Utilisation des entités nommées et des variantes terminologiques dans un système de question-réponse

Olivier Ferret (1), Brigitte Grau (1), Martine Hurault-Plantet (1),
Gabriel Illouz (1) et Christian Jacquemin (1)

(1) LIMSI-CNRS, BP 133, 91403 Orsay cedex
{ferret, bg, mhp, gabrieli, jacquemin}@limsi.fr

Résumé – Abstract

Nous présentons dans cet article le système QALC qui a participé à la tâche Question Answering de la conférence d'évaluation TREC. Ce système repose sur un ensemble de modules de Traitement Automatique des Langues (TAL) intervenant essentiellement en aval d'un moteur de recherche opérant sur un vaste ensemble de documents : typage des questions, reconnaissance des entités nommées, extraction et reconnaissance de termes, simples et complexes, et de leurs variantes. Ces traitements permettent soit de mieux sélectionner ces documents, soit de décider quelles sont les phrases susceptibles de contenir la réponse à une question.

We developed a system, QALC, that participated to the Question Answering track of the TREC evaluation conference. QALC exploits an analysis of documents, selected by a search engine, based on the search for multi-words terms and their variations both to select a minimal number of documents to be processed and to give indices for comparing question and sentence representations. This comparison also takes advantage of a question analysis module and a recognition of numeric and named entities in the documents.

Mots-clés

Système de question-réponse, entité nommée, variante terminologique, recherche d'information

1 Introduction

L'introduction de la tâche "Question Answering" lors de la conférence d'évaluation TREC8 (Text REtrieval Conference), en 1999, est révélatrice du besoin de développer des systèmes capables d'apporter l'information attendue par un utilisateur, celle qui répond le mieux à sa requête. C'est ainsi que les nouveaux systèmes de recherche d'information ne doivent plus s'arrêter à la seule proposition de documents, mais doivent en extraire les parties pertinentes, soit en proposant la réponse, s'il s'agit d'une question d'ordre factuel, ou un résumé si la requête est d'ordre thématique. Dans un système de question-réponse, la recherche de documents pertinents est complétée par la sélection de courts extraits de texte contenant la réponse à la question posée. Cette sélection est opérée par un ensemble de modules de TAL, à la fois de nature syntaxique et sémantique, devant posséder une grande couverture linguistique et s'appliquer indépendamment du domaine abordé. La problématique "question-réponse" a été introduite dès la fin des années 70 lorsque Lehnert (1977) a jeté les bases d'un

système de question-réponse avec le système QUALM. Plus récemment, Molla et al (2000) ont proposé EXTRANS, un système de question-réponse sur le manuel Unix. Même dans ce domaine limité, les auteurs associent une approche fondée sur une analyse syntaxique et sémantique des questions et du manuel à une approche fondée sur des mots-clés de manière à rendre leur système plus robuste.

Dans cet article, nous présentons notre système de question-réponse QALC, conçu pour traiter des questions factuelles ou encyclopédiques portant sur n'importe quel domaine. QALC a participé aux évaluations TREC8 et TREC9. La campagne 1999 consistait à proposer 5 réponses ordonnées, à trouver dans un corpus de 500 000 documents environ, pour chacune des 200 questions posées. En 2000, la même tâche s'appliquait à environ 700 questions et 1 million de documents. Chaque réponse proposée devait être un très court extrait (250 caractères maximum) d'un document du corpus. Les documents sont des articles de journaux américains, tels que le *Wall Street Journal*, le *Los Angeles Times*, etc.

Après avoir présenté l'architecture générale de QALC, nous détaillerons ses différents modules : reconnaissance des entités nommées, des termes et de leurs variantes dans les documents ainsi que l'utilisation de ces informations dans la sélection des documents pertinents et dans le module d'appariement entre une question et les phrases candidates à la réponse. La présentation des résultats obtenus lors de la dernière évaluation TREC sera suivie d'une discussion sur des travaux connexes, puis nous conclurons sur les évolutions envisagées pour notre système.

2 Architecture du système

Les composants TAL du système QALC (cf. Figure 1) visent soit à déduire le type attendu de la réponse afin de sélectionner la réponse précise à l'intérieur d'un document, soit à enrichir la description des documents sélectionnés afin que la recherche de la réponse s'appuie sur des indices allant au delà des simples mots des documents.

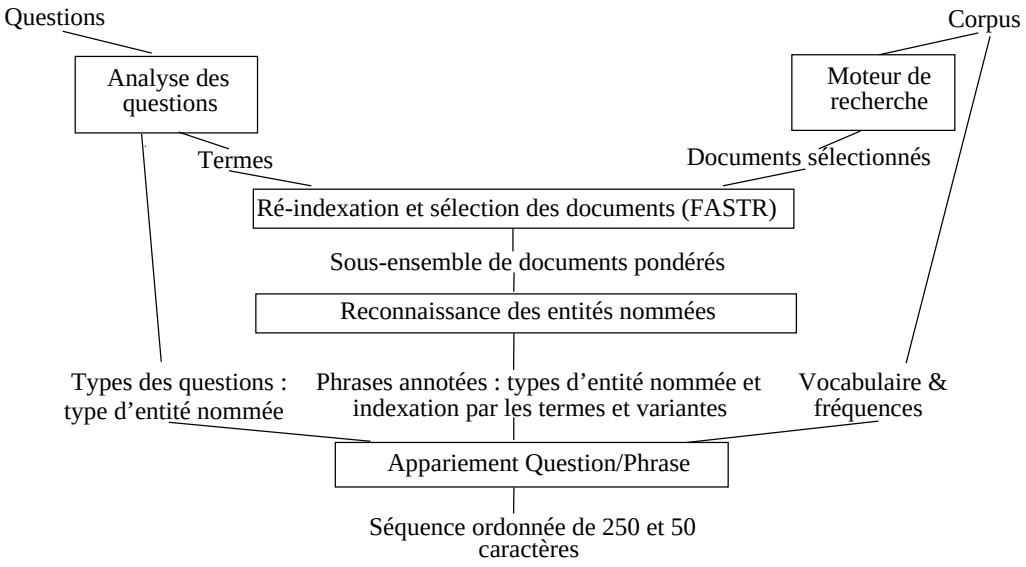


Figure 1. Architecture du système QALC

L'analyse des questions est réalisée par un analyseur partiel dédié qui attribue aux questions des catégories correspondant aux types d'entités nommées pouvant répondre à la question. La sélection d'un sous-ensemble de documents pertinents repose sur la reconnaissance des termes de la question ou de leurs variantes dans les documents sélectionnés par un moteur de

recherche classique. Cette reconnaissance est effectuée par FASTR (Jacquemin, 1999) sur la base des termes extraits de la question. Cette sélection revêt toute son importance lorsque le système applique les processus ultérieurs, à savoir la reconnaissance des entités nommées telles que les personnes, organisations, lieux et valeurs numériques, et la comparaison entre phrase et question, processus fortement consommateurs de temps de traitement.

Le dernier module, qui propose un ensemble limité de réponses à chaque question, met en œuvre un calcul de similarité entre une question, représentée par un vecteur contenant ses mots pleins lemmatisés, ses termes et le type attendu de sa réponse, et les phrases des documents retenus, représentées de manière analogue. Les réponses données par QALC sont des phrases, unités les plus significatives du point de vue de l'utilisateur final, mais peuvent être rendues plus concises grâce à un ensemble d'heuristiques permettant d'extraire de la phrase une réponse précise.

3 Les Entités Nommées

QALC utilise les entités nommées à la fois pour spécifier le type de la réponse attendue et pour détecter ensuite les entités de même type dans les documents afin d'aider à localiser la réponse.

3.1 Détermination du type de la réponse

Connaître le type de la réponse à une question permet au module d'appariement de privilégier, parmi plusieurs phrases candidates à la réponse, celles qui contiennent un groupe de mots qui correspond à ce type. L'analyse d'une question conduit à lui attribuer une étiquette s'identifiant au type de l'entité nommée qu'elle admet comme réponse. Par exemple :

Question : How many people live in the Falklands ? —> type = NUMBER
(Combien de personnes habitent les Falklands ?)
Réponse : Falkland population of <b_numex_TYPE=NUMBER> 2,100 <e_numex> is concentrated …
(La population des Falklands, de 2 100 habitants, est concentrée …)

Les étiquettes utilisées sont présentées Figure 2. Les étiquettes typant les réponses correspondent aux feuilles de l'arbre, auxquelles s'ajoutent les étiquettes *nomPropre* et *nombre* et sont similaires à ceux adoptés dans la tâche MUC (Grishman, Sundheim, 1995). Les étiquettes sont organisées en une hiérarchie afin d'avoir plus de souplesse dans la recherche de la réponse.

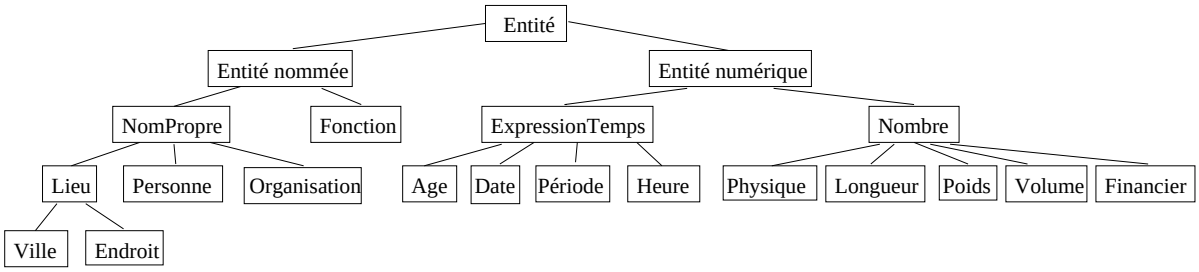


Figure 2 : Hiérarchie de types de réponses et de catégories sémantiques

L'analyse des questions est effectuée par un analyseur dédié fondé sur le déclenchement de règles décrivant différents types de questions. Les indices utilisés dans les règles pour décider de l'attribution d'une étiquette sont d'ordre lexical, avec la détection de mots spécifiques,

d'ordre syntaxique et sémantique. Les catégories sémantiques ont été constituées manuellement, et permettent d'étiqueter la très grande majorité des questions répondant aux types prévus, soit environ 60% des 700 questions de TREC9. Une amélioration prévue consiste à utiliser WordNet pour compléter les catégories sémantiques.

3.2 Reconnaissance des entités nommées

Les entités nommées sont marquées dans les documents par une balise dont le type correspond aux types de réponses présentés à la Figure 2. Les types retenus sont reconnus par des règles grâce à l'exploitation conjointe de deux sources d'information :

- des lexiques généraux permettant de trouver des traits syntaxiques et sémantiques associés aux mots simples en complément de traits lexicaux, et
- des dictionnaires d'entités nommées.

Les ressources utilisées sont CELEX (CELEX 1998), un lexique de 160 595 mots fléchis auxquels sont associés leur lemme et leur catégorie syntaxique, une liste de 8 070 prénoms (6 763 provenant de l'archive de CLR (CLR 1998)) et une liste de 211 587 noms de familles, provenant aussi de CLR, une liste de 22 095 entreprises provenant du "Wall Street Research Network" et 649 noms d'organisations obtenus à partir d'une acquisition lexicale sur Internet (Jacquemin 2000), deux listes dédiées aux noms de lieux : l'une de 7 813 villes et l'autre de 1 144 pays issus de CLR, plus des listes constituées manuellement sur les unités physiques et monétaires.

4 Les termes et leurs variantes

La reconnaissance et le marquage dans les documents des termes caractérisant la question et de leurs variantes servent dans un premier temps à réaliser une post-sélection des documents après celle effectuée par le moteur de recherche. Elle sert dans un second temps à donner des indices supplémentaires au module d'appariement question/réponse.

4.1 Extraction des termes

L'extraction automatique des termes à partir des questions utilise une technique simple de filtrage par des patrons de catégories syntaxiques. Les questions sont d'abord segmentées, étiquetées et lemmatisées par le *TreeTagger* (Schmid 1999). Des patrons de catégories syntaxiques sont ensuite utilisés pour extraire des termes des questions. Ces patrons ne diffèrent de ceux définis par Justeson et Katz (Justeson, Katz 1995) que par le fait que nous ne prenons pas en compte les syntagmes prépositionnels postposés. Les patrons utilisés sont synthétisés par l'expression régulière suivante[1] :

(((((JJ | NN | NP | VBG)) ? (JJ | NN | NP | VBG) (NP | NN))) | (VBD) | (NN) | (NP) | (CD))

La chaîne la plus longue est acquise en premier et les sous-chaînes ne peuvent être extraites que si elles ne commencent pas par le même mot que la surchaîne. Par exemple, dans la séquence $name_{NN}$ of_{IN} the_{DT} US_{NP} $helicopter_{NN}$ $pilot_{NN}$ $shot_{VBD}$ $down_{RP}$ (nom du pilote d'hélicoptère américain abattu), les quatre termes suivants sont extraits : *US helicopter pilot* (pilote d'hélicoptère américain), *helicopter pilot* (pilote d'hélicoptère), *pilot* (pilote) et *shoot* (abattre).

[1] JJ : adjectif, NN : nom, NP : nom propre, V_{BG} et V_{BD} : verbe au gérondif ou au participe passé, ? : indiquant au plus une occurrence de l'un des éléments entre parenthèses.

4.2 Reconnaissance et marquage des variantes par FASTR

Pour chaque question, nous ne retenons que les 200 premiers documents renvoyés par le moteur de recherche[2]. D'après les tests[3] que nous avons effectués (Ferret et al. 2000), c'est le nombre minimum de documents qui permet de conserver le maximum de documents contenant la réponse. Les performances des moteurs de recherche que nous avons utilisé sont très bonnes : ils conservent les documents pertinents, c'est à dire ceux qui contiennent les réponses correctes, pour plus de 95% des questions. Une indexation automatique de ces documents en fonction des termes de la question est ensuite faite par FASTR, un analyseur transformationnel de surface pour la reconnaissance de variantes terminologiques. Les termes extraits de la question sont transformés en règles de grammaire et les mots simples qui les composent sont stockés dans un lexique munis de liens morphologiques et sémantiques.

La *famille morphologique* d'un mot simple *m* est l'ensemble *M*(*m*) des mots simples de la base CELEX (CELEX 1998) qui ont la même racine que *m*. Par exemple, la famille morphologique du nom *maker* (fabricant) se compose des noms *maker*, *make* (marque) et *remake* (remake), et des verbes *to make* (faire) et *to remake* (refaire).

La *famille sémantique* d'un mot simple *m* est l'union *S*(*m*) des *synsets* de WordNet1.6 (Fellbaum 1998) auxquels ce mot *m* appartient. Un synset est l'ensemble des mots qui partagent un lien de synonymie sur une de leurs entrées sémantiques. Par exemple, la famille sémantique de *maker* se compose de trois noms : *maker*, *manufacturer* (fabricant), *shaper* (façonneur) et la famille sémantique de *car* (voiture) est *car, auto, automobile, machine* et *motorcar* (voiture à moteur).

Les patrons de variations qui reposent sur des familles morphologiques et sémantiques sont engendrés au moyen de métarègles. Le patron suivant[4], dénommé NtoVSemArg, extrait l'occurrence *making many automobiles* (fabricant de nombreuses voitures) comme variante de *car maker* (fabricant de voitures) :

VM ('*maker*') RP ? PREP ? (DT (NN | NP) ? PREP) ? DT ? (JJ | NN | NP | VBD | VBG) [0-3] NS ('*car*')

où VM('*maker*') est tout verbe de la famille morphologique du nom *maker* et NS('*car*') tout nom de la famille sémantique de *car*. En s'appuyant sur les familles morphologiques et sémantiques et sur le jeu de métarègles pour l'anglais, les occurrences suivantes sont extraites comme des variantes du terme d'origine *car maker* (fabricant de voitures) :

> *auto maker* (fabricant d'autos), *auto parts maker* (fabricant de pièces détachées automobiles), *car manufacturer* (fabricant de voitures), *make autos* (fabriquer des voitures) et *making many automobiles* (fabricant beaucoup de voitures).

5 Filtrage des documents

Le résultat de l'indexation des documents par FASTR est une liste d'occurrences de termes et de leurs variantes comprenant chacune un identificateur de document *d*, un identificateur de termes — une paire *t*(*q*, *i*) composée d'un numéro de question *q* et d'un indice unique *i* —, la

---

[2] Nous avons en particulier utilisé Indexal (de Loupy et al. 1998), moteur de recherche qui nous a été fourni par Bertin Technologie

[3] Ces tests ont été effectués grâce aux données fournies par le NIST après la campagne TREC8.

[4] RP sont les particules, PREP les prépositions, DT les déterminants et V les verbes.

variante reconnue et un identificateur de variation *v* (une métarègle). Par exemple, l'index suivant :

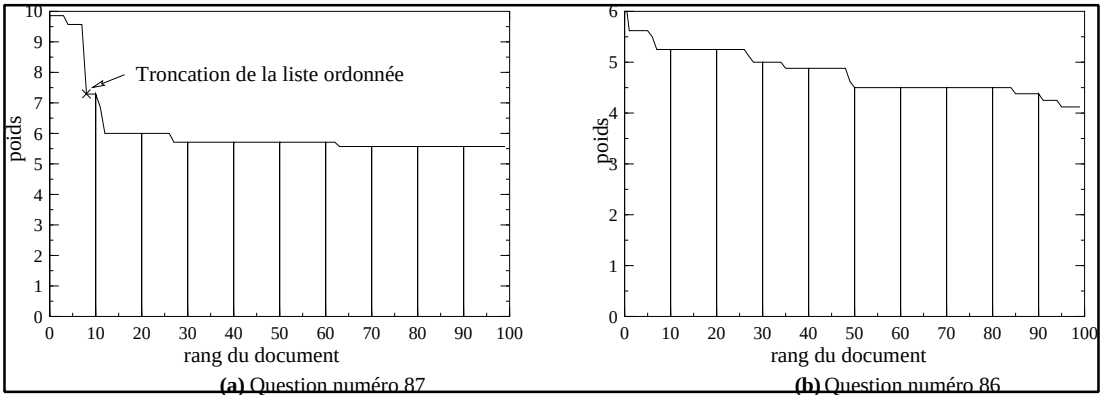
LA092690-0038 t(131,1) *making many automobiles* NtoVSemArg

signifie que l'occurrence *making many automobiles* du document n°LA092690-0038 est reconnue comme une variante du terme 1 (car maker) de la question *q*=131 au moyen de la variation NtoVSemArg donnée en section 4.2.

Chaque document sélectionné pour une question reçoit un poids. La fonction de pondération $W_q(d)$ (voir la formule (1)) repose sur une mesure de qualité des différentes familles de variations décrite dans (Jacquemin 1999) : le poids (w(v)) des occurrences de termes sans variation est 3, celui des variantes morphologiques et morpho-syntaxiques est 2 et celui des variantes sémantiques et morpho-sémantico-syntaxiques est 1. Les noms propres représentent des indices importants. Chaque terme $t(q, i)$ reçoit un poids $P(t(q, i))$ entre 0 et 1 correspondant à sa proportion de noms propres. Par exemple, *President Cleveland's wife* (la femme du président Cleveland) a un poids de 2/3=0,67 selon ce critère. Enfin, le dernier facteur de fiabilité est le nombre de mots du terme, représenté par la quantité $|t(q, i)|$ dans la formule (1). Ce facteur favorise donc les termes les plus longs. Le poids $W_q(d)$ d'un document d par rapport à une question q est alors donné par la formule (1). Les produits des pondérations de chaque terme identifié par FASTR sont sommés sur les index $I(d)$ extraits du document d et sont normalisés en fonction du nombre de termes $|T(q)|$ dans la question q.

$$W_q(d) = \sum_{(t(q,i),v)\in I(d)} \frac{w(v)\times(1+2P(t(q,i)))\times|t(q,i)|}{|T(q)|} \qquad (1)$$

Ce poids est calculé pour les 200 documents retenus par le moteur de recherche pour chaque question. La distribution de ces poids permet de réaliser un filtrage plus sélectif des documents. On observe principalement deux types de courbes de pondération des documents sélectionnés pour une question : les courbes avec un plateau et une chute brutale des valeurs des poids au-delà d'un certain rang (Figure 3.a) et les courbes avec des valeurs de poids en décroissance progressive (Figure 3.b).



(a) Question numéro 87 **(b)** Question numéro 86

Figure 3 : Deux types de courbes de pondération

Lorsque le système reconnaît une pente suffisamment forte, il sélectionne les documents apparaissant avant la chute, sinon il sélectionne un nombre de documents fixé a priori (nous avons fixé ce nombre à 100). Dans le cas d'une courbe ayant un profil semblable à celui de la Figure 3.a, le seuil correspondant à la chute du poids est détecté en analysant simultanément la pente de la courbe (la différence entre le poids d'un document et le poids du document précédent) et la variation de second ordre (la différence entre la pente à une étape et la pente à l'étape précédente). Dans le cas de la Figure 3.a, qui correspond à la question 87 des données

de TREC8 *Who followed Willy Brandt as chancellor of the Federal Republic of Germany?* (Qui a succédé à Willy Brandt comme chancelier de la République Fédérale d'Allemagne ?), QALC a trouvé un seuil égal à 8 documents. En revanche la courbe de la Figure 3b ne présente pas de seuil détectable. Pour la question 86 *Who won two gold medals in skiing in the Olympic Games in Calgary?* (Qui a gagné deux médailles d'or au ski aux Jeux Olympiques de Calgary ?), QALC a donc retenu 100 documents.

Nous avons évalué l'efficacité du filtrage en appliquant notre chaîne de traitement sur les données de TREC8, une fois avec le processus de filtrage, et une autre sans cette sélection. Sans filtrage, les 200 documents issus du moteur de recherche étaient donc conservés pour chacune des 200 questions. Nos tests ont donné un score de 0.463 dans le premier cas, et de 0.452 dans le second. Ces résultats montrent que les performances ne diminuent pas quand on traite moins de documents, car ce sont les documents les plus pertinents qui sont gardés. De plus, les performances de notre système sont en général meilleures pour les questions pour lesquelles moins de 100 documents sont gardés.

6 Appariement question/réponse

Le principe général de cet appariement consiste d'abord à comparer la question considérée avec chaque phrase des documents retenus pour cette question, et à conserver enfin les N_a phrases les plus similaires (N_a est égal à 5 pour la tâche Question Answering de TREC). Pour effectuer cette comparaison, QALC construit une même représentation pour la question et pour la phrase candidate à la réponse. Cette représentation consiste en un vecteur contenant trois types d'éléments : des mots pleins, des termes, et des entités nommées. Chacun de ces éléments est pondéré en fonction de son importance par rapport aux autres.

Les mots pleins sont essentiellement les adjectifs, les verbes et noms, sous leur forme lemmatisée, telle qu'elle est donnée par le *TreeTagger* (Schmid 1999). Les mots de la phrase qui ne sont pas présents dans la question reçoivent un poids nul. Les autres se voient attribuer un poids, de type *tf.idf*, en rapport avec leur fréquence dans un corpus de référence. Les termes proviennent de l'extracteur de termes décrit au paragraphe 4.1. Les termes de la question sont affectés d'un poids fixe. Pour la phrase, les termes considérés sont les variantes des termes de la question reconnues par FASTR dont le poids est celui qui permet de pondérer les documents et qui exprime la distance entre la variante et le terme correspondant (voir paragraphe 5). Les entités nommées sont, pour la question, celles qui correspondent au type attendu de la réponse, et pour la phrase, celles qui ont été reconnues dans la phrase par le module de reconnaissance des entités nommées. Les entités nommées correspondant au type attendu de la réponse reçoivent un poids fixe.

Finalement, la comparaison entre la question et la phrase est réalisée par le calcul de la similarité entre les vecteurs qui les représentent suivant la mesure suivante :

$$sim(V_q, V_d) = \frac{\sum_i wd_i}{\sum_j wq_j} \qquad (2)$$

où wq_j est le poids d'un élément du vecteur V_q représentant la question et wd_i est le poids d'un élément du vecteur V_d représentant la phrase. Le poids des termes et entités nommées est modéré par un coeficient afin de rester inférieur au poids des mots de la question. Lorsque la valeur de similarité est la même pour deux phrases différentes, QALC sélectionne la phrase où les mots pleins de la question sont le moins dispersés.

Nous allons montrer sur l'exemple de la question *What two US biochemists won the Nobel Prize in medicine in 1992?,* qui a été proposée à TREC8, comment chaque phrase est évaluée. La question est d'abord transformée en un vecteur, où <PERSON> est le type attendu de la

réponse, 16.01 est l'identificateur du terme *US biochemist* et 16.04 est l'identificateur du terme *Nobel Prize*[5] :

two (1.0)	us (1.0)	biochemist (0.9)	nobel (1.0)
prize (0,6)	medicine (0,5)	win (0,3)	1992 (1.0)
<PERSON> (0.5)	16.01 (0.5)	16.04 (0.5)	

Le même type de vecteur est construit pour chaque phrase du document FT924-14045, sélectionné pour cette question. Par exemple la phrase étiquetée par le module des entités nommées : *<NUMBER> Two </NUMBER> US biochemists, <PERSON> Edwin Krebs </PERSON> and <CITY> Edmond </CITY> Fischer, jointly won the <NUMBER> 1992 </NUMBER> Nobel Medicine Prize for work that could advance the search for an anti-cancer drug* donne le vecteur suivant :

two (1.0)	us (1.0)	biochemist (0.9)	nobel (1.0)
prize (0,6)	medicine (0,5)	win (0,3)	1992 (1.0)
edwin (0.0)	krebs (0.0)	edmond (0.0)	fischer (0.0)
work (0.0)	advance (0.0)	search (0.0)	anti-cancer (0.0)
jointly (0.0)	drug (0.0)	<PERSON> (0.5)	<NUMBER> (0.0)
<CITY> (0.0)	16.01 (0.5)	16.04 (0.3)	

où le poids 0.0 est donné aux éléments qui ne font pas partie du vecteur représentant la question. Le terme *US biochemist* est trouvé sans variation et *Nobel Prize* apparaît sous la forme de la variante syntaxique *Nobel Medicine Prize*. Finalement, en appliquant (2), on trouve une mesure de similarité de 0.974 entre les deux vecteurs.

Le module d'appariement considère la phrase en tant qu'unité de réponse de base mais, lorsque l'on souhaite une réponse plus concise, QALC s'appuie sur un ensemble d'heuristiques simples pour réduire la taille des phrases dépassant la limite fixée. Lorsqu'une entité nommée correspondant au type attendu de la réponse, ou d'un type proche, a été trouvée, QALC sélectionne la partie de la phrase entourant cette entité nommée. Dans le cas contraire, ou dans le cas où le type attendu de la réponse n'a pu être déterminé, il extrait une partie de la phrase contiguë à celle contenant les mots de la question. On suppose ainsi que la phrase contenant la réponse possède une structure comparable à ce que serait la forme affirmative de la question posée.

7 Résultats et discussion

Dans le cadre de TREC9, les résultats de QALC ont été évalués dans trois conditions différentes, les variations de l'une à l'autre concernant le moteur de recherche utilisé et la taille de la réponse (250 ou 50 caractères). Le meilleur de ces tests a obtenu un score de 0,407 avec 375 réponses trouvées sur 682, pour des réponses sur 250 caractères. Le calcul de ce score tient compte du rang de classement (de 1 à 5) de la réponse trouvée. Ce score nous a placé en 6^{ième} position sur 28 participants. Le premier (Harabagiu et al, 2000) a obtenu un score de 0,760, le deuxième (Kwok et al, 2000) un score de 0,464, et les trois suivants respectivement des scores de 0,460, 0,457 (Ittycheriah et al, 2000) et 0,425.

La conférence d'évaluation TREC offre une référence intéressante pour mesurer l'efficacité des méthodes utilisées par les différents systèmes de question-réponse. L'architecture de base généralement adoptée par les systèmes participants est conforme à celle de notre système QALC, avec éventuellement quelques variantes.

[5] Nous ne retenons ici que les expressions les plus longues des différents termes

Le typage de la réponse attendue est évidemment une fonctionnalité indispensable à un système de question-réponse, particulièrement lorsqu'il s'agit de donner une réponse courte (50 caractères). Parmi les questions posées à TREC, certaines attendent en réponse une entité nommée telle qu'une date, un nom de personne ou un nom d'organisation par exemple. Dans ce cas, le typage de la réponse est simple même si son exploitation nécessite un bon système de reconnaissance des entités nommées et si le nombre de catégories de types peut être augmenté de manière à raffiner le typage. En revanche, lorsque la réponse attendue est constituée d'un nom commun ou d'une phrase, son typage, plus complexe, est rarement réalisé. Certains systèmes, comme le système FALCON (Harabagiu et al, 2000), utilisent les hiérarchies des classes de mots dans WordNet pour typer les réponses. Pour sa part, le système développé par (Ittycheriah et al, 2000) se fonde sur un modèle de l'entropie maximum pour la classification des types de réponse. Sur les 682 questions de TREC9, 57,5% ont été analysées par QALC comme étant des questions à entité nommée, les autres n'ont pas été typées. Parmi les réponses correctes de notre meilleur test, 62,7% répondent à des questions à entité nommée. Par ailleurs, le test que nous avons fait pour des réponses plus courtes donne 84% de réponses possédant une entité nommée parmi les bonnes réponses. Typer la réponse permet donc de mieux cibler la portion de phrase qui peut la contenir.

Tous les systèmes ayant participé à TREC9 utilisent un moteur de recherche pour effectuer la sélection d'un sous-ensemble de documents dans la base d'environ un million de documents mise à disposition par le NIST. Dans le système QALC, nous conservons dans son intégralité chaque document retrouvé par le moteur. Mais d'autres sytèmes sélectionnent le ou les paragraphes pertinents de chaque document retrouvé. Comme dans QALC, la recherche des meilleures réponses est fondée sur un appariement entre question et réponse reposant sur la comparaison des mots des questions avec ceux des phrases sélectionnées et tenant compte du type attendu de la réponse et des entités nommées. Les critères retenus pour effectuer l'appariement peuvent varier d'un système à l'autre. Kwok et al (Kwok et al, 2000) par exemple utilisent entre autres un dictionnaire de synonymes qu'ils ont extrait manuellement de WordNet. Le système FALCON, quant à lui, (Harabagiu et al, 2000) utilise une approche sémantique pour réaliser cet appariement : une unification est recherchée entre la représentation sémantique de la question et les représentations sémantiques des paragraphes sélectionnés. C'est également le seul système à effectuer une justification de ses réponses.

8 Conclusion

Un système de question-réponse doit trouver la réponse à une question précise, dans un temps suffisamment court pour être compatible avec une éventuelle utilisation interactive. Cette réponse étant recherchée dans une grande masse de documents, il est tentant d'appliquer des méthodes essentiellement numériques pour la trouver. Néanmoins les expériences montrent que l'ajout de raisonnements fondés sur des connaissances sémantiques et pragmatiques est nécessaire si on veut obtenir, à terme, un système réellement efficace. En effet, le système qui a obtenu les meilleurs résultats à TREC9 est aussi celui qui utilise le plus largement les techniques d'analyse syntaxique et sémantique (Harabagiu et al, 2000). Les futures orientations de la tâche question-réponse de la conférence TREC vont d'ailleurs dans ce sens. En effet, à un horizon de 5 ans, les organisateurs prévoient des améliorations à la fois sur la rapidité de la réponse, la vérification de sa justesse, la possibilité de fusionner plusieurs réponses pour obtenir une réponse complète, et enfin des possibilités de dialogue permettant à l'utilisateur de préciser sa demande. Les futurs systèmes devront aussi pouvoir décider si la réponse se trouve ou non dans le corpus de recherche. De plus, ils devront être capable de déduire la réponse complète de réponses fragmentaires dispersées dans différents documents. Toutes ces améliorations ne pourront se faire sans une intégration plus importante des méthodes propres au traitement sémantique de la langue.

Les améliorations que nous voulons apporter à notre système relèvent donc essentiellement d'une approche sémantique et pragmatique. Ainsi, la base de connaissances WordNet, que

nous utilisons déjà pour trouver les variantes sémantiques d'un mot, pourra aussi être exploitée pour une classification plus fine des types de réponses. Nous utiliserons aussi une analyse syntaxico-sémantique robuste pour construire les représentations sémantiques de la question et de l'ensemble des réponses candidates, afin de sélectionner les réponses à la fois sur les termes de la question et sur les liens sémantiques que ces termes ont entre eux.

Références

CELEX, 1998, http://www.ldc.upenn.edu/readme_files/celex.readme.html, UPenns, Eds., Actes Consortium for Lexical Resources, (1998)

CLR, 1998, http://crl.nmsu.edu/cgi-bin/Tools/CLR/clrcat#D3, NMSUs, Eds., Actes Consortium for Lexical Resources, New Mexico (1998)

Fellbaum C., (1998) *WordNet: An Electronic Lexical Database*, Cambridge, MA, MIT Press.

Ferret O., Grau B., Hurault-Plantet M., Illouz G., Jacquemin C. (2000), QALC — the Question-Answering system of LIMSI-CNRS, pre-proceedings of TREC9, NIST.

Grishman R., Sundheim B., (1995), Design of the MUC-6 evaluation, Actes de *MUC-6*, NISTs, Eds., Morgan Kauffmann Publisher, Columbia, MD.

Harabagiu S., Pasca M., Maiorano J., (2000), Experiments with Open-Domain Textual Question Answering, Actes de *Coling'2000*, Saarbrucken, Germany.

Ittycheriah A., Franz M., Zhu W-J., Ratnaparkhi A., (2000), IBM's statistical Question Answering System, , Actes préliminaires de *TREC9*, Gaithersburg, MD, NIST Eds, 60-65.

Jacquemin C., (1999), Syntagmatic and paradigmatic representations of term variation, Actes de *ACL'99*, 341-348.

Jacquemin C., Bush C., Fouille du Web pour la collecte d'entités nommées, Actes de *TALN 2000*, Lausanne (2000), 187-196.

Justeson J., Katz S., (1995), Technical terminology: some linguistic properties and an algorithm for identification in texte, *Natural Language Engineering* , Vol 1, pp. 9-27.

Kwok K.L., Grunfeld L., Dinstl N., Chan M., (2000), TREC9 Cross Language, Web and Question-Answering Track experiments using PIRCS, Actes préliminaires de *TREC9*, Gaithersburg, MD, NIST Eds., 26-35.

Lehnert W., (1977), Human and computational question answering, *Cognitive Science* , 1, p. 47-63.

de Loupy C., Bellot P., El-Bèze M., Marteau P.-F. (1998). Query Expansion and Classification of Retrieved Documents, *TREC7* , p.382-389.

Mollà A. D. et al., (2000), EXTRANS, An Answer Extraction System, *Traitement automatique des langues* , 41, p. 495-522.

Schmid H., (1999), Improvements in Part-of-Speech Tagging with an Application To German, *Natural Language Processing Using Very Large Corpora*, Dordrecht, S. Armstrong, K. W. Chuch, P. Isabelle, E. Tzoukermann, D. Yarowski, Eds., Kluwer Academic Publisher.