurs de prédiction ne sont pas trop importantes pour cette valeur de ε.

(#, ε) / Echantillon	(10 000,0.01)	(10 000,0.1)	(5 000,0.01)	(5 000,0.1)
1	39,3	47,16	42,8	42,8
2	33,6	35,6	32,8	32
3	34,4	36,2	36,9	38,4
4	35,4	43,2	35,4	42,4
5	34,4	38,9	37,3	37,3
Moyenne	35,4	40,2	37	38,6
Moyenne corrigée	34,5	38,5	35,6	37,5

Tableau 1. *Erreurs de prédiction moyenne et par échantillon (en %)*

Le tableau 1 présente les erreurs de prédiction sur l'échantillon test pour l'ensemble des échantillons de validation croisée et des combinaisons de paramètres testés. Nous remarquons que l'erreur de prédiction pour l'échantillon 1 est élevée pour l'ensemble des couples de paramètres. Cet échantillon présente une répartition des types de clics atypiques. La proportion de clics sur le titre et la carte est faible, tandis que la proportion de clics « réservation » est élevée comparé aux autres échantillons. Cette caractéristique peut expliquer les valeurs observées ici. Par la suite, nous présentons les valeurs d'erreur et de MAP moyennes avec et sans l'échantillon 1 (moyenne corrigée), pour éviter de biaiser les interprétations.

Nous constatons qu'à nombre d'itérations fixé, l'erreur de prédiction est d'autant plus forte que ε est grand. L'erreur de prédiction est la plus forte pour 10 000 itérations et ε=0.1. L'erreur de prédiction est la plus faible pour ε=0.01. Néanmoins, nous avons constaté que les temps d'exécution étaient plus longs pour cette valeur de ε. Ils étaient instables pour ε=0.01 et #=5 000 itérations. Nous avons effectué la suite

[3] Nous utilisons une machine de 1 cœur et 36 Go de RAM

des analyses en fixant 5 000 itérations et ε=0.1. Ces valeurs de paramètres constituent un bon compromis entre temps d'exécution et erreur de prédiction.

(#, ε) / Echantillon	(10 000,0.01)	(10 000,0.1)	(5 000,0.01)	(5 000,0.1)
1	0,76	0,72	0,74	0,75
2	0,81	0,79	0,81	0,81
3	0,81	0,80	0,80	0,78
4	0,79	0,76	0,79	0,75
5	0,81	0,77	0,79	0,79
Moyenne	0,80	0,77	0,79	0,78
Moyenne corrigée	0,80	0,78	0,80	0,78

Tableau 2. *MAP moyenne et par échantillon*

Nous notons que les valeurs de la MAP (Tableau 2) sont élevées par rapport aux résultats connus (Liu, 2011). Cela confirme notre hypothèse selon laquelle la présence de nombreuses requêtes avec peu de documents retournés peut biaiser la valeur de la MAP. Elle peut toutefois être utilisée pour comparer les comportements des scores que nous souhaitons tester.

7. Evaluation du score αi = 1 pour tout i

Dans cette expérience, nous cherchons d'une part à comparer la performance du score $\alpha_i = 1$ pour tout i par rapport au cas où la pertinence est définie de façon aléatoire. D'autre part, nous souhaitons analyser l'évolution des valeurs de MAP. Celle-ci devrait diminuer dans le cas d'une pertinence aléatoire.

7.1. *Protocole expérimental*

Nous définissons un score pour lequel la pertinence est tirée aléatoirement dans l'ensemble {0,1} pour chaque triplet utilisateur-requête-document. Nous apprenons un modèle sur les données étiquetées avec ce score, et un modèle sur les données étiquetées avec le score $\alpha_i = 1$ pour tout *i*. Nous comparons les temps d'exécution des deux modèles. Pour la même combinaison des paramètres # et ε, un temps d'exécution plus grand indiquera une plus grande difficulté à apprendre un modèle et donc un score moins adapté. Nous comparons également l'erreur de prédiction et la MAP qui traduisent une bonne performance du modèle en apprentissage.

7.2. *Résultats*

Nous constatons que tous nos indicateurs sont meilleurs pour le score évalué (Tableau 4). Le temps d'exécution de l'étape de validation croisée est plus faible dans le cas du score $\alpha_i = 1$ pour tout i ce qui peut signifier que l'apprentissage d'un modèle est plus difficile dans le cas où les documents pertinents sont choisis de

façon aléatoire. Ceci est confirmé par la forte valeur de l'erreur de prédiction sur l'échantillon test : l'algorithme est incapable d'apprendre un modèle dans ce cas. Enfin, on constate que la valeur de la MAP est plus faible que pour le score proposé. L'approche qui considère la pertinence comme le nombre de clics sur le document semble cohérente.

	Temps d'exécution	**Erreur de prédiction**	**MAP**
Aléatoire	4 h	82,7%	0,65
α_i = 1 pour tout i	2 h 50 mn	38,6%	0,78

Tableau 3. *De meilleurs résultats pour le score score α_i = 1 pour tout i par rapport à un tirage aléatoire de la pertinence*

Nous constatons que la valeur de la MAP reste élevée dans le cas d'une pertinence aléatoire. Nous expliquons cette caractéristique par la présence importante de requêtes restituant peu de résultats, ce qui augmente les valeurs de MAP.

8. Influence des différents types de clics sur la pertinence

L'objectif de cette expérience est de montrer que chaque type d'action a une influence sur la pertinence. Certains clics traduiront une pertinence forte tandis que d'autres pourront au contraire introduire du bruit dans le calcul de la pertinence. Cette expérience permettra de déterminer l'importance de chaque action dans l'évaluation de la pertinence.

8.1. *Protocole expérimental*

Nous étudions cinq scores pour lesquels nous avons successivement annulé le poids d'une des actions. Nous appliquons l'algorithme RankingSVM sur les jeux de données étiquetés avec ces scores. Nous analysons l'influence de chaque action sur deux critères : l'erreur de prédiction et la MAP. Des valeurs plus fortes d'erreur de prédiction entre un score A et un score B indiquent que l'algorithme a plus de difficultés à apprendre un modèle pour le score B, donc que ce score est moins adapté. De façon similaire, une diminution de la MAP entre un score A et un score B indiquera que le score B est moins adapté.

8.2. *Résultats*

Dans cette section, nous présentons les résultats de l'analyse de l'influence de chaque type action sur l'erreur de prédiction. L'analyse de l'influence de chaque type d'action sur la MAP donne des résultats identiques, nous ne les présentons donc pas ici.

Nous avions constaté à la section 6 que l'échantillon 1 avait un comportement atypique pour le score α_i = 1 pour tout *i*, avec des valeurs d'erreur beaucoup plus

élevées que celles des autres échantillons. Ce comportement est vérifié pour l'ensemble des scores observés.

Nous constatons que l'erreur de prédiction sur l'échantillon test est la plus forte en moyenne pour le score $\alpha_{réservation} = 0$, c'est-à-dire lorsque les clics sur le bouton « Réserver » ne sont pas pris en compte. L'algorithme d'apprentissage a plus de difficultés à prédire correctement la pertinence dans ce cas. Les clics sur le lien de réservation jouent donc un rôle important pour la modélisation de la pertinence. D'un point de vue intuitif, cela paraît logique. Un utilisateur clique sur le bouton « Réserver » s'il envisage d'effectuer une réservation, donc si le lieu lui convient et répond à son besoin. Il s'agit donc d'un indicateur fort de satisfaction de l'utilisateur pour le résultat proposé. Par opposition, le résultat sur le score $\alpha_{téléphone} = 0$ est surprenant. L'action téléphone indique que l'utilisateur a cliqué sur le lien affichant le numéro de téléphone, donc qu'il cherche à contacter le lieu. Celui-ci semble donc répondre à son besoin. L'action téléphone paraît très similaire à l'action réservation. Nous pensions observer des résultats similaires pour les scores $\alpha_{réservation} = 0$ et $\alpha_{téléphone} = 0$. Or, nous constatons que l'erreur de prédiction est la plus faible lorsque l'action téléphone n'est pas prise en compte. Les clics sur le numéro de téléphone semblent donc introduire du bruit dans la modélisation de la pertinence. D'autres analyses sont en cours afin d'expliquer cette observation surprenante, mais nous n'avons pas à ce jour de résultats concluants. Nous constatons également que l'erreur de prédiction est plus forte lorsque les clics sur la carte sont négligés que lorsque les clics sur le titre sont négligés. Les clics sur la carte semblent avoir un impact plus fort sur la pertinence.

En conclusion, les actions peuvent être réparties en trois catégories de pertinence :

– Pertinence faible : clics sur le site Internet et sur le titre

– Pertinence moyenne : clics sur la carte

– Pertinence forte : clics sur le lien de réservation

L'action téléphone ne permet pas de dire si le résultat est pertinent. Elle semble même dégrader l'estimation de la pertinence.

Score / Echantillon	α_i=1 pour tout i	$\alpha_{réservation}$ = 0	$\alpha_{téléphone}$ = 0	α_{web} = 0	α_{carte} = 0	α_{titre} = 0
1	42,8	42,6	44,6	42	42,5	50,8
2	32	46,4	37,6	37	37,6	42,1
3	38,4	42,6	38,3	36,1	41,7	38
4	42,4	44,5	33,8	42,7	44,6	40,1
5	37,3	44,5	32,2	40,7	41,6	37,6
Moyenne	38,6	44,1	37,3	39,7	41,6	41,7
Etendue	10,8	3,8	12,4	6,6	7	13,2
Moyenne corrigée	37,5	44,5	35,5	39,1	41,4	39,5
Etendue corrigée	10,4	3,8	6,1	6,6	7	4,5

Tableau 4. ***Erreur de prédiction moyenne et par échantillon pour l'ensemble des scores***

9. Discussion

Nous avons proposé un modèle permettant de modéliser la pertinence à partir des clics utilisateurs dans le cas particulier où plusieurs types de clics sont possibles sur un même document. Nous avons montré que le score $\alpha_i = 1$ pour tout *i* donne des résultats cohérents et peut être utilisé comme première approche pour construire des collections d'apprentissage géoréférencées. Nous avons montré que les différents types de clics avaient une influence sur la qualité de l'apprentissage et donc sur la qualité du score. Nous avons mis en évidence les clics sur le bouton « Réserver » ou sur la carte traduisent une pertinence plus forte. Nous avons constaté que les clics sur le téléphone introduisent du bruit dans l'apprentissage. Nous proposons d'utiliser ces résultats afin de définir de nouvelles pondérations qui permettront de mieux modéliser la pertinence et de créer des collections de référence de meilleure qualité. Nous avons constaté que des échantillons présentaient des valeurs atypiques de l'erreur de prédiction et de la MAP. De premières analyses sur les pondérations semblent indiquer que la qualité d'apprentissage peut dépendre de l'échantillon.

Ces expériences ont été réalisées sur un faible volume de données afin de valider l'approche méthodologique envisagée. Une campagne de plus grande envergure utilisant les ressources du centre de calcul CalMip va être menée prochainement. Cette nouvelle étude sur des données volumineuses permettra d'analyser des scores prenant en compte l'ordre d'importance des clics. Des expériences auprès d'utilisateurs réels sont envisagées afin de comparer l'ordre d'importance déduit des expériences et le ressenti des utilisateurs. Nous effectuerons une analyse de la composition des échantillons, afin d'étudier l'influence de la proportion des types d'actions et de requêtes sur la qualité de l'apprentissage. Des travaux dans le domaine de la recommandation de pages web et de la création de profils d'utilisateur

se sont également intéressés à l'évaluation de l'intérêt utilisateur à partir de suivi de traces (Chan, 2000) (Esslimani, 2011) et ont proposé des modélisations similaires. Il sera intéressant d'évaluer si les modèles proposés sont applicables dans notre cadre.

Nous remercions les rapporteurs de cet article pour l'intérêt qu'ils ont porté à nos travaux, ainsi que la région Midi-Pyrénées, partenaire de ce projet via le financement 10009108.

10. Références

Burges C.J.C., Shaked T., Renshaw E., Lazier A., Deeds M., Hamilton N., Hullender G., « Learning to rank using gradient descent », *Proceedings of the 22nd International Conference on Machine Learning (ICML)*, 2005, p. 89-96.

Cao Z., Qin T., Liu T.Y., Tsai M.F., Li H., « Learning to rank : From pairwise approach to listwise approach », *Proceedings of the 24th International Conference on Machine Learning (ICML)*, 2007, p. 129-136.

Chan P., « Constructing web search profiles : a non-invasive learning approach», *Lecture notes in Computer Science*, 2000, vol. 1836/2000, p. 39-55.

Chapelle O., Zhang Y., « A dynamic bayesian network click model for web search ranking », *Proceedings of the 18th International Conference on World Wide Web WWW 2009,* 2009, p.1-10.

Cleverdon C.W., « The significance of the Cranfield test on index languages », *Proceedings of the 14th International ACM SIGIR Conference on Research and Development in Information Retrieval,* 1991, p. 3-13.

Cossock D., Zhang T., « Subset ranking using regression », *Proceedings of the 19th Conference on Learning Theory (COLT)*, 2006, p. 605-619.

Crasswell N., Zoeter O., Taylor M., Ramsey B., « An experimental comparison of click position-bias models », *Proceedings of the 1st International Conference on Web Search and Data Mining (WSDM),* 2008, p. 87-94.

Esslimani I., Vers une approche comportementale de recommandation : apport de l'analyse des usages dans un processus de personnalisation, Thèse de doctorat, Université de Nancy 2, 2010.

Guo F., Liu C., Wang Y.M., « Efficient multiple-clicks models in web search », *Proceedings of the 1st International Conference of Web Search and Data Mining (WSDM),* 2008, p. 124-131.

Joachims T., « Making large-scale SVM learning practical », In *Advances in kernel methods : support vector learning*, MIT Press, 1999.

Joachims T., « Optimizing search engines using clickthough data », *Proceedings of the 8th ACM Conference on Knowledge Discovery and Data Mining (KDD),* 2002, p. 133-142.

Joachims T., Granka L., Pan B., Hembrroke H., Gay G., « Accurately interpreting clickthrough data as implicit feedback », *Proceedings of the 28th annual international*

SIGIR Conference on Research and Development in Information Retrieval, 2005, p. 154-161.

Kumar C., « Relevance and ranking in geographic information retrieval », *Proceedings of the 4th Symposium on Future Directions in Information Access,* 2011.

Liu C., Faloutsos C., « BBM : Bayesian browsing model for petabyte-scale data », *Proceedings of the 15th ACM SIGKDD International Conference on Knowedge Discovery and Data Mining (KDD),* 2009, p. 537-546.

Liu T.Y., *Learning to Rank for Information Retrieval*, Springer, 2011.

Mandl T., Gey F., DiNunzio G., Ferro N., Larson R., Sanderson M., Santos D., Wonser-Hacker C. and Xie X., « GeoCLEF 2007 : the CLEF 2007 cross-language geographic information track overview », *Advances in Multilingual and Multimodal Information Retrieval*, 2007.

Martins B., Calado P., « Learning to rank for geographic information retrieval », *Proceedings on the 6th workshop on Geographic Information Retrieval*, 2010, p. 21.1-21.8.

Nallapati R., « Discriminative models for information retrieval », *Proceedings on the 27th annual international SIGIR Conference on Research and Development in Information Retrieval*, 2004, p. 64-71.

Radlinski F., Joachims T., « Query chains : Learning to rank from implicit feedback », *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD*), 2005, p. 239-248.

Spink A., Jansen J., Wolfram D., Saravecic T., « From e-sex to e-commerces : web search changes », *Computer*, vol. 35, n° 3, 2002, p. 107-109.

Yue Y., Finley T., Radlinski F., Joachims T., « A support vector method for optimizing average precision », *Proceedings of the 30th annual international SIGIR Conference on Research and Development in Information Retrieval,* 2007, p. 271-278.