

Le Web et la question-réponse : transformer une question en réponse

LUC PLAMONDON

RALI/DIRO, Université de Montréal

CP 6128, Succ. Centre-Ville

Montréal (Québec) Canada, H3C 3J7

Email : plamondl@iro.umontreal.ca

Tél : +1 514 343-6111 #3507 Fax : +1 514 343-2496

LEILA KOSSEIM

CLAC Laboratory, Concordia University

1455 de Maisonneuve Blvd. West

Montréal (Québec) Canada, H3G 1M8

Email : kosseim@cs.concordia.ca

Tél : +1 514 848-3074 Fax : +1 514 848-2830

Résumé

De plus en plus de systèmes de question-réponse comptent le Web parmi les outils qu'ils utilisent pour trouver une réponse courte et précise à une question posée en langue naturelle. Dans cet article, nous présentons comment des règles simples de transformation de questions permettent de générer des contextes de réponse suffisamment restrictifs pour repérer des réponses sur le Web. La recherche de ces contextes (seuls ou en conjonction) alliée à une vérification sémantique de base des réponses extraites nous a permis de trouver la bonne réponse à 25 % des questions des campagnes d'évaluation TREC-10 et TREC-11.

Abstract

An increasing number of Question Answering systems use the Web to find a short and precise answer to a natural language question. In this paper, we present how simple question re-write rules allow for the generation of answer contexts that are restrictive enough for finding answers on the Web. Searching for such contexts (alone or in conjunction) and then performing simple semantics checks on the extracted answers leads to a correct answer for 25 % of a set of questions from the TREC-10 and TREC-11 conferences.

1 Introduction

Les développements récents dans le domaine de la question-réponse (*question answering*) rendent maintenant possible la recherche, dans un ensemble de textes, de réponses précises à des questions posées en langue naturelle. Par exemple, un utilisateur qui interrogerait un système de question-réponse (QR) avec la question *Qui était le premier ministre du Canada en 1873 ?* se verrait proposer une réponse spécifique, telle que *Alexandre Mackenzie*, plutôt qu'un document entier dans lequel il devrait localiser lui-même la réponse. À cet égard, la QR peut être vue comme la prochaine génération d'outils de recherche dans d'imposantes collections de textes telles que le Web.

Les campagnes TREC (*Text Retrieval Conference*), sous les auspices de l'institut américain des standards et de la technologie (NIST), ont donné le coup d'envoi aux recherches en QR en 1999 en lançant la première campagne d'évaluation de systèmes de QR. Depuis, chaque année, NIST met à la disposition des équipes participantes un ensemble de questions et une collection de textes afin qu'elles mettent leur système à l'épreuve. Les réponses sont ensuite évaluées par NIST suivant une méthodologie standard. Les questions et les textes fournis par NIST sont exclusivement en anglais, c'est pourquoi des campagnes d'évaluation de systèmes traitant d'autres langues ont récemment vu le jour : EQueR (dans le cadre du projet EVALDA de Technolanguage/ELDA) pour la question-réponse en français et CLEF pour la question-réponse multilingue (textes en anglais et questions en d'autres langues, dont le français).

À ce jour, les campagnes TREC continuent d'être le foyer des principaux avancements en QR. On y a vu l'apparition de systèmes dérivés des moteurs de recherche classiques, de systèmes à apprentissage statistique, de systèmes faisant de l'analyse linguistique de surface et de systèmes faisant appel à des bases de connaissances et à des réseaux sémantiques tels que Wordnet [Miller, 1995]. Bien que les réponses suggérées par les systèmes doivent être présentes dans la collection de textes standard à TREC, de plus en plus de systèmes consultent aussi le Web : parfois pour départager plusieurs réponses prometteuses trouvées dans la collection de textes [Magnini et al., 2002], d'autres fois pour y puiser directement des réponses (quitte à les apparier ensuite avec des réponses similaires présentes dans la collection) [Brill et al., 2001, Brill et al., 2002, Clarke et al., 2001].

Afin d'améliorer notre système de QR, Quantum [Plamondon et al., 2002], nous avons privilégié la deuxième approche, c'est-à-dire utiliser le Web comme source primaire d'information. Quantum tente de prévoir, à partir de la question, le contexte (c'est-à-dire la séquence des mots environnants) dans lequel la réponse est susceptible d'être rencontrée, dans un texte quelconque. Il interroge ensuite un moteur de recherche Web avec ce contexte de réponse et, si le contexte est trouvé, il peut procéder à l'identification de la réponse exacte. Nous verrons qu'il est possible d'atteindre des résultats intéressants même avec un nombre limité de règles permettant de prévoir le contexte d'une réponse.

Deux stratégies de recherche sur le Web s'offrent à nous. La première vise un rappel élevé : elle consiste à obtenir une longue liste de candidats et à utiliser ensuite une méthode efficace d'évaluation des candidats afin de sélectionner les meilleurs. Cette stratégie est suivie notamment par les systèmes qui s'appuient sur la *redondance des réponses* et qui

considèrent que plus une réponse est fréquente, plus elle a de chances d'être la bonne. Nous avons plutôt opté pour la stratégie inverse : viser une précision élevée aux dépens du rappel. Il nous apparaissait en effet préférable, pour les besoins de Quantum, d'obtenir peu de candidats du Web mais d'avoir une confiance élevée en leur qualité. Pour ce faire, vu la quantité astronomique de textes accessibles sur le Web et leur impressionnante variété, nous avons choisi d'utiliser des critères de recherche très serrés qui garantiraient autant que possible la validité des réponses trouvées. Ces critères ne sont nul autres que le contexte exact de la réponse, que nous générons automatiquement à partir de la question.

2 Travaux antérieurs

Une première tentative de formulation automatique de contextes de réponse a été faite à TREC-10, où [Clarke et al., 2001] et [Brill et al., 2001] ont vu le Web comme une source gigantesque de textes supplémentaires pour améliorer l'extraction de réponses. En particulier, le système présenté par Microsoft à TREC-10 [Brill et al., 2001, Brill et al., 2002] recherche sur le Web une liste de formulations possibles d'une réponse, produites en permutant les mots de la question. Autrement dit, étant donné une question de la forme : *Who is $w_1 w_2 w_3 \dots w_n$?*, le système cherche :

```
" $w_1$  is  $w_2 w_3 \dots w_n$ "
" $w_1 w_2$  is  $w_3 \dots w_n$ "
" $w_1 w_2 w_3$  is  $\dots w_n$ "
...
```

Par exemple, étant donné la question : *Who is the world's richest man married to ?*, les requêtes suivantes sont construites :

```
"the is world's richest man married to"
"the world's is richest man married to"
"the world's richest man is married to"
...
```

Avec un peu de chance, au moins une des expressions (vraisemblablement la dernière de l'exemple ci-haut) sera trouvée dans une page Web et permettra d'identifier la réponse correcte. Bien que simple, cette stratégie s'avère très efficace. En utilisant cette méthode, [Brill et al., 2001] se sont classés 9^e sur 37 équipes à la campagne TREC-10. Suite à ces résultats, le Web est devenu une ressource presque standard pour la QR [Duclaye et al., 2002].

D'un autre côté, dans les travaux de [Agichtein and Gravano, 2000] et de [Lawrence and Giles, 1998], les formulations de réponses sont produites dans le but spécifique d'améliorer la recherche d'information sur le Web. Les formulations produites sont

précises mais elles sont employées pour l'expansion de requêtes, et non pas pour la recherche de réponses exactes. Tandis que [Lawrence and Giles, 1998] établissent à la main des règles de transformation pour générer des contextes de réponse tels que "NASDAQ stands for" et "NASDAQ means" à partir de questions comme *What does NASDAQ stand for?*, [Agichtein and Gravano, 2000] utilisent de l'apprentissage automatique.

3 La génération et la recherche Web de contextes de réponse

Dans Quantum, la génération d'un contexte de réponse se fait en transformant la question en sa forme déclarative à l'aide de règles établies manuellement. Par exemple, la question #1697 – *Where is the Statue of Liberty?*¹ est reformulée "the Statue of Liberty is <LIEU>". Nous nous sommes inspirés de 198 questions de TREC-8 et de 682 questions de TREC-9 pour bâtir les règles de transformation et nous avons utilisé 447 questions de TREC-10 et 454 de TREC-11 pour les tester. Nous avons éliminé de notre étude 112 questions pour lesquelles NIST ne fournit pas de réponse, soit parce qu'elles ont été retirées par NIST pour des raisons techniques, soit parce qu'elles n'ont pas de réponse dans la collection de textes de TREC.

Avant de procéder à la transformation, la forme grammaticale de la question est normalisée afin de limiter le nombre de cas à traiter. Par exemple, une question commençant par *What's ...* est normalisée en *What is ...*, *What is the name of ...* est changée en *Name ...*. Au total, 17 règles grammaticales sont utilisées pour la normalisation.

Le tableau 1 illustre un exemple de règle de transformation. La règle est formée de 2 catégories de patrons : des patrons qui déterminent à quelle forme de question s'applique la règle, et des patrons qui définissent la forme de la réponse à chercher. Les patrons mettent en jeu des mots spécifiques (ex. **when**), des chaînes de caractères quelconques (représentées dans le tableau par SYNTAGME) et des étiquettes grammaticales (ex. VERBE-SIMPLE). Les patrons de réponse, que nous appelons *contextes de réponse*, utilisent les mêmes types de traits, en plus d'une spécification de la classe sémantique de la réponse (ex. <TEMPS>). Les classes sémantiques sont employées plus tard, après l'extraction des réponses, afin de baisser le poids de celles qui ne correspondent pas au type de réponse attendu. Dans l'exemple du tableau 1, la réponse doit être, de préférence, une expression de <TEMPS>, ce qui élimine des réponses comme "The Jurassic period ended the same way the Tiasic period did" (classe sémantique différente) ou "The Jurassic period ended a long time ago" (réponse trop générale). La version actuelle de Quantum comprend 10 classes sémantiques.

La transposition d'un verbe de sa forme simple (lorsqu'il est utilisé dans une question avec l'auxiliaire du passé *did*) à sa forme passée (dans la réponse, sans auxiliaire de temps) est un phénomène particulier de l'anglais qui ne peut être traité par la seule application de patrons. Afin d'effectuer cette correspondance rapidement, nous avons au préalable extrait

¹Les questions précédées d'un numéro proviennent des campagnes TREC.

Règle de transformation	Exemple
Q : When did SYNTAGME VERBE-SIMPLE ?	Q : #22 - <i>When did the Jurassic Period end ?</i>
R : SYNTAGME VERBE-PASSÉ <TEMPS>	R : the Jurassic Period ended <TEMPS>
R : <TEMPS> SYNTAGME VERBE-PASSÉ	R : <TEMPS> the Jurassic Period ended
R : <TEMPS>, SYNTAGME VERBE-PASSÉ	R : <TEMPS>, the Jurassic Period ended

TAB. 1 – Exemple de règle de transformation

tous les verbes répertoriés par le réseau sémantique *Wordnet* et nous avons construit une table de hachage permettant de faire correspondre la forme simple d'un verbe à sa forme passée.

Afin d'augmenter les chances de succès, *Quantum* peut chercher des contextes en *conjonction*. Par exemple, la réponse à la question #670 - *What type of currency is used in Australia ?* doit apparaître à la fois dans "SYNTAGME is used in Australia" et dans "SYNTAGME is a type of currency", peu importe si ces deux contextes sont présents dans des pages Web différentes. Les conjonctions sont introduites dans le cas de questions syntaxiquement plus complexes qui généreraient des contextes plus longs ayant peu de chances d'être trouvés sur le Web. Dans la version actuelle du système, les conjonctions sont générées pour des patrons de questions pré-identifiés. Cependant, la décision de générer des conjonctions de contextes pour une question donnée pourrait être dynamique, par exemple en fonction de la longueur de la question ou du contexte.

Au total, 76 patrons de question ont été élaborés. Ils sont essayés l'un après l'autre sur une même question et tous ceux qui s'appliquent déclenchent la création des contextes de réponses correspondants. Le tableau 2 montre le nombre de patrons de question par type de question. Nous y apprenons que, par exemple, 6 patrons peuvent être employés pour transformer les questions de type *when* et que le nombre de contextes de réponse élaborés pour ce type de question s'élève à 9 (un patron de question de ce type engendre en moyenne 1,5 contexte de réponse). Les 76 patrons de question couvrent 93 % des 198 questions de TREC-8 et 89 % des 682 questions de TREC-9. Par couverture, nous entendons qu'au moins un patron de question est applicable. 409 contextes de réponse ont été produits suite à l'analyse des 185 questions de TREC-8 couvertes par les patrons de question, alors que 1209 contextes de réponse ont été produits suite à l'analyse des 610 questions de TREC-9 couvertes. Ainsi, en moyenne, nous obtenons 2 contextes de réponse par question.

Une fois les contextes de réponse générés, nous collectons les pages Web qui les contiennent à l'aide du moteur de recherche *Yahoo!*. Il suffit ensuite d'unifier l'inconnue d'un contexte avec un court extrait du texte. *Quantum* effectue ensuite des tests simples pour vérifier que le candidat ainsi extrait est du type sémantique recherché : par exemple, il met de côté les candidats qui ne débutent pas par une majuscule si le type de la réponse doit être un <LIEU>.

Afin de déterminer le meilleur de tous les candidats trouvés, un système de pointage a

Type de question	Patrons de question	Contextes de réponse	Moyenne
when	6	9	1.5
where	9	12	1.3
how many	12	13	1.1
how much	5	5	1.0
how (autres)	10	14	1.4
what	21	21	1.0
which	2	2	1.0
who	7	9	1.3
why	2	2	1.0
name	2	2	1.0
Total	76	89	1.2

TAB. 2 – Nombre de patrons de question et de réponse par type de question, et moyenne de contextes de réponse par patron de question

été mis en place. Un candidat qui est du type sémantique voulu débute avec un score de 0.6. Pour chaque occurrence additionnelle de ce candidat trouvée ailleurs dans un autre contexte de réponse, le score est augmenté ainsi :

$$score_{i+1} = score_i + \frac{1 - score_i}{2} \quad (1)$$

Ainsi, une deuxième occurrence du candidat fait grimper son score de 0.6 à 0.8, une troisième occurrence le fait grimper à 0.9, et ainsi de suite. Quant aux candidats qui échouent le test sémantique, ils débutent avec un score de 0.1. Ce score est augmenté de 0.1 à chaque nouvelle occurrence du candidat, et ce tant que le score n'excède pas 0.6 :

$$score_{i+1} = \min(0.6, score_i + 0.1) \quad (2)$$

De cette façon, un candidat qui est extrait à l'aide d'un contexte de réponse mais qui ne semble pas être du type voulu n'a jamais un score plus élevé qu'un candidat sémantiquement valide, si un tel candidat est trouvé par le système.

4 Évaluation et discussion

La recherche de contextes de réponse sur le Web est un ajout récent au système de QR Quantum. Ce dernier s'appuie aussi sur d'autres techniques telles que l'analyse syntaxique de surface des questions, la consultation du réseau sémantique Wordnet, l'extraction d'entités nommées à l'aide du module *NE Transducer* de GATE [Cunningham et al., 2002] et l'apprentissage du poids de chacun de ces traits [Plamondon et al., 2002, Plamondon and Kosseim, 2002]. Cependant, nous nous concentrons ici sur l'évaluation du module Web utilisé seul.

Nous avons utilisé les questions de TREC-10 et de TREC-11 pour lesquelles nous disposons de réponses fournies par NIST afin de tester l'efficacité de notre approche de

Corpus	Nombre de questions	Couverture	Nombre de contextes générés
TREC-8 (développement)	192	93 %	409
TREC-9 (développement)	682	89 %	1209
TREC-10 (test)	443	91 %	664
TREC-11 (test)	454	87 %	722

TAB. 3 – Couverture des patrons de question pour le corpus de développement et le corpus de test, et nombre de contextes de réponse générés

recherche de réponses sur le Web. Les patrons de question couvrent 89 % des 897 questions du corpus de test avec pour effet d'engendrer un total de 1386 contextes de réponse à chercher sur le Web, soit 1.7 contexte de réponse par question en moyenne (tableau 3). La couverture obtenue avec le corpus de test est conforme aux prévisions basées sur le corpus de développement. Elle est particulièrement satisfaisante, considérant que seulement 76 patrons de question ont suffi à couvrir presque 90 % de toutes les questions qu'un utilisateur potentiel pourrait se poser. Notons à ce sujet que les questions de TREC sont courtes (jamais plus d'une phrase), qu'elles sont la plupart du temps exemptes d'erreurs syntaxiques et orthographiques, qu'elles attendent une réponse factuelle d'ordre général, que la majorité d'entre elles ont été réellement posées par des internautes sur des sites Web mais que NIST s'est réservé le droit de les filtrer et de les adapter aux besoins de TREC (voir [Voorhees, 2002] pour une description détaillée de la plus récente piste QR). Hormis ces précisions, aucune contrainte imposée aux questions n'aurait permis de prévoir la teneur et la structure de celles du corpus de test à partir du corpus de développement.

Si nous effectuons la recherche des contextes de réponse ainsi générés dans la collection standard de textes de TREC, qui fait environ 3 gigaoctets, nous trouvons au moins un contexte pour seulement 83 (10 %) des 800 questions du corpus de test couvertes par les patrons de question. Cependant, si nous effectuons plutôt la recherche sur le Web (un corpus nettement plus vaste), nous trouvons au moins une occurrence d'un contexte de réponse pour 354 questions (44 % des questions couvertes). Déjà, le potentiel du Web est évident. Le système trouve au moins une réponse correcte pour 227 (64 %) de ces 354 questions. En moyenne, 15 réponses par question sont extraites et lorsqu'elles sont ordonnées par le système selon leur qualité présumée, une réponse correcte ne se trouve au premier rang que pour 108 (49 %) de ces 227 questions. En clair, si on ne permet au système qu'une seule suggestion de réponse, elle est correcte dans 12 % des cas, et cela simplement en cherchant des contextes de réponse sur le Web sans aucun traitement supplémentaire (hormis quelques tests de validité sémantique de base). Le tableau 4 indique la performance du système à chacune des étapes du processus.

Il est difficile de cibler quelle étape du processus complet il conviendrait d'améliorer en priorité car chacune diminue les possibilités de succès d'environ 50 %. Il est cependant clair que de nombreuses améliorations peuvent être apportées à la version actuelle de **Quantum** à tous les niveaux. Tout d'abord, l'élaboration de plus de contextes de réponse et l'utilisation de davantage de conjonctions de contextes de réponse permettraient d'aug-

Type de question	Nombre de questions (corpus test)	Couverture des règles (a)	Au moins un contexte trouvé (b)	Au moins une réponse correcte (c)	Réponse correcte en 1 ^{er} (d)
when	93 (100 %)	83 %	26 %	18 %	14 %
where	63 (100 %)	87 %	41 %	33 %	25 %
how many	9 (100 %)	56 %	33 %	33 %	22 %
how much	12 (100 %)	33 %	8 %	8 %	8 %
how (autres)	48 (100 %)	73 %	27 %	10 %	6 %
what	548 (100 %)	93 %	41 %	25 %	10 %
which	25 (100 %)	88 %	0 %	0 %	0 %
who	93 (100 %)	98 %	65 %	45 %	20 %
why	4 (100 %)	0 %	0 %	0 %	0 %
name	2 (100 %)	100 %	50 %	50 %	0 %
Total	897 (100 %)	89 %	39 %	25 %	12 %

TAB. 4 – Pourcentage de succès après chaque étape du processus d'extraction de réponse, par type de question. Les pourcentages représentent la proportion des questions du corpus de test, par rapport à toutes celles d'un type donné, qui satisfont de façon cumulative aux conditions suivantes : (a) il existe au moins un patron de question qui s'applique et qui génère un contexte de réponse (b) au moins un de ces contextes a été trouvé sur le Web (c) au moins une réponse correcte a été trouvée sur le Web (d) une fois les réponses ordonnées, la première suggérée s'avère être correcte

menter le nombre de candidats trouvés (colonne b du tableau 4) et de mieux identifier les candidats hautement pertinents (colonne d). Aussi, il faudrait mesurer précisément l'apport des conjonctions de contextes. Les conjonctions ont été introduites de façon intuitive pour scinder les contextes qui nous paraissaient trop longs pour être trouvés sur le Web. Il serait intéressant de mesurer formellement l'apport de ces conjonctions en fonction de la longueur du contexte original.

Il nous semble très important d'améliorer aussi l'algorithme de pointage pour discerner les bons candidats des mauvais (colonne d). Ceci inclut l'amélioration des tests sémantiques et possiblement l'introduction de tests syntaxiques séparés. En effet, les tests devraient être capables de faire une analyse syntaxique et sémantique plus fine de la phrase dans laquelle le contexte est trouvé (colonne b). Par exemple, pour répondre à la question #1697 – *Where is the Statue of Liberty?*, on cherche le contexte *the Statue of Liberty is <LIEU>* sur le Web et on peut trouver :

1. The arm of *the Statue of Liberty* is **42 feet long**
2. *The Statue of Liberty* is **recognized** as a symbol of freedom throughout the world
3. *The Statue of Liberty* is **a huge sculpture** that is located on Liberty Island.

Bien que les 3 phrases contiennent toutes le contexte de réponse cherché, aucun des candidats qui suivent le contexte ne constitue une réponse correcte. Le candidat de l'exemple 1 ne constitue pas une réponse appropriée car la tête du groupe nominal sujet n'est pas *Statue of Liberty* mais *arm*. Dans l'exemple 2, le type syntaxique du candidat ne correspond

pas au type syntaxique recherché : le candidat fait partie d'un groupe verbal alors que la réponse doit être un groupe nominal. Quant au candidat de l'exemple 3, c'est son type sémantique qui ne convient pas. Ces exemples montrent que la présence du contexte recherché dans la collection de textes ne garantit pas la validité du candidat, ce qui se traduit par une importante baisse de score entre les colonnes b et c du tableau 4.

La vérification de contraintes linguistiques est nécessaire pour bien distinguer les bons des mauvais candidats et mérite que l'on s'y attarde. En effet, lorsque l'on demande au système de choisir une seule réponse (colonne d) parmi tous les candidats qu'il a trouvés, il suggère une réponse correcte pour seulement 12 % des questions alors qu'il dispose d'au moins un candidat correct pour 25 % des questions.

Nous n'avons pas mesuré l'importance relative de chacune des sources d'erreurs mentionnées ci-haut, à savoir une mauvaise affectation ou vérification de la classe sémantique, la non-vérification de la classe syntaxique et de la tête du groupe nominal sujet. Cependant, l'affectation de la classe sémantique étant faite à l'aide de patrons développés manuellement, il nous semble que les erreurs proviennent surtout de la vérification *ad hoc*.

5 Conclusion

Nos expérimentations nous ont permis d'entrevoir le potentiel du Web pour la question-réponse. En effet, nous avons constaté que la taille gigantesque du Web rend possible la recherche de contextes de réponse très restrictifs, ce qui augmente d'autant les chances que la chaîne de caractères trouvée soit bel et bien la réponse à la question. Dans un corpus relativement petit comme celui de TREC (≈ 3 Go), cette stratégie ne pourrait porter fruit car la probabilité de trouver un contexte particulier est nettement trop inférieure.

Nous avons généré les contextes de réponse en transformant la question en sa forme déclarative à l'aide de règles de réécriture simples. Les quelque 76 patrons de question et 89 contextes de réponse ont suffi à couvrir 89 % d'un ensemble de 897 questions issues des campagnes d'évaluation TREC. Ils ont permis de trouver une bonne réponse à 25 % d'entre elles ; cependant, l'algorithme de pointage des candidats doit être amélioré car la bonne réponse n'est suggérée en première position que dans 12 % des cas.

Les questions de TREC étant d'ordre général, le Web constitue une base documentaire tout indiquée pour les questions sur lesquelles a porté notre étude. De plus, elles sont courtes et exemptes d'erreur d'orthographe et de syntaxe, donc plus faciles à analyser mais pas nécessairement représentatives des questions posées par un utilisateur moyen. Il serait intéressant de tester la robustesse de notre système de question-réponse sur des questions authentiques. Le fait que les réponses aux questions de TREC soient factuelles joue aussi en notre faveur car les *comment ?* et les *pourquoi ?* sont exclus, les réponses sont courtes (rarement plus longues qu'un syntagme nominal) et elles ne nécessitent pas de regrouper ni de synthétiser des éléments tirés de plusieurs documents.

Remerciements

Nous tenons à remercier Louis-Julien Guillemette pour sa programmation et les relecteurs de

JFT pour leur commentaires. Ce projet a été financièrement soutenu par les Laboratoires Universitaires Bell et le Conseil de recherche en sciences naturelles et en génie du Canada (CRSNG).

Références

- [Agichtein and Gravano, 2000] Agichtein, E. and Gravano, L. (2000). Snowball : Extracting Relations from Large Plain-Text Collections. In *Proceedings of the 5th ACM International Conference on Digital Libraries*, San Antonio, Texas.
- [Brill et al., 2002] Brill, E., Dumais, S., and Banko, M. (2002). An Analysis of the AskMSR Question-Answering System. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP-2002)*, Philadelphie, Pennsylvanie.
- [Brill et al., 2001] Brill, E., Lin, J., Banko, M., Dumais, S., and Ng, A. (2001). Data-Intensive Question Answering. In *Proceedings of The Tenth Text Retrieval Conference (TREC-10)*, pages 393–400, Gaithersburg, Maryland.
- [Clarke et al., 2001] Clarke, C. L. A., Cormack, G. V., Lynam, T. R., Li, C. M., and McLearn, G. L. (2001). Web Reinforced Question Answering (MultiText Experiments for TREC 2001). In *Proceedings of The Tenth Text Retrieval Conference (TREC-10)*, pages 673–679, Gaithersburg, Maryland.
- [Cunningham et al., 2002] Cunningham, H., Maynard, D., Bontcheva, K., and Tablan, V. (2002). GATE : A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, Philadelphie, Pennsylvanie.
- [Duclaye et al., 2002] Duclaye, F., Yvon, F., and Collin, O. (2002). Using the Web as a Linguistics Resource for Learning Reformulations Automatically. In *Proceedings of LREC*, pages 390–396, Las Palmas, Espagne.
- [Lawrence and Giles, 1998] Lawrence, S. and Giles, C. L. (1998). Context and Page Analysis for Improved Web Search. *IEEE Internet Computing*, 2(4) :38–46.
- [Magnini et al., 2002] Magnini, B., Negri, M., Prevete, R., and Tanev, H. (2002). Is It the Right Answer ? Exploiting Web Redundancy for Answer Validation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 425–432, Philadelphie, Pennsylvanie.
- [Miller, 1995] Miller, G. (1995). WordNet : a Lexical Database for English. *Communications of the ACM*, 38(1) :39–41.
- [Plamondon and Kosseim, 2002] Plamondon, L. and Kosseim, L. (2002). QUANTUM : A Function-Based Question Answering System. In *Proceedings of the 15th Conference of the Canadian Society for Computational Studies of Intelligence (AI 2002)*, pages 281–292, Calgary, Canada.
- [Plamondon et al., 2002] Plamondon, L., Lapalme, G., and Kosseim, L. (2002). The QUANTUM Question Answering System at TREC-11. In *Notebook Proceedings of The Eleventh Text Retrieval Conference (TREC-11)*, pages 157–165, Gaithersburg, Maryland.
- [Voorhees, 2002] Voorhees, E. M. (2002). Overview of the TREC 2002 Question Answering Track. In *Notebook Proceedings of The Eleventh Text Retrieval Conference (TREC-11)*, pages 115–123, Gaithersburg, Maryland.