MBOI: Un outil pour la veille d'opportunités sur l'Internet

François PARADIS, Quing MA, Jian-Yun NIE (*)
Stéphane VAUCHER, Jean-François GARNEAU, Robert GÉRIN-LAJOIE (**)
Arman TAJAROBI (***)

{paradifr,maqing,nie}@iro.umontreal.ca
{vauchers,garneauj,rgl}@cirano.qc.ca
arman.tajarobi@nstein.com

(*) Université de Montréal, Département d'Informatique et Recherche Opérationnelle, C.P. 6128, succursale Centre-ville, Local 2241, Montréal, Québec, Canada
(**) CIRANO, 2020 University , 25th Floor, Montréal, Québec, Canada
(***) Nstein Technologies, 75, Queen Street, Suite 4400, Montréal, Québec, Canada

Mots clés :
Veille d'opportunités d'affaires, Modèle XML, Extraction d'information, Classification

Keywords :
Business watch, XML model, Information extraction, Classification

Résumé :
Nous présentons un outil pour assister la veille d'opportunités d'affaires, basé sur la collecte et l'extraction d'information provenant de sites Web. Nous définissons d'abord un modèle XML pour représenter les appels d'offres et leur inférence à partir des documents Web. Nous présentons ensuite les grandes étapes de notre système, en insistant particulièrement sur l'extraction d'information et la classification. Nous mesurons et analysons la performance de ces deux tâches sur nos documents. Nous présentons enfin l'interface et les fonctionnalités de notre outil, à travers son application dans des environnements de veille, et montrons comment l'extraction et la classification peuvent aider à la construction de requêtes. L'expérience avec ces veilleurs nous permet de conclure que notre système a bien rempli son but premier: celui de faciliter l'accès aux appels d'offres.

1 Introduction

La veille d'opportunités d'affaires est une activité cruciale pour les entreprises, mais elles ont souvent peu de ressources à y consacrer. Plusieurs sites d'appels d'offres existent maintenant sur le Web pour faciliter leur tâche. Ces sites peuvent recevoir les appels d'offre directement des organismes émetteurs (comme par exemple dans le cas de TED[1]), ou par agrégation d'autres sites d'appels d'offres (par exemple, SourceCan[2]). Bien que l'approche centralisatrice permette de contrôler le contenu et la richesse de l'information, elle est difficile d'application dans certains domaines où il n'y a pas d'autorité reconnue, et souvent limitée à une zone géographique. De plus, l'information complémentaire aux appels d'offres qui pourrait exister sur le Web est ignorée. Par contre, en procédant par agrégation, il est difficile d'extraire les information pertinentes, puisque les documents ne suivent pas un format standard. On s'expose de plus à divers problèmes tels que les changements apportés aux sites, la reconnaissance de doublons, etc.

Dans le projet MBOI (*Matching Business Opportunities on the Internet*), nous proposons une approche d'agrégation, reposant sur un modèle de représentation nous permettant de synthétiser un appel d'offres à partir de plusieurs documents trouvés sur les sites. Nous employons également des techniques d'extraction d'information pour retrouver les caractéristiques des appels d'offres contenus dans les documents. Enfin nous offrons une interface de recherche et de navigation adaptée aux besoins de la veille.

Dans cet article, nous présentons ces trois aspects: le modèle de représentation, l'extraction d'information, et l'interface du programme de veille à travers quelques applications industrielles.

2 Un modèle pour les opportunités d'affaire

Le modèle que nous avons défini a pour but de représenter l'information relative aux opportunités d'affaires. Cette information provient de différents types de documents: communiqués de presse, avis d'appels d'offres, contrats adjugés, rapports trimestriels, etc.

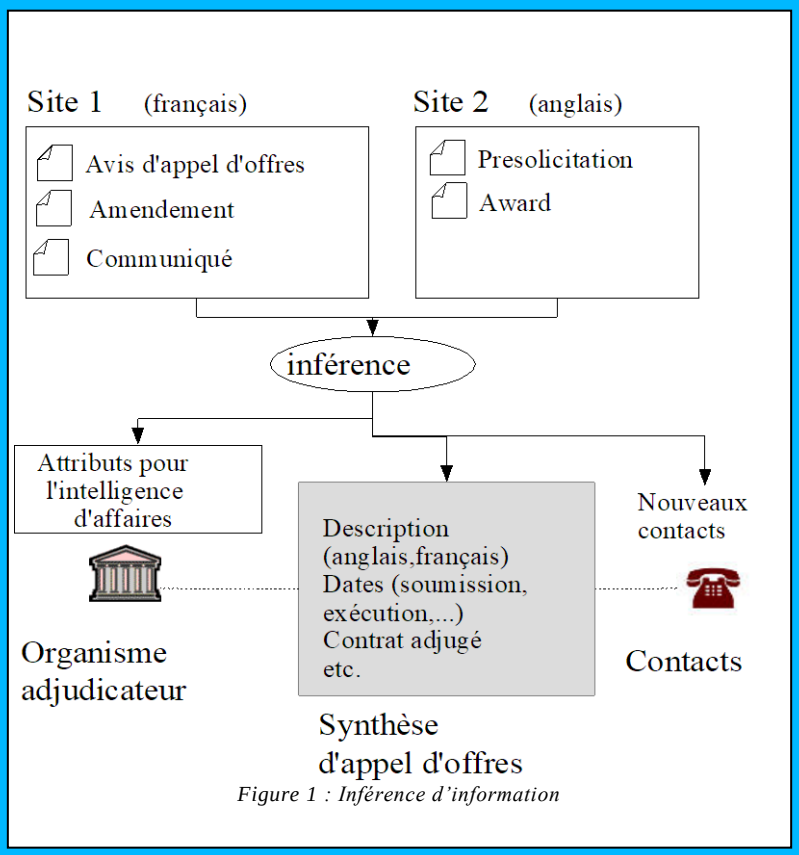

Figure 1 : Inférence d'information

[1]http://ted.publications.eu.int/
[2]https://www.sourcecan.com/

La figure 1 schématise le processus d'inférence de l'information. Au coeur du modèle se trouve la *synthèse d'appel d'offres*, qui combine l'évidence provenant de divers sites. Ainsi, en supposant que deux sites contiennent respectivement une version française et anglaise d'un même appel d'offres, la synthèse contiendra un titre et une description dans les deux langues. D'autres caractéristiques tels que les dates de soumission et d'exécution, la classification selon un code d'industrie, la procédure de soumission, etc. seront aussi déduites à partir des avis d'appels d'offres. Des amendements peuvent remplacer ou compléter des éléments de la synthèse. Des information connexes peuvent aussi venir s'ajouter aux connaissances sur les organismes adjudicateurs et leurs contacts, et pourraient plus tard être utilisées pour l'intelligence d'affaires. Liée au processus d'inférence, une mesure de confiance permet d'exprimer l'incertitude.

Nous avons choisi de représenter notre modèle dans le format XML. Il existe déjà des normes XML dans le domaine des affaires: notamment xCBL (*Common Business Language*) et plus récemment OASIS UBL (*Universal Business Language*) [Dum01]. Cependant ces normes visent surtout l'échange des données entre les organisations plutôt que les opportunités d'affaires. Plusieurs sites ont aussi défini leur propre format de données. Nous avons examiné deux de ces formats: l'un défini par l'OJEC (Journal officiel de l'Union Européenne) pour le site TED, et l'autre utilisé par le site FedBizOpps[3] (Opportunités d'affaires du gouvernement américain). Ces deux formats sont à l'opposé l'un de l'autre, puisque OJEC est très détaillé, et FedBizOpps plus minimaliste. Nous avons adopté une approche plus flexible, en permettant de représenter les éléments qui nous intéressent dans le détail, mais en spécifiant peu d'éléments obligatoires ou de règles de composition. Lorsque cela était possible, nous avons réutilisé les conventions xCBL.

La figure 2 montre la représentation abrégée d'une synthèse d'appel d'offres: il s'agit d'un contrat pour les fournitures de bureau du gouvernement de la Saskatchewan. L'appel d'offres a été inféré à partir de deux documents (représentés par des éléments `published-document`s): un avis d'appel d'offres trouvé sur le site Merx, ainsi que son amendement. L'appel d'offres est identifié par le gouvernement de la Saskatchewan par le numéro « 031021-5 » (élément `contracting-authority`...), qui est différent de l'identificateur donné par Merx, « CFAB4 » (élément `publisher-solicitation-id`). Il est classé dans le système NAICS sous le code « 418210 ». Un même appel d'offre pourrait être classé selon plusieurs codes ou même plusieurs systèmes.

La plupart du contenu textuel de l'appel d'offres se trouve dans le titre et la description: ici on a un titre anglais et français parce que l'avis d'appel d'offres publié sur Merx était bilingue. La date de clôture des soumissions (élément `date-closing`) apparaissait, elle, dans l'amendement à l'appel d'offres. Il y a deux dates d'exécution, chacune avec une mesure de confiance, parce qu'il y avait ambiguïté lors de l'extraction de cette information. D'autres informations, telles que la date de début de soumission, la valeur du contrat, etc. ont été omises de l'exemple.

On a aussi inclus dans cette représentation les informations connexes concernant l'organisme adjudicateur et les contacts.

3 Indexation et Recherche

La figure 3 présente l'architecture de notre système et ses deux grandes composantes: l'*indexation*, c'est-à-dire la collecte, l'inférence et la création d'index, et la *recherche / navigation*, qui est l'interface pour le veilleur. La construction de synthèses d'appels d'offres telles que décrites à la section précédente, a lieu lors de l'*inférence*. Cette tâche est divisée en trois: la reconnaissance des sections, l'extraction d'entités nommées, et la classification. Ces trois étapes peuvent chacune conduire à l'inférence d'informations, indépendamment l'une de l'autre. On note cependant un lien entre la reconnaissance des sections et l'extraction d'entités nommées: afin d'améliorer les résultats de cette dernière on utilise aussi les sections préalablement identifiées[4].

---

[3] http://www.fedbizopps.gov/

[4] Il serait aussi concevable d'utiliser cette information pour la classification.

```
<call-for-tender>
   <contracting-authority-solicitation-id>
      031021-5</...>
<title xml:lang="en">Office Supplies</title>
   <title xml:lang="fr">Fournitures de
       Bureau</...>
   <description xml:lang="en">A supplier is
      needed for...</description>
   <date-closing>2003-10-28</date-closing>
   <execution-date-start confidence="0.7">
      2003-11-05</...>
   <execution-date-start confidence="0.8">
      2003-12-05</...>

   <classification>
   <classification-system>NAICS</...>
   <code>418210</code>
   </classification>

   <published-document>
     <publisher-id>Merx</publisher-id>
     <publisher-solicitation-id>CFAB4</...>
     <original-url>...</original-url>
     <cached-url>...</cached-url>
     <date-published >2003-10-08</...>
     <date-cached >2003-10-09</date-cached>
     <language>eng</language>
     <format>text/html</format>
     <document-type>presol</document-type>
   </published-document>
   <published-document>...
     <document-type>amendment</document-type>
   </published-document>

   <contracting-authority>
   <name>Saskatchewan Government</name>
   <authority-type>provincial government</...>
   </contracting-authority>
   <contact><name>Bernie</name>
   <surname>Juneau</surname>
   <phone>(306) 381-1542</phone>
   </contact>
</call-for-tender>
```

Figure 2: Une synthèse d'appel d'offres

Nous décrivons maintenant chacune des étapes illustrée dans le diagramme.

3.1 Collecte de documents

Cette étape est réalisée par un robot-collecteur qui cherche des pages suivants certains patrons sur des sites Web. La liste des sites consultés et des patrons dépend de l'application. Le robot peut aussi se connecter aux sites sous un nom d'utilisateur (pour les sites à accès restreint), remplir les formulaires et suivre les liens au besoin.

3.2 Reconnaissance des sections

L'idée est d'exploiter les structures régulières dans les avis d'appel d'offres, c'est-à-dire le fait que ces documents contiennent souvent des marqueurs explicites de leur contenu, par exemple: « Date de soumission: 30 avril 2004 », « Numéro: 03021-5 », « Personne Contact: Bernie Juneau », etc. Cette

hypothèse est d'autant plus probable dans le cas des sites centralisés, où les pages Web sont générées automatiquement à partir de bases de données. Même quand ce n'est pas le cas, les appels d'offres sont souvent rédigés selon un modèle défini par le pouvoir adjudicateur.

Puisque les marqueurs sont relativement simples, nous cherchons les sections à l'aide d'*expressions régulières* (*regular expressions*), c'est-à-dire de patrons qui décrivent des chaînes de caractères à retrouver. Ces expressions peuvent référer soit uniquement au texte comme dans les exemples ci-dessus, ou combiner le texte avec des balises HTML ou XML, comme par exemple: « NAICS: .* », qui retrouverait des paragraphes commençant par « NAICS: ». Les sections retrouvées peuvent être directement des informations à extraire, comme par exemple un titre, une date, ou pour l'exemple de NAICS, un code de classification. Elles peuvent aussi permettre de retrouver le contenu pour l'étape d'extraction suivante.

Un problème de l'approche par expressions régulières est qu'elle est très sensible aux changements de forme ou de présentation sur le site, qui causent des « pannes ». Au cours de la dernière année et demie, nous avons téléchargé quotidiennement 40 sites Web contenant des avis d'appels d'offres. Durant cette période, nous avons observé 18 pannes.

Pour résoudre ce problème, certains ont proposés des techniques de vérification automatique [Kush00] ou d'induction de règles [Kush97]. Nous prévoyons examiner ces approches dans le futur.

3.3 Extraction d'entités

Type d'entité	*Préc.*	*Rappel*	*F1*
date	.679	.977	.801
lieu	.923	.724	.811
valeur monétaire	.923	.923	.923
organisme	.362	.811	.501
personne	.141	.833	.241

Table 1: Extraction d'entités nommées sur les documents FBO

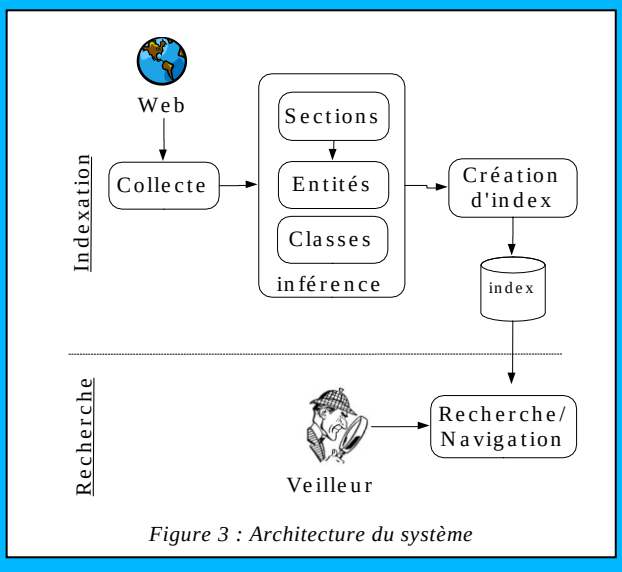

Figure 3 : Architecture du système

nommées

On appelle *entités nommées* des expressions qui contiennent les noms de personnes, organismes, lieux, temps ou quantités. Dans notre contexte, ces entités peuvent nous permettre de retrouver les dates de fermeture ou d'exécution, les lieux d'exécution, les contacts, les organismes adjudicateurs, les montants de contrats, etc.

L'extraction d'entités nommées procède généralement par une analyse syntaxique ou lexicale des mots ou des relations entre les mots, possiblement avec l'aide d'un dictionnaire. Contrairement à

l'identification de sections, qui donne des résultats quasi-parfaits mais qui est coûteuse à maintenir, ce type d'approche donne des résultats moins élevés. Par contre elle n'a pas à être adaptée à chaque site (bien qu'en pratique elle bénéficiera beaucoup de règles spécifiques au domaine). Dans notre système les deux approches peuvent être combinées, puisque l'extraction d'entités peut s'effectuer sur des sections identifiées. Par exemple, on peut avoir identifié la section de description ou sommaire au travers des annonces, liens, etc. qui apparaissaient sur la page Web. En ciblant le texte, on améliore les résultats d'extraction.

Nous avons mesuré l'efficacité de cette approche sur un petit sous-ensemble de 40 documents provenants d'un des sites téléchargés, le FedBizzOpps (FBO). Nous avons manuellement annoté les dates, lieux, valeurs monétaires, organismes et personnes dans ces 40 documents, et comparé avec les résultats d'extraction automatique. En pratique, notre système utilise *Nfinder* (développé par Nstein) pour l'extraction d'informations. Nous présentons ici les résultats d'expérimentation obtenus avec un autre système, *Gate/Annie* [Cun02], dans un but de comparaison avec l'état de l'art. Les résultats des deux systèmes sont comparables.

La table 1 résume les résultats par type d'entité. La précision représente le taux d'entités correctes parmi l'ensemble des entités extraites. Par exemple, pour « date », 67.9% des dates extraites étaient correctes, ou en d'autres termes, le système n'aurait pas dû extraire 32.1% des dates retournées. Le rappel, lui, mesure la complétude des résultats, ou le taux d'entités correctes extraites par rapport au total des entités correctes dans tous les documents. Ainsi, pour « date », le système a extrait 97.7% des dates existantes. La mesure F1, quant à elle, est une façon courante de combiner rappel et précision afin de faciliter la comparaison. Elle est égale à: 2*précision*rappel/(précision+rappel). Plus cette mesure est élevée, meilleurs sont les résultats.

Tel que prévu, l'extraction de valeurs monétaires, une tâche assez simple, a obtenu les meilleurs résultats, alors que l'extraction d'organismes et de personnes ont moins bien performé. Les valeurs obtenues sont considérablement plus basses que celles rapportées dans [Man01], qui a aussi extrait ces entités en utilisant Gate/Annie. Ceci est partiellement expliqué par la petite taille de notre collection, ainsi par certaines particularités des documents, pour lesquelles nous n'avons pas ajusté les règles d'extraction. Certaines de ces particularités sont discutées ci-dessous.

Pour l'extraction de dates, les faux positifs (i.e. incorrectement identifiées comme dates) ont causé un bas taux de précision. Plusieurs codes de classification SIC (*Standard Industrial Classification*) à quatre chiffres furent identifiés comme des années. Un problème similaire a affecté les organismes, où les acronymes ont aussi donné plusieurs faux positifs (e.g. « *frequency domain helicopter electromagnetic* (HEM) ») et les personnes, où une séquence de deux mots commençants par des majuscules était extraite si le premier mot était connu comme un prénom (e.g. « *Space Flight* » ou « *Will Result* »). Enfin, pour les lieux, les abréviations d'états américains n'étaient pas reconnues (e.g. « TX », « FL ») ce qui a causé un rappel plus bas.

L'extraction de dates et de lieux pourrait facilement être améliorée par l'addition de quelques règles couvrant les cas mentionnés ci-dessus. Il serait également possible d'ajouter des caractéristiques aux entités extraites, de façon à différencier par exemple les dates de soumission et d'exécution. Ceci pourrait être réalisé en recherchant des expressions régulières signalant ces caractéristiques, de façon similaire à la reconnaissance de sections.

3.4 Classification

Les avis d'appels d'offres sont souvent classés par type d'industrie, selon les nombreuses normes en vigueur: SIC (*Standard Industrial Classification*), NAICS (*North American Industry Classification System*), FCS (*Federal Supply Codes*), CPV (*Common Procurement Vocabulary*), etc. Ces codes de classification ne sont pas toujours extraits lors de la reconnaissance des sections, et même lorsqu'ils le sont, il est intéressant de classer le même appel d'offre selon d'autres normes. Par exemple, un veilleur américain sera sûrement familier avec la norme NAICS, mais peut-être pas avec la norme CPV en vigueur dans l'Union Européenne. De même, ces normes sont régulièrement mises à jour, et la différence de codes entre deux versions peut être la source d'erreurs. On peut donc rendre ces conversions explicites en classant les appels d'offres selon plusieurs normes.

La performance des méthodes de classification varie selon le type des données. Nous présentons donc dans cette section les résultats de différentes méthodes sur nos données. Afin de nous constituer

une collection test, nous avons considéré encore une fois les documents provenant de FedBizzOpps. Nous nous sommes limités aux documents qui contenaient deux codes de classification: FCS et NAICS (ceci afin de pouvoir mesurer plus tard la conversion de codes en utilisant la même base de documents). Nous avons ainsi obtenu 21945 appels d'offres, répartis dans la période de septembre 2000 à octobre 2003. Nous avons scindé cette collection en deux: 60% pour l'entraînement des méthodes de classification, et 40% pour les tests.

Les codes NAICS sont hiérarchiques: chacun des six chiffres formant le code représente un niveau de la hiérarchie. Ainsi, pour l'exemple de la figure 1, le code de classe d'industrie est « 418210 » (grossistes-distributeurs de papeterie et de fournitures de bureau), et le code de secteur, « 418 » (grossistes-distributeurs de produits divers). Les trois pays participants à la norme, le Canada, le Mexique et les États-Unis, ont chacun leur propre version de la norme, qui en principe différent surtout au niveau des classes d'industries (i.e. 5 ou 6 chiffres). Il y a cependant des exceptions, comme le montre notre exemple: la classe équivalente dans la version américaine serait « 424120 » (grossistes-marchands de papeterie et de fournitures de bureau) et le secteur, « 424 » (grossistes-marchands de biens non durables).

Nous avons réduit l'espace des catégories en ne retenant que les trois premiers chiffres, c'est-à-dire le secteur. Nous avons ainsi obtenu 92 catégories NAICS dans notre collection (vs. 101 pour les codes FCS). Nous n'avons pas normalisé pour la distribution inégale de ces catégories. Ainsi pour NAICS, 34% des documents sont dans les deux classes les plus courantes, alors que pour FCS, 33% se retrouvent dans les cinq premières classes.

Nous avons testé les méthodes de classification suivantes: KNN (*K-Nearest Neighbours*) [Au00], Bayes (*Naive Bayes*) [Ren03], SVM (*Support Vector Machines*) [Joa98] et Max (*Maximum Entropy*) [Nig99]. Le but de cette première expérience était de se faire une idée du comportement des données avec ces méthodes bien connues. C'est pourquoi, encore une fois, nous présentons les résultats obtenus avec un programme de l'état de l'art, *rainbow* [MC96], plutôt qu'avec l'outil de Nstein, Ncategorizer, qui implémente son propre algorithme de classification. De plus nous n'avons pas essayé d'optimiser les paramètres de ces algorithmes, de même que la sélection de termes (*feature selection*), qui permet de filtrer les termes des documents.[5] Il est bien connu que certains de ces algorithmes, notamment KNN, sont très sensibles à la sélection, tandis que d'autres, comme SVM, le sont beaucoup moins.

La table 2 montre les résultats de micro-F1 pour la classification des codes NAICS. En comparant ces résultats avec ceux publiés dans [Yan99a] sur la collection *Reuters* avec les mêmes algorithmes, on constate que la méthode SVM donne les meilleurs résultats dans les deux cas. Dans le cas de Reuters, la méthode KNN a donné d'aussi bons résultats, alors que sur FBO elle s'est classée dernière. Ceci peut partiellement s'expliquer par la différence entre ces collections, de même que par le fait que la sélection de termes a été l'objet de plusieurs expériences sur Reuters.

méthode	***titre & desc.***	***titre***	***desc.***
SVM	**.6445**	.5200	.6121
Maxent	**.5947**	.5569	.5766
Bayes	**.5110**	.5104	.4854
KNN (k=30, Chi-Square)	**.5007**	.4366	.4700

Table 2: Classification des codes NAICS sur la collection FBO (mesure micro-F1)

La table 2 montre aussi trois variantes de la collection, selon que nous ayons ou non inclus le titre et la description, ceci dans le but de voir si l'une ou l'autre de ces sections est plus porteuse d'information. Bien que l'on ne puisse déduire que le titre (ou la description) seul doive être utilisé(e), il est intéressant qu'il n'a pas produit une baisse si dramatique, bien qu'il ait simplifié la complexité de la classification par plusieurs ordres de magnitude. Ceci tend à confirmer le fait que la description n'est pas toujours très porteuse d'information dans les avis d'appels d'offres. Nous avons remarqué que plusieurs documents FBO utilisent une large portion de l'avis d'appel d'offres pour décrire la procédure plutôt que l'objet de la soumission. Ces descriptions procédurales ne sont d'aucune utilité pour la

[5] Ceci à l'exception de KNN, pour lequel nous avons appliqué d'emblée la sélection par *Chi-Square* (8000 termes).

classification, et auraient sans doute avantage à être filtrées. Nous examinons présentement cette question.

Des résultats préliminaires sur l'impact de la sélection de termes semblent indiquer que la sélection par *Information Gain* apporte une légère amélioration à la méthode Bayes (la micro-F1 passe de 0.5110 à 0.5310, soit une augmentation relative de 3.9%). Une formule similaire, dite du *Chi-Square*, a apporté quant à elle une augmentation spectaculaire à la classification de KNN (la micro-F1 passe alors de 0.2791 à 0.5007).

3.5 Création d'index et Recherche

La création d'index consiste à construire un index ou *fichier inverse*, qui donne, pour chaque *terme*, les documents où ce dernier apparaît, avec un *poids* qui représente son importance dans le document. La recherche utilise cet index afin de retourner une liste de documents pour une requête. Dans notre cas, les termes peuvent être des mots-clés, mais aussi des dates, codes de classification, etc. De plus afin de pouvoir retrouver un type d'information en particulier, l'index est organisé par *champs*, qui correspondent aux éléments définis dans le modèle. Par exemple, on a: titre, description, code (de classification), etc.

Pour ces deux étapes nous avons utilisé le moteur de recherche *lucene*. L'utilisateur ne voit bien entendu que l'interface de recherche, à laquelle se greffe des fonctionnalités de navigation. Nous discutons de cette interface à la section suivante.

4 Applications

Une première application de notre approche a été réalisée par l'entremise de SDTI (Société de Développement des Technologies de l'Information). SDTI avait pour mandat d'aider des entreprises de la région de Ste-Hyacinte (Québec) dans la recherche d'appels d'offres. Pour ce faire elle disposait de veilleurs, qui étaient chacun responsable d'un petit groupe d'entreprises participantes, et qui parcouraient régulièrement des sites Web pour des appels d'offres pertinents. Pour assister leur recherche sur ces sites, les veilleurs définissaient un vocabulaire décrivant les besoins de l'entreprise, qui pouvait aussi prendre en compte leur expertise et connaissance du domaine. Par exemple, pour une entreprise fabriquant des hottes, le vocabulaire pourrait aussi inclure des termes renvoyant à la construction ou à l'équipement de laboratoires chimiques, puisque ces derniers nécessiteront souvent ces équipements.

Notre système facilite la tâche du veilleur en lui présentant une liste d'appels d'offres pertinents pour une *requête* donnée, à partir d'informations colligées des sites Web. Il prévoit aussi une interface pour la construction et le raffinement des requêtes, où des termes sont proposés à partir des entités nommées et des codes de classification.

La figure 4 illustre le processus d'édition de requêtes, qui permet ici au veilleur de définir une requête concernant le déneigement. Les termes « snow » et « removal » sont recherchés; on pourrait aussi exiger qu'ils soient tous deux présents (opérateur de conjonction) ou rechercher le syntagme « snow removal ». [6] Les onglets à droite de la requête permettent le raffinement. Des concepts (i.e. sujets des appels d'offres), organismes, lieux et codes de classification sont présentés. Il s'agit d'informations extraites des documents retrouvés par la requête. Le veilleur peut alors ajouter un terme à la requête en le sélectionnant et en relançant la requête. Ainsi dans l'exemple de la figure 4, le concept « grounds maintenance » a été sélectionné. On peut aussi rechercher des appels d'offres ne contenant pas un terme donné, en le sélectionnant de nouveau (le crochet change alors pour une croix).

L'interface de requête permet également de sélectionner les *sources*, c'est-à-dire les sites d'où proviennent les appels d'offres, et de définir la portée selon les dates de téléchargement. Ceci est particulièrement utile pour les veilleurs, qui consultent le système de façon régulière, et qui ne veulent donc pas revoir des appels d'offres déjà consultés.

Les requêtes peuvent par ailleurs être sauvegardées et réutilisées.

La partie inférieure de l'écran affiche les résultats pour la requête en cours. Ces résultats sont présentés de façon très similaire à ceux d'un moteur de recherche Web, c'est-à-dire qu'on présente une liste d'appels d'offres classés par ordre de pertinence. On affiche pour chaque item un court résumé, où les termes de la requête sont surlignés, de même que certaines informations extraites: dans ce cas-ci,

[6] Les requêtes suivent la syntaxe de *lucene*, notre moteur de recherche, qui permet, en plus des opérateurs booléens, la recherche floue, par caractère de remplacement (*wildcard*), par proximités de termes, etc.

des dates de publication et de fermeture. Un lien permet de consulter l'avis d'appel d'offre original, tandis qu'un autre permet de commander l'appel d'offre officiel du pouvoir adjudicateur. Pour l'instant, ce dernier lien passe par notre entremise, mais il serait envisageable de renvoyer plutôt aux contacts ou instructions extraites.

Un arbre de navigation (à gauche des résultats) permet de parcourir les appels d'offre sans modifier la requête. Les rubriques sont composées des mêmes informations que pour le raffinement de requête (i.e. les concepts, lieux, organismes et codes de classification relatifs à la requête) et sont organisés en arborescence par un outil de Nstein, *Nretriever*. En sélectionnant un item, les documents correspondants s'affichent.

Notre système est facilement applicable à des secteurs ou industries spécifiques: il s'agit de définir les sites appropriés pour le robot collecteur, et d'adapter l'interface à l'application. Une de ces applications est le portail d'affaires de l'industrie du métal au Canada, NetMetal (http://www.netmetal.net/). Ici, plutôt que de demander aux veilleurs de définir un profil spécifique pour une entreprise, on offre plutôt des profils types (c'est-à-dire des requêtes) dans cette industrie. Ces profils ou requêtes peuvent être combinés pour exprimer un besoin plus complexe.

Notre système n'est pas non plus limité aux opportunités d'affaires, mais peut s'appliquer tout aussi bien à la veille stratégique, scientifique, commerciale, etc. Ainsi, nous réalisons présentement un système de veille d'informations pour le tourisme au Québec, en collaboration avec la chaire de tourisme de l'UQAM (Université du Québec à Montréal).

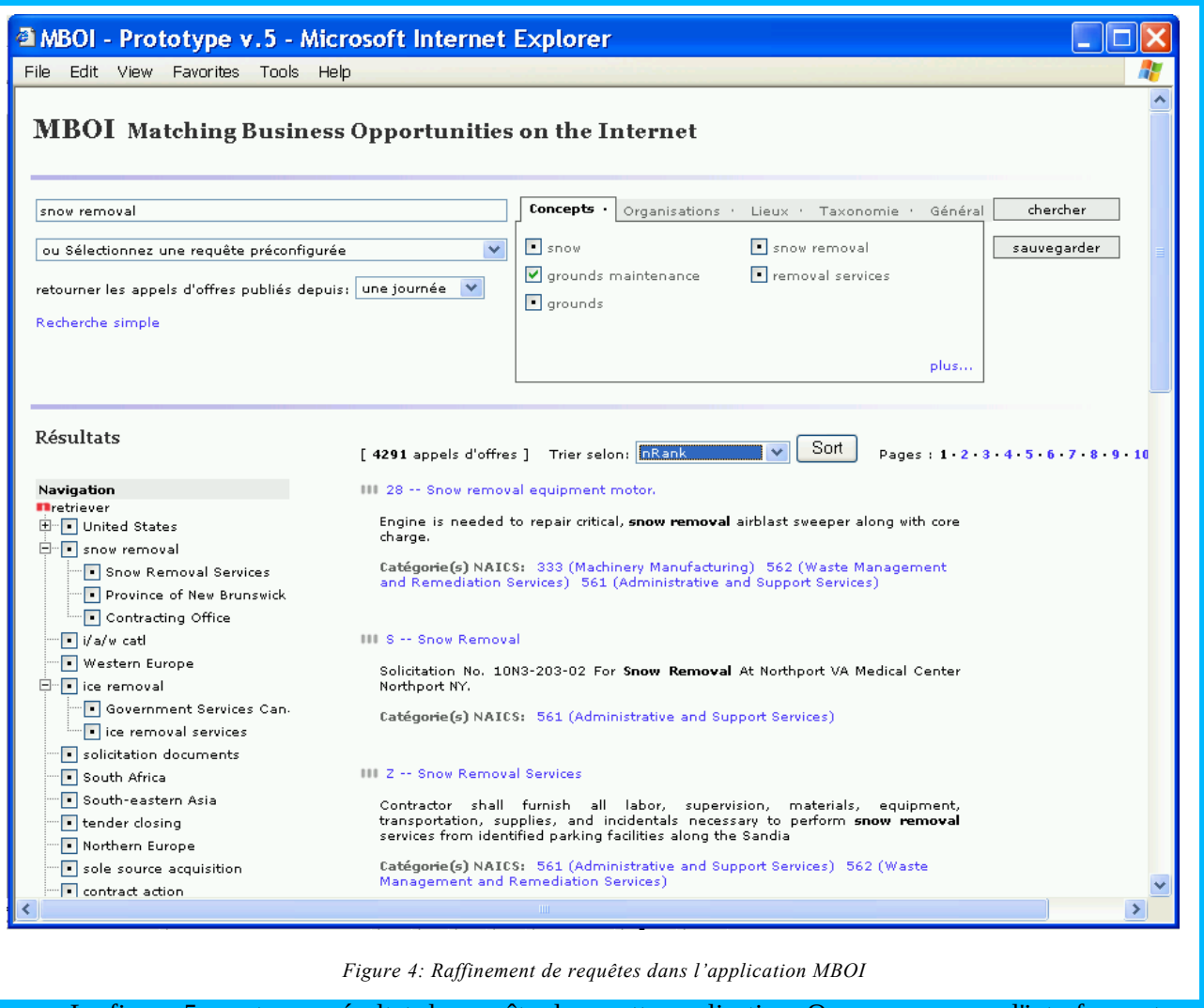

Figure 4: Raffinement de requêtes dans l'application MBOI

La figure 5 montre un résultat de requête dans cette application. On remarque que l'interface est légèrement différente. D'abord la date de téléchargement et la source de l'information sont mises en évidence, puisque dans cette application elles peuvent souvent déterminer à elles seules la pertinence du document pour le veilleur. On permet de plus au veilleur de sauvegarder les documents qui l'intéressent

dans des dossiers thématiques (dans l'exemple, le « marché américain »). Cela lui permet de se définir des zones de travail où il peut explorer différentes pistes de recherche, ce qui est particulièrement important en tourisme, où les besoins sont plus flous et changeants que pour les appels d'offres.

Figure 5: Sauvegarde de résultats (tourisme)

5 Conclusion

Dans le cadre du projet MBOI, nous avons défini une approche pour l'aide à la veille d'opportunités d'affaires sur le Web, et réalisé un outil pour les veilleurs. Cet outil a été appliqué dans plusieurs contextes depuis plus d'un an. Cette expérience nous a permis de confirmer le bien-fondé de notre approche, et de constater son impact positif dans les entreprises, en facilitant leur accès aux appels d'offres.

Ce travail nous a permis d'étudier les documents dans le domaine des affaires, et la performance de méthodes classiques d'extraction et de classification sur ces documents. Nous avons ainsi remarqué que les avis d'appels d'offres contiennent beaucoup de « bruit » comparativement aux collections standards. Le problème de la sélection de sections ou passages est donc plus important. Une autre caractéristique du domaine est la nature hiérarchique des codes de classification. Nous entendons exploiter cette hiérarchie dans des travaux futurs, notamment par le biais de méthodes de *shrinkage* [MC98], i.e. en utilisant le(s) parent(s) ou voisin(s) lors de la classification.

Dans le futur, nous souhaitons intégrer d'autres outils de Nstein au système. Ainsi, nos expériences préliminaires semblent indiquer que l'extraction de concepts réalisée par Nstein peut améliorer le résultats de classification et de recherche.

D'autres pistes de recherche sont la recherche d'information multilingue [Pet03], qui permettra la découverte d'appels d'offres dans d'autres langues, les robots fureteurs intelligents [Am97, Chau03] et le filtrage, qui est une problématique légèrement différente où l'on informe automatiquement les veilleurs de nouvelles opportunités à partir de leur profil.

6 Remerciements

Ce projet de recherche a été financé conjointement par Nstein Technologies et le CRSNG. Nous tenons de plus à remercier les personnes suivantes qui ont rendu possible ce projet: M. Claude Martineau de SDTI, M. Mario Girard, de Nstein Technologies, et MM. Jean-Marc Rousseau et Jacques Robert du Cirano.

7 Bibliographie

[Am97] Ambite, Jose Luis, Knoblock, Craig A., New Directions: Agents for Information Gathering, *IEEE Expert*, 12(5), pp2-4, September/October, 1997.

[Au00] Tom Ault, Yiming Yang, kNN at TREC-9, in *Ninth Text REtrieval Conference (TREC)*, Gaithersburg, Maryland, November 13-16, 2000.

[Chau03] Chau, Michael, Zeng, Daniel, Chen, Hsinchun, Huang, Michael, Hendriawan, David, Design and evaluation of a multi-agent collaborative Web mining system, *Decision Suport Systems*, 35, pp167-183, 2003.

[Cun02] H. Cunningham, D. Maynard, K. Bontcheva, V. Tablan. GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications, *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)*, Philadelphia, July 2002.

[Dum01] Dumbill, Edd, High Hopes for the Universal Business Language, *XML.com*, O'Reilly, November 07, 2001.

[Joa98] Thorsten Joachims, Text Categorization with Support Vector Machines: Learning with Many Relevant Features, in *ECML-98, 10th European Conference on Machine Learning*, 1998.

[Kush97] Kushmerick, Nicholas and Daniel S. Weld and Robert Doorenbos, Wrapper Induction for Information Extraction, *Proceedings of IJCAI-97*, 1997.

[Kush00] Nicholas Kushmerick, Wrapper Verification, *World Wide Web*, 3(2), pp79-94, 2000.

[Man01] D. Maynard, V. Tablan, C. Ursu, H. Cunningham, and Y. Wilks. Named Entity Recognition from Diverse Text Types. In *Recent Advances in Natural Language Processing*, pp 257-274, Tzigov Chark, Bulgaria, 2001.

[MC96] McCallum, Andrew Kachites, *Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering*, http://www.cs.cmu.edu/~mccallum/bow, 1996.

[MC98] Andrew McCallum, Ronald Rosenfeld, Tom Mitchell and Andrew Ng, Improving Text Classification by Shrinkage in a Hierarchy of Classes, in *ICML*, 1998.

[Nig99] Kamal Nigam, John Lafferty, Andrew McCallum, Using Maximum Entropy for Text Classification, *IJCAI'99 Workshop on Information Filtering*, 1999.

[Pet03] C. Peters, M. Braschler, J. Gonzalo and M. Kluck (ed), Advances in Cross-Language Information Retrieval Systems, CLEF 2002, *LNCS 2785*, Springer, 2003.

[Ren03] Jason D. M. Rennie, Lawrence Shih, Jaime Teevan and David R. Karger, Tackling the Poor Assumptions of Naive Bayes Text Classifiers, in Proceedings of the Twentieth International Conference on Machine Learning, 2003.

[Yan99a] Yiming Yang and Xin Liu, A Re-Examination of Text Categorization Methods, in *SIGIR '99: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Berkeley, CA, pp42-49, August 15-19, 1999.

[Yan99b] Yiming Yang, An evaluation of statistical approaches to text categorization, *Journal of Information Retrieval*, 1(1/2), pp67-88, 1999.

[Yan01] Yiming Yang, A Study on Thresholding Strategies for Text Categorization, in *Proceedings of SIGIR-01, 24th ACM International Conference on Research and Development in Information Retrieval*, 2001.