Utilisation de connaissances sémantiques pour l'analyse de justifications de réponses à des questions

Vincent Barbier

Utilisation de connaissances sémantiques
Pour l'analyse de justifications
de réponses à des questions

Vincent Barbier

LIMSI-CNRS

Université Paris XI – Orsay

Soutenue le 22 janvier 2009 devant la commission d'examen composée de :

Président du jury	M. Philippe Tarroux, Professeur à l'École Normale Supérieure, LIMSI
Rapporteurs	M. Mohand Boughanem, Professeur à l'Université Paul Sabatier, IRIT
	M. Michael Zock, Directeur de recherches au CNRS, LIF
Examinateurs	M. Olivier Ferret, Chercheur Au CEA-LIST
	Mme Anne Vilnat, Professeur à l'IUT d'Orsay, LIMSI
Directrice	Mme Brigitte Grau, Professeur à l'ENSIIE, LIMSI

Remerciements

Je tiens à remercier ma directrice de thèse, Mme Brigitte Grau, pour avoir encadré mes recherches, pour sa grande patience et pour les conseils précieux qui m'ont guidé pendant ces années de thèse, ainsi que pour son investissement lors de la rédaction du manuscrit.

Je remercie mes rapporteurs, MM. Mohand Boughanem et Michael Zock, pour avoir consacré du temps à lire mon manuscrit.

Je remercie également Mme Anne Vilnat et M. Olivier Ferret pour avoir accepté d'être examinateurs de mon jury.

Je remercie enfin ma famille, mes amis et ma copine Liza pour m'avoir soutenu pendant cette période difficile, ainsi que pour les corrections orthographiques.

Résumé

En raison de l'augmentation constante des quantités d'information disponibles sous format électronique; documents du Web, articles de journaux ou données privées, il est nécessaire de disposer d'outils adaptés pour retrouver les informations recherchées.

Deux voies principales tentent de remédier à ce problème. D'une part, la recherche d'information classique dans laquelle une requête composée de mots-clés est fournie à un moteur de recherche qui sélectionnera des documents en rapport avec ces mots clés. Ce type de recherche est plus approprié pour la recherche d'informations larges. D'autre part, la recherche par systèmes de questions-réponses, où des questions sont posées en langue naturelle et où le système cherche à apporter à l'utilisateur la réponse précise. Dans ce travail, nous nous plaçons dans de cadre des systèmes de questions-réponses et de la recherche d'informations précises factuelles.

Dans le premier chapitre, nous explicitons la notion de réponse à une question, qui doit être composée d'une part de la réponse courte (par exemple, un nom de personne ou une date), mais également d'un passage justificatif qui permette à l'utilisateur de vérifier la réponse. Nous décrivons ensuite le fonctionnement des systèmes de questions-réponses. Ces systèmes comportent généralement trois phases successives. La première, l'analyse de la question, permet d'extraire d'une question en langage naturel des éléments à rechercher dans les réponses. Parmi ces éléments, on trouve notamment le type attendu de la réponse, les termes de la question et les liens syntagmatiques entre ces termes. Les informations extraites sont alors utilisées pour rechercher des documents grâce à un moteur de recherche traditionnel, puis pour les filtrer. Le système effectue généralement plusieurs filtrages successifs : sélection de passages, puis sélection de phrases et enfin la sélection de la réponse exacte.

De plus, nous présentons les différentes campagnes d'évaluation de ces systèmes qui donnent des fils directeurs de l'évolution de la thématique de questions-réponses vers des systèmes capables d'analyser la validité de la justification apportée. Nous notons pour ces systèmes d'analyse une divergence importante entre des méthodes fortement structurées nécessitant d'importantes ressources linguistiques et des techniques fondées sur la réunion de traits hétérogènes mis en commun grâce à un système de décision utilisant l'apprentissage artificiel. Nous nous proposons donc de rechercher une approche médiane, visant à conserver au système la capacité à produire une justification structurée, tout en rendant possible l'intégration d'une grande hétérogénéité de traitements linguistiques de nature, de niveau de granularité et de fiabilité variés.

Dans le chapitre 2, nous décrivons les ressources sémantiques, morphologiques et pragmatiques, et leur utilisation dans les systèmes afin de réduire l'écart entre les formulations de la question et de la réponse. Nous distinguons dans ce chapitre trois types de phénomènes linguistiques constituants de la justification : les variations sémantiques paradigmatiques locales qui permettent de décrire les

variations d'expression entre un terme de la question et un terme de la réponse, la sémantique syntagmatique qui permet de décrire les relations entre les constituants d'une phrase, et enfin, une composante de sémantique énonciative, qui concerne les procédés permettant de relier des parties d'un énoncé situées dans des phrases distinctes, voire des documents distincts (anaphores, coréférences, thématisation).

Dans le chapitre 3, nous proposons un formalisme de représentation des justifications. Ce formalisme intègre les différents types de phénomènes linguistiques vus précédemment. Ce formalisme consiste en un graphe d'appariement entre les éléments extraits de la question et les éléments justificatifs correspondants de la réponse.

Nous décrivons également la construction d'un corpus de questions-réponses et l'annotation des justifications qu'il contient grâce au formalisme choisi. Nous analysons le corpus ainsi obtenu pour déterminer les phénomènes linguistiques présents dans les justifications et leur fréquence. Ce corpus est annoté grâce au formalisme choisi et nous étudions la configuration des justifications en termes de variation sémantique, d'étendue spatiale.

Ce corpus annoté nous permet de mettre en évidence la variété des types de phénomènes sémantiques et guide la conception d'un algorithme permettant d'implémenter les différents phénomènes linguistiques mis en évidence et d'extraire automatiquement les structures de justification. Nous décrivons les résultats de l'algorithme et montrons notamment une amélioration de l'étape de filtrage de documents grâce à la gestion de justifications multiphrases.

Table des matières

Table des figures

Liste des tableaux

Introduction

Devant l'augmentation constante des quantités d'information disponibles sous format électronique, qu'il s'agisse du Web, d'articles de journaux ou de données privées, se pose le problème de l'accès à une information souhaitée. En effet, il est difficile de retrouver une information particulière parmi la masse des informations disponibles.

Deux voies principales existent pour résoudre ce problème. D'une part, la recherche d'information classique (notée RI), s'appuyant sur une interrogation à l'aide de mots-clés plus appropriée pour la recherche d'informations larges et d'autre part, la recherche par systèmes de questions-réponses, où des questions sont posées en langue naturelle et où une réponse précise à la question est attendue. Cette réponse soit s'accompagner d'un contexte qui permette à l'utilisateur de vérifier que la réponse rapportée est correcte, c'est-à-dire, un passage justifiant la réponse. Notre thèse se situe dans le cadre de la recherche d'informations précises et en particulier des systèmes de questions-réponses.

De nombreux systèmes de questions-réponses comme le système QALC du LIMSI[1] sont constitués de trois étapes : analyse de la question, recherche de documents par un moteur de recherche traditionnel, puis filtrage des réponses. Dans de nombreux systèmes, comme dans le système QALC, l'étape de filtrage comprend plusieurs filtrages successifs. Ces filtrages consistent fréquemment en la sélection de passages, puis la sélection de phrases, et enfin la sélection de la réponse exacte.

Notre but est de proposer des traitements permettant d'améliorer les performances de l'étape de filtrage des passages candidats à fournir la bonne réponse, ainsi que d'estimer la fiabilité de ces traitements. Nous nous intéressons en particulier au problème de la mise en évidence, dans un passage candidat, de l'ensemble des informations de ce document permettant de vérifier si la réponse est exacte, en complétant si besoin est l'information du passage grâce à d'autres extraits de texte. Nous nommons justification un ensemble d'informations langagières permettant cette vérification. Elles doivent permettre de vérifier toutes les informations contenues dans la question. Cette justification peut être contenue dans une phrase, dans un paragraphe, dans une page ou dans plusieurs documents. Voici un exemple simple tiré de la campagne d'évaluation TREC11 où la justification (en caractères gras) est contenue dans une phrase :

> Q1394 : In what **country** did the **game** of **croquet originate** ?
> R : Croquet (pronounced cro KAY) **is a 15th century [REP French] sport** that has largely been ...

En revanche, voici un exemple tiré de cette même campagne, où la justification est étendue sur

1 Laboratoire d'Informatique pour la Mécanique et les Sciences de l'Ingénieur

deux phrases :

```
Q1414 : What was the length of the_Wright_brothers' first
flight ?
R : Orville and Wilbur Wright made just four brief flights ...
[...]
In a way , the poor stability of the flyer was a tribute to the_
Wrights ' flying skills .
`` The fact that their first flight was [REP120 feet] and their
last one was 852 feet shows they were learning , '' Watson
said .
```

Dans cet exemple on trouve une chaîne anaphorique qui permet de rattacher « Orville and Wilbur Wright » à « the Wrights », puis « the Wrights » à « their » et ainsi au reste de la justification. On remarquera également que le lien de fraternité de Orville et Wilbur n'est pas indiqué dans le passage. Cependant, comme cette phrase donne une grande partie des informations, il serait dommage de l'éliminer des réponses candidates. Une stratégie pourra consister à vérifier qu'Orville et Wilbur Wright sont frères dans d'autres documents, ou alternativement, que « Orville and Wilbur Wright » est bien une paraphrase de « the Wright brothers ». Dans le cas d'une justification répartie, un système de questions-réponses(SQR) devrait chercher à fournir à l'utilisateur un extrait du document centré autour des éléments de la justification. Ces éléments pourraient être mis en évidence comme dans l'exemple ci-dessus.

Deux types de phénomènes nous intéressent particulièrement. Le premier est le rattachement d'informations distantes. En effet, le filtrage effectué au niveau de la phrase pose un problème car il est difficile de trouver toutes les informations dans une seule phrase. On peut alors espérer trouver des compléments d'information dans d'autres phrases du même document ou encore dans d'autres documents. On peut également faire appel à des ressources encyclopédiques, par exemple Wikipédia. La question qui se pose est d'écrire des programmes capables d'aller chercher ces informations et d'intégrer ces informations dans la justification totale tout en tenant compte que la stratégie de rattachement utilisée peut avoir une fiabilité moindre.

La question suivante illustre le besoin de rattachement d'informations distantes

```
Q1444 : What female leader succeeded Ferdinand_Marcos as
president of the Philippines?
R : In 1986 , President Ferdinand_E._Marcos fled the Philippines
after 20 years of rule in the wake of a tainted election ;
[REP Corazon Aquino] assumed the presidency <.>
```

Dans cet exemple, la caractéristique « female » qui doit être vérifiée pour la réponse « Corazon Aquino » n'est pas présente dans le document. De même la caractéristique « leader » n'est présente dans le document que sous la forme du nom « presidency ». Il est donc nécessaire de vérifier, au moins le caractère « female » de Corazon Aquino, ce qui pourra être obtenu en fouillant l'article « Corazon Aquino » de Wikipédia.

Pour délimiter la justification, nous devons reconnaître les termes de la question dans les passages candidats. La variabilité sémantique entre les termes de la question et leur expression dans les passages est le second point abordé par cette thèse. Entre la question et la réponse, les mots peuvent se trouver substitués à d'autres de sens proches, par exemple, un synonyme ou un hyperonyme[2]. Par exemple, pour la question « How many chromosomes does the human zygote have? », plusieurs réponses utilisent un hyperonyme : « human cells ». De nombreuses relations de ce type sont par exemple fournies par la base de données lexicale WordNet.

2 Mot de sens plus large. *Table* est un hyperonyme de *meuble*.

Ces phénomènes de variation seront nommés sémantique locale. Ces variations peuvent prendre des formes très diverses, par exemple dans le passage précédent, le terme « president » de la question est repris par « presidency » dans le passage réponse, ce qui peut être traité comme une variation de morphologie dérivationnelle. Parfois, la variation est très complexe et demande une inférence plus ou moins poussée pour permettre le rattachement entre question et réponse. Par exemple la question « When did South Dakota become a state? » peut trouver une réponse sous la forme « ... joining the union in 1998 », ce qui nécessite une paraphrase entre deux fragments de texte d'une granularité supérieure au terme : « become a state » et « join the union ». Dans certains cas, cette inférence semble hors de portée d'un système informatique. Notamment, dans l'exemple de la question 1444 (Corazon Aquino) vu précédemment, l'information portée par le verbe « succeed » est présente par la succession de deux événements « president Ferdinand Marcos fled » et « Aquino assumed presidency ». Dans ce cas, un système sera vraisemblablement contraint de considérer l'information comme manquante.

Le problème de la recherche de variations sémantiques est le bruit qu'elle engendre. C'est-à-dire que lorsqu'on tient compte de la possibilité de reformulations au moment de la requête, cela entraîne l'augmentation du nombre de documents non pertinents rapportés et la baisse de la précision du système. Le choix de mots de sens plus distants permet de retrouver plus de documents pertinents mais également de nombreux documents non pertinents ce qui fait que l'information recherchée risque de disparaître dans la grande quantité des documents sans rapport avec la question posée.

Les deux points évoqués ci-dessus tentent de répondre au même but : retrouver une justification la plus complète possible de la réponse. Parfois, plusieurs phénomènes permettent de rapporter la même information. Par exemple, une information peut être trouvée dans le document avec une importante variation sémantique qui la rend peu fiable et dans d'autres textes sous une forme plus directe et donc plus sûre.

Si ce cas est fréquent, cela signifie qu'il est possible de réduire le besoin de reformulations sémantiques en utilisant au maximum les rattachements d'informations distantes. Les informations du contexte du document peuvent aussi aider le choix des variantes sémantiques et, on l'espère, réduire la perte de précision du système.

Le problème soulevé par notre thèse est de proposer une modélisation de la justification évoquée ci-dessus pour effectuer l'appariement questions-réponses, en tenant compte des différentes techniques d'appariement de l'information entre question et réponse, notamment les différentes techniques de rattachement d'informations distantes et les nombreuses techniques de reconnaissances des variations sémantiques des termes. Ces techniques impliquent divers types de ressources sémantiques ou morphologiques, et des niveaux de granularité variables.

Cela pose les questions de savoir quels procédés d'appariement sémantiques, locaux et distants, sont utiles et quelle est leur fiabilité pour l'utilisation dans un système d'appariement questions-réponses. Une question très importante est d'arriver à déterminer quelles variations sémantiques (synonymie, hyponymie) peuvent être utilisées sans dégrader la précision du système d'appariement.

Pour cela il sera nécessaire d'estimer la fréquence de chaque phénomène et savoir dans quelle proportion ils peuvent se substituer les uns aux autres. Notamment, à quel point peut-on éviter l'utilisation de variations sémantiques grâce à des techniques de rattachement d'informations distantes. Cette étude a été effectuée grâce à un corpus que nous avons constitué.

Nous proposerons un algorithme chargé de détecter les justifications en accord avec la modélisation proposée, intégrant de façon modulaire ces différentes connaissances sémantiques provenant de différentes sources et reconnaissant des variations sémantiques à des niveaux de granularité divers.

Dans le premier chapitre nous présenterons l'état de l'art des systèmes de questions réponse et de validation des réponses. Dans le second chapitre, nous traiterons des ressources linguistiques existantes pour l'appariement entre questions et réponses et de l'utilisation qui en est faite par les systèmes. Dans le troisième chapitre, nous présenterons la construction et l'analyse de notre corpus de questions-réponses. Enfin dans le quatrième chapitre, nous décrirons notre algorithme d'extraction de justifications et donnerons ses résultats.

Chapitre 1 : Présentation des systèmes d'extraction d'information

1.1 La recherche d'informations précises

Dans cette section, nous positionnons notre recherche par rapport aux approches de recherche d'information existantes, nous précisons de plus les notions de questions et de justifications dans le cadre des systèmes de questions-réponses.

1.1.1 Recherche d'information

Les techniques de recherche d'information se divisent en deux branches : la branche de la recherche d'information classique, qui consiste à rechercher des documents à partir de mots-clés fournis par l'utilisateur et les systèmes de questions réponses, qui effectuent des recherches à partir de questions formulées en langue naturelle.

Ces deux types de systèmes n'ont pas le même objectif : le but d'un système de recherche d'information classique est de ramener des documents contenant les mots-clés, ce qui fait que ces systèmes sont plus aptes à rapporter des documents couvrant le domaine représenté par ces mots-clés. Au contraire, les systèmes de questions-réponses servent à rechercher une information précise répondant exactement à la question posée.

Un choix qui s'impose lors de la réalisation d'un système de questions-réponses est de savoir si le système travaille sur un domaine fermé, comme le tourisme, la médecine, ou s'il travaille en domaine général. La recherche d'information en domaine fermé permet la construction de ressources fines couvrant le domaine, voire la réalisation d'une théorie du monde, c'est-à-dire un ensemble d'axiomes et de formules logiques permettant d'effectuer des inférences dans ce monde restreint. Dans le cas de la recherche en domaine général, les questions peuvent porter sur n'importe quel sujet. La question peut porter sur une connaissance scientifique, sur l'actualité politique, sur des faits de culture générale, comme par exemple « quelle est la hauteur de la tour Eiffel? ». La vaste étendue des thèmes abordables rend difficile voire impossible la réalisation de ressources couvrant complètement le lexique susceptible d'être rencontré dans les questions et les textes traités. Actuellement, une des seules ressources assurant une couverture étendue est WordNet, un réseau lexical sur l'anglais décrivant les relations sémantiques qu'entretiennent entre eux les mots de la

langue.

Les systèmes de questions-réponses sont censés faciliter l'accès à l'information dans le cas de la recherche d'une information ciblée, plutôt que d'une information générale sur un thème donné. Dominique Laurent compare dans [LAU05] l'efficacité du système de QR QRISTAL et du système de RI classique Google Desktop dans une tâche de recherche d'informations précises, en tenant compte du temps de saisie de la question plus élevé que celui d'une requête, de la vitesse moins grande avec laquelle le système rapporte les fragments de textes (snippets) et de la vitesse de lecture de l'utilisateur. Il obtient que le temps de recherche avec le système de QR est réduit par rapport à celui du système de RI. Dans le cas d'une vitesse de lecture correspondant à des utilisateurs d'un niveau d'études supérieures, estimées à 40 caractères par secondes, ils mettent en évidence un temps d'accès à la bonne réponse plus faible pour le système de questions-réponses (23,8s) que pour le système de RI classique (40,0s). Cet écart se réduit toutefois à 3 secondes dans le cas d'une vitesse de lecture de 100 caractères par seconde, considérée dans cet article comme une limite supérieure largement surévaluée.

Conclusion

Nous positionnons notre thèse dans le cadre de la recherche d'informations précises par le moyen de questions. Nous travaillerons plus précisément dans le cadre des systèmes de questions-réponses, qui sont un moyen adapté de rechercher ce type d'informations.

1.1.2 Qu'est-ce qu'une réponse

Nous étant placés dans le cadre d'un système de questions-réponses, nous devons apporter à l'utilisateur une réponse à la question qu'il a formulée. Or, cette notion qui semble intuitive pose des problèmes de définition. Tout d'abord, la notion de « réponse » n'est pas clairement définie dans le langage courant, pouvant référer à un très court fragment de texte aussi bien qu'à une longue phrase justificative. De plus, différentes personnes peuvent avoir une vision différente sur ce qui constitue une réponse acceptable à une même question.

Réponse et justifications

Premièrement, une réponse peut être considérée comme le fragment de texte minimal contenant l'information recherchée, c'est-à-dire, sans aucun complément, ni élément justificatif. Par exemple, la réponse à « 4 + 5? » est « 9 » ou « neuf », la réponse à « Qui est le président espagnol? » est « José Luis Zappatero » ou « José Luis Rodriguez Zappatero ». Dans le vocabulaire des systèmes de questions-réponses, ce fragment de texte est nommé réponse exacte ou réponse courte, par opposition à la notion de réponse longue, qu'on peut encore nommer phrase ou passage réponse, par laquelle on désigne un fragment de texte plus long, qui peut par exemple être déterminé par un nombre fixé de caractères (250 caractères dans TREC8).

Une réponse, courte ou longue, n'est valide que si le passage dont elle est extraite justifie la réponse. En effet, le but des systèmes de questions-réponses se définit par rapport à un besoin de l'utilisateur d'obtenir une réponse avec un contexte, qui lui permette de vérifier que la réponse proposée est la bonne.

Les campagnes d'évaluation des systèmes de questions-réponses sont fondées sur ces notions d'exactitude de la réponse et de pertinence de la justification. Dans ce cadre, une réponse est considérée comme correcte, ou pertinente, si elle est justifiée par le document dont elle est extraite.

On pourra se référer aux consignes d'une des nombreuses campagnes, par exemple dans CLEF 2004[3]. Le but de cette campagne est de rapporter des réponses courtes, c'est-à-dire uniquement le fragment de texte qui constitue l'élément d'information demandé. Les juges doivent noter ces réponses des systèmes selon la classification suivante :

- incorrect : la réponse courte rapportée ne correspond pas à une réponse valide (notion de correction de la réponse)
- unsupported : la réponse courte rapportée est valide mais n'est pas justifiée (supported) par le document.
- non-exact : la réponse courte rapportée contient trop ou trop peu d'éléments d'information. Par exemple si la question porte sur la date d'un événement, la réponse courte donnant le jour et le mois est considérée comme insuffisante.
- correct : La réponse courte rapportée est correcte et justifiée par le document.

La première difficulté pour déterminer si une réponse est justifiée est que certaines informations peuvent être omises. Si l'on se place du point de vue de l'utilisateur ayant posé la question, celui-ci dispose d'un certain nombre de connaissances préalables, toute connaissance possédée par l'utilisateur peut être omise. Dans les campagnes d'évaluation, cette connaissance présupposée est laissée au jugement des assesseurs humains, chargés de la validation des réponses rapportées par les participants.

Du point de vue du système, une justification partielle est acceptable à condition que tout élément manquant puisse être vérifié par quelque ressource. Certaines informations présentes dans la question, comme le type attendu de la réponse ou les modifieurs des noms sont souvent absents des réponses. Par exemple, dans la question : « Quel volcan a détruit Pompéi? », la réponse comportera rarement le mot « volcan », mais celui-ci pourra être obtenu par vérification du type de la réponse « Vesuve », par exemple en regardant dans la hiérarchie de WordNet ou dans l'entête de Wikipédia, qui fourniront l'information « le Vésuve est un volcan ». En conséquence, la justification d'une question peut s'étendre sur plusieurs documents.

Contexte et unicité de la réponse

Karen Sparck-Jones précise que la notion de réponse telle qu'elle est utilisée dans les campagnes de TREC repose sur des fondements approximatifs [SPA03]. Premièrement, la réponse attendue n'est pas toujours unique, ce qui pose le problème de savoir quelles réponses on doit accepter et rejeter.

Une première raison à ceci est que la question peut être ambiguë. Par exemple, en l'absence de tout contexte, la question « Where is the Taj Mahal? » est ambiguë car le Taj Mahal est à la fois le célèbre monument indien et un casino à Atlantic City, aux États-Unis. Le choix de la réponse ne peut être assuré que par la présence d'un contexte, qui précise les désirs de l'utilisateur.

Dans une situation réelle, ce contexte serait défini par la situation dans laquelle la question est posée, par exemple l'emplacement géographique des interlocuteurs, et la succession de la conversation. Par exemple, une phrase antérieure à la question « Où est le Taj Mahal? » pourrait être « Je suis en vacances à Atlantic City, pourriez-vous me renseigner sur les attractions touristiques des environs? ». La campagne TREC propose depuis 2004 de travailler sur des questions enchaînées, c'est-à-dire que les questions sont organisées par lots qui partagent un thème commun, chaque nouvelle question devant être traitée en tenant compte du contexte introduit par les précédentes.

3 http://clef-qa.itc.it/2004/guidelines.html

Cependant, dans le cadre des campagnes d'évaluation, le contexte n'est pas défini. Le contexte implicite de ces campagnes est celui de la recherche d'informations de culture générale. On peut y ajouter quelques informations contextuelles selon le pays où est organisée la campagne et la date de la campagne : par exemple, dans la campagne TREC11, on trouve les questions : « Where does the vice president live when in office? » « What year did Alaska become a state? », qui en France seraient posées en précisant : « le vice président des États-Unis » « un état américain ». Cependant, dans la grande majorité des questions le contexte géographique est précisé de sorte que les questions sont rarement ambiguës, comme par exemple, « Which U.S. state is the leading corn producer? ».

Précision attendue de la réponse

Le niveau de précision de la réponse à choisir pose aussi problème. Le choix qui a été traditionnellement fait est que le système doit apporter une réponse exacte. Cependant, Sparck-Jones objecte que dans le contexte d'une recherche de QR interactive, des réponses qui sont rejetées par la campagne TREC pourraient cependant aider l'utilisateur à combler son manque d'information et l'aider à trouver la réponse. Cette approche est d'autant plus nécessaire que certaines questions n'ont pas de réponse exacte dans la collection interrogée.

Par ex : "Who is John Sulton?" pourrait recevoir une réponse partielle pertinente telle que "Former director of the Sanger Institute" ou "the nematode genome man". La première ou la seconde réponse permettent par exemple à l'être humain de déduire que John Sulton travaille dans le domaine de la génétique, ce qui pourra être utilisé ultérieurement pour affiner la recherche. TREC 2003 introduit cette notion d'utilité d'une réponse approximative, mais seulement pour les questions appelant une définition.

Nous plaçant dans le cadre d'un système n'interagissant pas avec l'utilisateur, nous ne pouvons pas profiter d'un quelconque retour de celui-ci. Lors des campagnes d'évaluation, le niveau de précision attendu est évalué par des juges humains. On peut cependant donner la règle qu'une bonne réponse est une réponse qui permet de vérifier toutes les informations de la question. Par exemple, pour la question de CLEF 2005 :

```
Q26 :Quel politicien libéral était ministre de la santé en
Italie de 1989 à 1993 ?
```

Il faut trouver un document parlant d'une personne qui soit politicien libéral, ministre de la santé, et l'information doit donner l'intervalle de temps ou du moins une information temporelle incluse dans cet intervalle. Si une de ces informations manque, l'utilisateur du système devra vérifier celle-ci, soit par ses connaissances personnelles, soit en cherchant dans d'autres documents. Le système de questions-réponses devrait dans ce cas réunir plusieurs documents de façon à : 1) vérifier que la justification trouvée est valide; 2) fournir à l'utilisateur les indications qui lui manquent.

Conclusion

La conception d'un système de recherche d'information repose sur un certain nombre de choix. Nous nous positionnons dans le cadre des systèmes de questions-réponses, par rapport aux systèmes de RI classique, parce que les premiers sont adaptés de façon plus naturelle à la recherche d'information précises. Nous ne nous occuperons pas de la notion d'interactivité qui ne paraît pas nécessaire pour notre étude. Nous nous focaliserons sur la notion de justification, qui nous semble en effet la notion fondamentale en matière de questions-réponses.

Rappelons toutefois que parallèlement à ce but de recherche d'une justification, un système de

questions-réponses peut choisir d'autres alternatives, comme :

- Fournir un ordonnancement des passages rapportés. Cette optique laisse à l'utilisateur le soin de vérifier la présence d'une justification.
- Fournir des compléments d'information à l'utilisateur pour l'aider à reformuler sa requête.
- Poser des questions à l'utilisateur pour l'aider à affiner sa requête.

1.1.3 Évaluation des systèmes de questions-réponses

Le développement des systèmes de questions-réponses est encouragé et guidé par les campagnes d'évaluation. Celles-ci permettent de fixer des normes pour les méthodes d'évaluation. Tout d'abord, elles apportent comme nous l'avons vu, une définition précise et partagée de ce qu'est une réponse.

De plus, ces campagnes permettent de faire porter l'accent sur des besoins de perfectionnement particuliers des systèmes, et principalement, des étapes de filtrage. Nous allons dans cette section décrire les différentes tâches et mesures proposées dans ces campagnes et les traitements sur lesquels elles permettent d'insister.

Précision, Rappel, F-mesure

La précision et le rappel sont deux mesures très utilisées en recherche d'information. La première permet d'estimer la capacité du système à ramener dans les premières positions des documents importants et d'exclure de ces positions des documents non-pertinents. La seconde sert à mesurer la capacité à ramener la totalité des documents pertinents.

$$Precision = \frac{\text{nombre de documents pertinents rapportés par le système}}{\text{nombre de documents rapportés par le système}}$$

$$Rappel = \frac{\text{nombre de documents pertinents rapportés par le système}}{\text{nombre de documents pertinents existants}}$$

Le rappel nécessite la connaissance du nombre de documents pertinents, ce qui est difficilement faisable sur des tâches de recherche d'information réelles. Cependant, cela est possible dans le cadre d'une ressource de référence où les documents pertinents ont été recherchés et étiquetés. Bien que coûteuse en temps, la création d'une telle référence est envisageable, comme par exemple, [GIL05] sur le français utilisant le jeu de questions et les documents de la campagne EQUER.

Enfin, la F-Mesure donne une évaluation intermédiaire entre la précision et le rappel. Sa formule dépend d'un paramètre alpha qui permet de faire s'approcher cette mesure de la mesure de précision, $alpha = \infty$, ou de la mesure de rappel, $alpha = 0$.

$$F_{alpha} = \frac{(1 + alpha) * précision * rappel}{alpha * précision + rappel}$$

Le paramètre *alpha=1* donne une F-mesure faisant intervenir de façon symétrique la précision et le rappel.

MRR

Le MRR (Mean Reciprocal Rank, rang inverse moyen) est la mesure traditionnelle des campagnes d'évaluation TREC. Elle vise à favoriser les bonnes réponses apportées aux toutes premières positions. Elle consiste en effet en la moyenne sur l'ensemble des questions posées, de l'inverse du rang auquel apparaît la première réponse correcte. Ainsi, si pour une question donnée le

système ramène la bonne réponse en première position, elle obtiendra un score de 1; de 0,5 si la réponse arrive en deuxième position, etc. Les réponses au delà du rang 10 apportent moins de 0,1 points, soit peu de gain par rapport à l'absence de réponse.

$$MRR = \frac{1}{Nb\,questions} \sum_{i=1}^{nbquestions} \frac{1}{rank_i}$$

Incitation à l'auto-évaluation.

Les campagnes d'évaluation ont récemment adopté des mesures incitant à rendre les systèmes capables de s'auto-évaluer, évaluant les systèmes par des mesures qui les forcent d'estimer la fiabilité de leurs réponses. De plus, la campagne CLEF propose une sous-tâche dédiée à l'évaluation de la validité réponse : AVE. Des mesures comme le CWS(Confidence Weighted Score) ou les mesures K et K1 prennent en compte l'estimation de fiabilité du système.

Le CWS [VOO02] est une mesure utilisée lors des campagnes d'évaluation de systèmes de QR, TREC02 et CLEF04, ainsi que dans la campagne d'évaluation RTE(Recognising Textual Entailment, reconnaissance de l'implication textuelle) en 2005. Elle est conçue pour des systèmes renvoyant une unique réponse par question. Le CWS ne requiert pas que le système attribue un score de fiabilité à chaque réponse. Il suffit pour le système de classer ses réponses par ordre décroissant de fiabilité. Cet ordonnancement sera dénoté $(q1,r1)..(q_n,r_n)$ ci-dessous :

$$CWS = \frac{1}{Nb\ questions} \sum_{i=1}^{Nbquestions} \frac{Nb\ réponses\ correctes\ parmi\ r_1..r_i}{i}$$

Les mesures K et K1 (introduites dans CLEF en 2005) : la mesure K est conçue pour des questions acceptant plusieurs réponses et pour lesquelles les systèmes ramènent plusieurs propositions. Contrairement au CWS, ces mesures requièrent le calcul par le système d'un score de fiabilité du résultat compris entre 0 et 1. Elle encourage donc l'estimation correcte de la probabilité de réussite du système.

$$K(sys) = \frac{1}{Nb\ questions} \sum_{q \in questions} \frac{\sum_{r \in answers(q)} score(r).eval(r)}{max(Existing(i), Given(sys,i))}$$

avec :

sys : le système de QR évalué

eval(r) : 1 si la réponse retournée par le système est jugée correcte, -1 si elle est jugée incorrecte, 0 si c'est une réponse répétée.

score(r) : un score de confiance dans [0,1].

Existing(i) : nombre de réponses existantes dans la collection pour la question q;

Given(sys,i) : nombre de réponses rapportées par le système sys

La mesure K1 est une simplification de celle-ci pour le cas où le système ramène une seule réponse par question, comme pour le CWS.

$$K1(sys) = \frac{1}{nbquestions} \sum_{r \in answers} score(r).eval(r)$$

Malgré son intérêt, cette mesure n'a pas à notre connaissance été utilisée dans d'autres campagnes.

Conclusion

Les campagnes d'évaluation des systèmes de questions-réponses fournissent un cadre d'évaluation précis et reproductible, et permettent d'uniformiser les mesures de performance des systèmes afin de comparer les systèmes entre eux. Elles créent également une dynamique dans ce domaine, notamment en encourageant la création de systèmes, en guidant les développements effectués sur ces systèmes et en générant de nombreux jeux de données. Nous utilisons dans notre thèse principalement les jeux de données des campagnes TREC ainsi que la collection de documents utilisée pour ces campagnes, nommée AQUAINT.

L'activité des campagnes de questions-réponses met en évidence la complexité de la notion de justification d'une réponse, qui reste à l'heure actuelle peu définie et laissée au jugement humain. Par ailleurs les campagnes récentes mettent l'accent sur la capacité des systèmes à s'auto-évaluer.

Notre thèse a pour but de décrire la configuration des justifications et de permettre l'extraction automatique de celles-ci. Elle s'attache aussi à fournir un moyen d'évaluer la fiabilité des justifications extraites.

1.2 Fonctionnement d'un système de questions-réponses

Notre développement d'un système d'extraction de justifications s'intègre dans la problématique de questions-réponses. En effet, pour fournir une représentation judicieuse des éléments justifiant la réponse, ce système nécessite une analyse fine des éléments requis par la question. Il s'appuie sur la chaîne de questions-réponses QALC pour obtenir des passages de documents susceptibles de répondre à la requête. Il utilise et complète l'analyse que le système fait de ces passages et tente de mettre en relation les critères de validation de la justification que l'analyse de la question a mis en évidence, avec des éléments du passage susceptibles de les satisfaire.

Notre système peut être vu également comme un module d'un système de questions-réponses puisqu'il aide, par une estimation de la fiabilité des justifications extraites des passages, à estimer la pertinence des réponses rapportées, et par ce moyen, à classer les réponses par ordre de pertinence.

Nous présenterons ci-après l'architecture des systèmes de questions-réponses, en mettant l'accent sur le système QALC.

1.2.1 Architecture générale

Un système de questions-réponses est généralement constitué d'une chaîne de traitements partant de la question, et extrayant des informations linguistiques qui seront décrites en section 1.2.2. Ces informations contiennent des termes qui seront fournis à un moteur de recherche classique pour obtenir des documents. Ces documents seront ensuite filtrés par des techniques variées faisant intervenir des critères de filtrage très variables selon les systèmes. On peut trouver parmi celles-ci la vérification de la présence des termes de la question ou de leur reformulations, la vérification que le type de réponse requis par la question est bien présent dans le document et la vérification de la présence de relations syntaxiques identiques entre la question et la réponse. Les différents critères repérés reçoivent en général une pondération qui permet de classer les documents

selon un score de fiabilité permettant de placer en première position ceux qui semblent les plus susceptibles de contenir la question.

On peut prendre comme exemples de tels systèmes le système FALCON [HAR00] qui possède une chaîne de traitements très complète, intégrant notamment un système de preuves logiques, ou le système QALC du LIMSI[FER01a] présentant des techniques nécessitant moins de ressources. Les critères de filtrage se répartissent selon le découpage habituel en linguistique en critères paradigmatiques et syntagmatiques.

Nous utilisons cette division tout au long de thèse, c'est pourquoi nous prenons le temps de définir brièvement cette distinction fondamentale. Rappelons que les phénomènes linguistiques se décomposent en deux axes ; syntagmatique et paradigmatique. L'axe syntagmatique représente les liens entre les mots d'un énoncé. Les relations syntaxiques reliant deux mots ou encore le nombre de mots séparant deux mots d'un énoncé, sont des informations de nature syntagmatique. L'axe paradigmatique représente la possibilité de variation dans la langue.

La notion d'axe paradigmatique se fonde sur la notion de classe de substitution et permet de représenter des phénomènes divers. Par exemple, la notion grammaticale de syntagme verbal peut se définir et se justifier par la constatation de l'existence d'une classe paradigmatique contenant tous les syntagmes nominaux possibles, les éléments de cette classe étant substituables entre eux en préservant la validité syntaxique de la phrase, sans considération du sens ou de la plausibilité de l'énoncé, par exemple : *(Gainsbourg | Le chat | Le chanteur | Le deutérium | ...) s'est mis à boire*.

Dans le cadre de la sémantique qui nous intéresse plus, on peut définir l'ensemble des mots et expressions pouvant se substituer les uns aux autres dans un contexte donné. Par exemple : *(Gainsbourg | Serge Gainsbourg | L'auteur du Poinçonneur des Lilas | Le mari de Jane Birkin ...)* constitue une telle classe paradigmatique.

Dans le cadre des systèmes de questions-réponses, la composante paradigmatique du filtrage consiste à repérer dans la réponse les mots de la question et leurs variations. Les différents systèmes admettent des variations plus ou moins importantes. Il est nécessaire de restreindre les reformulations autorisées en raison du bruit qu'elles engendrent. En effet, plus on augmente le nombre de variantes admises pour les mots de la question et plus on accepte comme pertinent des passages sans rapport avec l'information voulue.

Les systèmes de questions-réponses doivent donc utiliser d'autres informations pour contrer le bruit. Les techniques de filtrage comptent notamment la prise en compte de contraintes syntagmatiques, que celles-ci soient représentées par une notion de densité des mots dans le passage [CHA02], par des relations syntaxiques [TAN05] ou des relations étiquetées sémantiquement [HART04].

1.2.2 Analyse de la question

L'analyse des questions est une étape indispensable des systèmes de questions-réponses puisqu'elle permet au système de déterminer les informations qui seront ensuite recherchées dans les documents. Nous décrirons les informations suivantes : catégorie de la question, type attendu de la réponse, termes de la question et focus.

1.2.2.1 La catégorie de la question

Pour répondre à une question posée, une information essentielle est le type de la réponse attendue. Un premier niveau de typologie sépare les grands types de questions :

- Questions factuelles simples : Ce sont des questions qui demandent comme réponse un élément d'information précis exprimable de façon concise, généralement sous la forme d'un groupe nominal ou d'une expression numérique. Parmi ces questions, on trouvera « Qui a écrit "la bicyclette bleue" », « Quel volcan a détruit Pompéi? » ou « Combien de films Ingmar Bergman a-t-il réalisé? »

- Questions booléennes : Ces questions attendent une réponse par oui ou par non, ce type de question nécessite d'extraire un passage justifiant la réponse.

- Questions de définition : Ces questions attendent généralement la caractéristique d'une personne, d'une organisation ou autres. Par exemple « Qui est Jacques Chirac », (Président français). Elles peuvent aussi demander la forme longue d'un sigle. « Qu'est-ce que l'OTAN? »

- Questions attendant une liste de réponses. Nous n'avons pas traité ce type de questions.

À côté de ces catégories acceptant des réponses courtes - ou des listes de réponses courtes dans le dernier cas - on trouve des questions requérant des réponses complexes sous formes d'une ou plusieurs phrases entières. On peut notamment citer les questions demandant une explication « Pourquoi...? », une procédure « Comment fait-on...? » ou un résumé « Que doit-on savoir sur...? ».

Les premières évaluations de QR ont porté sur les questions factuelles et de définition, comme c'est le cas dans TREC11. Les campagnes TREC comportent une évaluation dédiée aux questions de type liste depuis 2001. Les questions booléennes sont présentes dans la campagne EQUER, en 2004.

Notre but étant d'étudier les variations de formulation des mots, cette variabilité de la typologie des questions était pour nous secondaire et nous nous sommes limités à des questions de type factuel simple. Notons que les questions de type définition nous intéressent moins, car elles fournissent en général très peu de matériel à l'étude des variations sémantiques. Cela est dû à leur forme habituelle très succincte « Qui est X? », « Qu'est-ce que Y? ».

1.2.2.2 Type attendu de la réponse

Les questions factuelles simples se sous-catégorisent ensuite par le type attendu de la réponse. Ce type peut correspondre à un ou plusieurs types d'entités nommées. Si ce n'est pas le cas, ce type est catégorisé comme type général.

Entités nommées

Une entité nommée est dans son sens exact une expression figée visant à désigner un référent, c'est-à-dire une entité du monde, unique. Par exemple, les noms de personnes, d'organisations et de lieux sont des entités nommées.

On étend généralement cette définition aux entités numériques, telles que les longueurs, les différentes mesures de grandeurs physiques (longueur, vitesse, température,...) et les montants financiers, ainsi qu'aux entités temporelles, qui contiennent des dates absolues : « le 8 septembre 1980 », « en 2008 » ou relatives : « en juin [de cette année] », « le 13 », « ce vendredi ».

Les entités nommées utilisées par le système QALC sont répertoriées dans la figure 1.

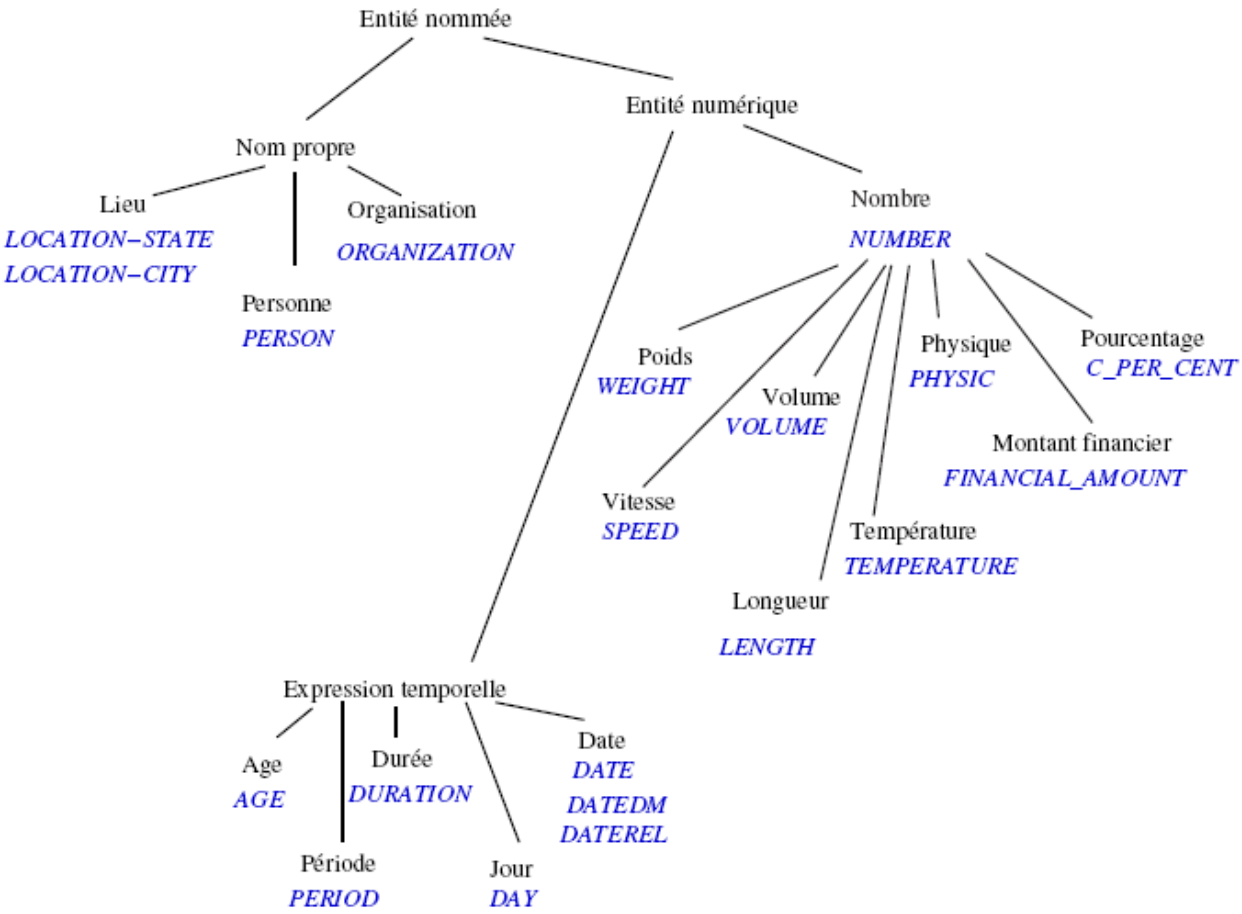


Fig. 1: Les entités nommées reconnues par le système QALC

Les systèmes de questions-réponses utilisent un ensemble plus ou moins fourni d'entités nommées. Par exemple, le système FALCON [HAR00] possède un jeu d'entités nommées similaire à celui du système QALC, auquel sont ajoutées d'autres catégories d'entités nommées, telles que nationality, continent, disease, alphabet, qui respectent la propriété de référent unique.

L'ensemble des catégories utilisées peut être étendu à des catégories de mot ne respectant pas cette propriété. Dans un but pragmatique, on pourra étendre les catégories dites d'entités nommées à toute catégorie de mots facilement identifiable, par exemple, le système FALCON possède les catégories bird, mammal et reptile, qui ne feront pas référence à des entités nommées, comme « Tom » ou « Jerry », mais à des hyponymes comme « cat » ou « mouse ».

Types généraux

Définition

Les catégories dites « type général » sont celles qui ne sont pas gérées par un type d'entités nommées. Les questions appelant un type général précisent explicitement le type attendu :

```
Q: Quel est l'animal le plus rapide du monde?
Q: Quel volcan a détruit Pompéi?
```

Une réponse appelée par un type général peut éventuellement être également traitée par une entité nommée. La réponse attendue peut aussi être un nom commun, simple ou composé, et par conséquent, ne servant pas à désigner de référent unique, celui-ci n'est pas une entité nommée. Des

exemples de tels types sont « animal », « device », qui correspondront respectivement à des termes comme « tiger », « jaguar » ou « telegraph », « violin ».

Toutefois, on peut aussi trouver sous cette étiquette les entités nommées qui ne bénéficieraient pas d'un traitement spécifique. La raison de cette absence de traitement particulier est qu'il existe un grand nombre de catégories d'entités nommées peu fréquentes. Par exemple la catégorie d'entités nommées « Volcano », contenant entre autres les entités nommées « Vesuvius » et « Etna », n'est généralement pas traitée par les systèmes comme une entité nommée.

La différence entre ces deux types est par conséquent principalement opérationnelle. Elle tient dans la méthode utilisée pour vérifier l'équivalence entre le type attendu et l'élément correspondant dans le document. Sont considérées comme entités nommées les types qui sont repérables par un programme dédié. Les entités nommées moins fréquentes, comme « Volcano » seront traitées comme un type général.

Utilisation d'un type général

Le type général est donné par un groupe nominal et de ce fait peut être compté parmi les termes de la question, cependant son comportement en terme d'appariement entre question et passages candidats est totalement différent des autres termes. En effet, le type général, comme le type EN, est mis en correspondance dans le passage réponse, avec l'élément réponse lui-même, qui doit être une instance[4] de ce type. Bien qu'il arrive parfois que le type soit précisé dans la phrase réponse, cet élément se trouve fréquemment omis[GRAU08]. Les deux exemples de recherche suivants illustrent ces deux cas, les termes en gras sont ceux possédant un lien sémantique avec le type général de la question :

Q1 : Quel est l'animal le plus rapide du monde?
R1 : Le ***guépard*** *est l'****animal*** *terrestre le plus rapide du monde.*

Q2: Quel est le nom du volcan qui a détruit Pompéi?
R2 : Le ***Vésuve*** *a enseveli Pompéi sous une avalanche de cendres brûlantes.*

Vérification de type général

Comme pour les entités nommées, il faut vérifier que le type général requis par la question se retrouve bien dans la réponse. Pour que cela soit vérifié, il faut qu'un des éléments de la réponse soit un hyponyme du type général. La difficulté est que, contrairement aux entités nommées qui possèdent des techniques de validation adaptées à chaque type, la vérification des types généraux doit se faire par des méthodes génériques. Il existe plusieurs méthodes pour les trouver.

La première méthode consiste à utiliser une ontologie telle que WordNet. Une telle ressource est constituée manuellement et fournit des relations d'hyponymie[5] et d'instanciation qui nous sont utiles. WordNet permet par exemple de retrouver les noms des monnaies des différents pays, qui seront les hyponymes de monetary_unit#n#15[6] ou ceux des couleurs : hyponymes de colour#n#6.

Le problème de cette méthode est qu'aucune ressource ne peut avoir une couverture exhaustive des termes employés dans un corpus. Par exemple, WordNet ne recense pas le lien entre « Michael

4 Dans WordNet, la relation d'instanciation relie un concept de l'ontologie à une entité nommée de ce type, par exemple : actor -(instance)-> Buster Keaton

5 Lien entre deux concepts de l'ontologie, le concept source étant plus général que le concept cible. Par exemple actor -(hypo)-> comedian

6 Nous utilisons pour représenter les entrées de la base de données lexicale WordNet la notation fort utile rencontrée dans le module perl d'accès à WordNet : WordNet::QueryData. L'entrée se constitue des éléments suivants : "lemme#catégorie_grammaticale#numéro_de_sens" , le sens servant à distinguer les différentes entrées de lemme et catégorie grammaticale identiques.

Jordan » et « basketball player ». Il faut de plus noter le danger l'utilisation de WN comme validateur du type général car certaines classes peuvent être couvertes partiellement. C'est le cas, par exemple, de la catégorie « symbol, emblem » de WordNet, qui a pour hyponymes les symboles les plus connus du point de vue du monde américain, mais qui ne contient pas les emblèmes d'un pays arbitraire. Par exemple, on y trouvera les symboles des partis républicain et démocrate des États-Unis, respectivement « elephant » et « donkey ». Par contre, « Marianne »,« Eiffel Tower » qui répondraient à la question « What is a symbol of France? » ne sont pas reliés au concept « symbol » de WordNet. Ils sont absents de WordNet dont le lexique a été principalement développé à partir des usages lexicaux américains. De ce fait, l'utilisation de WordNet comme validateur de type général peut favoriser certaines réponses par rapport à d'autres.

L'autre solution pour valider l'appartenance d'un terme à un type est de vérifier cette information dans des documents. D'une part, il existe des systèmes d'extraction de connaissances à partir de textes. On peut trouver des formulations, par exemple dans des articles de journaux, permettant d'inférer des types. La phrase suivante permet d'inférer l'appartenance de « Jackie Robinson » au type général « athlete » :

```
I have regarded Jackie Robinson as the greatest athlete I have
ever seen for one.
```

Il faut noter cependant que ces connaissances, de type ontologique, ne sont pas toujours exprimées dans les collections d'articles de journaux où sont recherchées les réponses, car elles appartiennent généralement aux domaines de la connaissance du monde ou de la culture générale. Une solution pour pallier ce problème est d'utiliser des ressources encyclopédiques, telles que Wikipédia.

1.2.2.3 Termes de la question

L'extraction des termes de la question consiste à éliminer les mots vides et à reconnaître les termes composés. Les mots vides sont d'une part les mots grammaticaux (déterminants, pronoms, conjonctions...), qui correspondent à toutes les catégories grammaticales à l'exclusion des noms, des verbes, des adjectifs et des adverbes. D'autre part, certains mots non vides ne sont pas considérés comme termes, soit parce qu'ils sont des mots outils servant à poser la question (en gras ci-dessous) ou bien, parce qu'ils sont traités par une technique adaptée : c'est le cas notamment du type général (souligné) :

« **Nommez** une ville ayant accueilli le tour de France en 1999. »
« Quel est le **nom** du volcan qui détruisit Pompéi? »

La reconnaissance des termes est effectuée dans le système QALC par un simple patron morphosyntaxique, c'est-à-dire, définissant une succession de catégories grammaticales. Pour l'anglais, il s'agit de :

```
(JJ|NN|NP|VBG)?(JJ|NN|NP|VBG)(NP|NN) | VBD | NN | NP | CD
avec JJ=adjectif, NN=nom commun, NP=nom propre, VBG=verbe en
-ing, VBD=verbe au prétérit, CD=nombre.
```

Les termes peuvent s'emboîter, par exemple : « US helicopter pilot » est un terme contenant le sous-terme « helicopter pilot » lui-même contenant les sous-termes « helicopter » et « pilot ». Les termes seront ensuite recherchés dans les documents avec ou sans variation sémantique et seront alors plus ou moins pondérés selon le type de la variation.

1.2.2.4 Focus

Le focus joue un rôle important dans les systèmes de questions-réponses [FER01b] car il s'agit d'un terme de la question considéré comme plus important que les autres. La notion de focus est définie de façon imprécise dans la littérature. Ce terme regroupe plusieurs éléments de définition. El Ayari [ELA07] en recense trois.

La première est la définition originale du focus comme donnée par Lehnert en 1978 [LEH78]. Il s'agit du concept de la question sur lequel porte le besoin d'information(1). Lehnert donne l'exemple de la question suivante et d'une réponse inattendue R1 :

```
Q: Why did John fly to Siberia?(focus=Siberia)
R1 : Because it's too far to walk. (focus=fly)
R2 : Because he wanted to see the Baikal lake. (focus=Siberia)
```

La première réponse ne répond vraisemblablement pas au besoin d'information de celui qui aurait posé la question parce que la personne qui répond s'est trompée dans la sélection du focus.

Cette première définition entraîne deux propriétés. D'une part, le focus peut aussi être considéré comme un des termes les plus importants de la question(2). Par exemple voici deux questions et leurs focus (en gras) :

```
« What year did Wilt Chamberlain score 100 points? »
« Who was the first governor of Alaska? »
```

De ce point de vue, le terme focus doit être retrouvé dans le passage candidat prioritairement aux autres termes, que ce soit sous une forme exacte ou avec variation. L'utilisation du focus pour la sélection des phrases est citée dans [FER01b].

D'autre part, le focus peut être considéré comme le terme qui devra se retrouver dans les passages candidats à proximité de la réponse(3). Cette propriété découle de la première définition, qui décrit le focus comme l'élément de la question sur lequel doit porter la réponse. Dans le système QALC, la détection des réponses de type général, qui comme nous l'avons vu ne jouit pas d'une technique spécifique de reconnaissance, se fait par l'utilisation de patrons syntaxiques permettant de relier la réponse au focus.

1.2.3 Extraction et filtrage des documents

Les informations extraites de la question sont ensuite utilisées pour sélectionner des documents puis des passages ou des phrases et enfin des réponses courtes supposées pertinentes. Pour ce faire, les systèmes de questions-réponses doivent effectuer des filtrages efficaces en intégrant des types de traitements et de connaissances linguistiques variées. L'extraction des justifications, qui est le but de notre recherche, peut être considérée comme une technique de sélection de passages et de la réponse courte pertinents. De plus notre but est de proposer un formalisme permettant de réunir les différents traitements de filtrage existants pour obtenir un modèle structuré de la justification de la réponse. Nous présenterons dans cette partie le cheminement de l'information textuelle de la collection de documents aux passages sélectionnés puis à la réponse courte. Nous détaillerons les différentes méthodes de filtrage et de sélection existantes.

1.2.3.1 Extraction des documents

À partir des termes extraits de l'analyse de la question, le système de questions-réponses

recherche des documents contenant ces termes. Ces documents peuvent être de grande taille. Par exemple, le système QALC récupère des fragments d'articles de journaux prédécoupés correspondant à une vingtaine de lignes de texte.

Contenu de la base de documents

Il existe deux types de collections de documents. Soit il s'agit d'une collection fermée, quoique éventuellement de grande taille, par exemple une collection d'articles de journaux, soit il s'agit d'une collection ouverte comme le Web. Dans ce dernier cas, les documents arrivent non traités et toutes les opérations doivent être effectuées après leur récupération.

Dans le cas d'une collection fermée, celle-ci est stockée de façon statique, ce qui autorise à effectuer des prétraitements sur le texte. Parmi les prétraitements les plus fréquents, on compte la normalisation du texte : on peut supprimer des caractères étrangers, qui ne manquent pas d'apparaître même dans une version informatisée d'articles de journaux, et restituer de la casse pour les mots présentés tout en majuscules.

On peut aussi ajouter à l'information textuelle des informations de différentes natures, à condition que l'analyse de celles-ci soit indépendante de la question posée.

L'utilité de cet ajout est double. Premièrement, permettre d'effectuer des requêtes sur ces éléments et deuxièmement, réduire la durée des traitements ultérieurs, qui s'effectuent après la formulation de la requête par l'utilisateur et qui par conséquent augmentent le temps d'attente de celui-ci.

Les informations ajoutées peuvent être par exemple :

- Meta-informations et contexte : les informations de contexte du document, telles que le titre, la date de l'article, la localisation géographique du contenu, comme dans [JOC03], ainsi que des informations permettant de garder trace de l'origine du document comme l'identifiant de l'article et un numéro de passage.
- Segmentation des mots (tokenisation) et des phrases.
- Étiquetage morphosyntaxique : présentation du texte en ajoutant les lemmes et les catégories grammaticales des mots. Notons que la présence de ces informations n'est pas toujours utile pour la récupération des documents étant donné qu'il est peu pertinent d'effectuer une requête sur une catégorie grammaticale et que, d'autre part, il est possible d'utiliser une fonctionnalité de racinisation (stemming) avec certains moteurs de recherche. Cependant, le moteur de recherche Lucene utilisé par la chaîne QALC pour les documents en français ne gère pas de racinisation ou de lemmatisation.
- Étiquetage des entités nommées : délimitation et typage des entités nommées présentes dans les documents. Cet étiquetage, que j'ai ajouté aux prétraitements de la collection utilisée par le système QALC, permet de réduire le temps de travail du système. Il pourrait également être utilisé pour préciser les requêtes en indiquant le type de réponse attendu, ce qui réduirait le nombre de documents rapportés et améliorerait de nouveau le temps de calcul.

On peut pousser loin ce processus d'indexation. Le système QRISTAL indexe notamment les relations syntaxiques, les champs thématiques, des éléments de désambiguïsation sémantique se fondant sur une ontologie à deux niveaux. Le système InSicht [HART05] pousse le processus à son maximum en stockant pour tous les documents de la collection, l'analyse de ses phrases sous la forme d'un réseau sémantique ou de plusieurs sous-réseaux en cas d'échec [HART04].

Ces informations peuvent servir pour l'interrogation ou comme simple prétraitement. La base de données peut contenir un nombre variable de prétraitements. Effectuer ceux-ci présente l'avantage

d'accélérer la phase de traitement lors de l'appel au système et de permettre de donner la réponse à l'utilisateur en un temps plus bref.

Découpage de la collection

Dans le cas de la collection utilisée par le système QALC les documents sont découpés en passages plus courts, d'une taille souhaitée entre 300 et 400 mots. Les passages obtenus sont constitués d'environ 5 paragraphes ou 11 phrases. Ce découpage permet d'une part de limiter la taille des textes rapportés, et d'autre part, permet de filtrer les documents pour lesquels un nombre important de mots de la requête sont assez proches les uns des autres.

Dans la chaîne QALC, les traitements ultérieurs sont effectués sur ces passages. Afin de ne pas perdre une réponse qui serait séparée en deux lors du découpage, les passages successifs se recouvrent partiellement. De plus, la collection elle-même possède des passages redondants ou presque identiques. Ces deux raisons font que l'on peut retrouver dans la collection 5 occurrences du même passage, éventuellement avec des variations minimes.

Requêtes booléennes et vectorielles

Requêtes booléennes

Sur la collection indexée, il est possible d'effectuer des requêtes booléennes ou vectorielles. Les requêtes booléennes consistent en une formule logique pouvant contenir les opérateurs ET, OU et NON, dans lesquels les opérandes sont les mots à rechercher dans les textes. Par exemple, à partir de la question « What is the name of the volcano that destroyed the ancient city of Pompeii?» on pourra constituer la requête :

```
(a) destroy AND city AND Pompeii
```

Si l'on veut augmenter le rappel de la requête, on peut introduire des reformulations, par exemple les synonymes :

```
(b) (destroy OR destruct) AND (city OR metropolis) AND Pompeii
```

Cette expansion de la requête nécessite une désambiguïsation des mots de la question, faute de quoi le nombre de reformulations erronées peut entraîner beaucoup de bruit dans les documents rapportés, c'est-à-dire l'apport de nombreux documents non-pertinents. Par exemple, les synonymes proposés pour le mot président dans WordNet sont : President of the United States, United States President, Chief Executive, chairman, chairwoman, chair, chairperson, prexy (dirigeant d'une université).

Les requêtes booléennes sont contraignantes car elles ne rapportent que les documents contenant tous les termes de la requête ou, dans le cas de la requête (b), un ensemble de variantes pour chaque terme, représenté par une opération OR. Il est possible de relâcher la requête en rajoutant des reformulations pour chaque terme de la question ou en supprimant des termes [MOL99].

Dans le système Diogene [MAG01], les mots-clés sont étendus en ajoutant d'une part, des variations morphologiques, ainsi le verbe *to invent* est rendu en italien par « inventato OR inventata OR inventò OR inventore OR invenzione » (inventé OR inventée OR inventa OR inventeur OR invention), d'autre part des synonymes. Pour choisir le bon ensemble de synonymes à ajouter, ce système effectue une désambiguïsation des mots de la question.

Requêtes vectorielles

Les requêtes de second type sont nommées vectorielles. Elles sont nommées ainsi car elles sont

fondées sur une mesure de similarité entre deux ensembles de mots : ceux de la requête et ceux d'un document. Ces deux ensembles sont représentés par des vecteurs de mots de grande dimension dont chaque dimension représente un des mots du corpus.

On attribue à chaque mot du vecteur de la requête et du document un poids entre 0 et 1 représentant sa significativité. Ce poids peut dépendre du nombre d'occurrences du mot dans l'ensemble de mots concerné (requête ou document) et de la fréquence dans l'ensemble des documents de la collection. Il existe plusieurs fonctions de pondération. Une mesure communément utilisée est le tf-idf :

Soit *t* un terme, *d* un document contenant le terme et *D* l'ensemble des documents :

$$w(t,d) = tf(t,d).\log\left(\frac{(D)}{(\{d \in D / t \in d\})}\right)$$

Les deux vecteurs ainsi construits sont comparés par une fonction associant un réel à une paire de vecteurs de l'espace de représentation. La mesure généralement utilisée pour cela est la mesure de cosinus, définie par :

$$\mathrm{sim}(x,y) = \frac{\sum_i x_i . y_i}{\sqrt{\sum_i x_i^2 + y_i^2}}$$

Ce type de requête est peu contraignant, puisque aussi éloignés que soient les requêtes des documents, une note sera attribuée. Il suffit de la présence d'un mot en commun entre les deux ensembles de mots comparés pour que le score soit non nul.

On notera enfin l'existence de méthodes hybrides comme le modèle booléen étendu (Extended Boolean Model) de Salton et Fox[SAL05].

Expansion et modification itérative de la requête

On a vu que les systèmes pouvaient avoir besoin de rendre une requête plus stricte ou plus permissive. Pour estimer dans quel sens la requête doit être modifiée, certains systèmes se fondent sur le nombre de documents rapportés par cette requête. Certains systèmes effectuent une expansion a priori, avant d'avoir interrogé la collection. C'est le cas de [HOVY01] qui introduit des variations pour certains termes considérés comme fortement pertinents (Russian → Soviet, The capital of the United States → Washington). On notera que les systèmes évitent ou restreignent généralement les reformulations a priori en raison du risque de ramener de nombreux documents non pertinents.

Dans le système QALC, une première requête booléenne est effectuée, fondée sur les termes invariants (noms propres et nombres) auxquels on ajoute les mots jugés les plus significatifs. Si le nombre de documents rapportés se situe en dessous d'un certain seuil, une seconde requête, vectorielle, est effectuée. Ce choix exploite le fait qu'une requête vectorielle est moins stricte qu'une requête booléenne. Si au contraire la première requête est trop peu filtrante, on la précise en y ajoutant des termes.

Dans le système FALCON[HAR00], trois boucles permettent d'interrompre la suite des traitements à différentes étapes pour revenir à l'étape de recherche de documents, si le nombre de documents ou de passages obtenus à l'étape concernée est trop faible ou trop élevé. Dans les cas où la requête sans variations ne permet pas d'apporter assez de documents à certaines étapes de la chaîne, le système effectue des reformulations sémantiques au niveau de la requête, par exemple (far → distance, like → prefer).

En conclusion, il est possible d'inclure la reconnaissance des variations sémantiques au niveau de la sélection des documents, mais celle-ci pose des problèmes techniques du fait de l'augmentation du bruit qu'elle engendre. Pour cette raison, de nombreux systèmes limitent la gestion des reformulations aux étapes ultérieures du traitement.

1.2.3.2 Filtrage des documents

Après avoir récupéré des documents ou de longs passages grâce à un moteur de recherche, les systèmes effectuent un filtrage qui consiste à réduire la taille des données à traiter. Celui-ci consiste à découper les longs passages en petites unités, généralement en phrases, et a sélectionner les meilleurs de ces passages courts. Cette sélection nécessite le classement des passages courts candidats par ordre de pertinence supposée.

La réduction de la taille des données est effectuée premièrement pour des raisons de temps de traitement. En effet les traitements effectués en dernier peuvent s'avérer lourds (analyses syntaxiques ou sémantiques, preuves logiques) et il est nécessaire de ne pas surcharger le système avec de trop nombreux candidats. Deuxièmement, les systèmes sont prévus pour travailler sur des extraits de texte courts, soit parce qu'ils utilisent la phrase comme unité d'appariement, comme par exemple [KOU05], [HART05], soit parce que cette taille réduite permet de s'assurer que les différents critères de la question se retrouvent à proximité les uns des autres dans le document candidat. Laurent Gillard décrit l'utilisation de scores de densités pour le système SquaLIA[GIL05].

Pour certains systèmes, ces filtrages se font principalement par la construction d'une représentation structurée de la question et du passage (généralement une phrase) candidat, ce filtrage par structuration pouvant être précédé d'une étape de réduction de la quantité de données à traiter. On peut citer par exemple le système d'appariement syntaxique [KOU05]. Les systèmes procédant par structuration peuvent être vus comme des systèmes de validation de la réponse. Ils seront décrits ultérieurement, dans la partie 1.2.5.

Pour d'autres systèmes, la sélection des documents est la composante principale permettant de décider de la présence de la réponse. Ces systèmes reposent en effet sur un ordonnancement des passages candidats par ordre de pertinence. Nous décrirons ci-dessous les différents aspects de l'étape de filtrage pour ce second type de systèmes.

Analyse de passages

Les systèmes prétraitent généralement le texte en effectuant un étiquetage morphosyntaxique et en reconnaissant les entités nommées qu'il contient, plus rarement la résolution des coréférences, c'est-à-dire le repérage des formes référant à une même entité du monde [HART05]. Le système QRISTAL effectue la résolution des anaphores pronominales [LAU05].

Le classement des passages nécessite l'appariement des informations de la question avec celles du passage réponse. Les informations recherchées dans le passage sont donc celles décrites dans la section 1.2.2, traitant de l'analyse de la question, et principalement le type attendu et les termes de la question.

Dans le système QALC, la recherche des termes de la question est effectuée en utilisant le programme de reconnaissance de termes Fastr[JAC99]. Ce système permet de reconnaître les variantes d'un mot, notamment les mots partageant la même forme lemmatisée, les mots reliés par un lien de synonymie et les mots dérivés morphologiquement l'un de l'autre. Le grand apport de Fastr est permettre la reconnaissance de termes composés et de leurs variantes grâce à des règles syntaxico-sémantiques, comme dans l'exemple suivant :

```
Q : ... business slogan ...
R : ... catchwords in the modern business ...
```

qui a recours à une relation de synonymie entre « slogan » et « catchwords », et à une règle de variation syntaxique :

```
xPermSemHead : N2 N3 -> N3 PREP ( ART? (A|N|Np|V)0,3 ( (N|Np) C Art? )? N2
```

Combinaison des critères

Dans le système QALC, les critères vus précédemment permettent de déterminer un poids pour les documents : ce poids est obtenu par la somme de poids partiels prenant en compte des aspects syntaxiques ou sémantiques. Participent à cette somme des informations sémantiques : la présence de chaque terme de la question, la présence d'une entité nommée du type attendu comme réponse, la présence du focus qui obtient ainsi un poids supplémentaire par rapport à un simple terme.

Participe également à cette somme une mesure de densité des critères de la question, c'est-à-dire, une mesure rendant compte de la proximité de ces termes dans la phrase réponse. Cette densité est calculée à partir de la distance dans la phrase entre les critères de la question. Deux mesures de densité ont été testées pour le système : une mesure linéaire et une mesure syntaxique, définie dans [BAR05a]. La mesure de densité syntaxique présente des résultats meilleurs, faisant passer le MRR du système de 0,268 à 0,293 [LIG06].

1.2.4 Validation de la réponse et implication textuelle

Plusieurs techniques peuvent être appliquées pour extraire et valider une réponse. Valider un passage candidat à être la réponse consiste à décider automatiquement si ce passage justifie la réponse, avec un score de précision qu'on souhaite le plus proche possible de ce que ferait un être humain méticuleux. Les premiers systèmes de QR se comportaient plus comme des moteurs de recherche améliorés, c'est-à-dire qu'ils se contentaient de ramener des courts passages susceptibles de contenir la réponse. Ainsi, les premières campagnes d'évaluation des systèmes de questions-réponses posaient comme tâche de ramener des passages de 250 caractères contenant la réponse, celle-ci devant être justifiée par le document.

Extraction de la réponse

Les campagnes récentes imposent l'extraction de la réponse exacte (TREC depuis 2003, CLEF depuis 2004, la tâche était déjà proposée en 2003, mais en parallèle avec une sous-tâche où l'on pouvait rapporter de courts passages de 50 caractères maximum).

À partir des phrases sélectionnées, les systèmes extraient la réponse exacte, c'est-à-dire le fragment de taille minimal contenant la réponse, comme nous l'avons vu en section 1.2. Plusieurs stratégies sont utilisées pour extraire la réponse. Dans le cas où le type attendu de la réponse est une entité nommée, on peut sélectionner dans la phrase le ou les candidats à être la réponse.

Dans le cas où le type attendu est un type général et que l'on n'est pas capable d'identifier les éléments de façon sûre, le système QALC a recours à des patrons syntaxiques pour l'extraction de la réponse. Par exemple :

```
Q : What do Knight Ridder publish?
Pattern : NPfocus Connecting-elements NPanswer passage
Phrase : Knight Ridder publishes 30 daily newspapers,
```

```
including ...
R : 30 daily newspapers
avec Connecting-elements correspondant au verbe de la question.
```

De plus, dans le cas de systèmes d'extraction structurée, le choix de la réponse peut être pris en charge par le module de validation de la réponse. Par exemple, dans le système InSicht[HART04], la réponse est extraite par un appariement de deux graphes sémantiques.

Validation de la réponse

En plus de la tâche d'extraction de la réponse, CLEF propose depuis 2006, pour plusieurs langues, un exercice de validation de réponse (AVE) dans lequel le système doit juger si la réponse est justifiée par le passage rapporté. Le but de cette campagne est d'amener les systèmes de QR à améliorer leur capacité à s'auto-évaluer.

Dans la campagne AVE, le jeu de données est l'ensemble des réponses fournies par les systèmes concurrents dans la tâche de questions-réponses de CLEF. On notera le biais entraîné par ce choix, puisque les systèmes participant aux deux tâches sont généralement les mêmes. Les systèmes d'AVE travaillent donc sur des données qu'ils ont aidées à obtenir, et pour lesquelles ils sont vraisemblablement adaptés. Ce biais est d'autant plus fort pour des langues où peu de systèmes participent, puisque le jeu de données de départ risque de se rapprocher très fortement du meilleur système en lice.

La campagne RTE (Recognising Textual Entailment), lancée en 2005, possède un but proche d'AVE. La tâche qu'elle propose consiste à vérifier qu'un texte T implique une hypothèse H. L'hypothèse est une phrase affirmative généralement plus courte que le texte. Leurs tailles moyennes respectives sont de 25 et 10 mots. Les couples Texte-Hypothèse ne proviennent pas uniquement de la problématique QR mais d'un éventail varié de tâches, telles que la comparaison de documents similaires ou la traduction automatique. Voici un exemple de couple T-H:

```
Tâche : Questions-Réponses
T : There are currently estimated to be somewhere between 60-100
million landmines in the ground worldwide, this remains a rough
estimate since few accurate records were kept when mines were
deployed.
H : There are about 100 million landmines in the world.
```

Cette évaluation, bien que ne faisant pas intervenir des questions, est fortement liée à la thématique de questions-réponses. En effet, les courts passages de l'hypothèse peuvent être vus comme des questions transformées à la forme affirmative en y remplaçant le syntagme interrogatif par la réponse. De ce fait, les méthodes utilisées dans les deux types de tâches diffèrent peu.

On notera que cette tâche propose des exemples de couples (T,H) tirés de problématiques différentes, par exemple, la recherche d'information, l'extraction d'information, les questions-réponses ou le résumé multi-document [BARH06]. Le mélange des tâches d'origine est intéressant du fait que certaines tâches apportent des jeux de données plus difficiles que d'autres. Les systèmes qui ont été testés séparément sur les différentes catégories montrent la disparité des tâches en terme de précision des résultats.

La présentation des techniques ci-dessous se fonde principalement sur les campagnes d'évaluation RTE, aussi, nous avons choisi d'utiliser le plus souvent possible la terminologie de RTE pour désigner les passages, c'est-à-dire « texte » et « hypothèse » plutôt que (respectivement) « passage candidat » et « question ».

Évaluation des systèmes

Dans ces campagnes sont proposées à la fois des exemples de couples (Q,R) ou (T,H) corrects et incorrects, qu'il faut valider ou invalider, c'est-à-dire pour lesquels il faut décider s'ils sont corrects ou non.

La campagne RTE2 propose deux façons d'évaluer les résultats. La première et principale correspond à une tâche de classification qui consiste à décider si les paires (T,H) présentent bien une implication. La seconde correspond à une tâche d'ordonnancement, où le système doit trier les paires dans l'ordre estimé de fiabilité de la présence d'une implication.

Pour estimer la fiabilité selon cette tâche, les campagnes RTE et AVE utilisent une mesure nommée « accuracy », que nous traduirons par « exactitude », qui est le pourcentage de paires correctement classées. Dans RTE, les exemples positifs et négatifs étant en nombre égal, la baseline d'un tirage aléatoire est de 50%. De plus, [ZAN06] obtient des scores de presque 60% avec plusieurs systèmes utilisant uniquement des traits de similarité lexicale mot à mot (mots identiques, lemmes et catégorie grammaticale identiques, similarité dans WordNet), ce qui définit une seconde baseline.

La campagne AVE utilise également la F-mesure pour évaluer la capacité de classification de chaque système. La F-mesure se calcule par la formule vue en section Error: Reference source not found1.1.3, avec le coefficient alpha=1 qui donne un rôle symétrique à la précision et au rappel. Dans le cadre de cette évaluation où les systèmes doivent décider si une réponse fournie est correcte, ces mesures s'expriment ainsi :

$$Précision = \frac{\text{Nb réponses estimées comme correctes à juste titre}}{\text{Nb réponses estimées comme correctes}}$$

$$Rappel = \frac{\text{Nb réponses estimées comme correctes à juste titre}}{\text{Nb réponses correctes}}$$

Pour la tâche d'ordonnancement, la mesure utilisée dans RTE1 était le CWS vu précédemment. Dans RTE2, cette mesure est remplacée par la précision moyenne, qui est la moyenne des précisions du système en tout point où le rappel augmente :

$$Précisionmoyenne = \frac{1}{R} \cdot \sum_{p \in C} \frac{\text{Nb correctes jusqu'à la paire}(p)}{\text{rang}(p)}$$

avec C l'ensemble des paires correctement évaluées par le système, toutes questions confondues et R le nombre total de paires correctement évaluées.

Approches

Il existe de nombreux systèmes de validation de questions mis en évidence par les campagnes d'évaluation citées ci-dessus. Ceux-ci peuvent se classifier en trois catégories dépendant notamment du type de dispositif assemblant les informations de similarité sémantiques élémentaires pour effectuer le jugement de validité.

Certains systèmes utilisent un système d'apprentissage automatique, comme par exemple des arbres de décision, qui prend en entrée des critères nombreux et hétérogènes de similarité entre la question et la réponse. D'autres utilisent une mesure de proximité, comme une distance d'édition, entre les arbres syntaxiques de la question et de la réponse. Enfin une troisième catégorie fait usage d'un système de preuves logiques qui essaye d'inférer la question ou l'hypothèse à partir de la réponse.

1.2.4.1 Validation par un système de décision

Le premier type de systèmes est fondé sur un système d'apprentissage automatique qui décidera si la réponse est valide en fonction d'un ensemble de mesures, ou traits, fournies en entrée. Les systèmes de décision les plus souvent utilisés sont les arbres de décision [ADA06].

Les traits sont obtenus par des modules de comparaison entre question et réponse. L'avantage de ce type de systèmes est de pouvoir aisément ajouter une mesure. De plus, les mesures peuvent être de nature hétérogène.

Types de traits utilisés dans les systèmes de décision

Nous allons détailler les différentes mesures proposées dans les systèmes. Nous distinguons les traits locaux, qui font intervenir une partie très restreinte des termes de la question - le plus souvent, il s'agit d'un appariement d'un terme de l'hypothèse avec un terme du texte – et les traits globaux, qui utilisent la totalité des termes des deux fragments textuels ou du moins une partie la plus grande possible.

Mesure de recouvrement de texte

Cette première catégorie regroupe des mesures globales qui donnent une évaluation de la similarité entre les deux textes, en faisant intervenir des mesures de similarité plus ou moins complexes entre les mots (égalité, relations lexicales) et une notion syntagmatique minimaliste résidant dans la succession des mots dans l'énonciation de la phrase.

Un des traits les plus utilisés est la plus longue chaîne commune (longest common chain) ou plus longue sous-chaîne commune (longest common subchain). Ce type de mesure, avec des variations sur la délimitation de la sous-chaîne et la façon de calculer son poids, est utilisé dans le système FRASQUES du LIMSI[GRA08] ou par Newman dans [NEW05].

Cette chaîne peut être une succession continue de mots comme dans [HIC06]. Dans ce cas on la nomme plutôt « plus longue chaîne commune » ou alors elle peut admettre des discontinuités, comme dans le module de validation des réponses du système FRASQUES.

[NEW05] utilise le score ROUGE fondé sur des recouvrements de n-grammes. Cette mesure est une comparaison des 1- à 4-grammes contenus dans le texte et l'hypothèse. L'ensemble des n-grammes possibles forme un espace vectoriel sur lequel est appliqué une fonction cosinus.

On note également la présence éventuelle de traits négatifs. Certains traits rendent compte de la présence de parties de l'hypothèse ne trouvant pas de correspondants dans le texte, comme dans [ADA06], qui utilise pour ce trait le nombre de mots à supprimer de l'hypothèse, ou [HIC06], qui comptabilise le nombre de chunks à supprimer.

Traits sémantiques

Les mesures présentées dans cette section définissent des traits locaux. Ces mesures attribuent un poids au lien sémantique résidant entre un mot de l'hypothèse et un mot du texte. De ce fait, il est nécessaire d'utiliser une fonction unificatrice pour les intégrer dans le système de décision. Notons qu'un moyen d'effectuer cette intégration est d'utiliser ces informations dans les mesures de recouvrement textuel vues ci-dessus.

Ces mesures seront plus amplement décrites dans le chapitre suivant sur les ressources sémantiques. Cependant, on peut les découper en deux types d'approches : approches fondées sur des ressources lexicales et approches sur corpus.

Parmi les approches utilisant des ressources lexicales, on trouve l'utilisation classique de synonymes ou de variations morphologiques. Certains systèmes [ADA06] utilisent une mesure de

similarité lexicale dans WordNet semblable à la mesure de chaînage lexical décrite dans [HIR97]. Cette mesure autorise la recherche de variations plus éloignées que les synonymes, par le biais de parcours dans le réseau sémantique de WordNet.

Quant aux méthodes sur corpus, elles comportent par exemple la mesure créée de Glickman et Dagan [GLI05], qui mesure la probabilité d'implication de deux mots par des mesures de fréquences de mots sur internet. Pour être exact, le score d'implication calculé pour un couple de mots (wH, wT) est le rapport du nombre de cooccurrences sur le net divisé par le nombre d'occurrences de wT soit une approximation de la p(wH | wT).

Un autre exemple de cette catégorie est l'Indexation Sémantique Latente, (Latent Semantic Indexing), utilisé notamment dans [NEW05]. Cette méthode, proposée par Derwester et al. [DER90], consiste à projeter les documents et les mots dans des espaces réduits[7] de traits sémantiques obtenus par analyse factorielle des correspondances sur la matrice des occurrences des mots dans les documents. Les dimensions de cette analyse factorielle représentent des noyaux de termes fortement cooccurrents, ou encore, du point de vue opposé, des groupes de documents possédant une affinité lexicale. Ces dimensions semblent pouvoir se rattacher à des traits sémantiques plus ou moins évidents. Comme dans toute analyse factorielle, le nombre de dimensions conservées pour la représentation sémantique centrale (l'espace réduit) peut être choisi par l'expérimentateur. Pour donner une idée, [DER90] préconise une dimension de l'ordre de 50 à 150 traits sémantiques. La comparaison terme à terme s'effectue par le produit scalaire des projections des deux termes dans l'espace de dimension réduite.

À côté de ces mesures sur des couples de mots se trouvent les mesures indiquant la présence d'éléments négatifs. Ces mesures sont introduites pour tenter de détecter des exemples où l'implication est impossible. Parmi ces critères on peut citer la présence d'une négation, la présence d'un antonyme. [NEW05] par exemple déclenche un trait de présence d'antonyme si celui ci est présent à l'intérieur de la plus longue sous-chaîne commune.

Traits syntagmatiques locaux

La composante syntagmatique des documents peut être traitée par une méthode d'appariement globale de graphes syntaxiques ou sémantiques représentant la question et la phrase ou le passage candidat. Ce type de méthodes structurées sera décrit en section 1.2.4.2. Dans le cadre d'un système de décision intégrant des traits hétérogènes, l'aspect syntagmatique de l'appariement peut être représenté par une série de traits locaux d'appariement des relations sémantiques entre prédicat et arguments de l'hypothèse et du texte, comme dans [HIC06].

Ces mesures reposent sur l'appariement des deux prédicats et de leurs arguments. Elles se déclinent en l'évaluation de la proximité des prédicats et l'évaluation plus ou moins stricte de la proximité des arguments.

Tâche

Comme nous l'avons vu, les jeux de données de RTE se divisent en plusieurs sous-jeux correspondant à une tâche de TAL particulière (QR, Résumé automatique,...). Dans RTE, le type de la tâche est indiqué pour chaque couple (texte, hypothèse). Fournir ce paramètre à un système, comme dans [NEW05] ou [ADA06], lui permet de prendre en compte les particularités de chaque tâche. Les mesures faites par [ADA06] montrent que les paires issues de la tâche QR profitaient grandement de cette distinction ce qui semble montrer que cette tâche se différencie fortement des 3 autres choisies dans RTE2.

7 Les espaces utilisés pour les comparaisons terme-terme, document-document, terme-document sont presque les mêmes. Comme expliqué dans [DER90], ils diffèrent par le produit d'une matrice carrée diagonale qui « étire » différemment chaque dimension de l'espace.

Autres

Cette catégorisation des traits utilisés se veut fidèle des systèmes utilisés dans les campagnes RTE. Cependant la liste est facilement extensible du fait de la grande souplesse du système de décision. Par exemple, Hickl utilise également un système de détection des paraphrases.

Introduction de données multiples dans un trait unique

Les techniques par système de décision posent le problème de l'intégration de données unitaires dans un système global. En effet, du côté de l'analyse des phrases, il est fréquent d'obtenir des mesures concernant un seul critère de l'hypothèse, comme la présence d'une similarité sémantique reconnue entre un mot de l'hypothèse et un mot du texte, alors que de l'autre côté, le système de décision attend une mesure globale. En effet, le nombre de traits en entrée de ce système ne peut pas varier en fonction du nombre de termes de l'hypothèse ou du texte.

Dans le système FRASQUES, les termes remarquables de la question sont représentés par des entrées séparées du système de décision. Il s'agit de l'information de présence du focus, du verbe principal et du type général.

Pour les autres informations, comme la présence des noms communs, noms propres et adjectifs, ou la présence de termes toutes catégories confondues il peut y avoir plusieurs occurrences de la catégorie dans la question.

Certaines techniques de comparaison lexicale, comme une mesure de cosinus ou les distances d'édition, permettent de passer d'un ensemble de mesures de similarité mot à mot, à un scalaire. Cependant les auteurs ne souhaitant pas utiliser ces techniques particulières sont amenés à rechercher des façons diverses de combiner ces mesures unitaires.

Par exemple le système FRASQUES utilise le rapport du nombre de critères trouvés dans le passage réponse sur le nombre de critères attendus dans la question. Dans RTE2, Adams [ADA06] effectue le produit des scores obtenus séparément pour chaque terme de l'hypothèse. Il remarque que le produit est une opération très contraignante puisqu'il suffit qu'un des termes ne trouve pas de correspondant dans le texte, ayant ainsi un score nul, pour que le produit devienne nul à son tour. Pour compenser ce risque, Adams utilise une mesure de similarité assez permissive fondée sur les mesures de fréquences sur la toile (cf ci-dessus).

Utilisation à des fins de test

Du fait de sa grande modularité, ce type de système permet de tester facilement quels traits sont les plus pertinents. Une méthode souvent employée est de faire tourner le système d'apprentissage en ne lui fournissant en entrée qu'un sous-ensemble des traits utilisables.

Plusieurs études utilisent cette propriété [HIC06][ADA06]. Par exemple, Hickl obtient par cette méthode une évaluation de l'utilité des grandes catégories de traits que son système utilise.

Traits	Sémantiques	Paraphrase	Dépendances	Alignement	Tous
Score	58,00	65,88	62,50	65,25	75,38

Tab. 1: Hickl@RTE2 : utilité des différents types de traits.

Une étude similaire est menée par Grappy et al. dans [GRA08]. Elle est utilisée pour tester la fiabilité des différents critères de la question dans le cadre de la campagne d'évaluation AVE. Les critères sont évalués séparément en utilisant un arbre de décision pour fixer un seuil au dessus

duquel le critère validera la question. Les critères qui apportent au moins 50% de précision sont : la proportion de termes retrouvés dans les passages toutes catégories confondues (50%) la proportion d'adjectifs retrouvés (52%)la présence du focus (50%), la LCC (53%). La présence des catégories grammaticales comme les noms communs, les noms propres, les nombres et les verbes donnent des précisions moindres, de l'ordre de 30%, mais apportent un très bon rappel (86% pour les noms communs, 88 pour les noms propres), ils sont donc nécessaires pour améliorer la couverture du système. Mis ensemble en entrée du système de décision, ces critères donnent une précision de 66%. Ce qui montre la nécessité de cumuler plusieurs critères.

Utilisation comme système de vote

La grande adaptabilité du système permet d'utiliser en entrée le résultat de systèmes complets. Le système de décision permet d'intégrer plusieurs variantes de stratégies complexes. Cette technique permet de dépasser le score des stratégies isolées. [TATU06] utilise cette méthode pour assembler deux variantes d'un système de preuve logique COGEX et une méthode d'alignement lexical. Kouleykov et Magnini [KOU06] utilisent cette technique pour faire voter 6 systèmes d'appariement syntaxique.

Conclusion

Nous avons vu que les méthodes par systèmes d'apprentissage étaient d'utilisation très pratique de par leur aspect fortement modulaire. De plus, elles peuvent atteindre d'excellents scores à condition d'intégrer des traits rendant compte de la similarité de structure entre le texte et l'hypothèse.

Cependant, cette méthode possède un aspect insatisfaisant du fait qu'elle ne repose pas sur une représentation structurée du phénomène d'implication. Ainsi, elle ne permet pas d'expliquer pourquoi une réponse est justifiée et se contente d'estimer la probabilité de la présence d'une implication entre texte et hypothèse.

1.2.4.2 Validation par une méthode structurée

La deuxième catégorie de méthodes se fonde sur une représentation structurée au centre de la décision. Ces méthodes se répartissent entre :

- appariement syntaxique [KOU05], [ZAN06]
- preuve logique [TATU06]
- appariement de représentations sémantiques.

Ce dernier type de méthodes est moins représenté que les deux autres. On peut toutefois noter la participation du système MAVE pour l'allemand à AVE 2007 [GLO07]. Le peu d'engouement pour ces méthodes peut provenir de la difficulté de l'analyse sémantique, qui n'est pas encore une technique rodée, contrairement à l'analyse syntaxique.

On peut affiner le classement de ces systèmes selon deux axes: d'une part, la technique de validation utilisée, qui peut être soit un appariement de graphes, soit une preuve logique, d'autre part, le niveau de représentation des phrases, syntaxique ou sémantique. Les systèmes par preuve logique utilisent une représentation plutôt sémantique et les méthodes d'appariement de graphes utilisent généralement la syntaxe, mais le système de MAVE utilise une représentation sémantique.

Appariement de graphes

Une première méthode d'appariement syntaxique est celle proposée par le système de l'ITC [KOU06]. Celui-ci effectue un appariement syntaxique grâce à un algorithme de distance d'édition sur les arbres décrit dans [ZHA90] et qui adapte aux arbres la distance d'édition de Wagner et Fischer sur les listes.

La distance d'édition de Zhang et Shasha, illustrée dans la figure 2, revient à calculer la séquence d'opérations la moins coûteuse pour transformer un arbre T1 en un arbre T2. Les opérations que l'on peut effectuer sont, comme dans la distance de Wagner et Fischer, l'insertion, la suppression et la substitution d'un nœud.

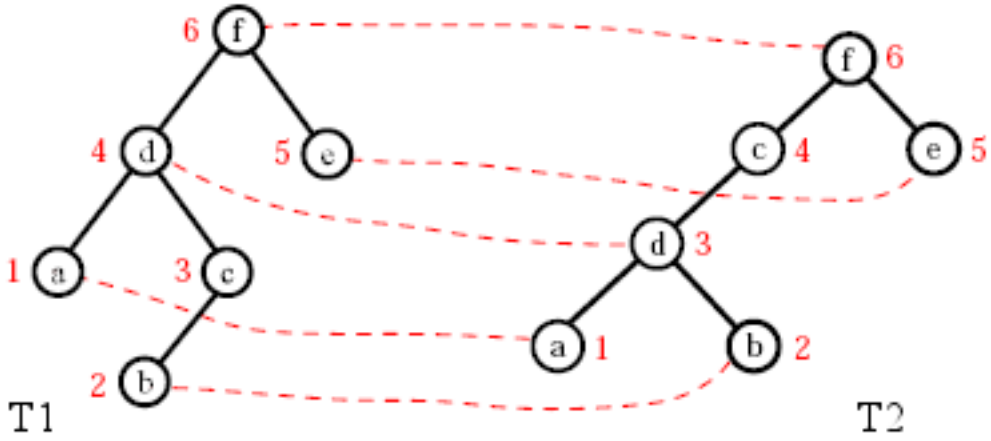


Fig. 2: Distance d'édition entre deux arbres

Par exemple, pour transformer l'arbre T1 en T2, on peut effectuer les opérations suivantes : *supprimer c au nœud 3, insérer c en dessous de f*. Le coût de cette suite d'opérations est la somme du coût de ces deux opérations. Le coût d'édition de T1 à T2 est celui de la suite d'opérations de coût minimum transformant T1 en T2.

L'algorithme de Zhang et Shasha peut être réglé en modifiant les paramètres de coût pour les trois opérations. Le système de l'ITC utilise une fonction de coût dépendante du type d'opération effectué ainsi que du ou des mots entrant en jeu. Ainsi, le coût de l'insertion d'un mot m1 peut être égal à son idf :

$$ins(m1) = idf(m1) = \log\left(\frac{N}{Nm1}\right)$$

où Nm1 le nombre de documents contenant le mot m1 dans un corpus de référence contenant N documents. Le coût d'insertion peut prendre des valeurs liées à sa position dans l'arbre syntaxique.

$$ins(m1) = \#\text{children}(m1)$$

ou $ins(m1) = 10 - \#\text{parents}(m1)$

Le principe de ces deux dernières formules est de décourager l'insertion d'éléments dans les nœuds centraux de l'arbre syntaxique. La formule utilisant le nombre de parents donne un meilleur résultat que les autres.

La substitution de m1 par m2 possédera un poids dépendant de la probabilité d'implication de

m2 par m1. Le poids utilisé par le système est :

$subst(m1,m2) = ins(m2) * (1 - Ent(w1,w2))$

Si m1 est lié à m2 dans WordNet par une relation parmi hyperonymie, synonymie, implication, pertainimy:

alors $Ent = \frac{1}{S_{m_1}} \cdot \frac{1}{S_{m_2}}$

avec S_{m_1} et S_{m_2} le nombre de sens que possède chacun des mots m_1 et m_2 dans WordNet.

sinon $Ent = 0$

finsi

Enfin, l'opération de suppression se voit assigner un coût nul, afin de rendre compte du fait que le texte, choisi comme arbre de départ, est plus long que l'hypothèse (arbre d'arrivée).

Limitations de la méthode

La méthode de par son algorithme central relativement figé, pose certains problèmes d'adaptabilité.

La distance ne permet pas les permutations de blocs, ce qui entraîne par exemple qu'un complément circonstanciel devra conserver sa place dans la phrase. De même, la méthode semble peu adaptée au cas de transformations d'un mot en plusieurs autres comme dans « X tue Y » → « Y est_mort »

Les coûts de suppression et d'insertion ne sont pas sensibles au contexte. Si l'on supprime un nœud à un endroit, puis on l'insère plus loin dans l'arbre, les deux opérations seront considérées comme indépendantes. On peut toutefois se demander si cette capacité est souhaitable car elle peut s'avérer utile dans le cas du déplacement d'un complément circonstanciel ou pour transformer une phrase de la voix active à la voix passive, et nuisible si elle autorise un échange de place entre le sujet et un complément.

Autoriser les transformations

Une solution à ce problème de variabilité de l'ordre des reformulations consiste à produire plusieurs représentations équivalentes de l'un des deux arbres. Cette méthode est utilisée dans le système [SAL05], qui utilise une représentation hiérarchique réunissant plusieurs niveaux de relations : rôles sémantiques, relations syntaxiques, et succession linéaire des mots. Cette méthode pourrait très probablement être utilisée pour des structures syntaxiques. Ce système obtient de meilleurs résultats que COGEX à RTE1, mais n'a pas concouru à RTE2.

Dans cette méthode, après avoir généré plusieurs variantes de la première phrase grâce à des patrons de paraphrase, le système recherche la présence d'une « subsomption », c'est à dire une implication logique, entre la seconde phrase et chacune de ces variantes.

Apprentissage de patrons de reformulation syntaxique

Une autre solution, décrite dans [ZAN06], appréhende la variabilité de la syntaxe, en faisant apprendre au système les variations de structure syntaxique correspondant à un cas d'implication ou, au contraire, à une absence d'implication.

Dans cette méthode, ce ne sont plus des arbres qui sont appariés, mais des couples d'arbres (H,T). Le but de cet algorithme est de rechercher dans la liste des exemples de couples (H',T') déjà connus les couples présentant une structure et une transformation similaire au couple courant.

Cette similarité entre paires de couples d'arbres utilise également une mesure de similarité consistant à effectuer une mesure de proximité par cosinus dans un espace de très grande dimension

ou chaque dimension correspond à un sous-arbre d'une base de référence. L'énumération des sous-arbres est évitée par l'utilisation d'une technique à base de fonctions noyaux. Pour plus de détails, on se référera à l'article de Zanzotto.

Vérification par preuve logique

Les systèmes de vérification par preuve logique sont pour la plupart développés par la société *Language and Computer Corporation*. Le principe de ce type de système est de convertir les deux passages (Texte et Hypothèse) ou (Texte et Question) en une formule logique. Puis d'effectuer une preuve d'implication entre ces deux formules.

Le système COGEX [MOL03] fait partie du système proposé par Tatu à la campagne RTE2 [TATU06]. En raison de son manque de robustesse, le programme est utilisé en collaboration avec un système plus simple d'alignement textuel. Le système présenté dans [SAL05], utilisé dans RTE1, utilise lui aussi une méthode fondée sur une preuve logique.

Nature de la représentation

La représentation utilisée par ces systèmes est de nature plutôt sémantique. Le système COGEX utilise une représentation proche de la syntaxe, notamment par le fait que les groupes prépositionnels ne sont pas transformés en un prédicat sémantique correspondant. Par exemple, « heavy selling of [stocks] in Chicago », exemple tiré de [MOL03] et simplifié, serait transcrit :

```
heavy_JJ(x1) & selling_NN(x1) & of_IN(x1,x6) & stocks_NN(x6) &
in_IN(x1,x8) & Chicago_NN(x8)
```

Dans une représentation sémantique plus éloignée de la syntaxe, le nom « selling » serait converti en prédicat verbal sell_VB(x1,?,x6) où '?' est un joker pour le sujet non spécifié.

L'avantage d'une représentation proche de la syntaxe est de simplifier les traitements pour la construire, et de limiter ainsi les erreurs d'analyse qui pourraient rendre l'analyse inopérante. Notons pour illustrer ce propos que le système COGEX possède une fonctionnalité de désambiguïsation des mots mais que cette analyse provoquait trop d'erreurs dans la campagne RTE1, d'après les auteurs.

Une version plus récente du système utilise une version avec une représentation plus sémantique. Premièrement, l'analyse des entités nommées est introduite dans la formule logique, ainsi qu'une représentation plus fine des indications temporelles sous forme de sextuplets (année, mois, jour, heure, minute, seconde). La négation est gérée en rendant les adverbes « never » et « not » par la négation logique.

Description de la preuve logique

Dans le système COGEX, la preuve que le texte T implique l'hypothèse H se fait en prouvant que la formule logique « T & nonH » est insatisfiable, ce qui est la technique généralement utilisée par les algorithmes de preuve.

Pour effectuer cette preuve, le système dispose de connaissances sur le monde et de règles d'inférence sous forme d'axiomes. Un exemple de règle d'inférence, rendant compte d'une relation d'hyponymie tirée de WordNet, est :

```
murder_VB(e1,x1,x2) → kill_VB(e1,x1,x2)
```

où e1 est l'identifiant de l'événement (kill/murder), x1 l'identifiant de son agent et x2 l'identifiant de son objet.

Problème de robustesse

Il faut noter que le système COGEX à RTE1 utilisé seul, n'avait obtenu que 55,1% d'exactitude. Couplé avec une méthode d'alignement lexical, il atteint 73,8% soit le second meilleur score à l'évaluation.

Ces résultats, conjoints à la première position des systèmes de preuve logique dans l'évaluation des questions-réponses TREC, laissent à penser que ce système permet de mettre en évidence de nombreuses implications dans des cas compliqués, mais est peu robuste et peut échouer, y compris sur des cas simples. Ce qui fait que cette technique nécessite d'être secondée par des méthodes moins complexes et plus robustes.

1.3 Conclusion

Au vu des différentes méthodes existantes, nous remarquons qu'il existe un écart important entre, d'une part, des méthodes fortement intégrées et structurées nécessitant d'importantes ressources linguistiques structurées comme l'ontologie WordNet, et d'autre part des techniques fondées sur la réunion de traits hétérogènes, de fiabilités variables.

L'avantage des méthodes structurées est de fournir des informations précises sur la justification et, en général, une représentation de l'appariement entre les éléments de la question et ceux de la réponse. Leur inconvénient principal est la grande quantité de ressources qui leur sont nécessaires.

L'avantage des secondes méthodes est leur capacité à intégrer automatiquement par un système d'apprentissage des éléments de justification hétérogènes, offrant des indices sur la fiabilité du document allant des plus grandes précisions aux plus faibles, ainsi que de la granularité la plus fine, le terme, à la granularité la plus grosse, celle par exemple de la plus longue chaîne commune. L'inconvénient qui en résulte est l'absence d'une représentation précise de la justification, laquelle est composée d'une succession d'indices hétérogènes, telle une mesure de recouvrement textuel, de nombres de termes communs, d'appariement de structures casuelles, etc.

Chapitre 2 : Application des connaissances linguistiques en QR

Nous avons vu dans le chapitre précédent que les systèmes de questions-réponses avaient besoin pour évaluer la fiabilité des documents rapportés, de nombreuses connaissances linguistiques. On peut distinguer deux grandes familles de connaissances. D'une part les informations sur lesquelles portent les questions auxquelles le SQR doit répondre, il peut s'agir d'informations factuelles évoquées dans des journaux ou de connaissances encyclopédiques. D'autre part les connaissances permettant d'effectuer l'appariement, c'est-à-dire de rapprocher les éléments de justification de la question et de la réponse. Notons que la séparation entre ces deux types de connaissances n'est pas exclusive et un système peut avoir besoin de faire appel à des connaissances encyclopédiques. « Miguel Indurain est espagnol » est la réponse à une question :

```
Quelle est la nationalité du coureur cycliste Miguel Indurain ?
```

Mais constitue un simple élément de justification dans une autre question plus complexe :

```
Quel coureur cycliste espagnol est surnommé « Périco » ?
```

Les connaissances permettant de mettre en évidence le lien entre les informations de la question et celles de la réponse sont très variées, couvrant tous les champs de la linguistique. On compte parmi celles-ci les connaissances permettant de reconnaître des reformulations entre un terme de la question et un terme de la réponse, qui peuvent être de nature sémantique (par exemple un lien entre deux mots dans une ontologie) ou de nature morphologique (inventeur – invention ou inventer – inventé). D'autres connaissances apportent des règles de reformulation syntagmatique, formulées au niveau syntaxique ou sémantique ou au niveau logique, sous forme de règles d'inférence.

Un système de questions-réponses a notamment besoin de connaître les reformulations possibles pour les termes de la question afin d'augmenter son rappel, ainsi que d'évaluer la fiabilité de ces reformulations, de façon, par exemple, à améliorer la précision de son classement.

Nous nous intéressons donc tout particulièrement aux ressources disponibles pour le TAL et dresserons dans ce chapitre un état de l'art des ressources existantes. Pour chaque type de connaissance, nous donnerons des exemples d'utilisation, effective ou supposée possible, pour le domaine des questions-réponses et plus particulièrement dans notre travail.

Notre but est de mettre en évidence l'éventail des connaissances existantes et utilisées par les SQR, de montrer leurs limites en termes de disponibilité et de couverture et les limitations que cela entraîne sur les performances des systèmes de questions-réponses.

Différents types de sémantique

Nous entendons par la notion de sémantique tout procédé permettant de rattacher un sens à un élément du discours. Cela contient d'une part les phénomènes permettant de rattacher un signifiant à un signifié, d'autre part l'attribution d'une valeur sémantique à des liens de nature syntaxique entre deux éléments du discours. Nous avons classifié ces phénomènes de la façon suivante :

1 Sémantique locale paradigmatique : ce sont les reformulations des termes de la question telles que l'identité (variation nulle), la synonymie, l'hyperonymie, l'hyponymie, les transformations morphologiques, mais aussi, toute variation plus étendue comme la métonymie ou la métaphore.

2 Sémantique locale syntagmatique : sémantique des relations prédicat-argument dans la phrase.

3 Sémantique distante ou sémantique de l'énonciation : ce sont tous les procédés permettant de relier deux parties distantes d'un énoncé : thématisation, anaphore, coréférence, auxquelles on peut ajouter la recherche d'informations dans des ressources distantes.

Les termes composés, comme centrale nucléaire cumulent une facette paradigmatique et une facette syntagmatique. Du point de vue du système qui les utilise, ces termes jouent le même rôle que des termes simples et doivent faire partie d'un appariement entre terme de la question et terme de la réponse. Leur gestion appartient donc au champ de la sémantique paradigmatique. Par contre, en interne, ces termes combinent une composante syntagmatique et paradigmatique. Par exemple, une reformulation du terme « car maker » (fabricant automobile) est « making many automobiles ». La détection de cette variante nécessite un appariement paradigmatique entre « car » et « automobiles » et entre « maker » et « making ». D'autre part, il possède une composante syntagmatique puisque les relations syntagmatiques entre les mots du terme varient. On passe d'une relation de modifieur (« car » est un modifieur de « maker ») à une relation d'objet (« automobiles » est l'objet du verbe « to make »)

2.1 Sémantique locale paradigmatique

La sémantique paradigmatique concerne les reformulations entre termes de la question et de la réponse.

2.1.1 Ressources non hiérarchiques

Parmi les connaissances utilisées par les systèmes de QR on compte des données présentées sous la forme de listes non structurées. Ces ressources comptent notamment des dictionnaires de synonymes, de morphologie, comme celles utilisées par le programme Fastr, que nous utilisons pour la reconnaissance de termes simples et composés.

Ces systèmes utilisent aussi certaines listes de mots consacrées à des tâches spécifiques, tels que des marqueurs d'entités nommées (Monsieur, Professeur, Unités de mesures) ou les listes de mots à ignorer.

Par exemple dans l'analyse des questions on ne tiendra pas compte des mots tels que *nom, nommez, date*, qui servent à spécifier le type de la réponse, dans des questions comme : « À quelle date a eu lieu... Nommez un plat... Quel est le nom de... »

2.1.2 Données structurées, ontologies.

La plupart des données utilisées sont sous la forme de hiérarchies et de réseaux. WordNet[FEL98], FrameNet[RUP06] combinent une architecture hiérarchique et des relations transverses.

2.1.2.1 Les bases de données lexicales WordNet et EuroWordNet

La ressource ontologique la plus complète pour l'anglais et la plus utilisée en TAL est la base de données lexicale WordNet. Pour d'autres langues dont le français, il existe un équivalent EuroWordNet[VOS03] présentant hélas pour le français une couverture moindre. La base de données lexicales WordNet est la plus aboutie et nombre de travaux l'utilisent de façon centrale. Aussi, il nous a semblé nécessaire de fournir une vision d'ensemble de cette base de données pour clarifier les parties suivantes.

Base lexicale, synsets comme unité lexicale

WordNet est une base de données lexicale recensant les mots de la langue anglaise avec une excellente couverture. EuroWordNet est une ressource similaire pour un bon nombre de langues européennes, dont le hollandais, l'italien, l'espagnol, l'allemand et le français.

Ces ressources peuvent être vues comme des ontologies, c'est-à-dire une définition des mots et des concepts qu'ils désignent, par le biais de différentes relations entre ces mots ou ces concepts. Une ontologie requiert la couverture d'un domaine spécifié, il s'agit souvent d'un domaine restreint (tourisme : Harmonise, médecine : Ontomed). Les ontologies présentées ici sont destinées à couvrir la langue générale. Notamment, elles couvrent ou visent à couvrir la presque totalité du lexique que l'on rencontre dans des articles de journaux, en exceptant les termes trop spécialisés qui pourraient apparaître occasionnellement, comme Lebesgue_integral ou permittivity.

On peut également voir ces ressources comme des bases de données lexicales, c'est-à-dire comme un dictionnaire structuré fournissant des informations, formalisées ou en langue naturelle, sur ses entrées, ainsi que des liens entre les différentes entrées.

Chacun de ces points de vue implique que la ressource se présente sous la forme d'un graphe de nœuds lexicaux reliés les uns aux autres par des relations que nous présentons ci-après.

Dans les ontologies WordNet et EuroWordNet, l'unité lexicale de base, qui correspond aux nœuds du graphe, est un ensemble de synonymes (nommés synset pour synonym set). Ce synset peut être assimilé à un concept puisqu'il rassemble ou vise à rassembler tous les termes permettant de désigner ce concept. À chaque synset est associé un certain nombre d'informations propres au synset, dont la plus utilisée est une définition (gloss).

Nous détaillons ci-après les liens qui relient ces synsets et en précisent le sens. Chaque catégorie grammaticale est structurée par des liens entre synsets selon une logique qui lui est propre. Le choix des liens est guidé par des considérations psycholinguistiques, c'est-à-dire qu'elles tentent de rendre compte du lexique mental.

Structuration des différentes catégories grammaticales.

Noms

L'ontologie des noms est structurée autour d'une relation principale, l'hyponymie, qui relie un concept général à des concepts plus spécifiques, et son inverse, l'hyperonymie. Ce couple de

relations constitue la principale structuration de l'ontologie noms. Les noms sont aussi organisés selon la relation d'holonymie qui relie un élément à un ensemble le contenant (par exemple, *argentin* est un méronyme de *Argentine*).

Verbes

Les verbes possèdent également une hiérarchie fondée sur la relation d'hyponymie, nommée également troponymie dans le cas des verbes. Un verbe X est un hyponyme d'un autre verbe Y si l'assertion suivante est vraie : *« inf(X), c'est inf(Y) d'une certaine manière »* où *inf()* donne l'infinitif du verbe. Par exemple : « **courir**, c'est **se déplacer** à pied, rapidement en ne posant jamais les deux pieds en même temps sur le sol. ». « courir » est donc un hyponyme de « se déplacer ». Si l'on trouve un terme correspondant à la définition « X, c'est se déplacer à pied. » ou « ... se déplacer rapidement. », par exemple « se presser » dans le contexte « un homme se presse dans la rue », ce terme sera un hyperonyme de courir et un hyponyme de se déplacer.

La hiérarchie obtenue est moins grande que celle des noms. En effet, les profondeurs dans la taxonomie des verbes sont souvent assez faibles, les feuilles arrivant souvent à une profondeur de 3 et dépassant rarement 5.

Entailment, Cause

En plus de la relation d'hyponymie, deux autres relations structurent principalement le lexique des verbes : la relation de cause et d'implication (entailment). La relation de cause relie sans surprise des couples de verbes désignant deux actions, processus ou états, le premier entraînant le second. (to give/to have, to fell/to fall, to kill/to die).

Toutes ces relations, y compris la troponymie, peuvent être regroupées dans une notion générale d'implication. Cette relation générale est présente entre deux verbes X et Y si le fait que l'action X se déroule implique que l'action Y se déroule aussi.

Cette relation se décline dans les relations vues précédemment : l'hyponymie, aussi appelée troponymie, les relations de cause, d'inclusion temporelle et de prérequis, qu'on peut classer selon la hiérarchie suivante [FEL98] :

```
implication (entailment)
                  inclusion temporelle
                       troponymie (hyponymie)
                       inclusion temporelle proprement dite
                  pas d'inclusion temporelle
                       orientée vers le futur : cause
                       orientée vers le passé : prérequis
```

Dans WordNet, ces différents sous-types sont répartis en trois classes : hyponymie, cause et une troisième classe générale étiquetée « entailment » regroupant les cas qui ne correspondent pas aux deux premières classes. On peut reprocher à ce jeu d'étiquettes de rapprocher indûment deux relations ayant peu de choses en commun : l'inclusion temporelle (to sleep → to snore, to walk → to step) et le prérequis (to try → to succeed)

Dans EuroWordNet, on dénombre trois relations d'entailment : hyponymie, sous-événement et cause. La relation nommée « cause » dans EuroWordNet couvre les relations de cause et présupposition de WordNet. La différence est faite par une étiquette « factive/non-factive ». L'étiquette factive indique le sens dans lequel s'effectue l'implication. L'étiquette non-factive signifie que la relation suivie dans ce sens ne permet pas d'inférer avec certitude que le verbe d'origine implique le verbe cible. Ainsi on a :

```
WordNet :
```

```
                    to_succeed  ENTAILS  to_try
EWN :
                    to_succeed  IS_CAUSED_BY  to_try  factive
                    to_try      CAUSES    to_succeed  non-factive

WordNet :
       to_kill  CAUSES  to_die
EWN :
       to_kill  CAUSES    to_die  factive
       to_die   IS_CAUSED_BY  to_kill  non-factive
```

Notons de plus que contrairement à WordNet, EuroWordNet prévoit de recenser les liens inverses, ce qui est utile car les reformulations nécessaires à un SQR peuvent se faire dans les deux sens.

Structuration du lexique des adjectifs et des adverbes

Les deux dernières catégories grammaticales de WordNet sont beaucoup moins structurées que les noms et les verbes. Ce manque de structuration, allié à la plus faible significativité des termes de nature adjectivale et adverbiale pour la recherche d'informations, explique la faible utilisation de la ressource pour ces catégories.

Les adjectifs sont structurés principalement autour de la relations d'antonymie, ex : big (vs large). À cela est ajoutée la relation de quasi-synonymie (big → ample, astronomic, bigger, biggest, largish , blown-up, puffy , hulky , vast, Brobdingnagian,...)

La relation la plus fructueuse sur les adjectifs est la relation « attribut » qui rattache l'adjectif au nom de la qualité ou de la grandeur qu'il représente. (deep → depth, beautiful → beauty) qui permet par ce biais d'accéder au reste du réseau.

Le réseau des adverbes utilise le même type de relations que les adjectifs, excepté l'antonymie. La relation la plus fructueuse est la relation pertainimy qui rattache l'adverbe à l'adjectif dont il est dérivé et par là au reste du réseau.

Couverture et relations manquantes dans WordNet et EuroWordNet

La couverture d'une ressource dépend de l'usage que l'on veut en faire, notamment des mots qu'on y cherche et des relations dont on a besoin pour la tâche. Pour une ontologie, nous avons besoin que les différents lexèmes rencontrés dans les articles de journaux soient présents dans la base de données. Il faut également que les différents sens de ces mots soient énumérés, avec le moins d'omissions possibles.

Il n'est pas envisageable qu'une ressource couvre tous les mots de la langue. Dans la pratique, il y a toujours des mots et ses sens absents, qu'il s'agisse d'une omission, d'un terme provenant d'un domaine spécialisé ou d'un néologisme. Par exemple, le mot *airdate* présent dans la collection de documents AQUAINT, qui désigne la date de diffusion d'un programme à la télé ou la radio, n'est pas présent de WordNet 2.1. Cependant, en pratique la couverture de WordNet est excellente en terme de nombre de mots couverts.

Pour donner un exemple, nous pouvons donner le taux de couverture pour les concepts présents dans les questions. Nous avons désambiguïsé manuellement les 500 questions de la campagne d'évaluation TREC11, rattachant chacune au sens qui nous semblait être le bon. Dans ces 500 questions, nous avons désambiguïsé 1013 mots, sans prendre en considération certains termes jugés peu flexibles, comme les noms propres. Si nous enlevons les doublons, il reste 678 concepts parmi lesquels sept n'ont pas été rattachés à un sens de WordNet 2.1, soit une couverture de 99%. Voici

quelques exemples de mots manquant : *rate* dans le sens de *taux*, *H.R.* (*human resources*), *dogsledding* (course de traîneaux tirés par des chiens), *to crack* dans le sens de résoudre une énigme. Dans certains cas comme pour *rate* ou *come from*, on trouve des sens plus ou moins approchants, mais les gloss restreignent le sens et interdisent de ce fait le rattachement.

La couverture des relations est plus difficile à établir. En effet il nous semble plus difficile de juger des relations qui devraient être présentes ou non. On peut signaler quelques-uns des problèmes suivants :

- Doubles héritages : il arrive qu'un concept de WordNet descende de deux hyperonymes, par exemple *geisha* descend de *woman* et *Japanese*. Par contre, *samurai*, placé en dessous de *warrior* ne possède pas de lien d'hyperonymie avec *Japanese*. On notera toutefois que le terme *Japanese* est présent dans la définition de samurai, ce qui permet de relier les deux termes, bien que cela implique le passage par une relation non formalisée donc plus dangereuse.
- Les relations entre les verbes ne sont données que si elles sont vraies dans 100% des situations. Pour cette raison : les relations de cause et d'entailment sont peu fréquentes.
- La polysémie des verbes est fort importante ; deux fois plus que celle des noms. De nombreux verbes en anglais (run, drive) proposent un nombre de sens impressionnant notamment à cause du phénomène d'«Argument alternation», c'est-à-dire l'existence de plusieurs structures d'arguments pour un verbe permettant de décrire le même concept, c'est-à-dire le même événement. On peut donner l'exemple du verbe « to drive » :
    - diriger un véhicule : drive a car.
    - se déplacer quelque part par un moyen de transport : we drove by bus to university.
    - transporter quelqu'un : he drove me to school.

  qui donne souvent naissance à trois sens distincts dans WordNet.
- Dans un même ordre d'idées, les noms et les verbes pourraient bénéficier de relations dites converses[LAU05], comme les verbes vendre-acheter ou les noms père-fils, professeur-élève. Ceci améliorerait la détection de la réponse pour la question :

```
Q : Who is Tom Cruise's wife?
R : Nicole Kidman's husband, Tom Cruise.
```

- Les relations de gradation pourraient être utiles autant pour les noms (par exemple village, town, city, megalopolis) que pour les adjectifs (large, huge, enormous)

Pour ce qui est d'EuroWordNet en français, la couverture est bien moins satisfaisante. Les mots absents sont plus nombreux et on compte parmi les absents des mots relativement courants (exportation, islamique, autel, nonne, bosnien, bosniaque, etc.). Jacquin et al. [JAC06] pointent les problèmes d'une hiérarchie des noms comportant trop peu de niveaux. Certains mots ne sont pas reliés à leur hyperonyme. [JAC06] propose des techniques pour réparer les défauts d'EuroWordNet.

2.1.2.2 Utilisation de ressources lexicales structurées

Proximité sémantique

La question qui se pose pour WordNet est de quelle façon combiner les relations qu'il propose. L'utilisation de WordNet requiert en effet le parcours de chemins lexicaux, c'est le cas dans la plupart des approches que nous avons rencontrées, si l'on excepte l'utilisation triviale de WordNet comme dictionnaire de synonymes ou de morphologie dérivationnelle.

De nombreuses approches sont testées dans [BUD99] sur une tâche de détection des « malapropisms » (mots employés à tort). Cette tâche permet le test de nombreuses techniques utilisant différents aspects de WordNet.

Ces heuristiques peuvent s'utiliser dans différents domaines comme le résumé automatique[BARZ97] et les questions-réponses. Le système de Jijkoun présenté à RTE 2005[JIJ05] utilise deux mesures de similarité lexicale, celle de Lin[LIN98], basée sur des dépendances, et celle de Hirst et St-Onge utilisant une mesure de chaînage lexical. Les chaînes lexicales sont également utilisées dans le système FALCON [MOL02].

Reconnaissance de la similarité lexicale

Nous alors décrire la mesure de proximité sémantique de Hirst et St-Onge[HIR97] qui se base sur les liens de WordNet. Et permet d'évaluer la force de la relation entre deux mots par une mesure sur la forme des chemins de liens qui relient leurs synsets dans WordNet. Cette mesure est utilisée par Hirst et St-Onge pour détecter des chaînes lexicales dans les textes. Ces chaînes sont des ensembles de mots d'un document liés par une relation de similarité lexicale.

L'algorithme de la mesure de similarité est le suivant. Il s'appuie sur trois types de liens unitaires :

- les liens descendants : hyponymie, holonymie, relations d'entailment.
- montants : hyperonymie et méronymie
- horizontaux : antonymie, quasi-synonymie, liens morphologiques entre catégories grammaticales

Si la plupart des choix de cette classification apparaît plutôt naturelle, on peut se demander ce qui fait pourquoi la relation d'holonymie a été classée dans les relations descendantes, plutôt que son inverse, la relation de méronymie. À partir de ces liens unitaires, la distance définit trois forces de relation différentes entre les noms de WordNet.

- La relation entre deux mots est dite extra-forte (extra strong) s'ils sont strictement identiques
- La relation est dite forte (strong) dans les cas suivants :
  - Les deux mots appartiennent au même synset de WordNet (ex : human, person)
  - Les deux mots appartiennent à deux synsets différents qui sont connectés par un lien horizontal, par exemple : precursor – successor, qui sont des antonymes.
  - Il y a un lien quelconque entre les deux termes, mais en plus, l'un des mots est un composé de l'autre, (par exemple, school et private school).
- La relation est dite moyenne (medium strong) s'il existe un chemin de relations dans WordNet les reliant, qui suive un motif particulier que l'on peut retrouver dans [HIR97]. Ils admettent des successions de liens allant dans différentes directions, par exemple : une succession de liens montants suivie d'une succession de liens descendants. Les chemins ainsi créés peuvent donc être d'une taille relativement longue.

Les relations sont classées selon le principe suivant :

les relations extra-fortes sont supérieures aux relations fortes, elles-mêmes supérieures aux relations moyennes, ces dernières étant départagées par le poids suivant :

```
poids = C – longueur du chemin – k * nombre de changements de
direction
```

où C et k sont des constantes.

Utilisation pour la segmentation du discours

Ces chaînes sont utilisées de façon prometteuse par Morris et Hirst pour effectuer des segmentations de textes. Notons que d'autres techniques peuvent être utilisées pour repérer les segments qui correspondent à une thématique homogène. On peut citer notamment le système de segmentation thématique par utilisation de la cohésion lexicale d'Olivier Ferret, SEGCOHLEX [FER98]. Elle s'appuie sur la connaissance d'unités thématiques regroupant les mots ayant un lien thématique. Ces unités thématiques sont calculées grâce aux cooccurrences des mots dans un corpus.

Ce type de techniques pourrait s'appliquer à la problématique de QR dans le cadre d'un travail qui ne s'arrête pas à la limite de la phrase. En effet, un même article peut changer de thématique ou aborder des aspects différents d'un même sujet. Dans le pire des cas, un article peut être une collection de faits sans points communs. On peut citer les exemples du New York Times citant la liste des faits ayant eu lieu à la même date. Un SQR travaillant sur des passages plus grands que la phrase pourrait bénéficier d'un système de segmentation, qui mettrait en évidence la faiblesse des liens lexicaux dans ce genre d'articles. Inversement, la segmentation permettrait de rapprocher des passages distants d'un même article partageant une thématique commune.

Utilisation pour l'appariement des termes

Différentes approches de reconnaissance de la similarité lexicale ont été utilisées pour l'appariement de termes. Par exemple, la mesure de Hirst et St-Onge[HIR97] est utilisée par Moldovan et al.[MOL02]. Dans cet article, ils montrent que de nombreux chemins trouvés sont au moins de longueur 2. Ils assignent de manière empirique un poids à chaque type de liens. [BAR05c] note que de nombreux liens sémantiques en QR utilisent des chaînes de liens de types opposés, par exemple, l'association d'une hyperonymie suivie d'une hyponymie, d'une méronymie suivie d'une holonymie.

L'utilisation de telles relations permet parfois d'obtenir la mise en relation souhaitée, comme montré dans l'exemple suivant :

```
Q : How much folic acid should an expectant mother get daily ?
R : ... Also, the term « good source » is based on the RDA of
400 micrograms daily for a pregnant woman.
```

Pour ce couple de phrases, les mots *expectant* et *pregnant* sont liés par une relation de quasi-synonymie, les termes *mother* et *woman* sont reliés car *woman* se trouve dans la définition du mot *mother*. On notera encore à ce sujet le manque de double héritage puisque mother pourrait hériter à la fois de parent et de woman.

Vérification du type de la réponse

Une utilisation particulière de WordNet consiste à vérifier le type de la réponse attendue, dans le cas où ce type n'est pas géré par un mécanisme spécifique comme les entités nommées. C'est le cas pour les questions du type « What + nom... », par exemple « What volcano destroyed Pompeii? » ou encore « What is the democratic party's symbol? »

Dans le premier cas, on notera que le Vésuve ainsi que bon nombre de ses confrères, sont présents comme hyponymes du synset *volcano*. En revanche, pour ce qui est des symboles, assez peu sont présents dans WordNet. Il y a alors un risque de mal classer les bonnes réponses si celles-ci ne sont pas reliées à leur type dans WordNet. Pour cette raison, des méthodes utilisant des ressources encyclopédiques comme Wikipédia sont développées. Une autre méthode consiste à rechercher les informations de typage en posant la question à un système de questions-réponses, par

exemple, à celui qui était chargé de répondre à la question initiale. Cependant, le risque d'erreur étant important, il faut se méfier d'une telle méthode.

Utilisation comme source des réponses

Certaines questions peuvent se trouver dans WordNet directement. C'est le cas des questions de définition, qui peuvent trouver un élément de réponse grâce à la relation d'hyperonymie de WordNet (ex : cassowary -> bird) mais surtout grâce à la gloss associée au mot (cassowary : large black flightless bird of Australia and New Guinea having a horny head crest)

D'autres questions, par exemple sur les emplacements géographiques ou les constituants d'un objet composite, peuvent trouver une réponse grâce aux relations de méronymie. Par exemple :

```
Q : What does a human cell contain ?
```

La relation de méronymie renverra la liste suivante :

```
 R : cell wall, cell membrane, cytomembrane, plasma membrane ,
energid, protoplast , cytoplasm , nucleus, cell nucleus,
karyon , organelle, cell organ , vacuole .
```

Une autre technique encore consiste à utiliser ces informations non comme source de réponses, mais pour confirmer une réponse extraite par ailleurs d'un corpus textuel, comme décrit dans [LIN02].

2.2 Sémantique locale syntagmatique

Les systèmes de question-réponses possèdent tous un moyen d'estimer si les mots de la réponse candidate se retrouvent à proximité dans la réponse. L'idéal est de vérifier que les relations syntagmatiques entre les termes de la question s'apparient avec celles de la réponse, que ce soit au plan syntaxique ou syntagmatique. Dans les cas les plus modestes, il s'agit d'une simple mesure de proximité des mots.

2.2.1 La syntaxe et la sémantique syntagmatique

L'analyse morphosyntaxique, syntaxique ou sémantique peut se faire par des ensembles de règles ou par apprentissage statistique sur un corpus annoté. L'analyseur XIP donne un exemple d'analyseur morphosyntaxique et syntaxique basé sur des règles pour le français et l'anglais.

Des ressources ont été développées, comme Penn Tree Bank pour la morphosyntaxe et la syntaxe, qui ont permis d'entraîner de nombreux analyseurs syntaxiques statistiques pour l'anglais(Stanford parser, Charniak parser). La création de ressources comme PropBank : Corpus où les relations entre un verbe et ses arguments sont annotées sémantiquement[KIN02] ou FrameNet facilitent la création d'outils d'étiquetage sémantique [PRA04], [SHI04].

Ces analyseurs sont ensuite utilisés par des systèmes de questions-réponses (ex: [NAR04]) pour créer une représentation structurée de la question et l'apparier avec les réponses candidates. La différence entre les structurations syntaxiques et sémantiques tient à ce que la structuration

syntaxique est fortement liée à la forme du texte. Ce type de représentation permet en général de reconstituer la phrase mot pour mot en conservant l'ordre de ceux-ci. La représentation sémantique est plus distante de la structure de la phrase. Idéalement, deux phrases portant le même sens doivent être analysées de façon identique. Dans la pratique, la qualité du rapprochement dépend de la qualité de l'analyse sémantique effectuée. Certains systèmes se contentent d'une représentation proche de la syntaxe, faute de disponibilité d'un analyseur sémantique fiable.

De nombreux problèmes se posent pour obtenir une analyse fiable. Un problème commun à l'analyse syntaxique et sémantique est celui du rattachement correct des arguments à leur gouverneur, notamment pour les groupes prépositionnels. Si l'on veut obtenir une analyse sémantique de qualité, il est de plus utile de désambiguïser les verbes du texte afin de connaître leur structure sémantique et effectuer un rattachement correct de leurs différents arguments. Par exemple les phrases suivantes pourraient être analysées d'une manière similaire. Ci dessous l'analyse est représentée sous le formalisme de FrameNet, chaque étiquette d'un terme désigne le rôle qui le relie au prédicat *Buy,* formant ainsi un graphe en étoile :

```
Q : [TIME:What year] was [GOODS:Alaska] [Predicat_buy:purchased]?
R1: But [TIME:by 1867] , when [BUYER:Secretary of State William
H. Seward] negotiated the [Predicat_buy:purchase] of
[GOODS:Alaska] [SELLER:from the Russians].
R2: In Seward , the town named for [BUYER:Secretary of State
William Seward] , who [Predicat_buy:bought] [GOODS:Alaska]
[MONEY:for $ 7.2 million] [TIME:in 1867] , a multimillion ....
```

L'intérêt d'une analyse sémantique poussée par rapport à l'analyse syntaxique est, en réduisant la distance entre les représentations des phrases de même sens, de simplifier le processus d'appariement ultérieur entre question et réponse. Les programmes utilisant un appariement syntaxique doivent se contenter d'un appariement partiel ou implémenter des règles de reformulation, comme dans [ZAN05].

L'appariement des représentations de la question et de la réponse peut constituer le centre du système d'appariement [HART04],[BAR05a] ou donner une mesure de similarité syntagmatique qui sera utilisée comme un critère parmi ceux fournis à un système de décision [HIC06]. Dans le second cas, le système d'analyse sémantique peut se contenter de représentations partielles de la question et de la réponse. Par exemple, dans l'article de Hickl, il s'agit de vérifier la structure casuelle du verbe principal de la question.

2.2.2 FrameNet et rôles sémantiques

La frame est une description de la structure casuelle d'un événement, action ou processus. Elle décrit les différents rôles sémantiques acceptables pour décrire cet événement, en faisant la distinction entre les rôles CORE, qui sont les arguments fondamentaux de l'événement. Parmi ceux-ci on reconnaît en général un agent, un patient, un objet, etc., bien que les étiquettes utilisées pour ces rôles soient spécifiques à chaque événement. Ainsi, dans l'événement concernant les transactions commerciales (Commercial Transaction), les arguments fondamentaux seront étiquetés : Acheteur, Vendeur, Bien, Argent.

Les rôles NON-CORE donnent des arguments accessoires de l'événement, qui contiennent entre autres les compléments circonstanciels. Outre ceux-ci, on trouvera des caractéristiques facultatives de l'action, par exemple, le prix d'une transaction commerciale peut être fixé par un taux (ex : des tomates vendues ***1€99 le kilo***)

Cette description de concept est lexicalisée par une liste de lexèmes décrivant l'événement, par exemple, pour l'événement *Change of Leadership*, les unités lexicales associées sont :

```
appoint.v, coronate.v, coup.n, crown.v, depose.v, dethrone.v,
elect.v, election.n, enthrone.v, install.v, insurrection.n,
mutiny.n, mutiny.v, name.v, oust.v, overthrow.n, overthrow.v,
rebellion.n, revolt.n, revolt.v, revolution.n, revolutionary.n,
take over.v, throne.v, topple.v, uprising.n, vest.v
```

L'extension donnée au mot (.v, .n ou .a) indique la catégorie lexicale de celui-ci. On notera donc que les noms et les verbes décrivant l'événement sont réunis dans la même liste d'unités lexicales, ce qui donne une famille élargie de quasi-synonymes, intéressante pour fournir des reformulations.

2.2.3 Connaissances pragmatiques de type script/schème

Certaines connaissances sortent du cadre de la sémantique. Nous entrons alors dans le vaste champ des connaissances pragmatiques qui regroupent toutes les connaissances sur le monde qui permettent la compréhension d'un énoncé. Les connaissances permettant d'établir le lien entre deux actions liées logiquement ou temporellement entre elles sont du domaine de la pragmatique. Prenons par exemple la paire de questions suivante :

```
Q1 : Quelle entreprise a remporté le marché de l'avion de chasse
européen ?
Q2 : Qui fabriquera l'avion de chasse européen?
```

Ces questions demandent la même réponse, mais pour s'en assurer, un système doit être capable de déterminer que « fabriquer » et « remporter le marché » sont très proches. En effet on peut voir une relation d'entailment de type prérequis ou à la rigueur de cause[8] entre remporter_marché(X,Y) et fabriquer(X,Y).

D'après Schank[SCH77] ces connaissances peuvent être organisées sous forme de scripts, qui décrivent les enchaînements logiques de ces actions. Un exemple courant d'un tel script décrivant la procédure d'un repas dans un restaurant, pour laquelle les opérations successives seront précisées :

```
Entrer dans le restaurant, passer commande, manger, [demander
la note], payer, sortir.
```

Scripts dans FrameNet

Le réseau de frames de FrameNet apporte une grande quantité d'informations de nature pragmatique sur les liens logiques entre deux événements ou actions. Ces multiples liens produisent des voisinages d'événements pouvant se produire dans le cadre d'un scénario particulier. Les scénarios sont représentés par des frames particulières auxquelles sont reliées des frames représentant, par exemple, les actions participant de ce scénario.

8 Il ne peut s'agir ici d'une relation de cause car le fait de remporter un marché de l'avion n'implique pas nécessairement sa fabrication, une faillite de l'entreprise étant envisageable.

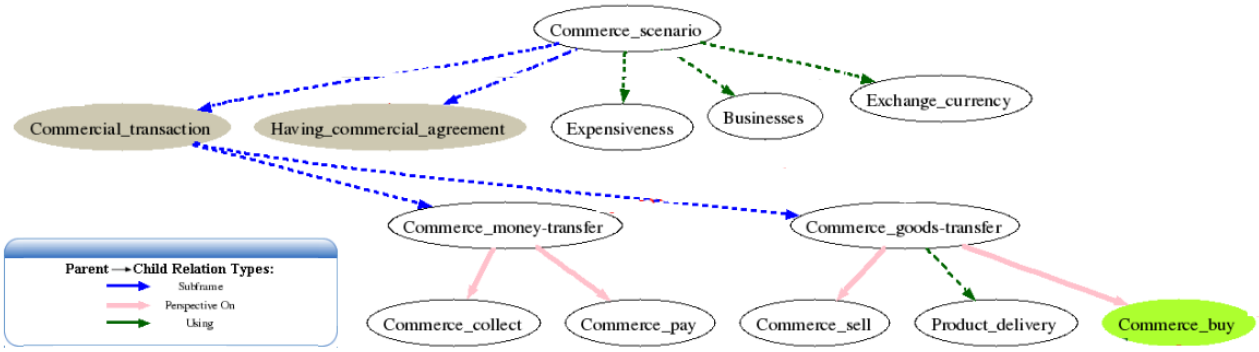


Fig. 3: Un scénario et les frames dépendantes

Cette figure montre l'étendue du scénario de transaction commerciale obtenue en descendant la hiérarchie des frames à partir de la frame de scénario. Ce scénario contient les actions de paiement et de réception de l'argent, de vente et de livraison de produit, représentées par des unités lexicales telles que : to buy, to purchase, to sell, to charge(faire payer une somme).

Les différentes relations possibles sont les suivantes :

- Inherits / Is Inherited By: La relation d'héritage est la plus forte de toutes les relations. Elle implique que tout ce qui est vrai dans la frame parente se retrouve dans la frame fille. Notamment les structures casuelles de la frame parente auront toutes un équivalent dans la frame fille.
- Perspective on / Is perspectivized in: Permet de distinguer des points de vue différents d'un même événement. Hiring et Get_a_job sont des perspectives de Begin_employment.
- Using / Is Used By: Using est une relation plus générale que la précédente, indiquant qu'une partie de l'événement fils fait référence à l'événement père, par exemple (Communication is used by Discussion is used by Jury_Deliberation, Point_of_dispute)
- Subframe of: / Has Subframes: Indique les sous-événements d'un événement plus vaste.
- Precedes:/ Is Preceded by: Indique la succession temporelle des événements
- Causative / Is caused By: Relation de cause à effet (Killing is causative of Death)
- Inchoative / Is Inchoative of: Un processus est inchoatif d'un état s'il permet d'entrer dans cet état (Becoming_member is_inchoative_of Membership)
- See Also: Relation non typée.

Relations de scripts dans WordNet et EuroWordNet

Il est à noter que WordNet et EWN proposent dans le réseau des verbes, des relations d'entailment pouvant être assimilées à des relations de scripts. On considérera notamment les différentes relations d'entailment de WordNet qui se répartissent, comme nous l'avons vu précédemment, en relations de cause à effet, de succession ou d'inclusion temporelle. Cependant, ce type de relations est somme toute assez peu présent dans WordNet et est limité à des cas où le lien d'implication est fortement affirmé, par exemple (ronfler => dormir)[9].

Dans ce cadre il est difficile de mettre en relation « recevoir_un_cadeau » et « être joyeux »

9 Pour cette raison, les relations de WordNet de ce type sont unidirectionnelles (dormir n'implique pas ronfler)

puisque chacune des deux situations peut être la cause (respectivement la conséquence) d'une grande variété d'événements. Une autre contrainte des relations de WordNet est qu'elles doivent être établies entre deux unités lexicales, ce qui est rarement le cas, notamment si l'on considère que les situations des scripts sont souvent décrites par des locutions verbales : remporter un marché, aller en prison, aller au restaurant, être joyeux.

Utilisation des relations de FrameNet dans un appariement de termes.

Les relations entre frames permettent de faire le lien entre des événements partageant un contexte commun comme une transaction commerciale ou un procès.

Cette connaissance permet de trouver des variations dans la description d'un événement, comme par exemple d'établir un lien de succession temporelle entre trial.n (frame Trial) et sentence.v (frame Sentencing), comme montré dans la figure 4.

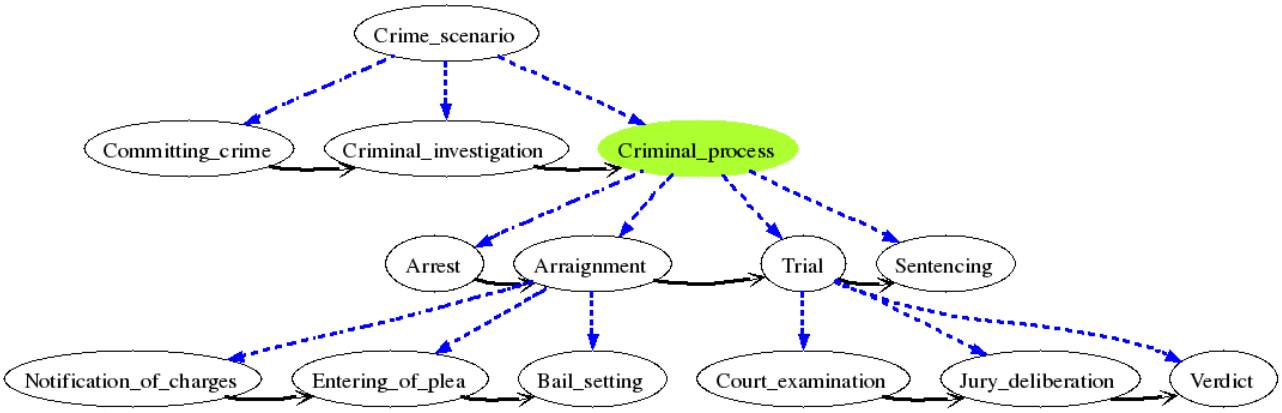


Fig. 4: Événements liés dans le cadre d'un scénario « Procédure judiciaire »

Ceci permettrait en théorie de rapprocher les formulations suivantes :

```
Q: When did Dorothy Williams' trial take place?
R : Dorothy Williams, 44, was sentenced in 1991 to die for the
1989 strangling of 97-year-old Mary Harris.
```

Le système QuaLiM [KAI05] fondé sur l'appariement de structures grâce à FrameNet, illustre la recherche de reformulations en utilisant le corpus d'exemples de chaque frame. Dans ce système, les documents sont obtenus par des requêtes à divers moteurs de recherche (Google, Lucene). Pour une même question le système propose plusieurs requêtes prenant en compte les variations de la réalisation du prédicat et de ses arguments indiquées par FrameNet[KAI07]. Par exemple, pour la question :

```
Q : Who purchased YouTube ?
```

La liste de patrons de requête formulés d'après les variations proposées par la frame « Purchase » contiendra :

```
ANSWER[NP] purchased YouTube
YouTube was purchased by ANSWER[NP]
ANSWER[NP] had purchased YouTube
YouTube was bought by ANSWER[NP]
ANSWER[NP] purchase of YouTube
```

Les requêtes utilisées correspondent à ces patrons privés de la partie ANSWER[NP], qui ne sera utilisée que lors l'extraction de la réponse.

Utilisation des rôles sémantiques en questions-réponses

Un des intérêts de FrameNet est de fournir un corpus de phrases permettant de faciliter le rattachement d'une structure prédicat arguments à sa structure casuelle. Ce corpus donne notamment pour chaque rôle sémantique que l'événement d'une frame autorise, un ou plusieurs exemples de réalisation. Ces exemples font intervenir différentes unités lexicales de la frame. Cependant, le lien entre une unité lexicale et les rôles sémantiques qu'elle admet n'est pas fourni.

Ces exemples permettent de rattacher les structures prédicat-argument des questions et des réponses à FrameNet, d'utiliser ensuite les informations contenues dans la frame, ainsi que le réseau de relations entre frames. Par exemple [NAR04] utilise FrameNet dans l'analyse de la question pour enrichir la représentation sémantique de l'événement par ses rôles sémantiques potentiels. La réponse recherchée est représentée sous la forme d'une frame dont certains rôles sémantiques ne sont pas instanciés, par exemple :

```
Q : What kind of [GOODS:nuclear goods] were [target-predicate:stolen]
[VICTIM:from the Russian Navy]?
R : [VICTIM(P1):Russia's Pacific Fleet] has fallen prey to
[GOODS:nuclear] [target-predicate(P1):theft];
in 1/69 [GOODS(P2):7 kg of HEU] was reportedly [target-
predicate(P2):stolen] [VICTIM:from a naval base] [SOURCE(P2):in
Sovetskaya Gavan].
```

Ici, la représentation de la question instancie les rôles présents dans celle-ci : GOODS et VICTIM, mais n'instancie pas les autres rôles possibles pour la frame, comme PERPETRATOR (celui qui comment le crime) et SOURCE (l'emplacement initial des produits). De plus, la réponse recherchée « What kind of nuclear good » correspond au prédicat sémantique GOODS. La réponse est constituée de deux phrases et donc de deux prédicats (P1,P2) correspondant à celui de la question (theft, steal). La première permet de rattacher l'argument de rôle sémantique VICTIM, sous forme de la variation sémantique (Russia's navy - Russia's Fleet). La seconde phrase permet de rattacher le reste des informations recherchées, notamment la réponse, reconnue comme le rôle sémantique GOODS : « 7 kg of HEU ».

Problèmes de couverture de FrameNet

La ressource FrameNet pose toutefois des problèmes de couverture du lexique et d'absence de liens souhaités entre les différentes frames. Prenons pour exemple le couple de question-réponse :

```
Q : What female leader succeeded Ferdinand Marcos as the
president of Philippines ?
R : .....revolution that toppled Ferdinand Marcos . Corazon
Aquino new president ....
```

Pour procéder à un appariement des questions on a besoin de faire le lien entre « topple » et « succeed », ce qui doit pouvoir se faire par l'inférence suivante : *to topple* (renverser) est une unité lexicale de la frame Change_of_Leadership (Changement de dirigeants) cet événement implique un ancien et un nouveau dirigeant, le nouveau dirigeant est successeur de l'ancien.

Cet exemple pose toutefois plusieurs problèmes. Premièrement, le verbe *succeed* est présent dans FrameNet seulement dans le sens de réussir et non dans celui de succession temporelle.

Cependant, WordNet nous informe que *succeed* est synonyme de *follow*, ce qui permet de rattacher le verbe à la frame Relative_time.

Le verbe topple est plus facile à mettre en relation car il est présent dans la base d'unités lexicales de FrameNet. Il correspond à la frame : Change_of_Leadership. De plus, des mots de la question (leader,president) sont également présents dans la frame.

Ensuite, il n'y a pas de lien évident entre les deux frames sélectionnées, ce qui fait que FrameNet dans son état d'élaboration actuel ne permettrait pas de résoudre ce problème.

Les problèmes de faible couverture lexicale peuvent être palliés en combinant FrameNet à d'autres ressources. Ainsi, la plupart des utilisations combinent FrameNet avec PropBank qui constitue également un corpus annoté de rôles sémantiques, sans posséder de réseau de relations entre frames, mais il fournit une meilleure couverture du lexique et un plus grand nombre d'exemples.

Conclusion sur la sémantique

Nous avons vu que la conjugaison du repérage des variations de termes (sémantique paradigmatique) et des variations de formulations des différents rôles sémantiques (sémantique syntagmatique) permettait un appariement fin et sémantiquement typé entre la structure sémantique de la question et de la réponse. Cette structuration permet notamment, si le type des arguments est correctement reconnu, de mettre en évidence la réponse recherchée. De plus le repérage précis des éléments correspondant à la question, permet d'expliciter à l'utilisateur la justification de la réponse rapportée.

Nous notons que ce type d'appariement sémantique, encore plus que l'appariement syntaxique, nécessite une grande quantité de ressources. Dans le cas du français, aucune ressource du type de FrameNet n'existe encore. De plus, l'analyse sémantique requiert la construction d'un analyseur sémantique qui présente un coût de développement important. Ceci pousse certains systèmes à explorer des techniques moins gourmandes en ressources.

2.2.4 Approximations syntaxiques

Pour résoudre le problème du coût de développement de ressources et d'un analyseur sémantique, il est possible de développer des méthodes plus approximatives.

Vérification de relations syntaxiques particulières.

Un pis-aller à l'approche précédente consiste à prendre en compte seulement certaines relations au sein de la phrase. Par exemple, [LIG06,section 6.2.2] propose une technique pour l'extraction des réponses fondée sur les relations syntaxiques environnant la réponse. Cette technique peut s'appliquer que le type attendu de la réponse soit une entité nommée ou un type général.

De plus, nous avons intégré au système de validation de réponses du LIMSI, un module de vérification des relations temporelles basé sur la reconnaissance des entités d'expressions temporelles présentée dans [BAT08]. La question est analysée sémantiquement par un analyseur spécifique pour rattacher les compléments de temps à l'événement qu'ils caractérisent. Ce traitement sera décrit au chapitre 3, en section 3.3.1.4. La vérification de la présence du lien dans la réponse se fait par une approximation, consistant à vérifier que les correspondants dans la réponse du complément de temps et de son gouverneur sont situés dans une fenêtre de 10 ou 20 mots...

Les systèmes QATAR[BOU05] tient compte des relations de dépendance syntaxique appariées

entre la question et la réponse, sous forme d'un poids de similarité syntaxique défini comme la proportion de relations reconnues. De cette façon, il autorise une reconnaissance partielle des relations syntaxiques. Ce poids est combiné ensuite à d'autres critères par une somme pondérée en fonction de l'importance de ces différents critères.

Approximation de l'appariement syntaxique par des mesures de densités

De nombreux systèmes n'effectuent pas d'appariement des relations syntagmatiques car cela demande de disposer d'un analyseur syntaxique fiable. De plus la conception d'un algorithme d'appariement entre questions et réponses est un problème ouvert qui demande donc un temps de développement non négligeable. Pour ces raisons, les systèmes approximent la composante syntagmatique par des mesures telles que :

- Densité linéaire : Une mesure du rapprochement des mots du passage sans considération du type de relations unissant les mots, comme dans le système QRISTAL[LAU05].
- Densité syntaxique[BAR05a][LIG06,p106] : Mesure qui reprend l'idée précédente en mesurant la densité sur une structure syntaxique arborescente. L'intérêt de cette mesure est de permettre l'omission des modifieurs (adjectifs, groupes prépositionnels) et des propositions incises qui éloignent les éléments de justification dans le cas d'une distance linéaire. Cette mesure de densité améliore un peu les résultats par rapport à la mesure linéaire [BAR05a].
- D'autres méthodes se fondent sur le repérage de la plus longue sous-chaîne (LCS) commune entre la question et la réponse, permettant ainsi de reconnaître des passages répondant à la question en conservant une formulation assez similaire. De nombreux systèmes de reconnaissance de l'implication utilisent ce genre de mesures, parfois sans utiliser d'autres mesures de proximité[KOZ06], parfois en collaboration avec des mesures syntaxiques ou de sémantique syntagmatique précises[HIC06]. Toutefois l'utilisation d'une mesure de LCS comme seule estimation syntagmatique de la réponse permet seulement d'atteindre des scores d'exactitude médiocres, de l'ordre de 55%.

2.3 Connaissances pragmatiques et encyclopédiques

En linguistique, la pragmatique peut être définie de façon générale comme l'étude des phénomènes dont la signification ne peut être comprise qu'en replaçant l'énoncé dans son contexte, c'est-à-dire comme une action sur le monde.

Cette définition permet d'englober deux aspects importants. D'une part la pragmatique étudie la logique du langage comme instrument d'interaction entre les individus, traitant de l'existence d'informations présupposées et d'actes de langages dans un énoncé. D'autre part, elle traite de l'utilisation de connaissances sur le fonctionnement du monde dans l'analyse d'un énoncé, celles-ci permettant par exemple de faire des inférences sur le texte et d'éviter des erreurs de compréhension.

Le premier aspect de la pragmatique, orienté vers le langage lui-même, est peu abordé dans notre thèse, à l'exception de la notion de présupposés. Les présupposés d'un énoncé sont des informations que l'auteur d'un énoncé suppose comme partagées avec son interlocuteur. De ce fait, il s'attend à ce qu'il n'y ait pas de remise en cause de celles-ci. Par exemple, l'énoncé « Prête-moi ta montre » indique que le locuteur croit que son destinataire possède l'objet. Dans « Pierre ignorait que Paul était là » L'ignorance de Pierre est simplement affirmée, mais l'information sur la présence de Paul est présupposée. En raison de cette présupposition, l'énoncé suivant est surprenant, et l'on ne peut l'interpréter :

```
Pierre ignorait que Paul était là. Pourtant, Paul était parti.
```

Cette connaissance permet d'effectuer certains raisonnements utiles pour répondre à des questions. D'une part ces informations supposées vraies par l'utilisateur et ce fait, la justification qui lui est apportée n'a pas besoin d'apporter la preuve de leur validité. Un système pourrait même se passer de vérifier ces présupposés, à condition de faire confiance à leur détection ainsi qu'aux connaissances de l'utilisateur qui a posé la question. Un exemple d'information facultative est illustré par le couple question-réponse suivant tiré de TREC11:

```
Q1395 : Who is Tom Cruise married to?
R : The celebrity couple Tom Cruise and Nicole Kidman are a very
big deal in London and have been for their entire run here.
```

Ici la question demande de trouver le nom de l'épouse de Tom Cruise. Cette question sous-entend que l'auteur pense que Tom Cruise est marié avec quelqu'un. La réponse *Nicole Kidman* (réponse datée, car la collection utilisée pour la campagne date des années 1998 à 2000) est presque totalement justifiée par le passage réponse. À ceci près que l'article ne précise pas si le couple est marié. L'information présupposée vient donc combler ce manque.

Le second aspect de la pragmatique concerne les connaissances de sens commun. Celles-ci, partagées par tout être humain, comportent notamment la connaissance du fonctionnement du monde. On peut citer les ordres de grandeur des objets ou les connaissances des règles physiques. Ces connaissances permettent aux êtres humains de détecter des erreurs d'interprétation. Ainsi, dans l'exemple :

```
L'enfant a mis la voiture dans sa poche.
```

elles permettent d'inférer que la voiture est un jouet, en prenant en considération l'ordre de grandeur des objets ainsi que le fait que les voitures sont des jouets usuels. Ces connaissances de sens commun sont censées être connues de tous et, pour cette raison, ne sont que très rarement rappelées dans les ressources textuelles telles que les articles de journaux ou les pages de la toile.

Dans des encyclopédies, comme Wikipédia, on peut trouver certaines de ces connaissances, par exemple : la Terre tourne autour du soleil, un réfrigérateur sert à conserver la nourriture. Mais on ne peut pas trouver des connaissances trop basiques et trop universellement connues sur le fonctionnement du monde, comme les ordres de grandeur, et les lois de la physique qui régissent notre vie quotidienne.

Ceci met en évidence une gradation parmi les connaissances sur le monde, des connaissances les plus élémentaires, les mieux connues et partagées, dont l'évocation dans un document semblerait superflue et ridicule, jusqu'aux connaissances les moins partagées, qui méritent d'être rappelées dans des articles, par exemple les connaissances d'un spécialiste.

Cyc et OpenCyc

Cyc/OpenCyc[10], produits par l'entreprise Cycorp, s'attaquent plutôt aux connaissances de sens commun, tandis que les encyclopédies classiques visent des connaissances moins partagées. Elles répondent donc à un manque d'information.

Cyc comporte des constantes représentant des individus, et des axiomes sur ces individus utilisant des quantificateurs (#$forAll, #$thereExists) et des opérateurs logiques ($#and, $#or, $#not, $#implies). Elle présente également des fonctions telles que #$FruitFn qui, à une liste de plantes associera la liste de leurs fruits. Cyc contient également des prédicats pour gérer les notions de croyance et de connaissance.

10 http://www.cyc.com/cycdoc/vocab/vocab-toc.html

Utilisation de Cyc en QR

Les utilisations de Cyc en QR sont presque inexistantes. Elles portent en général sur des points particuliers tels que les règles fondamentales du monde comme le temps ou l'espace.

On notera l'utilisation d'OpenCyc comme outil de vérification par le système PIQUANT[CHU02] à TREC11 (score : 0,588; 5ième). Celui-ci utilise les ordres de grandeur pour vérifier les réponses. [MOL05] utilise une autre ontologie de connaissances pragmatiques SUMO pour effectuer des raisonnements temporels.

Enfin, nous noterons l'utilisation de cette ressource pour le système de questions-réponses MySentient Answers, développé justement par la société Cycorp qui développe Cyc. Ce système consiste à interroger la base de Cyc pour en extraire directement les réponses. Il se rapproche donc de la problématique de fouille de données structurées et s'éloigne de celle des questions-réponses classiques, pour laquelle les réponses sont cherchées dans des textes non structurés.

2.4 La sémantique distante ou énonciative

Bien que de nombreux systèmes limitent leur portée au niveau de la phrase, nous voulons attirer l'attention sur les informations qui peuvent être rapportées par des parties éloignées de l'énoncé, situées au delà de la phrase.

Nous définissons la sémantique distante comme l'ensemble des procédés permettant de relier deux parties distantes d'un énoncé. Ce sont des procédés qui permettent de transmettre à une partie des informations sémantiques présentes dans la seconde. On compte parmi ces phénomènes la thématisation d'une idée ou d'un concept, les anaphores et coréférences. Dans le cadre de notre recherche, nous étendrons également cette définition à la recherche d'informations dans des ressources distantes.

2.4.1 Anaphores et coréférences

Les anaphores sont des phénomènes consistant en la présence d'une partie de l'énoncé faisant référence à un élément du contexte. La partie référée est en général un autre élément de l'énoncé :

> Je suis sorti ce matin avec mes **clefs.** Je ne **les** avais plus en rentrant chez moi le soir.

La coréférence est le phénomène où deux parties d'un énoncé réfèrent à la même entité. Par exemple :

> **Michael Jordan** was the best athlete ever. « **Michael** has always tried is best » says his trainer ....
> **George Bush** a déclaré que l'économie pouvait supporter le coût de la guerre en Irak. **Le président** s'est défendu de ...

Ici, *Michael Jordan* et *Michael* réfèrent à la même entité qui est un joueur de basket-ball américain, de même pour *George Bush* et *le président*.

2.4.2 Thématisation et contexte

Nous nommons thématisation le fait pour une information d'être valide au delà du contexte immédiat, c'est-à-dire au delà de la phrase où se situe son signifiant. L'étendue d'une information thématisée peut être le document entier. C'est le cas pour le titre d'un article de journaux, sa date de publication. Certaines informations circonstancielles définies dans l'article, en général à son début, (date, lieu) peuvent aussi être invariantes d'un bout à l'autre de l'article. Enfin le sujet de l'article (Personne, Événement, etc.) peut lui aussi avoir une portée globale.

Une thématisation locale peut aussi avoir lieu. Ceci est à rapprocher de la thématique de segmentation et de structuration des textes. Une thématisation locale peut venir masquer une information générale. Par exemple, elle peut venir redéfinir le contexte temporel global.

Une thématisation si elle est détectée peut permettre de restituer des informations manquantes dans une phrase donnée, cette absence dans la phrase pouvant être due à un échec de la reconnaissance du terme, comme ici :

```
Q1396 : What is the name of the volcano that destroyed the
ancient city of Pompeii ?
R: POMPEII, Italy In 62 A.D. it was devastated by an earthquake.
<just>In 79 A.D., it was buried in an avalanche of hot ash from
[REP Mount Vesuvius].</just>
Now, 2,000 years later, Pompeii is being destroyed again a
victim of its own popularity, creeping neglect and generations
of archaeologists more interested in digging up the past than
preserving it.
Twenty centuries after the city died, its cobblestone streets
are still ...
```

Dans la phrase réponse, le terme évoquant la destruction (« buried ») n'est pas directement relié au concept de destruction. L'enfouissement n'est pas particulièrement lié au concept de destruction. En effet, on peut par exemple enterrer quelque chose pour le préserver ou le protéger. Il s'agit d'un raisonnement sur le monde qui permet de savoir que la ville a été brûlée et rendue impropre à la vie, donc détruite. Cependant, si l'on s'était placé du point de vue de la richesse archéologique plutôt que la vie, on aurait plutôt déduit que cet enfouissement avait permis une conservation.

Malheureusement ce type d'inférence est très difficile à réaliser, demandant des représentations très fines des connaissances du monde. Il est donc utile de substituer à cette inférence d'autres techniques de rattachement.

La technique de rattachement par thématisation est ici un substitut possible à l'inférence. Le champ lexical de la destruction est en effet présent dans tout le paragraphe. Seule sa présence dans la phrase réponse n'est pas reconnaissable par un système automatique. Par contre, les deux autres variations de « destroy » le sont : « devastated » dans la phrase précédente et « destroy » dans la phrase suivante. Cependant ce type de rattachement semble dangereux à effectuer. En effet on peut imaginer que la phrase centrale ne traite pas de la destruction de la ville. Le paragraphe suivant propose quelques heuristiques pour évaluer le chaînage lexical.

2.4.3 Constructions parallèles

Il est assez fréquent que la justification de la réponse soit divisée en deux parties voire plus, on rencontre fréquemment un découpage en deux phrases ou propositions. Ces deux phrases partagent souvent plus que les informations requises par la justification. Par exemple pour la question 1400 de

la campagne TREC11, « When was telegraph invented », le passage suivant justifie la réponse (1844) en deux parties :

```
R : When <just>Samuel Finley Breese Morse opened the wire
between the Capitol and Baltimore on May 24, [REP1844]</just>, he
already knew he had successfully harnessed a force of nature
[...]
The challenge was to prove to the public and to Congress , which
had paid $ 30,000 <just>to build the line to Baltimore , that
the Morse magnetic telegraph</just> had practical applications .
```

Bien que ces phrases soient séparées par deux autres phrase totalisant 39 mots, on peut repérer un fort lien entre celles-ci. On peut en effet trouver des correspondances pour les termes :

- Samuel Morse (nom complet – nom seul)
- Baltimore (repris à l'identique)
- wire - line - telegraph

2.4.4 Coopération d'informations

Il est possible qu'un même rattachement puisse être obtenu dans un même passage grâce à plusieurs techniques de rattachement. Les techniques de rattachement sont sujettes à erreurs et peuvent également échouer et ne pas indiquer un lien existant. Il est donc utile de faire coopérer ces différentes techniques pour confirmer les rattachements et réduire le nombre de rattachements omis.

Par exemple dans le passage suivant, les méthodes de chaînage lexical et d'anaphores sont intriquées et peuvent coopérer à relier la phrase de justification (entre balises `just`) au nom du joueur Michael Jordan. Par exemple, le pronom « He » et le «verbe « played » permettent tous les deux de rattacher le nom « player » de la phrase précédente, lequel est rattaché par anaphore au nom **Jordan.**

```
Q1397 : What was the largest crowd to ever come see Michael
Jordan?
R : Just as frightening is what the loss of Jordan means to the
NBA , which is coming off a protracted lockout that alienated
its fans . The one commodity that could have mended the sides
was Jordan , the league 's marquee player for the last decade .
<just>He played before a sold out arena every night , even
bringing out an NBA record crowd of [REP62,046] last season at
the Georgia Dome against the Hawks .</just>
```

L'utilisation de techniques de rattachement de fragments de textes peut bien sûr s'avérer dangereuse, notamment dans le cas d'énumérations. Ainsi dans l'index Yahoo du 12 sep 2007 on trouve la liste des articles du jour :

```
Poutine a accepté la démission de son Premier ministre(1)
Paris confirme l'accélération de la réforme des régimes
spéciaux(2).
Le délit d'abus de biens sociaux ne sera pas supprimé(3)
Le premier ministre japonais Shinzo Abe démissionne(4)
Fradkoy présente la démission de son gouvernement en Russie(5)
```

On peut relier fortement les phrases 1 et 4 grâce aux termes démission et Premier Ministre. Ce cas peut se trouver dans des articles énumérant brièvement des faits liés par exemple par une même date, comme c'est le cas des articles d'éphéméride du Los Angeles Times, qui commémorent des

faits hétéroclites qui se sont produits à la même date, mais à des années différentes.

2.4.5 Informations manquantes

Malgré toutes les ressources dont nous pouvons disposer, il est inévitable que certaines informations soient manquantes. Dans ce cas il peut être nécessaire d'avoir recours à des ressources complémentaires qu'il s'agisse d'articles de journaux ou d'articles d'encyclopédies qui seront examinés par un système de Recherche d'Information pour extraire l'information partielle, ou encore de ressources structurées.

Il faut toutefois estimer qu'il est possible malgré cela qu'une information reste introuvable. C'est pourquoi un système doit se montrer tolérant à l'absence d'un terme.

2.4.6 Conclusion sur la sémantique distante.

Nous avons donc vu que la sémantique distante était un moyen de rapporter des informations provenant de parties différentes d'un discours. Ce type de liens est rarement utilisé dans les systèmes de questions-réponses qui se limitent souvent à la portée de la phrase ou d'un passage fait de quelques phrases consécutives, ce qui peut être considéré comme une approximation des phénomènes d'anaphore, de chaînage lexical et de thématisation. Nous voulons exploiter pleinement les informations de cette nature et les introduire dans notre formalisme de justification de manière à étudier dans quelle mesure elles peuvent fournir des informations avec une fiabilité suffisante.

2.5 Conclusion générale

Les ressources présentées dans ce chapitre apportent de nombreux moyens de repérer les phénomènes sémantiques dans les textes. La recherche de l'ensemble des informations données dans une question peut amener à analyser plusieurs phrases du document. Une information peut éventuellement être trouvée dans un autre document ou être totalement absente. Certaines connaissances font appel à des ressources structurées qui n'existent qu'en anglais, ou dont la couverture est actuellement limitée. Les phénomènes présentés ont une granularité variable, pouvant être le terme, la phrase, le groupe de phrases ou le passage entier. Aussi, un SQR ne peut envisager, pour traiter l'ensemble des problèmes, d'utiliser un seul type de ressource ou un seul type de granularité d'analyse. Ses choix doivent pouvoir s'adapter aux différents phénomènes et ressources disponibles.

Nous allons maintenant étudier précisément ces variations de formulation grâce à un corpus de questions-réponses que nous avons créé et dont nous expliquerons la technique de création.

Pour cela, nous définirons un modèle de la justification qui sera utilisé, d'une part pour l'annotation des documents (Ch4), d'autre part pour la réalisation d'un programme d'extraction des justifications (Ch5). Ce modèle a pour but de permettre d'intégrer les différents types de connaissances hétérogènes, traitées dans ce chapitre. Il gérera principalement : 1) La variation sémantique des termes et la paraphrase de fragments d'énoncés de taille supérieure au terme, introduisant ainsi différents niveaux de granularité. 2) L'appariement entre les liens syntagmatiques de la question et de la réponse.

Chapitre 3 : Étude de corpus

Dans le premier chapitre, nous avons décrit l'état de l'art des systèmes de questions-réponses et mis l'accent sur un fossé à combler entre deux types de systèmes : d'une part, les systèmes produisant des représentations structurées de la justification mais ayant pour inconvénient, soit d'être liées à un type d'analyse (ex : syntaxique), soit de requérir de nombreuses ressources structurées (preuves logiques), et d'autre part, des systèmes permettant d'intégrer des données hétérogènes mais peu adaptés pour fournir une représentation précise des éléments justifiant la réponse.

Dans le second chapitre, nous avons présenté les connaissances sémantiques disponibles et mis en évidence la grande diversité des connaissances sémantiques permettant de détecter dans les documents candidats à fournir la réponse des variations des termes de la question et des liens syntagmatiques entre ces termes.

Nous désirons à présent proposer un formalisme des justifications permettant d'allier à une représentation fine de la justification la gestion de ressources sémantiques et lexicales hétérogènes, et d'intégrer des techniques d'appariement question-réponse de natures diverses également. Nous allons fonder nos choix sur une étude de corpus qui nous permettra de mieux connaître la « physionomie » d'une justification, pour guider la construction du formalisme son implémentation sous forme d'un programme.

Nous décrirons premièrement la méthode semi-automatique qui nous a permis de rassembler ce corpus de questions-réponses. Nous décrirons ensuite la façon dont nous avons décidé d'annoter les justifications de ce corpus, ainsi que les choix faits et les outils créés pour faciliter cette lourde tâche d'annotation.

Nous décrirons ensuite l'annotation des structures de justification et le schéma d'annotation utilisé à cet effet. Ces annotations nous ont servi à effectuer des mesures de fréquences des phénomènes, qui nous permettent de mieux connaître la “physionomie” des justifications, qui seront détaillées dans la dernière section de ce chapitre. Ultérieurement, elles pourront servir à évaluer les résultats de nos modules d'appariement question-réponse.

3.1 Utilité du corpus

3.1.1 Étude des justifications

L'étude du corpus permet de savoir quelles variations sémantiques sont présentes dans les justifications, avec quelles fréquences elles apparaissent, de savoir si les variations syntaxico-sémantiques se retrouvent dans les justifications, et si oui, avec quel type de relations. Une meilleure connaissance de ces phénomènes nous a permis de diriger nos choix lors de la création de l'algorithme décrit au chapitre 4.

3.1.2 Validation du système

Un grand intérêt du corpus est de permettre de valider les résultats produits par le système et notamment de pondérer les modules d'appariement sémantique. Nous discutons ci-dessous deux méthodes de pondération.

Fréquence des occurrences pertinentes d'un phénomène

La première méthode consiste à comparer le nombre d'occurrences d'un phénomène parmi les documents pertinents et non pertinents. On calcule une mesure de précision du phénomène étudié :

$$précision = \frac{Nb\ occurrences\ dans\ un\ contexte\ pertinent}{Nb\ occurrences\ dans\ les\ contextes\ pertinent\ et\ non\text{-}pertinent}$$

Cette mesure a été appliquée dans des expériences précédentes, sur les passages entiers, les différences obtenues étaient faibles en raison du bruit. On peut envisager de réduire la fenêtre à un ensemble de 3 phrases ou encore à une fenêtre syntaxico-sémantique plus réduite reproduisant le comportement du programme d'appariement et tenant compte des anaphores et des relations syntaxiques autorisées par l'analyse de la question.

Ce type de mesure pose un problème car il peut y avoir très peu de différences entre un document pertinent et non pertinent. Il peut y avoir un seul élément de justification non pertinent. Dans ce cas il faut repérer manuellement l'élément de justification fautif si l'on veut mesurer la fiabilité des éléments.

Estimation de la fiabilité des phénomènes

Un module est capable en général de mesurer ou de combiner plusieurs types de phénomènes. C'est le cas par exemple pour la détection des variations, comme elle est faite par Fastr, ou pour une mesure de proximité entre deux fragments d'énoncés fondés sur un chaînage lexical. Dans ces deux cas, la fiabilité de l'appariement dépend des types de phénomènes qui le constituent. Pour donner un exemple, deux termes reconnus par Fastr, l'un sans variation, l'autre avec une variation sémantique, n'ont pas la même probabilité de porter le sens qu'on attend d'eux. En général, les outils sont capables d'attribuer à leurs résultats une pondération rendant compte de la fiabilité de ceux-ci, mais ils ne le font pas souvent sous la forme d'une probabilité. Par exemple, Fastr donne une note aux termes reconnus en fonction du nombre de mots dans le terme, de la significativité des mots du terme et de la fiabilité des variations de termes estimée par des mesures sur un corpus de résultats.

Nous avons besoin d'estimer une fonction de probabilité tenant compte le mieux possible de la

diversité des résultats fournis par le module.

Apprendre une fonction de conversion

Dans le cas où il y a assez d'exemples pour chaque type de variation sémantique, il est possible d'estimer la fiabilité de chacune séparément. Au contraire, si le nombre d'exemples est trop faible, une solution alternative est d'apprendre une fonction de conversion entre l'estimation de fiabilité effectuée par le module et la probabilité que nous mesurons grâce au corpus.

Nous disposons en effet pour chaque module d'une liste d'éléments de justification rapportés. Il peut s'agir par exemple d'une liste de termes avec indication de leur identifiant de passage et de leur position dans le passage.

Pour chacun de ces éléments, nous avons la connaissance d'un couple de valeurs composé de l'estimation faite par le système et de la note de pertinence apportée par le corpus.

Il est possible par exemple d'effectuer une régression linéaire entre ces deux valeurs qui donnera une fonction simple de conversion. Nous avons utilisé dans ce but le programme WeKa.

Utiliser le corpus comme collection de référence

Une seconde méthode consiste à utiliser le corpus comme collection d'entrée du système de questions-réponses. On récupère ainsi un jeu de justifications candidates. Pour chaque candidate, on peut noter quelles heuristiques ont été utilisées pour ramener chaque élément justificatif. En comptant le pourcentage d'utilisation de chaque heuristique dans les passages pertinents et non pertinents, on peut avoir une idée de sa fiabilité.

3.1.3 Enrichissement de l'annotation

La description que nous avons choisi d'annoter est relativement riche et par conséquent, l'annotation requiert du temps. Pour faciliter la tâche, nous avons décidé de n'annoter qu'une justification par passage et pour cette justification de choisir si possible un seul élément justificatif par critère de la question. Il est demandé à l'annotateur de choisir l'élément qui lui semble s'intégrer avec le plus de simplicité ou de certitude dans l'annotation.

Ce choix nous permet de limiter le nombre d'opérations d'annotation pour un passage. De plus elle permet à l'annotateur de se concentrer sur les zones du passage qui lui semblent les plus prometteuses. Il est naturel par exemple de commencer à annoter la phrase qui contient le plus d'éléments puis de rechercher les éléments manquants dans les phrases avoisinantes. Cette méthode permet de laisser l'exploration des zones éloignées du passage aux cas difficiles.

L'inconvénient est que deux annotateurs peuvent effectuer dans une certaine mesure des choix différents d'annotation, selon leur perception de la fiabilité de chaque rattachement. La solution serait d'annoter la totalité des phénomènes du passage, ce qui là encore demande du temps.

La difficulté d'obtenir l'exhaustivité pose un problème pour l'évaluation des modules, pour lesquels nous devons mesurer la fréquence des éléments pertinents rapportés.

3.2 Sélection des documents

3.2.1 Description du corpus brut

Le corpus contient 115 questions provenant de la campagne d'évaluation TREC11. À chaque question sont associés des passages d'articles de journaux extraits de la collection AQUAINT qui était utilisée pour la campagne TREC11. Nos passages consistent en des fragments de documents d'une taille moyenne de 5 paragraphes et lignes de titres, soit environ 11 phrases et 224 mots.

Ce choix ne provient pas de nous, mais reprend celui fait pour la campagne d'évaluation, pour laquelle ces documents étaient ensuite analysés par la chaîne de questions-réponses QALC. On peut toutefois noter que cette taille paraît tout à fait appropriée à notre tâche d'annotation car elle présente le double avantage de permettre une étude de la configuration spatiale des justifications sur une plage de texte raisonnablement étendue, tout en évitant d'inclure dans le corpus des articles entiers qui alourdiraient considérablement sa taille et rendraient plus difficile la tâche d'annotation.

Nous avons réuni dans le corpus un ensemble de documents pertinents et non-pertinents. Ces derniers devraient permettre d'effectuer les mesures de fiabilité des modules. Cependant nous avons préféré effectuer toutes nos mesures sur les documents pertinents, étant donné que ceux-ci contiennent à la fois des reformulations pertinentes et non pertinentes.

Ce lot de passages non pertinents a été rapporté en utilisant la même requête que pour les documents pertinents, afin de permettre des comparaisons non biaisées entre les éléments présents entre les passages pertinents et non pertinents.

Le corpus compte en moyenne 4,6 passages pertinents par question et jusqu'à 1000 passages non pertinents. Une validation a permis d'annoter pour chaque document s'il contient la réponse ou non. Cette validation s'est effectuée de façon partiellement automatique, du fait que les 1000 documents non pertinents ont été reconnus automatiquement par vérification de l'absence d'un patron de la réponse. Les documents contenant le patron réponse ont ensuite été filtrés manuellement pour déterminer s'ils étaient réellement pertinents.

Nous avons initialement examiné 190 questions. 40% ont été supprimées parce qu'elles avaient trop peu de termes, par exemple « What is X? », ou parce qu'elles étaient trop semblables à d'autres questions déjà sélectionnées. Par exemple, nous avons trouvé beaucoup de questions de la forme « When was X born? » et avons supprimé la plupart. Cette étape nous a laissé une sélection de 115 questions.

3.2.2 Extraction par des systèmes de questions-réponses.

Une solution utilisée pour obtenir un des corpus de questions-réponses consiste à rassembler les résultats de plusieurs systèmes de questions-réponses ayant concouru sur les mêmes questions. L'inconvénient de la méthode est que ces systèmes introduisent souvent les mêmes biais dans les justifications, notamment : la forte préférence des termes exacts ou de leurs synonymes, et la préférence de termes fortement rapprochés, voire situés dans la même phrase. La nécessité de fournir la bonne réponse parmi les premiers documents conduit donc à négliger certains phénomènes au profit des techniques les mieux connues et les plus implémentées. Un corpus obtenu de cette façon est donc biaisé en fonction de ces phénomènes fréquents. Il nous a donc semblé préférable de concevoir une méthode de création de corpus propre à conserver la variété des phénomènes linguistiques. Nous expliquons cette méthode dans la section suivante.

3.2.3 Construction du corpus brut

Afin de faciliter et d'automatiser partiellement la tâche de collecte de corpus sans tomber dans le biais précédemment évoqué, nous avons créé un outil de construction de corpus mettant l'accent sur le rappel et non la précision, ceci étant au prix d'une plus longue étape de validation manuelle.

Nous avons toutefois pu faciliter la reconnaissance des documents pertinents en utilisant dans notre filtrage le patron des réponses collectés par les organisateurs de la campagne TREC11 à partir des résultats des systèmes participants.

Les étapes du programme sont les suivantes. Elles seront suivies d'un A pour automatique et d'un M pour manuel. “A puis M” signifie que l'opération est commencée automatiquement et que l'utilisateur peut ensuite modifier le résultat :

- étiquetage morphosyntaxique des questions (A, une correction manuelle est toutefois souhaitable)
- désambiguïsation des mots dans WordNet (si l'expansion des requêtes est souhaitée) (**M**)
- classement des mots de la question par ordre décroissant de significativité (A puis **M**)
- production de plusieurs requêtes par question (A, une retouche manuelle peut être faite)
- expansion éventuelle de la question (A)
- sélection des documents grâce à ces requêtes (A)
- séparation des documents non-pertinents et des documents candidats (patron de la réponse) (A)
- validation manuelle des candidats (**M**)

Création des requêtes

La création de la requête est fondamentale dans notre outil. Nous avons cherché à créer des requêtes peu contraignantes pour tenter de ramener de nombreuses variations sémantiques. Afin de rendre possible la présence de reformulations de chaque mot de la question, nous avons décidé que pour chaque mot plein de la question excepté les noms propres, il doit exister au moins une requête dans laquelle il n'apparaît pas.

Afin de générer ces requêtes "à trou” en utilisant les mots les plus significatifs, nous extrayons les mots pleins de la question et les ordonnons selon l'heuristique suivante :

noms propres > *noms communs* > *verbes* > *adjectifs* > *adverbes*

L'utilisateur peut alors modifier l'ordre des mots s'il le souhaite. De plus, s'il souhaite utiliser l'expansion des termes, il devra désambiguïser ceux-ci par rapport aux entrées de la ressource qu'il souhaite utiliser pour l'expansion. Pour l'instant, cela est implémenté seulement pour l'anglais en utilisant WordNet et ses relations entre synsets pour l'expansion.

Donnons un exemple de production de requêtes. Pour la question suivante :

```
Q : What is the name of the volcano that destroyed the ancient city
of Pompeii ?
Mots (automatique): Pompeii > volcano > city > destroyed > ancient
Mots (correction manuelle) : Pompeii > destroyed > volcano >
ancient > city
Requête(Mot omis) : mots de la requête
Req1 (destroyed) : Pompeii volcano ancient
```

```
Req2 (volcano) : Pompeii destroy ancient
Req3 (ancient) : Pompeii destroy volcano
```

Dans cet exemple, le mot *name* a été éliminé automatiquement par le programme d'extraction des mots clés. Nous avons modifié manuellement l'ordre de pertinence des mots clés choisi par le programme, considérant que les termes *volcano* et *city* ne sont pas nécessaires dans une réponse à cette question.

Le nombre de mots à conserver pour chaque requête est choisi automatiquement grâce à une heuristique simple qui consiste à sélectionner les mots dans l'ordre défini précédemment tant que le score de significativité de la requête ne dépasse pas un certain seuil. Le score de la requête est obtenu par somme des scores de ses mots, qui sont définis de la façon suivante :

- 1 pour un nom propre
- 0,75 pour un nom commun
- 0,5 pour toute autre catégorie

La sélection s'arrête avant que le score de la requête ne dépasse un seuil de 2.

On notera qu'il y a presque tout le temps plus d'un mot omis par requête, ce qui fait qu'on ne doit pas générer autant de requêtes qu'il y a de termes à omettre.

Cet algorithme devrait être modifié notamment au niveau de l'ordonnancement des termes, pour lui apporter l'ordonnancement des mots de même catégorie syntaxique en utilisant une pondération de type tf-idf.

Validation des documents

Les requêtes sont ensuite envoyées à un moteur de recherche qui ramène les documents. Pour l'anglais, c'est MG qui a été utilisé avec une requête vectorielle.

3.2.4 Description de l'outil de construction de corpus

Les petits traitements jusqu'à la création des requêtes sont effectués par des scripts, les modifications sont effectuées par l'utilisateur dans les fichiers de résultat. La validation des documents, en revanche, est faite dans une interface graphique codée en perl/cgi-bin (Fig. 5).

Cette interface permet à l'utilisateur de distinguer si le passage justifie la question entièrement ou partiellement, ou si la question n'est pas du tout justifiée. Dans une interface d'annotation, il est conseillé de toujours laisser une certaine liberté à l'annotateur, pour qu'il puisse traiter des cas imprévus. Ou apporter des remarques constructives. Nous avons donc ajouté un champ textuel libre pour l'ajout de commentaires.

L'interface présente à l'utilisateur les informations nécessaires pour vérifier la validité du passage : la question et les patrons de réponse acceptés. Les réponses sont mises en évidence dans le texte par la couleur rouge et des liens hypertexte permettent de se rendre directement au niveau de chaque occurrence de la réponse.

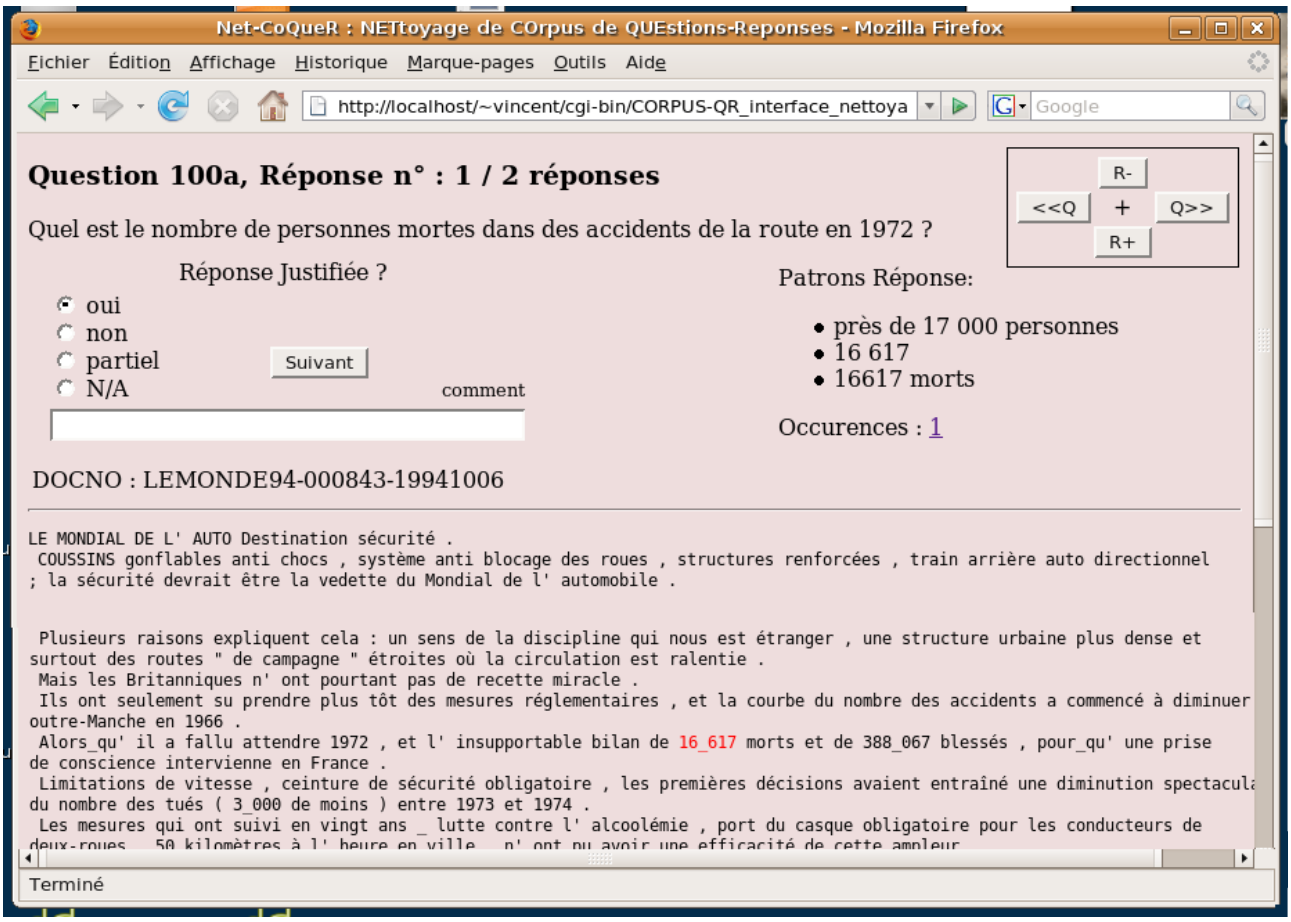


Fig. 5: Interface de validation de corpus

Cette interface a été réutilisée pour le projet CONIQUE moyennant l'ajout de quelques fonctionnalités pour la description des justifications partielles et la gestion de plusieurs justifications par passage. Elle a servi à annoter, en plus de la pertinence de la réponse, le type de traitement qui permet de compléter la justification si celle-ci est partielle, parmi les choix suivants :

- Vérification du type de la réponse
- Reconnaissance d'une variation sémantique ou d'une paraphrase
- Résolution d'une coréférence
- Rattachement d'une information donnée par le contexte
- Curiosité : les autres cas, souvent trop complexes pour un système.

3.3 Justification

L'étape suivante a consisté à annoter les éléments justificatifs dans les passages pertinents. Pour cela, nous avons conçu un modèle de la justification en tenant compte des éléments de sémantique vus dans le chapitre précédent (variations sémantiques, éléments syntagmatiques, rattachement d'informations distantes, problème de l'information manquante) et des exemples de passages pertinents tirés du corpus. Ce modèle a ensuite permis de procéder à l'annotation sur la totalité du corpus puis pour la conception d'un programme d'extraction de justifications qui sera présenté au

chapitre 4.

Le modèle que nous présentons dans cette section est le plus complet possible. Nous simplifierons ensuite ce modèle idéal selon l'utilisation souhaitée, d'une part pour l'annotation des structures de justification du corpus, d'autre part pour la réalisation du programme.

3.3.1 Représentation de la question

Nous proposons premièrement un modèle sémantique de la question incluant les différents types de sémantique vus précédemment (à savoir : type de la question, relations de sémantique locale, relations sémantiques syntagmatiques). Ce modèle de la question vise à modéliser la totalité des informations sémantiques que celle-ci contient. Il est cependant possible d'implémenter seulement une partie du modèle en approximant les parties non traitées. C'est le cas par exemple des relations syntagmatiques qui sont actuellement traitées de façon approchée, par des mesures de proximité entre mots et entre phrases.

3.3.1.1 Les termes de la question et le type attendu

Il est important pour la clarté de notre exposé de préciser la différence que nous faisons entre un mot et un terme. Un mot est pour nous l'élément obtenu par la segmentation utilisée dans le TreeTagger. Il s'agit en général de successions ininterrompues de lettres, parfois entrecoupées d'une apostrophe ou de points : « chat », « aujourd'hui », « U.S.A. » La notion de mot composé n'est pas utile pour nous, car les successions de mots sont gérées au niveau du choix des termes. Nous définissons les termes comme des mots(termes simples) ou successions de mots(termes composés) de la question qu'il est nécessaire de rechercher pour répondre à cette question. Cette définition très restrictive et fondée sur l'usage pratique que nous avons des termes, a le mérite de nous affranchir de considérations linguistiques et lexicologiques.

Nous avons réutilisé l'analyse de la question de la chaîne QALC. Celle-ci permet de reconnaître le type attendu, qui peut être soit un type d'entités nommées, soit un type général, comme « volcano »dans la phrase suivante :

```
Q : What is the name of the volcano that destroyed the ancient
city of Pompeii?
```

Nous avons ajouté à cette analyse la reconnaissance des chaînes placées entre guillemets, comme dans les questions suivantes :

```
Q1472 : How do you say "house" in Spanish?
Q1890 : Who is the author of the poem "The Midnight Ride of Paul
Revere?"
```

Ces chaînes, de taille variable, doivent être le plus souvent retrouvées telles quelles dans les documents. Elles peuvent cependant subir des altérations. Pour ces raisons nous les nommons chaînes littérales. L'analyse de la question permet de reconnaître également les termes de type nom propre en se fiant à l'analyse morphosyntaxique du programme TreeTagger. Les successions ininterrompues de noms propres sont reconnues comme des termes. Cette reconnaissance succincte des entités nommées pourrait être remplacée ultérieurement par un algorithme plus élaboré.

Enfin les mots de catégorie nom commun, verbe, adjectif, adjectif numérique(catégorie séparée pour le TreeTagger), adverbe, qui ne sont pas inclus dans une chaîne littérale sont destinés à devenir des termes de la question. Ils le seront sauf s'ils sont reconnus comme mots-outils.

3.3.1.2 Mots-outils

Les mots reconnus comme tels sont exclus de la représentation de la question. Un mot appartenant à une des quatre catégories grammaticales(nom, verbe, adjectif, adverbe) est considéré comme un mot-outil s'il appartient à une liste de mots vides connus, fournie par la chaîne QALC. Cette liste contient des mots utilisés pour indiquer un type d'entité nommée attendu, tels que « person », « country », « location », « year », « high »... D'autres mots comme « name », « appellation », « call » utilisés pour formuler des questions commençant par : « What is the name of... », « How do you call a... »

Nous avons amélioré la détection des mots vides de la chaîne QALC en restreignant la détection de ceux-ci à la partie de la question portant la définition du type attendu de la réponse.

Dans un but de simplicité, nous considérons que cette zone de définition du type de la question s'étend du pronom ou de l'adverbe interrogatif au premier nom commun de la question. Par exemple dans la question :

```
Q1396: What is the name of the volcano that destroyed the
ancient city of Pompeii?
```

La zone où le filtrage est effectué va de « What » à « name ». Le mot « name » est alors exclu des termes de la question. En revanche, dans la question :

```
Q1399 : What mythical Scottish town appears for one day every 100
years?
```

Le mot « year » n'est pas considéré comme un mot vide car il est situé en dehors de la zone de définition du type de la question (partie soulignée).

3.3.1.3 Représentation de la question

La représentation de la question est l'union de critères paradigmatiques et relations syntagmatiques. La partie paradigmatique de l'analyse de la question pourrait se résumer donc à :

- Un type attendu (éventuellement absent)
- Un ensemble de termes parmi lesquelles des chaînes littérales, des termes noms propres et des mots communs.

Pour la question à propos de « The Midnight Ride of Paul Revere », ces critères pourraient être représentés de la façon suivante. On notera chaque critère par la lettre 'C' suivie d'un numéro. Chaque critère représente un élément à trouver dans un passage réponse.

```
carac(C1, author), sem(C2,poem), litteral(C3, « The Midnight
Ride of Paul Revere »)
```

où carac représente une relation d'hyponymie entre le mot trouvé et « author », sem(C2,poem) indique le mot trouvé pour le critère C2 doit être une reformulation quelconque du terme « poem » et litteral(C3,...) indique que la chaîne doit être retrouvée telle quelle.

Bien que cela ne soit pas utilisé dans l'implémentation courante, ces termes peuvent être reliés entre eux par des relations syntagmatiques. Nous envisagions de relier les termes par des relations sémantiques assez proches de la syntaxe. Par exemple, pour la question sur Pompéi, la représentation aurait pu être, pour les critères paradigmatiques :

```
carac(C1,volcano), sem(C2 destroy), sem(C3, ancient),
```

```
sem(C4,city), semEN(C5,Pompeii)
```

Les critères syntagmatiques établissent des liens entre des critères paradigmatiques :

- agent(C2,C1) : indique que l'agent de destroy est C1, le volcan que l'on recherche.
- objet(C2,C5) : indique que l'objet de la destruction est Pompéi
- carac(C3,C4), carac(C4,C5) : indique que Pompéi doit avoir les caractéristiques d'être une ville et que cette ville doit avoir la propriété d'être ancienne.

On pourra compléter cette liste par le prédicat cplt_v(verbe, nœud) qui permettra de spécifier d'autres arguments du verbe dont la valeur sémantique est plus difficile à déterminer.

D'autres liens pourraient être introduits pour compléter cette représentation, mais comme cette représentation de la composante syntagmatique n'est pas utilisée actuellement, nous nous restreindrons à cette description sommaire, qui est suffisante pour décrire notre programme.

3.3.1.4 Rattachement des compléments temporels

Les compléments temporels ont la particularité de pouvoir se trouver loin de l'événement qu'ils concernent. En effet, ils sont souvent rejetés en début ou en fin de phrase. Pour cette raison, il semble préférable de les traiter séparément pour les rattacher à leur gouverneur sémantique. Ce gouverneur est le mot référent à l'action que le complément de temps vient spécifier.

Nous avons développé un module d'analyse des dépendances pour le français, dans le cadre du projet CONIQUE[11]. Un tel programme pourrait être adapté à l'anglais pour permettre d'ajouter à la représentation de la question les relations sémantiques entre événement et complément temporel.

L'implémentation est la suivante : Pour obtenir ce gouverneur, nous recherchons dans un premier temps le verbe, si celui-ci n'est pas un mot vide, celui-ci est le gouverneur recherché. Si celui-ci est un mot vide, nous effectuons une recherche selon le type de verbe. Les verbes vides sont découpés en plusieurs classes entraînant deux comportements différents :

Le premier comportement consiste à rechercher le premier complément c'est-à-dire, soit le complément d'objet direct, soit le complément du verbe. Voici des exemples de ces verbes : être, sembler, paraître, devenir / avoir, faire / estimer, penser, croire, prévoir.

Le second comportement consiste à rechercher le sujet du verbe. Ceci concerne des verbes dénotant le déroulement d'un événement. Dans ce cas, l'événement recherché est le sujet du verbe. Les verbes concernés sont entre autres : se_passer, avoir_lieu, se_dérouler, se_produire.

Ainsi dans la question Q3, le complément de temps « avant la puberté » qui devrait être rattaché à « faire », est rattaché à « tentative », dans Q26, le complément « de 1989 à 1993 »est rattaché au nom « ministre » :

```
Q3 : Quelle est la probabilité qu'un enfant fasse une tentative
de suicide avant la puberté ?
Q26 : Quel politicien libéral était ministre de la santé en
Italie de 1989 à 1993 ?
```

3.3.1.5 Gestion de différents niveaux de granularité

Notre système se caractérise par une grande permissivité au niveau de l'analyse des termes.

11 Projet ANR sur le traitement de l'inférence en questions-réponses

Nous avons en effet décidé d'accepter de faire cohabiter différentes analyses des termes, et ce, afin de permettre l'intégration éventuelle de plusieurs modules indépendants.

Dans les passages candidats, les termes composés peuvent se trouver exprimés de différentes façons, c'est-à-dire, soit en un seul bloc reprenant tout le terme, soit en plusieurs éléments séparés. Par exemple, « Général Charles de Gaulle" pourra se trouver décomposé en « Général" et « Charles de Gaulle" qui peut à son tour être décomposé en « Charles" et « de Gaulle". Les noms communs sont aussi concernés par cette représentation arborescente, dans le cas de termes composés comme « centrale nucléaire ».

Fig. 6: Termes arborescents

Une telle représentation arborescente a été utilisée pour l'annotation manuelle du corpus, ainsi que par le module Fastr.

Nous avons voulu traiter également du phénomène de paraphrase. Nous voulions représenter principalement la possibilité de repérer des paraphrases de fragments de la question dans les documents, voire de paraphrases de la question entière. Ces phénomènes se situent à un niveau de granularité plus étendue que celui du terme. Les paraphrases présentent un intérêt double :

D'une part, certains fragments de phrase sont présents dans un texte sous cette forme. C'est le cas dans le passage suivant :

```
Q : What year did South_Dakota become a state?
R : No doubt folks in South_Dakota are pretty excited each Nov. 2
about celebrating the anniversary of their state joining the
union in[REP1889] .
```

Ici, « become a state » doit être pris dans le sens de devenir un état fédéral des États-Unis, c'est en effet l'information qui semble intéresser le public américain, comme en atteste le fait que ce fait soit cité dans de nombreux articles. Par conséquent, « joining the union » est une paraphrase de « become a state ». Cette paraphrase est du ressort des systèmes d'inférences, elle ne sera pas traitée par notre programme mais se trouve annotée dans le corpus.

Le second avantage de la paraphrase est d'ordre pratique : Elle permet de traiter d'une façon plus spécifique des fragments de texte de taille supérieure à celle des termes. Elle devrait apporter une plus grande fiabilité qu'un appariement mot à mot.

Choix d'une représentation minimale commune

Pour pouvoir traiter tous les cas qui se présentent, nous avons dû choisir une base commune la plus générale possible. Dans les cas les plus fréquents, les termes et les fragments de texte de la question forment une structure d'arbre. Cependant il n'est pas exclu que certains termes se chevauchent.

Dans le cas d'une annotation manuelle, il est par exemple envisageable, bien que le cas ne se soit pas présenté, que deux paraphrases distinctes aient un terme en commun, et les autres termes

différents. Dans le cas de la reconnaissance par un programme, le chevauchement est encore plus aisé. En effet, le fait d'utiliser plusieurs analyses de la question, possédant des légères différences, ou tout simplement introduisant des divergences par erreur d'analyse, peut amener un chevauchement. Il serait par exemple possible qu'un terme reconnu par Fastr chevauche un mot composé issu de WordNet. Par exemple dans la question suivante,

```
Q: When did World War II begin ?
```

« World War II » est un terme de Fastr et de WordNet, pour raison d'une erreur dans l'extraction des termes pour Fastr le lemme inférieur est « War II », par contre WordNet permet de trouver le terme « World War »

```
R2: ... World War II ...
R1: ... the second World War ...
```

Nous avons donc choisi une base commune qui permette de gérer ce type de phénomène. Notre choix consiste à représenter tout terme repéré comme l'union de ses sous-termes minimaux, ceux-ci correspondant à l'ensemble des mots de catégorie nom commun, nom propre, verbe, adjectif (dont numérique) ou adverbe appartenant à un terme de la question. Ceux-ci correspondent aux feuilles représentées dans l'arborescence de la figure 6.

Du fait de cette absence de contrainte au niveau de la représentation commune, la liste des termes et fragments de la question pris en compte pour l'appariement dépend uniquement des choix de l'annotateur dans le cas de l'annotation du corpus ou des analyses de la question effectuées par les différents modules dans le cas du programme d'extraction de justifications.

Nous recommandons toutefois une certaine autodiscipline dans le cadre de cette liberté presque absolue. Lors de l'annotation du corpus et de la conception du programme, nous avons respecté la règle suivante.

Notre ligne de conduite générale consiste à utiliser si possible la même liste de termes pour tous les modules. Dans notre cas, c'est l'analyse arborescente de Fastr qui a été utilisée pour les différents modules ainsi que lors de l'annotation du corpus. Il était donc recommandé de respecter tant que possible l'arborescence des termes. Toutefois, l'annotateur était autorisé à créer de nouveaux termes en cas de besoin, notamment pour annoter des paraphrases, tout en essayant si possible de ne pas rompre les termes composés déterminés par Fastr. De même, tout module est libre de redéfinir les fragments de la question qu'il apparie, indépendamment des termes définis par l'analyse de la question conseillée.

3.3.2 Modèle d'une justification

La justification d'une réponse est représentée sous la forme d'un graphe mettant en relation les éléments de la question avec ceux du passage réponse. L'analyse de la question permet de mettre en évidence un certain nombre de critères dont principalement : les termes de la question, le type attendu de la réponse, les relations syntagmatiques entre ces éléments.

Les éléments de justification sont extraits d'un passage réponse de plusieurs phrases (rappelons que dans nos expérimentations il s'agit de fragments d'articles constitués d'environ 5 paragraphes). La représentation de cet extrait justificatif dans le formalisme est constituée d'un ensemble de termes reliés entre eux par des relations syntagmatiques diverses. Les termes d'une même phrase sont reliés entre eux par des relations syntagmatiques internes à la phrase et les termes de différentes phrases sont reliés entre eux par des relations de sémantique énonciative, par exemple

des anaphores ou des coréférences.

3.3.2.1 Appariement entre question et réponse

À chaque critère de la question, paradigmatique ou syntagmatique, on cherche à faire correspondre un unique élément du passage réponse.

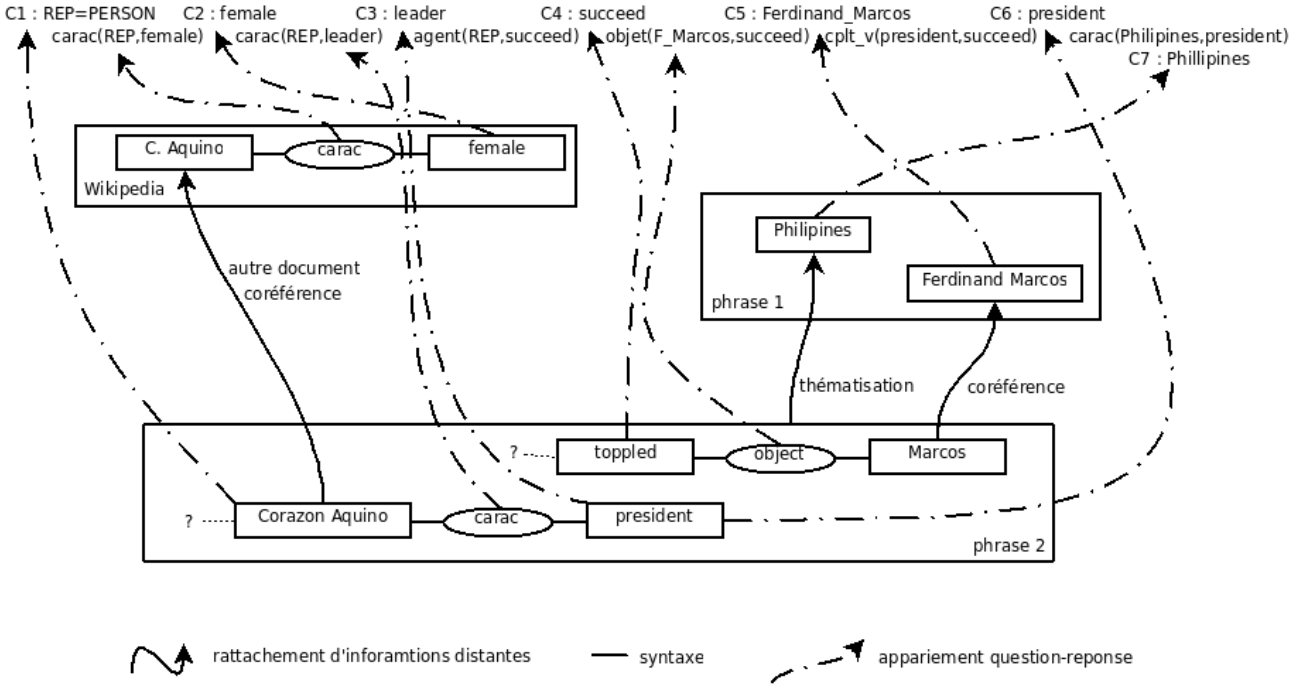


Fig. 7: Représentation de la justification

Ainsi, pour notre couple Q-R fétiche, les analyses de la question et de la réponse sont :

```
Q1444 : What female leader succeeded Ferdinand Marcos as
president of the Philippines?

Critères syntagmatiques :
C1 : Entité nommée de type PERSON (la réponse cherchée)
C2 : female
C3 : leader
C4 : succeed
C5 : Ferdinand_Marcos
C6 : president
C7 : Philippines

Critères syntagmatiques :
S1 : carac(C2,C1)
S2 : carac(C3,C1)
S3 : agent(C1,C4)
S4 : objet(C5,C4)
S5 : cplt_v(C6,C4)
S6 : carac(C7,C6).

Passage Réponse :
```

(Phrase1) : MANILA , **Philippines(C7)** (AP) -- Raul Manglapus , a former foreign secretary who organized Filipinos in the United States to fight the rule of **Ferdinand_Marcos(C5)** , died Sunday . He was 80 .
[...]
(Phrase2) : After a 1986 revolt **toppled(C4) Marcos** , Manglapus returned home and was appointed foreign secretary by **President(C7&C3)[REP Corazon Aquino](C1).**

Complément : (Wikipédia, article Corazon Aquino)
Title : **Corazon Aquino**
Text : María Corazón Cojuangco-Aquino (born María Corazón Sumulong Cojuangco on January 25, 1933), widely known as Cory Aquino, was the 11th President of the Philippines, serving from 1986 to 1992. She was the first **female(C2)** president of the Philippines.

L'appariement effectué devrait être celui présenté dans le graphique suivant. Pour conserver la lisibilité du graphique, je n'ai pas reporté les liens. Les deux graphes présentés sont superposables, les nœuds portant le même numéro de critère paradigmatique sont appariés, de même que les liens reliant ces termes, deux liens appariés étant présentés au même endroit dans chacun des deux graphes graphe.

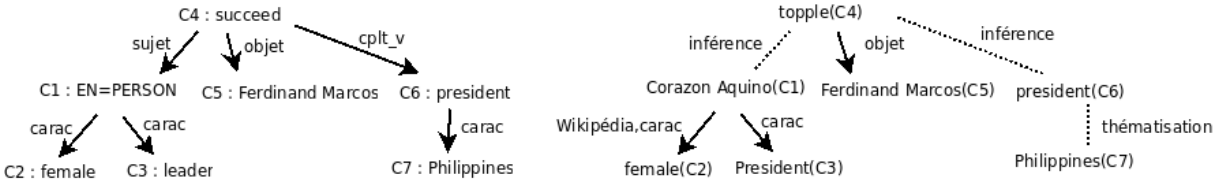


Fig. 8: Appariement entre question et réponse

Dans cet exemple certains des critères syntagmatiques de la question ne peuvent pas être appariés. C'est le cas de la relation de sujet entre « Aquino » et « succeed ». Dans ce cas, il faut faire une inférence complexe au niveau du sens de la phrase pour comprendre que Aquino succède à Marcos. La relation de complément entre « succeed » et « président » ne peut être trouvée. En effet, dans le document, président est rattaché à « Aquino ». Il est difficile d'obtenir un appariement correct des compléments de type groupe prépositionnel dans cet exemple, la formulation de l'information « succeed as president » peut se retrouver sous la forme « Aquino became president » et « Marcos was ousted from presidency ».

Dans notre thèse, ce problème n'est pas abordé car nous n'avons pas tenu compte des relations syntagmatiques mot à mot, nous contentant de mesures portant sur la proximité globale des mots de la justification.

3.3.2.2 Réponses étendues sur plusieurs phrases ou plusieurs documents.

Dans notre représentation, chaque critère paradigmatique de la réponse est relié à un autre critère par un lien syntagmatique, ce qui cache des phénomènes différents et parfois complexes. Dans notre exemple, le lien entre les termes « topple » et « Ferdinand_Marcos » s'effectue en fait par la composition de deux liens :

- Un lien de sémantique syntagmatique, entre « topple » et son objet « Marcos » tiré du fragment de phrase « After a revolt **toppled Marcos** , ... »
- Un lien de coréférence qui relie Marcos à la forme complète de l'entité nommée « Ferdinand Marcos » qui se situe dans une autre phrase du passage.

Le lien entre Corazon Aquino et female cache également plusieurs liens :

- La mise en relation du terme « Corazon Aquino » dans le passage avec une occurrence de « Corazon Aquino »dans l'article de Wikipédia.
- La liaison par une chaîne d'anaphores et de coréférences à l'intérieur de l'article de Wikipédia entre Corazon Aquino et le pronom « she » dans « **She** was the first **female** president »
- La liaison dans cette phrase de « she » et female par une relation de type carac.

Dans l'annotation du corpus, les sous-liens seront détaillés, alors que dans la construction des justifications, les étapes de mise en relation seront masquées afin de permettre au programme de manipuler une représentation en graphe simplifiée.

3.3.2.3 Niveau de granularité et recouvrement

La justification doit couvrir tous les termes de la question, sauf absences. Pour cela on essayera de sélectionner un ensemble de termes du passage réponse couvrant tous les termes de la question sans redondance. Étant donné que nous avons choisi comme base commune de représentation le sous-terme minimal, la couverture de la question se définit comme la couverture de tous les termes minimaux. De plus, l'annotateur veillera à éviter toute redondance dans son annotation, c'est-à-dire qu'aucun des termes sélectionnés dans la justification ne pourrait être enlevé sans perdre la couverture d'un des sous-termes minimaux.

Il est donc nécessaire, pour l'annotateur et pour le programme de gérer la redondance, notamment dans le cas de deux termes emboîtés, pour lequel nous avons choisi de toujours utiliser le terme le plus englobant possible.

3.3.2.4 Éléments manquants

Il est possible que certains éléments de la question ne trouvent pas de correspondant dans le passage. Par exemple, dans la question,

```
Q1444 : What female leader succeeded Ferdinand Marcos as
president of the Philippines?
```

les modifieurs « female » et « leader », qui sont des attributs de la réponse courte (« Corazon Aquino ») doivent être vérifiés. Or ces attributs ne sont pas souvent présents dans les documents. Il faut alors vérifier ces informations dans d'autres ressources, comme par exemple Wikipédia. Prenons l'exemple d'un passage candidat, qui répond approximativement à la réponse :

```
R : MANILA , Philippines ( AP ) -- Raul Manglapus , a former
foreign secretary who organized Filipinos in the United States to
fight the rule of Ferdinand Marcos , died Sunday . He was 80 .
[...]
After a 1986 revolt toppled Marcos , Manglapus returned home and
was appointed foreign secretary by President[REP Corazon Aquino].
```

Dans ce passage, l'information leader est présente puisqu'on apprend que Corazon Aquino est un

président. Or, les termes « president” et « leader” peuvent être mis en relation grâce à WordNet, dans lequel ces deux lexèmes sont liés par une relation d'hyperonymie (a president ISA leader).

Il nous reste une information manquante à vérifier : c'est que Corazon Aquino est une femme, notée carac(Aquino,« female ») dans la représentation de la question. On peut retrouver l'information manquante, ainsi que l'information que Corazon Aquino est un chef/dirigeant(« leader ») dans l'article de Wikipédia sur Corazon Aquino, qui contient la phrase suivante : « She was the first female President of Philippines ». « female » est directement présent comme attribut de Corazon Aquino et « leader » se retrouve de la même manière que précédemment. La structure totale de la justification est donnée dans la figure 7 (page 69).

Dans notre modèle nous décrivons les cas suivants :

1 L'information peut être trouvée dans un autre document, comme des articles de journaux ou une ressource encyclopédique telle que Wikipédia. Par exemple : « Michael Jordan” est un basketteur, est un sportif. Dans ce cas, le modèle permet d'intégrer cette information à la justification comme s'il s'agissait d'un lien de sémantique énonciative. La figure 7 donne un exemple d'un rattachement vers Wikipédia.

2 L'information peut être obtenue par inférence. Cette notion d'inférence peut couvrir de nombreuses sources de connaissances et mécanismes. Définissons l'inférence sur un plan logique comme un procédé faisant intervenir des axiomes (connaissances de départ) et des règles de production (transformation et combinaison de connaissances entre elles pour en inférer de nouvelles). Le type d'inférence est caractérisé par l'origine des axiomes (passages et autres documents, connaissance basique sur le monde), qui seront combinées entre elles pour fournir l'information manquante. Voici un exemple d'inférence:

```
Q1449: What college did Magic Johnson attend ?
R: [Magic] Johnson, who led [REP Michigan State] to the NCAA
championship in 1979...
```

Le terme *attend* n'est pas présent ici. Une fois repéré que Michigan State est un *college*, il faut faire l'inférence suivante : Magic Johnson mène une équipe à la victoire lors d'un tournoi, or Magic Johnson est basketteur, donc, Magic Johnson a joué dans cette équipe. Or les équipes des *colleges* sont constituées d'élèves, donc Johnson était élève du Michigan State College.

Certaines inférences peuvent être vues comme des transformations lexicales. Par exemple le passage de « How did Hitler die » à « Hitler committed suicide » peut être obtenu par la chaîne lexicale tirée de WordNet 3.0 : commit_suicide-(hype)->kill-(caus)-> die.

3 Les connaissances peuvent également venir de la question elle-même. En effet, toute question comporte un ensemble de présupposés pragmatiques. Il s'agit d'un ensemble de connaissances que la personne qui pose la question considère comme implicitement admis et que le système, par conséquent, peut se dispenser de vérifier. Par exemple, dans la question « What Spanish cyclist won the Tour de France in 1994? », « Spanish » et « cyclist » sont facultatifs. « Spanish”, parce que le gagnant d'une course est unique. Or, la question introduit le présupposé que ce vainqueur est espagnol. Par conséquent, si l'on ne remet pas en cause ce présupposé, il vient que « X a gagné le Tour de France » implique « X est espagnol ». Le critère « cyclist » est vérifié de façon implicite par le fait que cette personne a gagné une course cycliste.

Bien qu'il soit difficile de détecter automatiquement le caractère facultatif d'un critère, il est intéressant de mesurer la fréquence du phénomène.

3.4 Annotation des justifications dans le corpus

Nous avons annoté les documents pertinents du corpus de questions-réponses afin de faire apparaître la structure des justifications.

3.4.1 Faciliter L'annotation

Annoter la totalité de la justification prend du temps et requiert l'annotation de nombreux éléments. Pour cette raison, il était nécessaire de ne traiter que l'essentiel. Nous avons décidé pour cette raison de n'annoter que les documents pertinents. Les passages présentés sont trop grands pour être entièrement annotés. Bien qu'un passage contienne souvent plusieurs éléments correspondant au même terme de la question, nous avons décidé de nous limiter à sélectionner un seul de ces éléments comme appartenant à la justification. Dans le cas où l'élément est utilisé dans un lien de coréférence qui permet de relier deux zones de la justification entre elles, il y a pour le critère faisant le lien deux éléments du passage utiles à la justification. Là encore, un seul sera sélectionné dans la justification. L'autre ne sera présent que sous forme de point de départ ou d'arrivée d'un lien de coréférence.

Préparation du corpus

Pour faciliter la tâche de l'annotateur, nous avons annoté automatiquement les éléments qui pouvaient l'être de façon assez fiable. De cette façon, l'utilisateur n'a plus qu'à ajouter les termes qui n'ont pas été repérés. Il doit également procéder à l'annotation des relations sur tous ces éléments. Les éléments pré-annotés ont de plus l'avantage de guider l'annotateur lorsqu'il recherche la justification, l'orientant vers les passages qui semblent les plus favorables. Les informations mises à disposition de l'annotateur sont :

1 La liste des termes de la question, incluant les termes composés et leurs sous-termes. Si le terme complet apparaît, l'annotateur choisira celui-ci pour l'annotation, sinon il devra utiliser le sous-terme

2 La présence d'une justification, information obtenue manuellement à l'étape de validation précédente.

3 Le découpage des documents en phrases (réalisé automatiquement grâce à TreeTagger)

4 Le repérage de la réponse courte dans les textes (automatique grâce au patron de la réponse)

5 Les liens repérés par Fastr : reconnaissance de termes simples et composés, que ce soient des termes exacts, des variations morphologiques, de synonymie et leurs combinaisons.

La résolution des coréférences est laissée à la charge de l'annotateur.

Notons que les éléments d'analyse présentés pour la question et la réponse ne contraignent pas l'annotation de l'utilisateur qui peut ajouter les termes qu'il désire aussi bien au niveau de l'analyse de la question que dans le passage réponse. De même, tout terme proposé pour l'analyse de la question ou de la réponse peut être ignoré. En ce qui concerne la question, l'annotateur peut annoter comme non pertinent un terme proposé et justifier son choix en précisant s'il s'agit d'une erreur ou d'un terme facultatif. En ce qui concerne la réponse, les termes n'appartiennent pas par défaut à la justification, il n'y a donc rien à faire pour ignorer un terme reconnu à tort, le schéma d'annotation prévoit toutefois de distinguer parmi les termes non utilisés dans la justification les termes sémantiquement non pertinents de ceux qui sont pertinents mais n'ont pas été retenus pour la

justification..

3.4.2 Simplification du modèle pour l'annotation

Pour des raisons de temps d'annotation, nous avons décidé de ne pas annoter les relations syntagmatiques de la justification. Dans cette approche, les critères restants sont donc le type de la question et les termes simples ou composés.

Nous tenions cependant à ce que l'annotation de justification produise un graphe connexe. Pour ce faire, nous avons décidé de remplacer les relations syntagmatiques de la phrase par l'annotation d'une zone de justification. Cette zone de justification essaie de représenter le fragment de phrase, apportant l'information utile.

Par exemple, dans le passage suivant :

```
Q : When did Alexandra Graham Bell invent the telephone ?
R : <just> In [REP 1876] , the first successful voice transmission
over Alexander Graham Bell 's telephone took place </just> in
Boston as his assistant heard Bell say , ` ` Mr . Watson , come
here . I want you. ' '
```

Le fragment est souvent de la taille d'une proposition, comme c'est le cas ici. La zone contient en effet le verbe « take place » et ses arguments : sujet (« the first successful voice transmission over Alexander Graham Bell 's telephone ») et complément de temps(in 1876).

Structure spatiale et connexité

Parfois, les éléments de justification sont répartis sur plusieurs phrases. Au début de l'annotation nous avions tendance à sélectionner plutôt de grands blocs souvent de la taille de la phrase, puis au fil du temps, nous avons affiné notre façon de découper, pour ne sélectionner que les propositions utiles. Il faut homogénéiser l'annotation de ce point de vue.

Nos justifications se représentent donc de la façon suivante : une ou plusieurs zones principales (souvent une ou deux), reliées entre elles par des phénomènes de sémantique énonciative. À celles-ci s'ajoutent des termes isolés, c'est-à-dire non inclus dans une zone de justification, mais reliés à celle-ci par une chaîne de phénomènes de sémantique énonciative.

Contrairement au modèle simplifié utilisé pour le programme de recherche des justifications, nous annotons ici avec une grande précision les chaînes de phénomènes de sémantique énonciative, en notant tous les nœuds intermédiaires permettant de relier les éléments de justification, il arrive par exemple d'avoir des chaînes de longueur 3, comme dans cet exemple tiré du corpus :

```
Q:What was the name of Stonewall_Jackson 's horse ?
R:      -- Stonewall Jackson* Memorial Cemetery : Do n't be
surprised if ..... Stonewall Jackson ......... Jackson 's grave
[...]
[...]
On the lower level of Jackson* Memorial Hall on the VMI campus ,
where he taught for a decade before the war , < just> is where
you 'll find his* preserved horse . ` `[REP Little Sorrel] is
central to the museum both figuratively and physically , ' '
says director Keith Gibson . ` ` It compresses history for folks
when they think that horse was an eyewitness to war. ' '< /just>
```

Dans cet exemple, le terme « Stonewall Jackson » est mis en relation avec le reste de la justification par la chaîne d'anaphores et de coréférences suivantes :

```
his -(anaphore)- Jackson -(coréférence)- Stonewall Jackson
```

Nous nous efforçons d'obtenir les justifications les plus connexes possibles en essayant de trouver le plus souvent une chaîne lexicale, anaphorique ou coréférentielle rattachant le mot au reste de la justification. Si un mot du passage sémantiquement pertinent pour la question ne peut être relié au corps de la justification par aucune chaîne ou que cette chaîne semble trop peu sûre, il faut rechercher d'une relation thématique, c'est-à-dire décider si ce terme est valide en tant qu'élément du contexte pour une des zones de justification. S'il semble à l'annotateur qu'aucune thématisation ne peut être trouvée, il faut en conclure que ce terme ne fait pas partie de la justification. L'élément de justification concerné sera alors catégorisé comme information manquante.

Deux zones peuvent être reliées par plus d'une relation, ceci permettant d'étudier comment s'effectue le lien entre deux passages, de mesurer si la redondance des liens est un phénomène courant ou peu fréquent. Ceci permettra d'étudier les possibilités de collaboration entre informations sémantiques pour le rapprochement des passages distants.

Annotation des phénomènes de sémantique distante.

L'étude du corpus a permis de fixer la liste des phénomènes fréquents, que nous souhaitions annoter. L'annotation se fait de façon manuelle. Aucun phénomène de sémantique distante n'est pré-annoté automatiquement. Nous avons classifié les phénomènes de sémantique énonciative de la façon suivante :

1 direct : Toute relation entre deux références désignant un même référent, en d'autres termes vers un même objet du monde. Elle se sous-catégorise en :

- 1.1 exact : Les mots sont les mêmes (lemme identique)
- 1.2 pronominal : Lien d'un pronom vers son antécédent.
- 1.3 ne-approx : La répétition avec variation d'une même entité nommée. C'est le plus souvent un lien entre le nom entier de la personne et son prénom ou nom de famille. Le cas d'entités reprises sans variation est étiqueté par la catégorie « exact » au même titre que les termes non entités nommées. Ce qui, il faut le reconnaître, complique le travail pour faire une étude indépendante sur les entités nommées. Cependant, cela peut se corriger automatiquement, en utilisant le programme de repérage des entités nommées de la chaîne QALC.
- 1.4 synonym/abbreviation : Les deux mots sont synonymes ou l'un est une abréviation de l'autre, par exemple, (« Colorado», « CO »). Nous avons décidé de fusionner ces deux catégories car nous ne voulions pas introduire une difficulté de choix supplémentaire. De plus, les abréviations sont traitées comme les synonymes dans WordNet, appartenant au même synset que le mot d'origine.
- 1.5 is-a : Chaînage lexical entre un mot et un de ses hyperonymes, hyponymes, instances.
- 1.6 wn-others : Un autre relation sémantique qui peut être calculée grâce à WordNet (par exemple, deux concepts frères dans WordNet, pouvant représenter le même référent city, megalopolis)
- 1.7 dir-other: Les cas de reformulations directes non prévus par le schéma d'annotation.

2 bridging : Les deux références correspondent à des référents différents du monde.

2.1 part-whole : Un des éléments est une partie, une matière, un élément de l'autre (méronymie)

2.2 entity-attribute : Relie un concept à une des ses caractéristiques (ex Microsoft/Bill Gates)

2.3 cause-effect : Relation de cause à effet et inversement entre deux verbes. (par exemple : “to kill” cause “to die”)

2.4 temporal succession : Les deux événements se suivent généralement l'un l'autre (“to walk” “to arrive”)

2.5 temporal inclusion : Deux événements qui se produisent généralement simultanément

2.6 entailment (implication) : a implique b si a n'est pas possible sans b : (par exemple “ronfler” et “dormir” ; ronfler n'est pas possible sans dormir, donc ronfler implique dormir.
Les quatre dernières catégories sont très proches et il est parfois difficile de choisir, par exemple entre “cause” et “temporal inclusion” ou “temporal succession”; Elles ont été regroupées dans les analyses sous la catégorie générale “scheme”. Cette dénomination évoque le fait que les deux variantes sont reliées entre elles par le fait de leur appartenance à un scénario commun.

2.7 collocation : Relation prédicat argument, par exemple un verbe et son sujet ou objet habituel. Un exemple est le lien entre « soigner » et « médecin », entre « arme à feu » et « fusillade ».

2.8 para-meta : Une catégorie pour regrouper les phénomènes pour lequel un des termes appariés est un fragment d'énoncé complexe (paraphrases, métaphores, périphrases, etc.).

2.9 transcat : Changement de catégorie grammatical complexes conservant au moins une partie de l'idée et de la valeur justificative (« new » ~ « succeed », « invent » ~ « first » dans « he **first** managed to... »). Ce cas correspond en fait à une inférence complexe.

3 derivational: Relation de morphologie dérivationnelle entre les deux mots. (par ex.: “win” “winner”)

4 other: Si aucune des étiquettes précédentes ne correspond.

Les variations de morphologie flexionnelle, c'est-à-dire le changement de genre, de nombre, la conjugaison, etc., ont été considérées comme peu porteuses d'informations sémantiques et n'ont pas été prises en considération dans notre étude.

Appariement entre la question et la réponse

L'annotateur doit sélectionner un ensemble de termes du passage réponse constituant une justification de la réponse. Certains termes sont repérés par un prétraitement automatique qui consiste à marquer dans les passages candidats les termes reconnus par Fastr.

De plus, l'utilisateur doit pour chaque terme sélectionné indiquer le type de variation sémantique qui le relie au terme de la question. La tâche est entièrement manuelle car il n'y a pas de pré-annotation automatique des types de variations. Ces variations entre passage réponse et question sont annotées avec le même jeu d'étiquettes que pour les phénomènes de sémantique distante, excepté le type pronominal qui n'a de sens que pour un lien interne à un document.

Informations manquantes

Nous avons repris la notion d'information manquante décrite dans la section 3.3.2.4. Pour l'annotation du corpus, nous avons détaillé les cas les plus fréquents, en estimant pour chaque absence de termes la façon la plus probable de la rattacher à la justification. Les étiquettes du schéma d'annotation sont :

1 Autre document : Si l'information devrait être recherchée dans un autre document.

2 Inférence grâce au document: Si le critère, bien qu'absent du document, peut être inféré par un être humain. Nous noterons toutefois que cette étiquette est souvent négligée au profit de l'annotation d'une relation sémantique comme (entité-attribut ou une relation de type scheme pour les verbes)

3 Connaissance basique sur le monde: si l'information manquante est connue de tous et n'a pas besoin d'être précisée.

4 Information facultative : Si l'information paraît inutile. Ce cas se produit principalement si la présence du critère peut être inférée à partir des autres critères de la question.

3.4.3 Outils d'annotation

Pour effectuer l'annotation, nous avons choisi l'outil d'annotation MMAX disponible sur le site de l'European Media Laboratory GmbH[12].

Nous avons aussi examiné le logiciel GATE, qui pourrait être utilisé pour cette tâche. Nous avons cependant préféré utiliser une interface plus légère et plus rapide. Cependant il faut noter que GATE propose une grande variété d'outils intégrés qui faciliteraient les traitements sur le corpus, et qu'il dispose d'une interface plus jolie. C'est donc plutôt la rapidité de l'interface qui nous a guidé vers MMAX.

Pour ce faire, nous avons redéfini le schéma d'annotation des relations et transformé notre corpus au format MMAX et converti notre corpus au format utilisé par cet outil.

12 **MMAX Annotation Tool** Homepage : http://www.eml-research.de/english/research/nlp/download/mmax.php

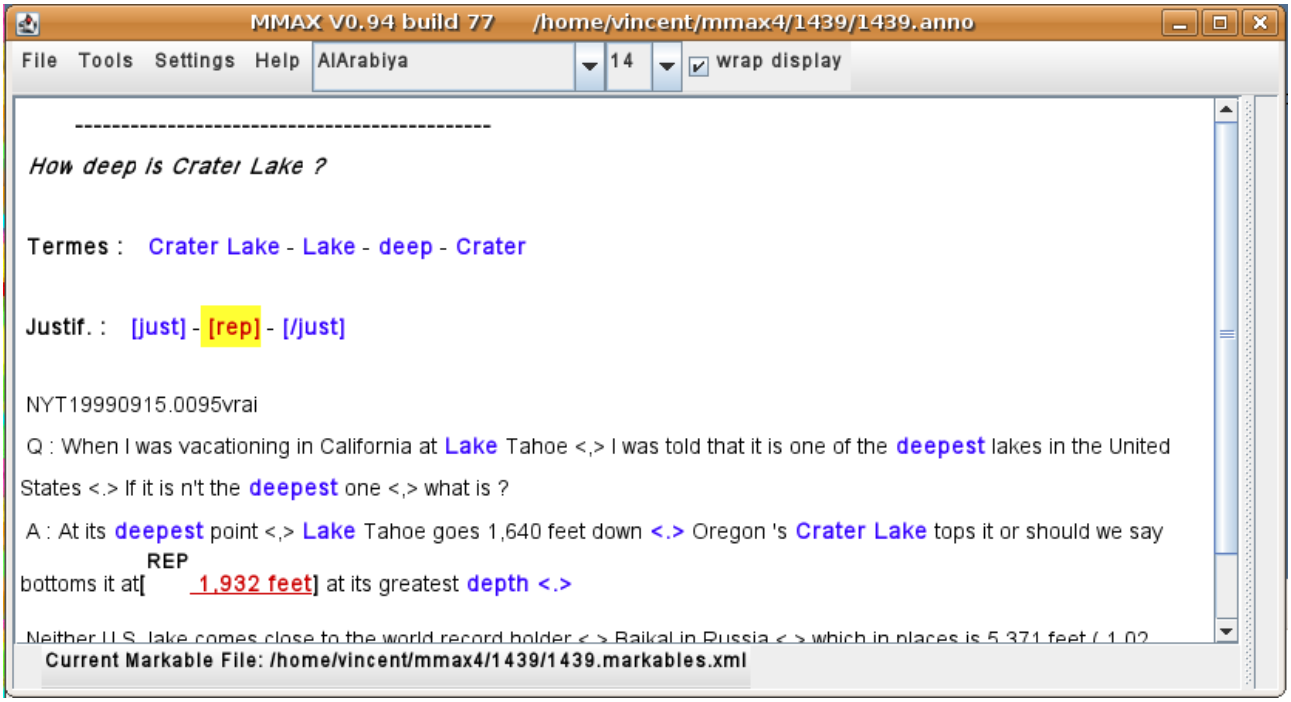


Fig. 9: Interface MMAX d'annotation de corpus

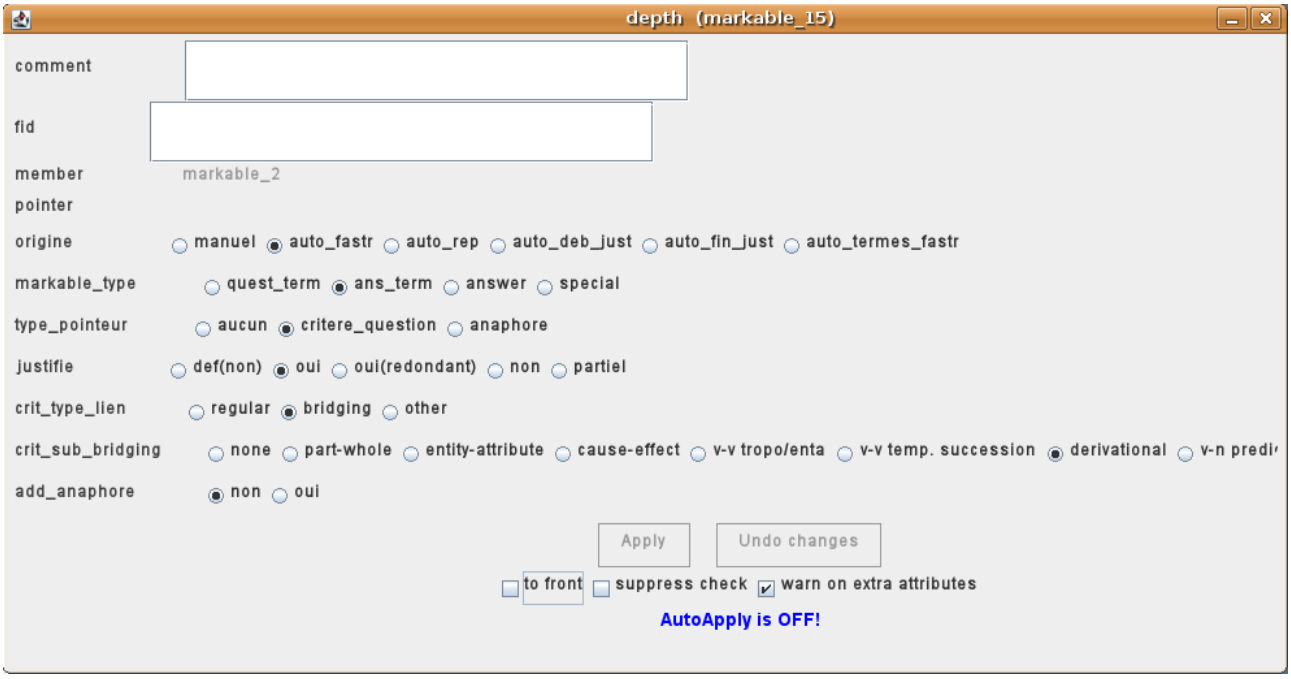


Fig. 10: Fenêtre d'annotation de MMAX

3.4.4 Conclusion

Le corpus annoté en suivant le schéma décrit dans cette section a donné lieu à des mesures permettant de donner une idée de l'aspect des justifications. Nous présentons ces mesures dans la section suivante.

3.5 Analyse du corpus

Cette section regroupe les analyses qualitatives et quantitatives sur le corpus. Son but est de donner une description de la conformation des justifications, notamment au niveau de leur étendue dans les passages réponses et du type de variations sémantiques utilisées.

3.5.1 Mesures de variation sémantique

3.5.1.1 Fréquence des types de variation sémantique

Nous avons dans un premier temps mesuré la fréquence des différents types de variation sémantique présents dans les justifications. Afin de faciliter la lecture et l'analyse des résultats, sous avons choisi de classer les types de variations de celles qui nous semblent les plus fiables et les plus simples à mettre en œuvre(exact, synonymie), jusqu'à celles qui demandent des ressources ou des traitements plus complexes (paraphrase, métaphore). Cet ordonnancement suit l'état de l'art de l'utilisation de ces connaissances en QR, plaçant en premier les types de variations les plus utilisées dans les systèmes (mots exacts, synonymie, hyponymie, morphologie) et pour lesquelles il existe des ressources clairement définies. Nous avons placé en dernier les variations impliquant un glissement sémantique(scheme, collocation) ou requérant des traitements complexes d'inférence (transcat, para-meta).

Lors de l'annotation, nous avons effectué un étiquetage détaillé des différents phénomènes d'appariement entre verbes. Afin de visualiser des résultats de manière plus lisible et plus synthétique, nous utilisons ici la catégorie générale scheme pour regrouper ces phénomènes. Pour la même raison, une seule catégorie regroupe ici les relations lexicales fiables de WordNet que sont la synonymie, l'hyponymie et l'hyperonymie (catégorie syn-abbr-isa). Dans notre corpus, la catégorie hyponymie regroupe à la fois des relations nom à nom et des relations verbe à verbe, bien que ces dernières soient beaucoup moins fréquentes.

exact	62%		scheme	8,6%
ne-approx	2,5%		collocation	4,0%
syn-abbr-isa	7,6%		transcat	0,7%
derivational	2,5%		para-meta	2,5%
wn-others	1,1%		others	7,8%
dir-others	0,7%			

Tab. 2: Fréquences des classes de variation

Nous remarquons tout d'abord la forte proportion de termes (62%) qui se retrouvent à l'identique entre la question et la réponse. En deuxième vient la catégorie scheme(8,6%), qui donne des relations entre verbes et en troisième vient la catégorie syn-abbr-isa avec 7,6% d'occurrences, qui bien qu'ouverte aux verbes concerne principalement les noms. La catégorie syn-abbr-isa contient :

Synonymes et abréviations	5,1%
Hyperonymie et hyponymie	2,5%

Tab. 3: Fréquences dans la catégorie syn-abbr-isa

La catégorie *scheme* se subdivise en les sous-catégories suivantes :

temp. succession	2.9%
temp. inclusion	0,36%
cause-effect	2.2%
entailment	2.9%

Tab. 4: Fréquence dans la catégorie scheme

Des variations de morphologie dérivationnelle ont été trouvées seules (catégorie morpho). Cependant, dans certains cas elles ont été trouvées combinées à d'autres relations. Par exemple :

```
Q : Where was the first McDonald's built ?
R : Ray Kroc , who died in 1984 at 82 , bought out McDonald 's in
1960 from its founders in[REP San Bernardino] , Calif. , and turned
the hamburger restaurant into the ubiquitous international fast
food chain .
Lien Annoté : build#v –(succ. temporelle)– found#v –(morpho)–
founder#n
```

Cette mesure de fréquence montre que la variation sémantique est la réunion de nombreux phénomènes qui, à part pour certaines classes de relations comme la synonymie et l'hyponymie, se trouvent présents dans des proportions comparables (entre 4 et 2,5%). Cela met en évidence la nécessité pour les systèmes de combiner différentes ressources sémantiques, morphologiques et lexicales pour trouver les variations rencontrées entre question et passage réponse.

Cette approche donne des résultats similaires à une étude des variations sémantiques entre les questions et les phrases-réponses rapportées par le système de QR FRASQUES[BAR05b]. Cette étude donne le pourcentage de mots identiques et de synonymes entre question et réponse, qui sont comparables à ceux de notre corpus (TREC11).

corpus	FRASQUES	TREC11
exact	69%	62%
synonymes	4,9%	5,1%

Tab. 5: Fréquence des variations synonymiques selon deux études

La plus grande quantité de mots exacts dans le corpus FRASQUES semble confirmer que notre objectif de construction d'un corpus riche en variations sémantiques est au moins en partie atteint.

3.5.1.2 Constitution des justifications

Nous avons également mesuré la fréquence des phénomènes au niveau de la justification. Nous avons essayé de donner une idée de la couverture lexicale que pouvait apporter un système en fonction du type de phénomènes linguistiques qu'il est capable de couvrir. Les colonnes donnent le niveau de capacité de reconnaissance des phénomènes. Pour chaque colonne, les lignes donnent le nombre de termes non reconnus dans les justifications en utilisant ce niveau de capacité de reconnaissance. C'est-à-dire qu'il indique le pourcentage de justifications du corpus dans lesquelles

tous les termes seraient trouvés et les pourcentage de termes dont un ou plusieurs termes seraient considérés comme absents.

Pour donner une représentation lisible des données, nous utilisons notre ordre « a priori » de difficulté d'utilisation des phénomènes. Ainsi, dans le tableau 6, la colonne « exact » donne une estimation de la couverture sur notre corpus d'un système qui ne reconnaîtrait que les mots sans variation. La colonne <=neappr donne le pourcentage de termes non appariés lorsque l'on reconnaît uniquement les catégories exact + ne-approx, c'est-à-dire les mots repris sans variation + les variations sur les entités nommées. <=*is-a*, concerne l'ensemble de relations *is-a* + *ne-approx* + *exact*, soient les relations précédentes plus les relations de synonymie, d'abréviation, d'hyponymie et d'hyperonymie.

non reconnus	**exact**	**<=neappr**	**<= is-a**	**toutes**
0	33%	34%	45%	85%
1	45%	45%	37%	15%
2	13%	13%	11%	0
3	8,9%	8,1%	6,5%	0,8%
4	0,8%	0	0	0
0 à 4	100%	100%	100%	100%

Tab. 6: Nombre de critères manquants par passage selon les capacités du système.

La dernière colonne indique que 85% des passages contiennent tous les termes de la question, bien que celles-ci ne soient pas toutes repérables par les programmes actuels. On constate également que pour l'annotateur humain, il n'y a en général pas plus d'un critère absent.

Si l'on considère les catégories (<= *is-a)*, une grande proportion des documents sont complets ou ont un seul critère manquant (83%). Ce niveau de reformulation pourrait être un bon compromis entre le nombre de critères rapportés et le bruit apporté par le repérage de variantes sémantiques.

3.5.2 Répartition spatiale des éléments

Dans cette mesure, nous considérons les zones de justification sans compter les éléments isolés. Les zones de justification sont d'une longueur moyenne de 13,8 mots. Il apparaît que sur 123 justifications, 18 (15%) sont composées de deux zones de justification séparées, et que les autres 85% sont composés d'un seul bloc.

Nous avons mesuré l'étendue de la justification en termes de nombre de phrases entre le premier élément de justification et le dernier. Une étendue de 1 indique que la justification est réduite à une phrase. Une étendue de 2 indique que les zones sont situées dans deux phrases consécutives. Dans la première mesure, nous avons tenu compte uniquement des zones, c'est-à-dire les blocs principaux de la justification. Dans la seconde mesure, nous avons calculé l'étendue totale des justifications en tenant compte des éléments isolés. La seconde mesure donne donc nécessairement pour chaque justification une étendue plus grande que la première.

Nb phrases	1	2	3	4	5	6
zones	80%	10%	4,1%	1,6%	2,4%	0,8%
zones+elts	72%	11%	5,7%	2,4%	4,9%	0,8%

Tab. 7: Étendue spatiale des justifications

Il est remarquable que seulement 72% des justifications soient limitées à une phrase. Cela démontre que l'approche de recherche d'une réponse dans une seule phrase n'est pas suffisante, et devrait être étendue pour traiter de courts passages. Par exemple, des passages de 3 phrases pourraient couvrir la totalité de la justification dans 88,7% des cas. De plus, une telle étendue couvrirait le corps de la justification, c'est-à-dire les zones, dans 94,1% des cas.

3.6 Conclusion

L'étude du corpus nous a permis de mettre en évidence la nécessité de traiter un grand nombre de types de variations sémantiques, ainsi que d'effectuer l'analyse des justifications sur une étendue de plusieurs phrases.

Nous allons dans le chapitre suivant décrire le système d'extraction de justification que nous avons réalisé, lequel essaie de rendre compte des phénomènes observés.

Chapitre 4 : Algorithme d'extraction de justifications

Dans le chapitre précédent, l'étude d'un corpus de questions-réponses nous a permis de mettre en évidence les caractéristiques principales des justifications. Cette étude montre que les justifications des réponses sont souvent étendues sur plusieurs phrases et font appel à de nombreux phénomènes de reformulation dans des proportions relativement réduites pour chacun des phénomènes.

Le problème central que nous posons est de trouver un moyen de réunir dans un même système des connaissances hétérogènes. Comme nous l'avons vu, la technique actuellement utilisée par les systèmes capables de faire face à l'hétérogénéité est l'utilisation d'un classifieur qui détermine le résultat en fonction de mesures calculées au préalable.

Le problème qui se pose est que ces systèmes sont incapables d'intégrer les éléments de justification qui leur ont été fournis en entrée dans une représentation de la justification. Nous avons vu également que certains systèmes sont limités au niveau de la phrase. Au contraire, nous proposons une approche ne limitant pas a priori l'étendue de la justification.

4.1 Le cœur de l'algorithme

Nous présentons à présent l'algorithme d'extraction de justifications que nous avons implémenté en accord avec ce modèle.

4.1.1 Les enjeux du programme

4.1.1.1 Intégration modulaire de connaissances hétérogènes

Pour pouvoir intégrer des connaissances hétérogènes, nous avions besoin d'une architecture générale qui soit assez peu contraignante pour s'adapter à chacun des modules. Pour cela, nous avons défini deux grandes catégories de modules :

- Les Paraphraseurs, qui ont pour rôle de repérer les variations de sémantique locale décrites au chapitre 2 dans la section 2.4 et d'attribuer une note de fiabilité à ces éléments paradigmatiques. Ces éléments peuvent être aussi bien des termes que des fragments d'énoncé

plus importants. Par conséquent, les éléments reconnus par ces modules peuvent couvrir un ou plusieurs termes de la question. L'utilisation qui est faite de ces modules paraphraseurs doit tenir compte de ces différents niveaux de granularité.

- Les Lieurs de Termes, qui sont chargés d'établir la fiabilité d'un lien entre deux éléments de justification (termes ou réponse). Les modules de ce type sont en charge à la fois de la sémantique syntagmatique et de la sémantique énonciative. Ils sont en charge de donner une note aux critères syntagmatiques vus également au chapitre 2, section 2.4.

Le système peut comporter un nombre indéfini de Paraphraseurs et de Lieurs. L'algorithme, sera détaillé en section 4.1.2. Pour fixer les idées, nous ferons ici une description rapide de son fonctionnement. Dans un premier temps, cet algorithme demande à chaque Paraphraseur la liste des termes qu'il a reconnu. Les modules de type Paraphraseur peuvent être de natures très diverses.

Chaque type de modules est utilisé d'une façon unifiée par le module central, décrite par les deux interfaces Paraphraser et TermLinker. Pour permettre d'adapter simplement de nombreux modules, ces deux interfaces sont très simples. Outre quelques fonctions permettant de configurer les modules (init, set_justifieur, set_docno), l'appariement n'a recours qu'à une fonction pour chaque type de modules.

Pour les paraphraseurs, il s'agit d'une fonction getTerms() qui rapporte la liste des appariements entre éléments de la question et de la réponse que le module a détectés. Cette représentation unifiée est constituée des informations suivantes :

- Terme(s) de la question : Indique quel ou quels termes de la question sont couverts par l'appariement. Pour simplifier le problème, nous avons décidé de n'autoriser que l'appariement de segments continus de termes de la question. Ces segments seront représentés par les positions du premier et du dernier mot inclus.

- Position dans la réponse : À l'instar de ce qui est fait pour la question, la position dans la réponse est indiquée comme un fragment continu de texte, représenté par la position de début et de fin du fragment dans le texte.

- Étiquette : Une étiquette renseignant comment a été trouvé l'appariement. Par exemple, si un terme a été rattaché à un critère de la question par le module utilisant le programme Fastr, avec une variation sémantique, son étiquette sera composée de « FASTR: » suivi de l'étiquette attribuée par Fastr : « X,1,Sem ».

- Probabilité de l'appariement. Une grandeur par laquelle le module estime la fiabilité de son appariement. Cette grandeur sera discutée dans la section suivante.

Une fois obtenue la liste des appariements, le programme cherche à effectuer des liens les plus significatifs possibles entre les termes ou fragments d'énoncés du passage qui sont donnés par cette liste. Étant donnés deux termes ou fragments de texte du passage, le programme fera appel à chaque Lieur par le biais de sa fonction cost(), qui prend en entrée deux termes décrits par les informations énumérées ci-dessus et renvoie son estimation de la fiabilité de leur liaison syntagmatique dans le passage sous la forme de la probabilité de la mise en relation de ces deux termes.

Ces informations sont utilisées par un algorithme d'optimisation implémenté dans le module central Justifieur comme le montre le schéma de l'architecture du système (Fig. 11). Le module central choisit un ensemble de termes couvrant tous les termes de la question. L'ensemble de termes et de liens entre les termes doit être choisi de façon à optimiser une mesure de probabilité de la justification, calculée par le produit des probabilités de chaque critère syntagmatique et paradigmatique. Le calcul de la probabilité totale de la justification et l'algorithme d'optimisation utilisés seront décrits et discutés dans la section 4.1.2.

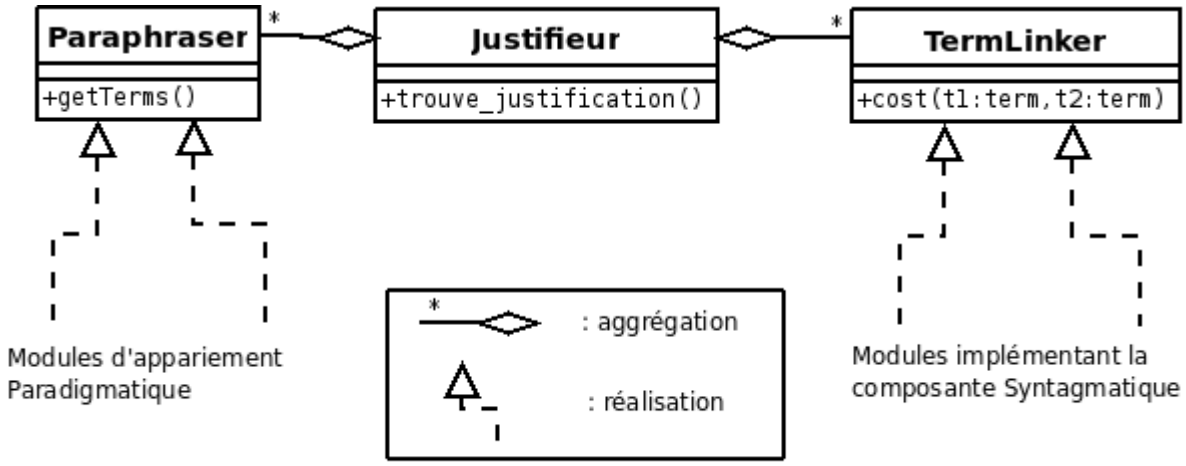


Fig. 11: Architecture actuelle du système de justifications

Ce schéma UML utilise les notations suivantes : l'agrégation désigne le fait qu'une classe du côté du losange contient un ou plusieurs éléments de la classe de l'autre côté. Ici le Justifieur possède une liste de paraphraseurs et une liste de TermLinkers. Les classes Paraphraser et TermLinker sont des classes abstraites, réalisées par différentes classes filles que nous appelons modules dans notre thèse.

Représentation des différents types de sémantique

Ce découpage en deux catégories se conforme au modèle théorique dans lequel les phénomènes de sémantique syntagmatique et de sémantique énonciative permettant de relier syntagmatiquement deux termes sont regroupés sous un même lien. Au niveau de l'algorithme, nous justifions ce choix par deux raisons :

Premièrement, il apporte une bien plus grande simplicité de représentation de la justification. Le choix de dissocier les modules de sémantique syntagmatique et de sémantique énonciative entraînerait l'apparition de nœuds intermédiaires dans la représentation de la justification, par exemple :

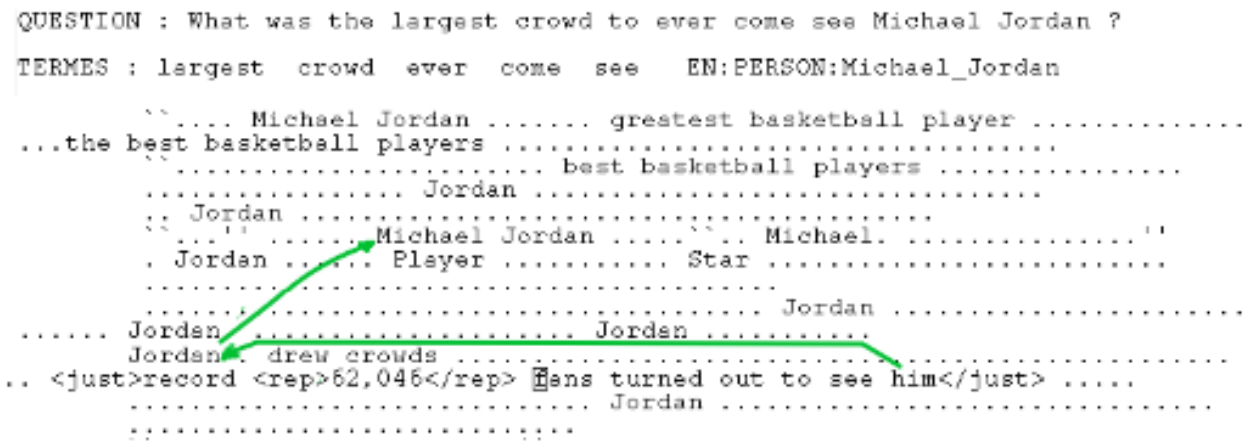


Fig. 12: Exemple de rattachement anaphorique

Dans l'exemple de la figure 12, nous aurions besoin de créer deux nœuds outils (« him » et « Jordan ») pour relier le prédicat « to see » au terme « Michael Jordan », par la succession d'une relation syntagmatique prédicat-objet de « to see » à « him », puis une relation énonciative

d'anaphore de « him » à « Jordan » et enfin une relation de coréférence de « Jordan » à « Michael Jordan ». Nous avons préféré ne garder dans la justification que les extrémités de la chaîne anaphorique. Un module lieur gérant les anaphores devra conserver secret le lien entre « to see » à « Michael Jordan ».

Cette approche demande un effort de programmation supplémentaire important et complique l'écriture des modules syntagmatiques au profit de la clarté de l'algorithme principal. L'algorithme central et la représentation utilisés fixent les interfaces Paraphraser et TermLinker. Par contre, il sera possible d'affiner ultérieurement l'interface TermLinker pour lui permettre de faire collaborer de manière modulaire des modules de sémantique syntagmatique et de sémantique énonciative. L'architecture du système deviendrait alors celle de la figure 13.

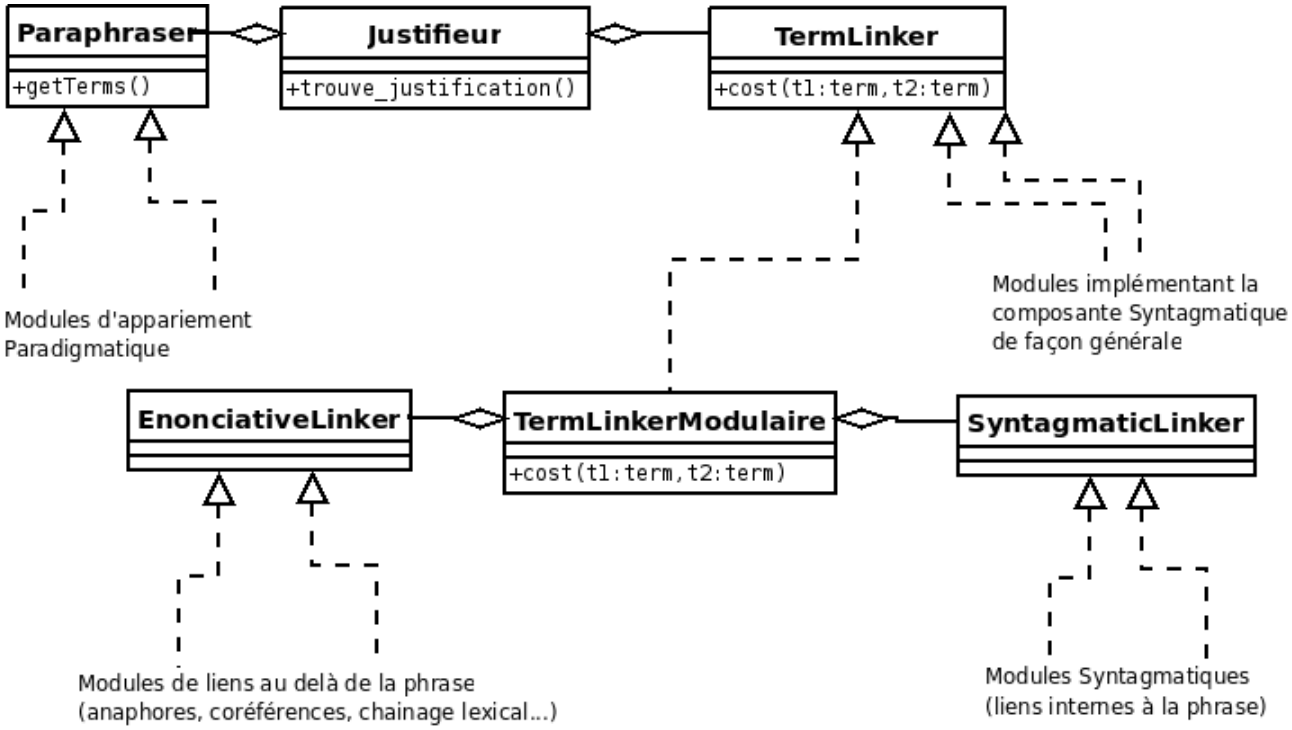


Fig. 13: Architecture envisageable pour le système d'extraction des justifications

Informations externes

L'apport d'informations externes au document se représente également dans ce formalisme. Pour rattacher une information externe, un module devra effectuer un lien entre cette information et un terme de la justification.

Pour ce faire, il pourra suivre la décomposition entre EnonciativeLinker et SyntagmaticLinker. Le module de type EnonciativeLinker devra vérifier qu'un terme du passage réponse est coréférent à un terme du passage distant. Cette coréférence peut dans le cas le plus fiable et le plus simple consister à relier deux entités nommées référent à la même entité. On peut supposer que ce type de rattachements comporte des cas plus complexes, comme par exemple relier deux expressions représentant le même événement. L'évaluation de la fiabilité de ce type de mise en relation pourrait alors s'appuyer sur la reconnaissance d'un contexte commun aux deux passages.

Le second module, de type SyntagmaticLinker, devra estimer la probabilité d'un lien syntagmatique entre ce terme du passage distant et l'information manquante. Si l'on reprend l'exemple de la vérification des caractéristiques d'une personne dans Wikipédia, cela pourrait donner

le rattachement suivant :

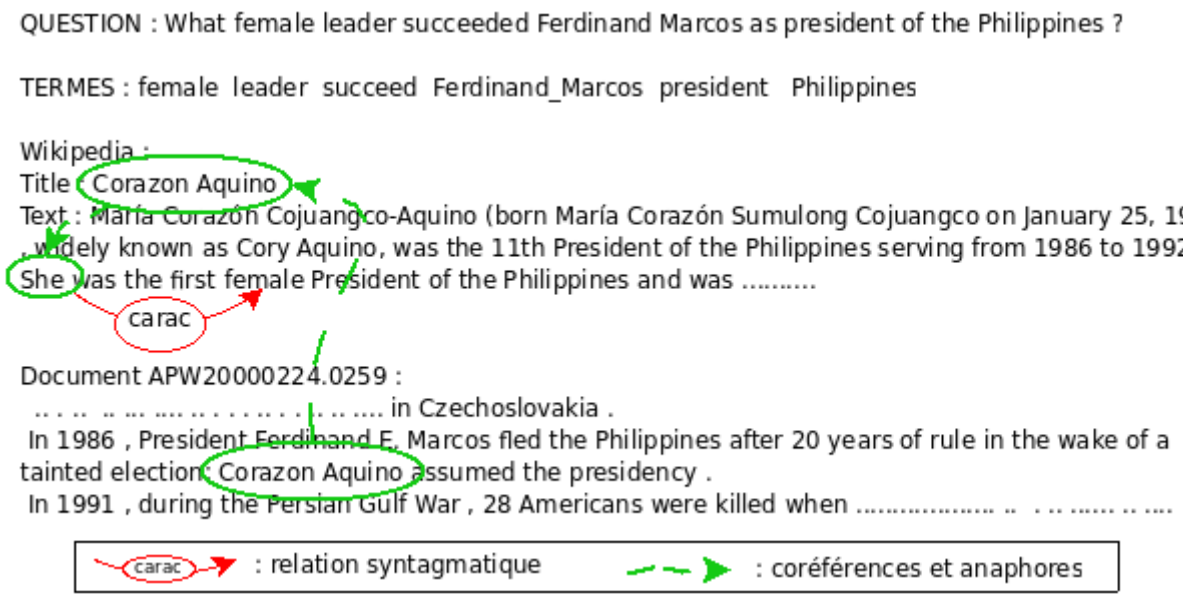


Fig. 14: Rattachement d'une information d'un document externe

Couverture des termes de la question

Le but de l'algorithme est de trouver une sélection de termes couvrant les termes de la question. Chaque module apporte au système central une liste de termes qu'il a repérés. Pour chacun de ces termes est fournie sa couverture de la question sous forme d'un intervalle de mots de la question.

Nous avons vu dans le chapitre précédent, à la section traitant de l'analyse de la question, que bien que nous souhaitions conserver le plus souvent possible les termes composés et notamment les noms de personnes, nous avons considéré que le système central pouvait avoir besoin d'apparier des zones de texte ne respectant pas la notion de terme composé. Ceci nous a amené à choisir comme plus petite unité d'appariement, utilisée pour le calcul de couverture, le mot plutôt que le terme. Nous présentons ici les aspects de notre algorithme qui justifient ce choix.

Premièrement, un nom de personne peut être retrouvé de manière partielle dans un passage. Bien qu'en général, le nom complet apparaisse au moins une fois dans le passage, le nom d'une personne (par exemple « Michael Jordan ») pourra ensuite être repris par le nom de famille seul, voire par le prénom seul.

Un module de résolution de coréférences pourra détecter que « Jordan » est en fait une occurrence elliptique de l'entité nommée « Michael Jordan ». Cette connaissance pourrait alors être intégrée dans un module de type Paraphraseur, qui attribuerait à l'occurrence elliptique de l'entité nommée les propriétés de couverture de l'entité nommée totale, ou encore par un module de type Lieur, qui accepterait de relier sans coût supplémentaire l'entité complète à des termes distants si ceux-ci sont liés de façon locale avec une de ses occurrences elliptiques.

L'utilisation d'une de ces deux techniques permettrait d'empêcher l'apparition de l'entité elliptique dans la justification finale, résolvant ainsi le problème de la gestion de fragments d'énoncés en deçà du terme. Cependant, nous ne pouvons pas attendre ce comportement de chaque module implémenté. De plus, on peut supposer que certains passages candidats contiendront seulement des occurrences elliptiques, notamment à cause du fait que nous travaillons sur des fragments de documents. Ceci nous contraint donc à concevoir un système capable de traiter aussi les sous-termes « Michael » et « Jordan ».

De plus nous utilisons dans notre programme un module de détection des paraphrases basé sur la reconnaissance de la plus longue chaîne de critères de la question trouvable dans la phrase. Ce module accepte lui aussi de détecter des sous-termes. Il est donc possible qu'une zone de texte repérée par le module de paraphrases chevauche un terme comme dans l'exemple suivant :

```
Q1823 : What number did Michael Jordan wear ?
R : Jordan returns wearing number 45, the same number he wore
```

L'expression soulignée constitue la plus longue sous chaîne commune, incluant le sous-terme Jordan ainsi que d'autres critères de la question.

Ces exemples montrent donc que malgré des efforts pour traiter les termes composés comme un tout, le programme aura vraisemblablement besoin de traiter dans certains cas des entités partielles, ce qui justifie le choix du mot comme unité d'appariement.

Dans ce cadre, la notion de couverture de la question s'exprime de la façon suivante. Chaque terme de la question est converti en un ensemble de sous-termes minimaux qui sont les mots pleins qu'il contient, en comptant comme mot plein les catégories grammaticales suivantes : nom propre, nom commun, verbes, adjectifs, adjectifs, adverbes. Tout module de type Paraphraseur fournit, pour chaque variation d'un terme de la question qu'il repère dans un passage, l'intervalle de mots de la question couvert. On peut grâce à cette information calculer la liste des sous-termes minimaux de la question couverts. Une sélection de termes de la question couvre tous les termes de la question si tout sous-terme de la question est recouvert par au moins un terme de la sélection.

Une objection à ce choix est que l'analyse des termes de la question, qui traite les termes composés comme une structure arborescente, semble inutilisée. Cependant, cette analyse est disponible pour tout module souhaitant l'utiliser, et nous considérons qu'il est du ressort de chaque module de vérifier l'adéquation des termes qu'il repère aux termes présents dans la question. C'est notamment ce que fait le module Fastr. Ce choix, bien que critiquable, permet une grande indépendance dans l'implémentation des modules.

4.1.1.2 Les probabilités comme échelle commune

Afin d'autoriser la collaboration de plusieurs sources de connaissances sémantiques dans un même programme, le système doit posséder une fonction de mise en commun des poids apportés par chaque module. De plus, nous voulons que cette fonction permette de faire varier librement le nombre de modules, et que les modules soient réellement modulaires, c'est-à-dire que leur phase de programmation soit réalisable indépendamment des autres modules. Ce problème d'indépendance n'apparaît pas pour la plus grande partie du développement des modules. Chaque module peut en effet faire appel aux ressources qu'il désire, faire les calculs qu'il désire sur le texte et sur ces ressources. C'est surtout au moment de la mise en commun de la pondération que la question de la modularité du développement se pose. En effet, il est nécessaire pour faire fonctionner un programme d'appariement, d'évaluer l'apport de chaque module - ou mesure - au fonctionnement du système total. Certaines mesures recevront une confiance plus importante que d'autres.

Cette confiance se traduira de différentes façons selon le programme unificateur. Ainsi dans le système QALC, le programme de classement des phrases utilise une unification de plusieurs mesures sous forme d'une somme de poids partiels apportés par différents modules. L'importance relative de ces modules a été réglée par l'expérimentation, par des coefficients propres à chaque module réglant l'ordre de grandeur des poids partiels. Ainsi la mesure de présence des termes de la question a un ordre de grandeur de 1000, ceux des entités nommées et de la proximité syntagmatique de l'ordre de 50.

Cet ajustement final nécessite de tester le module dans le système total. Ce qui complique son intégration. Une solution pour pallier ce problème consiste à automatiser le réglage par un module d'apprentissage. Nous avons présenté des exemples de tels systèmes dans le Chapitre 1. Nous avons opté pour un choix différent qui consiste à choisir une échelle commune de représentation de la fiabilité des modules.

Choix d'une représentation commune de la fiabilité

Nous avons choisi comme échelle commune les probabilités. Chaque module attribue aux phénomènes qu'il gère une évaluation de la probabilité que le phénomène soit sémantiquement pertinent vis-à-vis de l'élément de la question auquel il correspond.

Pour que le modèle permette une intégration peu contraignante, nous sommes contraints de faire l'hypothèse réductrice de l'indépendance des mesures de fiabilité de chaque élément entre elles. Cela signifie qu'un module implémentant une mesure particulière n'est pas capable de profiter d'informations extérieures pour affiner son évaluation.

De plus notre programme ne permet pas de combiner plusieurs indications partielles portant sur un même élément de justification pour affiner l'estimation de la probabilité de cet élément.

Calcul des probabilités par les modules

L'avantage d'une mesure de probabilité est qu'elle peut être calculée indépendamment pour chaque module par étude de corpus. On suppose, toujours selon l'hypothèse d'indépendance des modules, que la probabilité qu'une information rapportée soit correcte est approximée correctement par la pertinence sémantique de l'élément de justification. Par exemple, dans le cas d'un appariement de termes, il s'agit de mesurer si le terme de la réponse est sémantiquement lié au terme de la question, sans se préoccuper du contexte. On peut exprimer cela en disant que les termes de la question et de la réponse qui sont appariés doivent dénoter le même concept ou des concepts proches tels que le concept de la réponse permette d'inférer le concept de la question. (« to commit suicide » => « to die », « chirurgien » a pour rôle « opérer »)

Actuellement, les estimations de fiabilité de nos modules ne sont pas fondées sur des mesures sur corpus. Les probabilités sont estimées par des fonctions créées pour donner un résultat vraisemblable. Ces fonctions seront décrites pour chaque module.

Nous avons pour projet d'effectuer la pondération des différents phénomènes repérés par les modules grâce au corpus. La méthode escomptée consisterait à exécuter l'algorithme de sélection de passages sur le corpus afin de ramener des justifications contenant chacune un certain nombre d'appariements. On comptera alors pour chaque phénomène le nombre d'occurrences où il est sémantiquement pertinent.

Cette tâche de pondération des modules est fort lourde si elle est effectuée manuellement. Le corpus de questions-réponses décrit au chapitre précédent devrait permettre d'automatiser l'estimation des probabilités, en conservant l'annotation de pertinence sémantique des appariements validés précédemment. De cette façon, dans le cas de l'ajout ou de la modification d'un module, le calcul de sa fonction de pondération bénéficiera des annotations précédentes. Si ce module repère de nouveaux appariements, la pertinence sémantique de ceux-ci sera annotée et viendra enrichir le corpus.

4.1.2 Algorithme central

Nous décrivons maintenant l'algorithme qui effectue le calcul de la justification optimale à partir d'une analyse de la question et d'un passage de la réponse. Comme nous l'avons dit précédemment,

la justification consiste en la liste des appariements élément à élément entre un critère de la question et son correspondant dans le passage réponse. Les critères et donc leurs éléments correspondants sont de deux sortes : paradigmatiques, repérés par les modules de type Paraphraseur et syntagmatiques, repérés par les modules de type Lieur. Dans l'algorithme, les justifications candidates sont représentées par l'ensemble des termes choisis, nommé sélection. La sélection des éléments de la réponse suffit à caractériser la justification car les liens entre ces éléments découlent de façon déterministe de cette sélection.

L'algorithme est le suivant :

Soit T la liste des termes et de fragments de textes obtenus par les différents paraphraseurs. On cherche à trouver la sélection S qui optimise la probabilité de la justification tout en vérifiant les critères suivants :

- S couvre entièrement la question.
- Il n'y a pas de terme t dans S tel que S \ {t} couvre t. Dans notre algorithme, tous les cas ne sont pas vérifiés mais nous veillons à ce qu'il n'y ait pas trop de termes redondants.

Cette probabilité est calculée de la manière suivante :

Soit S une sélection de termes. Il existe de nombreuses façons de relier ces termes entre eux pour former un arbre. Pour chaque arbre possible, on peut calculer sa probabilité qui est le produit des probabilités d'appariement des critères sémantiques et des critères paradigmatiques. Ces probabilités, nous le rappelons, sont fournies par le module qui détecte l'élément de justification correspondant au critère dans le document. La probabilité associée à la sélection est le maximum des probabilités des arbres possibles.

*4.1.2.1 Implémentation par l'algorithme A**

Nous avons utilisé pour implémenter notre algorithme d'optimisation un algorithme A* (« A étoile », « A star ») . L'algorithme A* est un algorithme qui sert à résoudre le problème représenté de la manière suivante. Soient :

- Un graphe G (implicite ou explicite) dont chaque nœud est nommé un état, Chaque arc du graphe indique qu'on peut passer de l'état source à l'état cible. Cet arc est étiqueté par le coût du changement d'état. Nous noterons ces coûts unitaires par une fonction cout(e1,e2), où e1 et e2 sont des états contigus.
- Le nœud de départ D et le nœud d'arrivée A.

Le coût total du trajet est défini comme la somme des coûts de chacune de ses transitions. L'algorithme cherche une solution qui est un chemin allant de D à A en minimisant le coût total du trajet.

Pour ce faire, l'algorithme A* utilise une heuristique h(n1) qui pour un nœud du graphe donne une évaluation de la distance qui reste à parcourir jusqu'à A. On peut démontrer que si l'heuristique utilisée est optimiste, c'est-à-dire si elle sous-évalue toujours le coût réel de déplacement, alors le premier chemin trouvé sera le chemin optimal. Une fonction h optimiste est dite admissible.

En plus de la fonction d'heuristique h, une fonction g(E) donne le coût minimal pour parcourir le chemin de D jusqu'à l'état courant E. Ce coût est le minimum du coût de tous les chemins déjà parcourus jusqu'à l'état courant. À chaque pas de l'algorithme, chaque état E qui a déjà été atteint possède donc une estimation de la distance totale de D à A en passant par E. Cette estimation est nommée f = g(E) + h(E). Notons que la fonction g n'est jamais calculée directement. Comme les distances s'obtiennent par somme des coûts des transitions la fonction g(E), celle-ci est calculée à

partir des valeurs de g pour les cases voisines déjà parcourues. Ainsi la fonction g d'un nouvel état E est calculé comme le minimum pour E' sens voisines déjà parcourues de g(E') + cost(E',E).

Un exemple simple d'utilisation de A* est la recherche du plus court chemin dans un espace quadrillé où sont placés des murs. Dans ce cas, un état peut posséder jusqu'à 4 transitions : le déplacement d'une case sur l'axe des abscisses de façon positive ou négative et le déplacement d'une case sur l'axe des ordonnées également dans chaque sens. Si une des transitions débouche sur un mur, elle est alors impossible. Pour cet exemple, nous considérerons que les coûts des transitions sont homogènes, c'est-à-dire qu'on se déplace sur le quadrillage partout à la même vitesse. Dans ce cas, une heuristique simple du coût de déplacement consiste à évaluer le coût de déplacement dans les murs. Dans un quadrillage, la distance sans les murs entre les états e1(x1,y1) et e2(x2,y2) est abs(x1-x2) + abs(y1-y2). Cette heuristique est toujours inférieure ou égale à la réalité. Par conséquent, elle permet de trouver le chemin optimal dans le quadrillage.

On trouvera une description plus complète de l'algorithme A étoile sur Internet[13] ou dans un livre traitant d'intelligence artificielle tel que [RUS06].

4.1.2.2 Application de l'algorithme à notre cas

Dans notre cas, nous avons utilisé comme ensemble d'états du graphe l'ensemble des sélections S parmi la liste de termes T disponibles. C'est donc l'ensemble des parties de T, P(T).

Chaque transition part d'une sélection S1 à une sélection S2 qui contient un élément de plus que S1.

Les transitions autorisées s'efforcent de respecter la prétention de l'algorithme à choisir des sélections de termes non redondants. Pour ce faire, nous n'autorisons que les transitions pour lesquelles le terme ajouté augmente la couverture de la question. De plus, afin de limiter la combinatoire de construction de la sélection, nous forçons les termes à être introduits dans un ordre précis, par ordre décroissant de leurs positions dans la phrase réponse. Cela signifie que si un état contient les termes $\{t_1,t_2...,t_k\}$ le nouveau terme t_{k+1} devra être positionné dans le passage réponse avant les termes t_1 à t_k.

La fonction g doit être calculée par additions successives, mais doit correspondre au produit de probabilités du sous-graphe déjà construit. Par conséquent nous convertissons les probabilités en un coût selon la formule cout = -log(proba). Notons que comme proba appartient à [0,1], -log(proba) est positif ou nul.

Lorsque la probabilité croît de 0 à 1, le coût décroît de l'infini à 0. Par conséquent, au lieu de maximiser la probabilité, nous minimiserons le coût correspondant, ce qui est justement le but de l'algorithme A*.

Décrivons maintenant le calcul de la fonction g. Elle est calculée à partir de l'état d'origine S', qui représente le sous graphe optimal avec un terme de moins. En ajoutant un nouveau terme t, nous devons :

- ajouter le coût de l'appariement paradigmatique de ce terme t, c(t).
- rattacher ce terme à un des termes déjà sélectionnés, c'est-à-dire, puisque chaque état de notre graphe est un ensemble S de termes sélectionnés, il s'agit de choisir le terme t', élément de S' qui minimise le coût du lien syntagmatique : cl(t',t). L'évolution de la valeur de g lors du parcours de l'espace des états est illustrée par la figure 15.

13 http://en.wikipedia.org/wiki/A_star

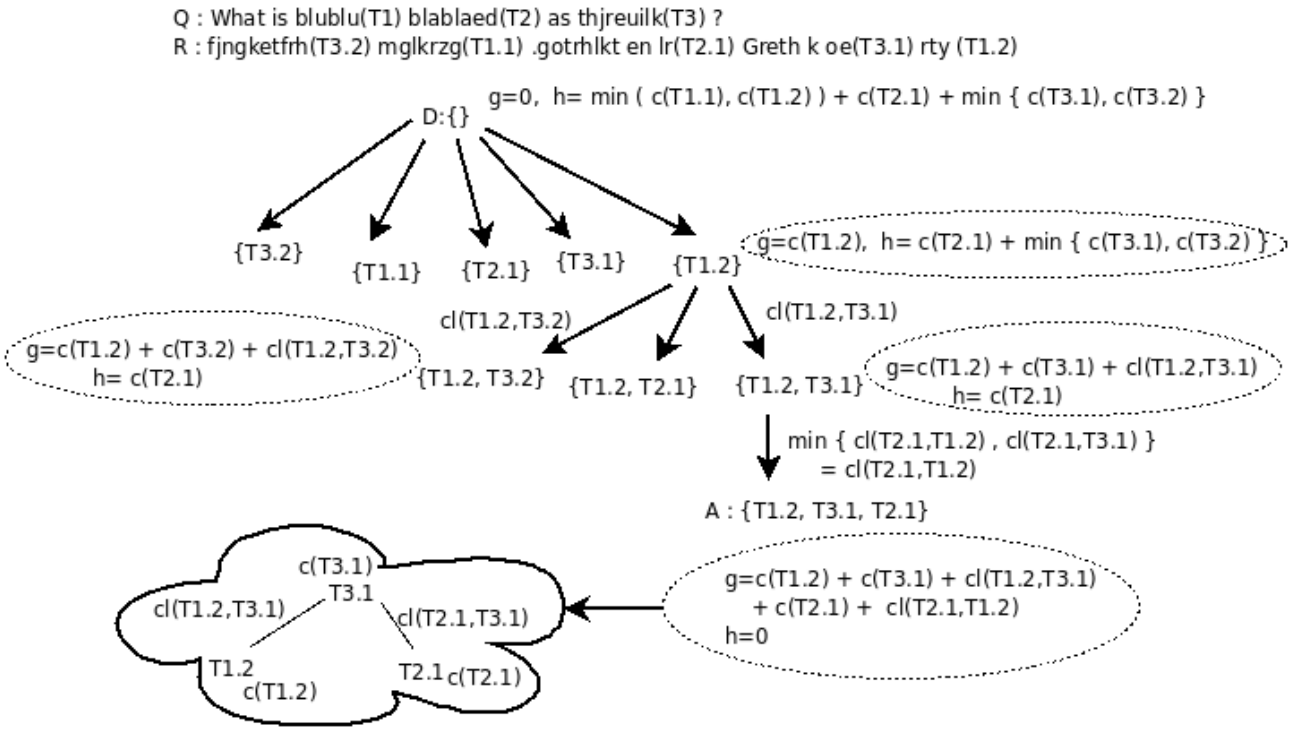


*Fig. 15: Sélection de l'arbre de justification par un algorithme A**

Pour la fonction h qui évalue le coût du reste des termes, nous avons décidé d'utiliser une évaluation optimiste du coût de couverture des termes manquants par les termes restants (positionnés avant t_k). L'estimation de ce coût minimum tient compte des différents niveaux de granularité des termes. Elle est effectuée de la façon suivante :

Algo cost_of_missing(liste de termes S)

soit H une table associative associant à chaque terme minimal de la question le coût minimal rencontré. (syntaxe H(clé) := valeur)

pour chaque terme t du document positionné avant tous les termes de S, **faire**

soit L la liste des sous-termes minimaux de la question couverts par t

soit n le cardinal de L

si n vaut 0 **alors** on passe au terme suivant (t ne couvre aucun élément de la question)

soit c le coût associé à t (nous rappelons que tout module qui fournit un terme au système indique sa probabilité de pertinence, et donc ce coût est connu)

soit cr := c / n (il s'agit d'un coût du terme réparti sur chaque sous-terme minimal couvert)

pour chaque sous-terme s de L, **faire**

si H(clé) n'est pas encore défini ou H(clé) > cr, alors

H(clé) := cr (H(clé) stocke le coût minimal rencontré jusqu'ici)

finsi

finpour

s'il existe un sous-terme minimal de la question dont le coût n'est pas défini dans H **alors**

(cela signifie qu'il ne reste plus assez de termes de la réponse pour couvrir tous les sous-termes

minimaux de la question.)

retourner +INFINI

sinon

retourner la somme des coûts unitaires.

finalgo.

Il est vraisemblablement possible de prouver que cet algorithme renvoie un coût optimiste. Cependant, pour ne pas alourdir l'exposé, nous nous contenterons de donner un exemple :

supposons qu'il reste 5 termes à utiliser :

- 3 termes a, b et c de coût 3, couvrant chacun un mot de la question.
- 2 termes composés a-b et b-c, couvrant chacun deux des précédents et coûtant chacun 5.

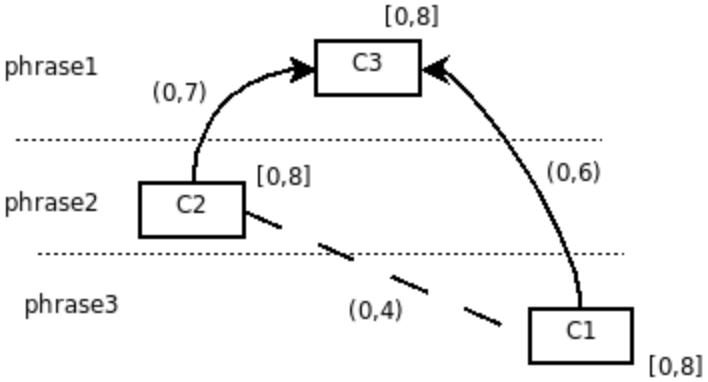
le coût minimal réel pourra être obtenu en combinant un terme composé, par exemple « a-b », avec le terme simple complémentaire, ici « c ». Le coût minimal réel vaut donc 5 + 3 = 8.

L'algorithme produit l'estimation suivante :

Les termes a, b et c apportent les estimations suivantes : H(a) = 3, H(b) = 3, H(c) = 3. Les termes « a-b » et « b-c » apportent un coût réparti de 5 / 2 = 2,5 pour chacun de leurs sous-termes. H(a), H(b) et H(c) passent donc tous trois à 2,5 . Le coût total estimé vaudra donc 2,5 +2,5 + 2,5 = 7,5 .

4.1.2.3 Caractère approximatif de l'algorithme d'optimisation

Cet algorithme ne donne pas nécessairement le meilleur appariement. Cela ne vient pas de la fonction h, qui est optimiste, donc admissible, mais de la restriction faite au parcours des termes. Comme nous l'avons dit, ces termes sont inclus dans le graphe par ordre décroissant de leur position dans la phrase. Or, comme le lien avec le terme entrant est nécessairement fait avec un terme déjà sélectionné, il en découle que certains graphes ne peuvent pas être générés par l'algorithme. Nous allons présenter un exemple théorique pour illustrer ce problème.



Cet exemple ne contient que trois termes, portant chacun sur un critère différent de la question. La justification attendue doit relier C1 et C2 à C3. L 'algorithme se déroulera cependant de la façon suivante :

1) Introduction du terme C1. S := {C1}, proba := 0,8
2) Introduction du terme C2. Le terme est relié au seul élément disponible, C1. La

probabilité est modifiée par la probabilité du nouveau terme(0,8) et celle du lien entre C2 et C1(0,5). proba := 0,8 * 0,8 * 0,5 = 0,32

3) Introduction de C3 : deux liens sont possibles, C3 – C1 ou C2 – C1. Le meilleur lien est C2-C3, de probabilité 0,7. La probabilité du terme ajouté C3 est 0,8. La probabilité devient donc : proba := 0,32 * 0,7 * 0,8 = 0,179

Le meilleur graphe, qui choisirait les liens C1-C3 et C2-C3, aurait une probabilité de $0,8^3$ * 0,7 * 0,6 = 0,215.

En pratique, cette configuration est rare car les liens entre phrases plus proches sont généralement mieux notés. Nous avons choisi cet algorithme pour élaguer au maximum le parcours de l'espace des sous-graphes. Notons que l'algorithme donne des résultats tout à fait satisfaisants comme en attestent les justifications qu'il extrait, dont certaines seront données en illustrations à la section 4.3.

Il est cependant envisageable de modifier l'algorithme d'optimisation si le besoin venait à s'en faire sentir.

4.1.3 Limitations de l'algorithme

4.1.3.1 Collaboration limitée entre modules

L'indépendance des modules est problématique car des modules différents gagneraient collaborer entre eux. Par exemple, dans la chaîne de questions réponse QALC, plusieurs mesures de similarité sont sommées pour évaluer le recouvrement d'un terme. En effet, le poids apporté par un terme de la question est la somme suivante :

poids := poidsMotExact + poidsVariationFastr.

Ces poids sont obtenus par des méthodes différentes (analyse de la sortie de TreeTagger d'une part, programme Fastr d'autre part. L'intégration modulaire idéale consisterait à intégrer deux modules séparés réunis par un système collaboratif.

Cependant, le système actuel se borne à sélectionner pour chaque terme de la question le meilleur appariement trouvé par l'un des modules. De ce fait, la collaboration entre informations diverses n'est possible que dans le cadre d'un même module, qui devra accéder aux résultats de plusieurs analyses hétérogènes ou encore à plusieurs ressources linguistiques.

Une possibilité est d’améliorer la fonction de sélection de la meilleure probabilité pour qu'elle augmente une probabilité dans le cas de nombreuses informations concourant à sélectionner un même passage de la question. Mais ce problème n'est pas trivial puisqu'il faut choisir une fonction appropriée. Plusieurs pistes sont envisageables. Nous avons notamment vu que certains systèmes utilisaient un système d'apprentissage au niveau le plus englobant du système, pour réunir les informations hétérogènes. Bien que dans ces systèmes le décideur soit placé tout en aval de la chaîne de traitements, on peut envisager d'utiliser un décideur pour calculer un score synthétique au niveau d'un terme donné. On peut envisager également de tester différentes fonctions de synthèse plus mathématiques, notamment un calcul de plausibilités suivant le formalisme mathématique de Dempster-Shafer.

4.1.3.2 Contraintes algorithmiques sur les modules

Ce formalisme de questions-réponses est en grande partie compatible avec une chaîne de Questions-Réponses classique. Notre formalisme possède deux limitations majeures pour l'intégration de modules de QR.

La première limitation tient à la difficile intégration de mesures donnant une pondération globale du passage. Prenons le cas de la mesure de proximité syntagmatique par densité de termes utilisée par le système de questions-réponses QALC, qui sera décrite dans la section 4.2.2.3. Cette mesure donne une pondération qui porte sur une phrase entière, obtenue par somme des pondérations de chaque couple de termes de la phrase. Or, notre algorithme ne permet pas naturellement de considérer tous les couples de termes d'une phrase. Dans notre représentation de la justification, la réponse est en effet représentée par un arbre où les nœuds sont les critères paradigmatiques et les arêtes sont les critères syntagmatiques. Une propriété bien connue des arbres indique que si un arbre possède N nœuds, alors il possède N-1 arêtes. Par conséquent, seuls N-1 couples seront examinés lors du calcul de la pondération d'une justification.

Il est également possible de faire porter la mesure globale sur un grand bloc de termes de la question, voire sur la totalité de ceux-ci. Dans ce cas, la mesure doit fournir une probabilité de pertinence sémantique du bloc total en tenant compte de ses éléments de sémantique paradigmatique et syntagmatique interne. Ce bloc sera traité par l'algorithme comme un élément paradigmatique, c'est-à-dire de la même façon qu'un terme ou une paraphrase.

La seconde limitation tient à l'algorithme A* utilisé pour la recherche du graphe optimal dans l'univers des graphes possibles. Cet algorithme impose d'utiliser un poids sous forme d'une somme de coûts d'utilisation de chaque élément. Dans notre cas, le système calcule une probabilité de la question, laquelle se calcule par produit des probabilités des différents éléments de justification, ce qui se traduit en une somme des opposés des logarithmes : cout = -log(proba).

L'algorithme A* est donc compatible avec une mesure de fiabilité additive ou multiplicative, mais pas nécessairement à d'autres méthodes combinaisons. Pour y remédier, il sera envisageable de conserver l'algorithme A* en adaptant la mesure ou d'utiliser un autre algorithme d'optimisation de graphe ne possédant pas cette limitation.

4.2 Des modules pour le modèle

Nous voyons ci-après les modules qui ont été réalisés pour « nourrir » l'algorithme central. Le fonctionnement conjoint des modules sera ensuite illustré par des exemples dans la section 4.3.

4.2.1 Modules paradigmatiques

4.2.1.1 Appariement de termes avec Fastr

Nous avons implémenté un module de paraphrase permettant l'utilisation du programme de reconnaissance de termes Fastr dans notre système, qui permet de reconnaître les variantes morphologiques et sémantiques des mots ainsi que les termes composés.

Le programme Fastr attribue un poids à chaque variante calculée selon un certain nombre de critères, comme la présence de noms propres, le nombre de mots composant le terme, la fréquence des termes en corpus. Cette mesure a été déterminée par l'expérimentation.

Dans un premier temps, nous avons décidé d'approximer la probabilité en appliquant une fonction simple sur la valeur apportée par Fastr. Ces poids prenant une valeur située entre 0 et 3. Nous avons appliqué la fonction suivante :

```
proba := poids * 2 / 3
si proba < 0 alors proba := 0 finsi
si proba > 1 alors proba := 1 finsi
```

Il serait cependant préférable d'affiner cette mesure sur corpus.

4.2.1.2 Module de détection de la réponse EN

Ce module utilise l'analyse des entités nommées de la chaîne MUSCLEF, pour la question et la réponse. Son but est d'évaluer si un terme de la réponse correspond à un type d'entité nommée attendu, fourni par l'analyse de la question.

En raison de l'hypothèse d'indépendance des modules, ce module à uniquement la charge d'évaluer la pertinence sémantique du type de l'entité nommée, et non la probabilité que le terme repéré comme entité nommée soit effectivement la bonne réponse.

Pour cette raison, étant donné que l'analyse des entités nommées est assez sûre, nous avons jugé raisonnable en première approximation de donner aux termes de type entité nommée une probabilité de 1.

4.2.1.3 Module de plus longue sous-chaîne commune

Nous avons vu à différents endroits de cet exposé que notre but était de gérer des variations de nature hétérogène et notamment des appariements de granularités différentes. Pour tester ce principe, nous avons implémenté une mesure de plus longue sous-chaîne commune (Longest Common Subchain, LCS) [GRA08].

Le principe de cette mesure est d'identifier, pour une phrase du document candidat, une séquence de mots de la question contenant de nombreux termes de la question fortement rapprochés les uns des autres. Cette technique permet de pallier l'absence de gestion fine de la syntaxe en repérant des paraphrases entre un fragment des termes de la question et de la réponse.

L'algorithme utilisé pour pondérer les phrases est le suivant :

1 : analyse des mots et termes du passage

les mots sont classifiés selon 4 catégories :

- terme : si le mot est reconnu par Fastr
- réponse : dans AVE la réponse est donnée en entrée du système. Dans notre cas, cette catégorie est utilisée dans le cas où une entité nommée a été repérée.

  Nous représenterons les termes et la réponse par la même lettre (C) pour désigner les critères de la question.
- mot vide (mv): si le mot est un déterminant, une préposition, un pronom personnel, une conjonction de coordination ou un signe de ponctuation (étiquette TreeTagger parmi

{'DT','IN','PP','CC','PUN'})

- mot plein sinon (mp).

2 : détermination de la plus longue chaîne de termes du passage, répondant aux critères suivants :

- deux critères de la question consécutifs appartiennent à la même chaîne s'ils sont séparés par au plus un mot plein, ce qui correspond au patron : `mv* (mp mv*)?`
- le même terme n'est pas répété.

Par conséquent, dans un passage de la forme :

`mv mp mp mv C1` **`C2 mv C1 mv mp mv mv C4`** `mp mp C5 mv mv mp`

On peut construire la chaîne indiquée en gras, mais pas inclure la première occurrence du critère C1 qui introduirait une répétition, ni le critère C5, qui est séparé du reste de la chaîne par deux mots pleins.

3 : Calcul du nombre de mots communs entre la sous-chaîne et la question :

Soit NbC le nombre de mots de la chaîne qui sont soit un critère, soit un mot vide présent dans la question.
Soit NbQ le nombre de termes et de mots vides de la question + 1 pour compter la réponse.
Le poids est NbC / NbQ.

Le fait que les mots vides soient comptés avec la même valeur nous paraissait inapproprié et nous avons donc modifié la mesure pour ne compter que les termes et la réponse éventuelle.

$$poids = \frac{Nombre\ de\ critères\ de\ la\ chaîne}{Nombre\ de\ critères\ de\ la\ question}$$

On peut critiquer le classement de cette mesure comme module paradigmatique. En effet, cette mesure fait intervenir un élément de sémantique paradigmatique (la reconnaissance des termes et de l'entité nommée) associé à un élément de nature syntagmatique (la proximité des mots dans la phrase). Ce module mixte est classé dans la catégorie Paradigmatique en raison de la façon dont il est utilisé par le système. En effet, la chaîne maximale obtenue est ensuite représentée comme un « super terme » qui peut être utilisé par le système comme tel.

L'avantage d'un tel module doit provenir de la gestion ad hoc des liens internes à la sous-chaîne.

En effet si l'on prend un exemple suivant tiré de [GRA08].

Q : « Qui est le père de la reine Elisabeth 2 ? ».
Terme réponse : George VI
hypothèse *: George_VI(C1) est le père(C2) de la reine(C3) Elisabeth(C4) 2(C5)*
passage : « **George VI, le père** d' **Elisabeth 2,** l' actuelle **reine** d'Angleterre ... »
forme du passage : C1 mv(le) C2 mv(de) C4 C5 mv(la) mp C3 ...

Et

Terme réponse : Noël
hypothèse : Noël(C1) est le père de la reine Elisabeth 2
passage : « **La reine Elisabeth 2** était hier en visite dans une école qui fêtait la venue prochaine du **père Noël».**
forme du passage : mv(la) C3 C4 C5 mp mp mv C2 C1

Les deux passages ont le même nombre de termes communs avec la question, mais le premier est valide et ses termes sont plus proches. Montrons comment le mécanisme de probabilités de notre système classera les deux réponses dans l'exemple abstrait suivant :

```
Q : qui est le père(T1) de la reine(T2) Elisabeth(T3) II(T4) ?
R1 : .... T2 ..... T1 T3 T4 ....
R2 : ... T2 ..... T3 T4 ..... T1
```

Dans le premier cas, la plus longue sous-chaîne est de taille 3, que nous noterons T234, le calcul de probabilités effectué par le système est : p(Q ~ R1) = p(T234) * p_lien(T1,T234) * p(T1)

Nous noterons de plus que la grande fiabilité de la sous-chaîne T234 lui donne une probabilité élevée, meilleure que celle des termes séparés.

Dans le deuxième phrase la probabilité se calcule p(Q ~ R2) = p(T1) * p_lien(T1,T34) * p(T34) * p_lien(T34,T2) * p(T2). La plus longue sous-chaîne de R2 (T34) étant plus courte que celle de R1 (T234) elle a une probabilité inférieure à celle-ci. On supposera que les probabilités de liens sont comparables. Les deux modules syntagmatiques implémentés actuellement donnent des poids identiques que nous noterons p_lien. On a donc avec cette approximation

$$\frac{p(Q\ \ R1)}{p(Q\ \ R2)} = \frac{p(T234) * p(T1) * p_{lien}}{p(T1) * p(T34) * p(T2) * p_{lien}{}^2} = \frac{p(T234)}{p(T34) * p(T2) * p_{lien}} .$$

et comme p(T234) / p(T34) > 1, on a : p(Q ~ R1) / p(Q ~ R2) > 1 d'où p(Q ~ R1) > p(Q ~ R2)

Seuil :

Dans le cas où la question comporte peu de termes, la mesure de LCC est peu fiable. En effet, si la question comporte seulement deux critères, la présence de ces deux critères dans un passage candidat, à proximité l'un de l'autre, donnera le score maximal, 1, alors que la pertinence sémantique ne semble pas aussi assurée. Pour éviter ce problème nous avons proposé une nouvelle version de ce module nommée LCC_seuil, qui ne détecte une plus longue sous-chaîne que si celle-ci est de contient au moins 3 critères.

Respect de l'ordre des termes

Les tests du module de LCC avec seuil ont montré que celui-ci apparie beaucoup de chaînes non pertinentes en raison de l'absence d'ordre des termes dans la chaîne. Pour cette raison, nous avons créé une troisième variante de ce module imposant en plus d'un seuil un certain respect de l'ordre et de la continuité des termes de la question.

L'algorithme que nous avons utilisé reprend la notion de sous chaîne du précédent. Elle autorise la présence de mots vides et de mots pleins intermédiaires de la même façon. Par contre, elle impose que les termes apparaissent dans l'ordre des termes de la question en interdisant la suppression d'un terme et en autorisant l'insertion d'un terme que nous nommerons l'impureté.

Ainsi si l'ordre d'apparition des termes dans la question est (T1 T2 T3 T4), seront admises les chaînes T1 T2 T4 T3; T1 T4 T2; T1 T3 T2; mais pas les chaînes T1 T3 T4 ni T1 T4 T3 T2. La chaîne T1 T2 T4 n'est pas autorisée, dans ce cas c'est la chaîne T1 T2 qui sera sélectionnée.

La fonction d'évaluation de la probabilité est toujours le rapport entre le nombre de critères de la chaîne en excluant l'impureté et le nombre de critères de la question.

4.2.1.4 Clone du système QALC

Nous indiquerons ci-après comment le programme permet de copier le système de pondération

des passages(phrases) de la chaîne de QR. Cette mise en relation de la chaîne de QR permet de tester les deux modules dans le cadre unifié de notre formalisme. Elle nous permettra de comparer les performances de notre système d'extraction de justifications à celui d'un système de questions-réponses existant.

Le poids des documents du système QALC est un bon exemple des limitations du système. Cependant, une implémentation proche de l'identique peut être réalisée, malgré la forte différence du module d'appariement centrale.

La différence vient de ce que le système QALC s'appuie sur une somme de poids partiels. Alors que notre système s'appuie sur un produit de probabilités. Par chance, la somme de poids peut être remplacée par le produit des exponentielles de ces poids, selon la formule suivante :

```
Soit Pmax une borne supérieure des poids obtenus par un module
QALC.

Conversion d'un poids en probabilité :
Cout := Pmax – poids
proba = exp(–cout)

Conversion d'une probabilité en coût:
si proba = 0, alors
 cout := +INF
sinon
 cout := – log(proba);
finsi

Conversion d'un coût en poids:
poids := Pmax – cout
```

4.2.1.5 Termes manquants

L'absence d'un terme est gérée par le système comme la présence d'un pseudo-terme possédant une probabilité très basse de pertinence sémantique. Ainsi, l'algorithme ne sélectionnera ce terme que si le passage n'en propose pas d'autres ou que ceux-ci sont trop éloignés.

Nous avons assigné à tout terme manquant une probabilité pénalisante de 0,01. De plus, une entité nommée absente sera pénalisée d'une probabilité de 0,0001, considérant qu'il est très rare que la réponse ne soit pas repérée par l'analyse des entités nommées.

Du point de vue de l'implémentation, notons que ce module est indispensable car il assure que chaque critère de la question possède au moins un terme lui correspondant dans le document, ce qui est une précondition au fonctionnement de notre algorithme d'appariement de graphe.

4.2.2 Modules syntagmatiques

4.2.2.1 Intégrer une analyse des relations sémantiques

Notre implémentation ne contient pas actuellement de module d'appariement des relations syntagmatiques sémantique. Pour y remédier, il serait nécessaire d'effectuer une analyse sémantique de la question et de la réponse. Un idéal serait d'utiliser une ressource comme FrameNet pour

reconnaître précisément si les deux couples prédicat-arguments de la question et de la réponse correspondent au même rôle sémantique. Un autre idéal serait d'utiliser un analyseur sémantique pour la question et la réponse. Cependant, nous n'avons pas pour l'instant de telle ressource ni de tel analyseur à disposition pour le français. Or bien que nous ayons travaillé la plupart du temps en anglais, nous ne perdons pas de vue l'idée de conserver à notre programme son adaptabilité à d'autres langues dont la nôtre. En l'absence d'un tel analyseur, nous pourrions nous contenter une analyse syntaxique à laquelle nous appliquerions des traitements pour la rapprocher de la représentation sémantique vue au chapitre précédent.

Le principe de cet algorithme serait de calculer pour deux termes de la réponse, si une relation syntagmatique les relie. Si une telle relation n'existe pas, ce module se taira et laissera aux autres modules moins précis et plus robustes le soin d'attribuer un poids.

La correspondance des relations pourrait être parfaite, si l'analyse donne le même résultat pour la question et la réponse, ou approchée dans certains cas. Par exemple, une relation de la question de type cplt_v<preposition>(n1,n2), avec n1 est un nœud verbe et n2 un nœud nom, pourrait se voir apparier une relation cplt_v utilisant une préposition différente.

Nous devons avouer que la limite de cette approche tient dans la faiblesse de profondeur sémantique et nécessiterait encore un développement. Le système par exemple ne semble pas apte à reconnaître une variation entre un verbe et sa nominalisation, lorsque « manger des pommes » se transforme en « la consommation de pommes », ce qui se traduirait dans le modèle comme une règle de paraphrase entre les formes

```
objet(verbe,nom) → prep<de>(nomin(verbe),nom)
```

avec *nomin* une fonction lexicale de nominalisation.

Prémices d'implémentation

Dans le cadre du projet conique, nous avons implémenté avec l'aide du MoDyCo un outil de rattachement des relations temporelles pour le français. Son implémentation utilise une reconnaissance d'expressions temporelles développée par l'équipe de MoDyCo prenant en compte la granularité de l'expression (Année,Mois,Jour) et le type d'intervalle (pendant,avant,de x à y) dans les questions et les passages correspondants.

Notre algorithme repère l'événement de temps auquel se rattache la proposition en suivant l'algorithme décrit dans le chapitre 3, section 3.3.1.4.

Précisons que cet événement choisi peut être référé par un verbe, comme dans (a **marché** sur la lune) ou par un nom (la **démolition** du mur de Berlin) et que le complément de temps peut être soit la réponse attendue à la question (« **Quand**... », « **En quelle année...** ») ou un complément de temps présent dans la question.

La dernière étape consiste en la vérification du rattachement des deux éléments, le verbe et son complément, dans la réponse. Pour ce faire, nous n'avons pas utilisé de critère syntaxique, mais à nouveau une mesure de distance linéaire, considérant que les deux éléments étaient liés s'ils étaient présents à une distance d'au plus N mots. Les fenêtres utilisées étaient de 10 ou 20 mots.

Cette mesure a été intégrée dans la campagne AVE 2006, comme critère d'entrée d'un arbre de décision. Il serait intéressant d'adapter cet outil à l'anglais pour l'intégrer à notre système.

4.2.2.2 Module de distance entre phrase

Nous avons implémenté un premier module pour donner une baseline sur l'appariement multiphrase. Ce module donne à deux termes d'un document une probabilité fondée sur le nombre de phrases de décalage. Les poids attribués sont les suivants :

```
Soit d la distance entre les deux phrases (différence des
positions des phrases dans le document)
proba := 0,8/(d²+1)
```

De cette façon, la probabilité de lien de termes d'une même phrase est égale à 0,8. La probabilité diminue rapidement avec l'éloignement.

```
d=1 -> proba=0,4; d=2 -> proba=0,16; d=3 -> proba=0,08 ..
```

4.2.2.3 Module mono-phrase avec pondération par densité linéaire

Nous avons implémenté un module reproduisant la mesure de proximité sémantique de la chaîne QALC.

Cette pondération suit l'algorithme suivant :

```
pdsdist := 0
Pour chaque couple de termes (tq1, tq2) de la question
Faire
 s'il existe des correspondants (tr1, tr2) dans la réponse tels
que tr1 et tr2 soient au plus séparés par 1 mot
 alors
      pdsdist := pdsdist + 0,02
 finsi
finpour
```

L'implémentation de la chaîne QALC calcule ainsi une grandeur pour la totalité de la phrase. Or, notre système d'appariement effectue un appel de fonction par couple de termes. Ce découpage semble facile à effectuer étant donné que le poids total se décompose comme une somme de sous-poids correspondant chacun à un couple de termes.

Il reste cependant une difficulté : en effet l'algorithme de QALC attribue un poids en testant tous les liens possibles entre deux termes de la question, c'est-à-dire N (N-1) liens avec N le nombre de termes de la question. Or, notre algorithme représente la justification sous forme d'un arbre, ce qui signifie que le nombre de liens intégrés sera au plus de (N-1).

Pour obtenir la même mesure, nous avons été contraints de modifier l'algorithme de la façon suivante :

```
Fonction distance(terme t1, terme t2)
 retourne le nombre de mots ou signes de ponctuation
s'intercalant entre t1 et t2.
Fin fonction


Fonction poids(terme t1, terme t2)
 d := distance(t1,t2);
 Si d > 1, alors retourner 0
 sinon
 poids := 20
```

```
 pour chaque terme t3 différent de t1 et t2, tel que
distance(t1,t3) < = 1 et distance(t2,t3) < = 1
 faire poids := poids + 10
 finpour
 finsi
 retourner poids
Fin fonction
```

4.3 Illustration de l'algorithme

Dans cette partie, nous illustrons le fonctionnement de l'algorithme par des exemples. Nous mettrons l'accent principalement sur les principes de granularité variable avec la mise en collaboration de différents modules de paraphrase.

4.3.1 Versions du système

Nous avons décidé de tester la collaboration des différents modules de Paraphrase en comparant les résultats de plusieurs versions de notre système dans lesquels seront utilisés plus ou moins de modules. Afin de pouvoir décrire de façon succincte les différentes combinaisons de modules effectuées, nous utilisons la nomenclature suivante :

Les versions de notre système utilisent un noyau commun, composé des modules suivants :

- QALC_TermLinker_Multiphrase : un lieur de termes permettant de relier des termes dans la même phrase ou dans des phrases distantes. Il permet d'extraire des justifications étendues sur plusieurs phrases, éventuellement séparées par d'autres phrases qui seront sautées.
- TermesManquants : un paraphraseur nécessaire au bon fonctionnement du programme, qui fixe la probabilité de pertinence accordée à un terme absent.
- QALC_ReponseEN : Un paraphraseur spécialisé dans le repérage des entités nommées du type attendu, utilisé uniquement dans le cas des questions attendant une réponse entité nommée.

Ce noyau est dit Noyau multiphrase(abrégé Multi) puisqu'il permet de repérer des justifications s'étendant sur plusieurs phrases. Pour effectuer des comparaisons, nous avons également utilisé un noyau monophrase(abrégé Mono), obtenu en substituant le module QALC_TermLinker au module QALC_TermLinker_Multiphrase. À ce noyau pourront être ajoutés un ou plusieurs des modules paraphraseurs suivants :

- TermesSansVariation(Fastr0) : Un module de baseline restreignant les appariements du module Fastr aux termes reproduits à l'identique ou possédant même lemme et même catégorie grammaticale. Les termes correspondant à une entité nommée sans variation sont également repérés par ce module.
- Fastr(F) : Un paraphraseur utilisant les termes reconnus par Fastr et les poids attribués par Fastr pour établir une probabilité sur des critères sémantiques.
- LCC, LCCseuil(LS), LCCseuilordre(LSO) : Différentes versions d'un paraphraseur utilisant les termes reconnus par Fastr pour reconnaître des morceaux de phrases de la réponse similaires à la question, c'est-à-dire des paraphrases supposées.

4.3.2 Variation de la justification suivant les modules utilisés

Un grand intérêt de notre système est de permettre d'intégrer différents modules afin de produire une justification de la réponse qui variera selon les informations qu'elle possède. Illustrons cette variation de la justification par l'exemple de la question 1444 : « What female leader succeeded Ferdinand Marcos as president of the Philippines? »

Nous nous intéresserons aux classements rapportés par trois versions de notre système : Multi+F, Multi+F+LS et Multi+F+LSO. Les trois premières réponses rapportées par ces différents systèmes sont les mêmes mais classées dans un ordre variable. Les exemples suivants ne sont pas liés à une version particulière du système. Les termes en gras sont ceux qui constituent la justification et que l'on espère voir repérés par les systèmes :

R1 : Some people here fear that under Estrada s relaxed style of leadership , the **Philippines** is drifting back into corruption and cronyism , a hallmark of the country s deposed dictator , **Ferdinand Marcos** , and an occasional weakness of his **successors** , **Corazon Aquino** and Fidel Ramos .
Since taking office in June , Estrada has helped nudge one of his main campaign contributors into the top job at San Miguel Corp. , the biggest company in the Philippines .

R2 : MANILA , August 21 (Xinhua) -- The Joseph Estrada administration Friday expressed willingness to reopen the Aquino Galman double murder case if a crony of the late President Ferdinand Marcos can provide new evidence to the case . In a press briefing , presidential spokesman Fernando Barican told reporters that it has been the judicial practice not only in the **Philippines** but in other countries that cases are reopened whenever new evidence providing a new twist is presented . [...] Aquino s wife , **Corazon Aquino** , **succeeded Marcos as the president** .

R3 : In 1986 , President **Ferdinand E. Marcos** fled the **Philippines** after 20 years of rule in the wake of a tainted election ; **Corazon Aquino** assumed the **presidency** .

Les rangs obtenus sont les suivants. Comme on le voit, l'ordre obtenu pour la version Multi+LS diffère des deux autres.

	Multi+F	**Multi+LS**	**Multi+LSO**
R1	23	44	25
R2	45	38	54
R3	118	60	76

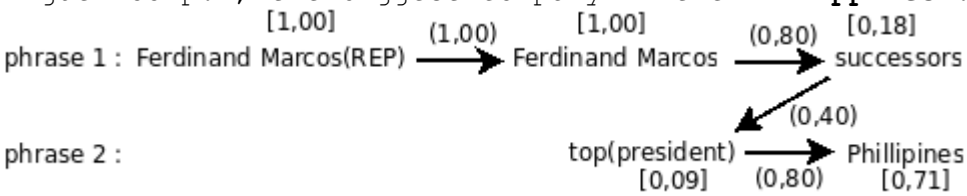
Tab. 8: Rangs des bonnes réponses pour différentes versions du système

Voyons à présent les différentes justifications obtenues pour R1. Dans nos exemples, la ligne J donne la structure de la justification rapportée par le système. Les probabilités entre parenthèses représentent les probabilités des liens syntagmatiques. Les liens entre crochets représentent la probabilité d'appariement au critère de la question correspondant.

On notera que l'entité nommée réponse(Corazon Aquino) est mal reconnue à cause de la présence d'une entité de même type(Ferdinand Marcos) dans la question. Il semble possible de

corriger le problème en interdisant que la réponse soit superposée avec un terme de la question.

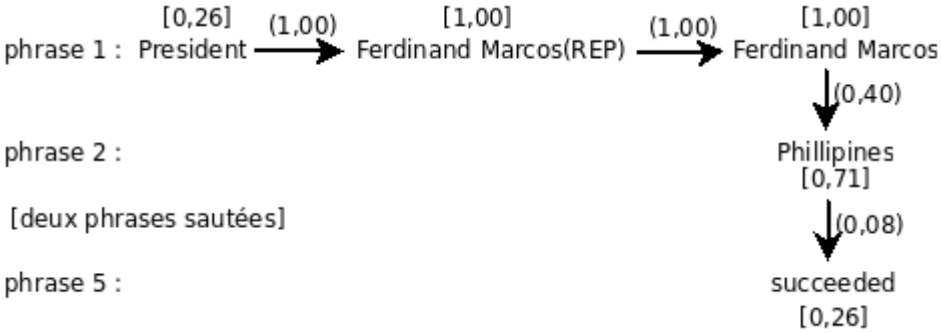
Q1444 : What **female leader succeeded Ferdinand_Marcos** as **president** of the **Philippines**?
Version=Identique pour toutes les versions
R1 : Some people here fear that under Estrada s relaxed style of leadership , the Philippines is drifting back into corruption and cronyism , a hallmark of the country s deposed dictator , ***[REP Ferdinand Marcos]*** , and an occasional weakness of his **successors** , *[REP Corazon Aquino]* and Fidel Ramos .
Since taking office in June , Estrada has helped nudge one of his main campaign contributors into the ***top****{president}* job at San Miguel Corp. , the biggest company in the **Philippines** .



J :
Probabilité totale : $2,89.10^{-07}$

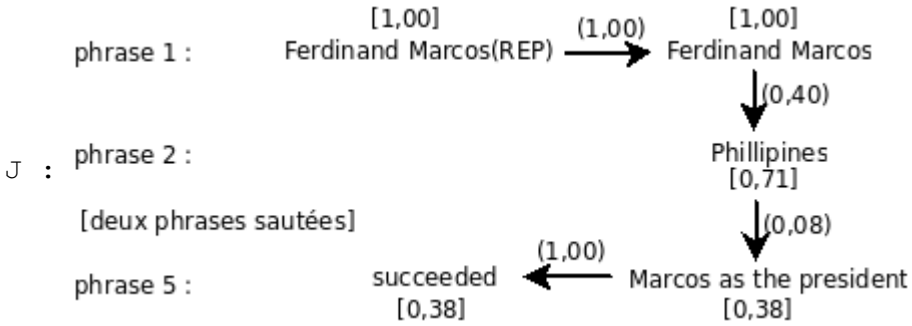
La faible probabilité s'explique par le fait que plusieurs termes ne sont pas reconnus (female, leader) qui infligent chacun une probabilité malus de 10^{-2}.

Q1444 : What female leader **succeeded Ferdinand_Marcos** as **president** of the **Philippines**?
Version=Multi+F
R2 : VINCENT MANILA , August 21 (Xinhua) -- The Joseph Estrada administration Friday expressed willingness to reopen the Aquino Galman double murder case if a crony of the late **President** Ferdinand Marcos can provide new evidence to the case .
In a press briefing , presidential spokesman Fernando Barican told reporters that it has been the judicial practice not only in the **Philippines** but in other countries that cases are reopened whenever new evidence providing a new twist is presented .
[...]
Aquino s wife , Corazon Aquino , **succeeded** Marcos as the president .



J :
Probabilité totale : $1.54.10^{-07}$

version = Multi+F+LS, Rang=38



Probabilité totale : 3,20416323332903.10^{-07}

« succeed » et « Marcos as president » forment une même LCC qui a été divisée en deux parties car le système admet seulement l'appariement de fragments continus de la question. Toutefois, le système mémorise que les deux fragments correspondent à une même LCC dans la réponse, ce qui permet au module lieur d'accorder une probabilité plus élevée(1.00 contre 0,89) au lien syntagmatique qui les unit. De plus, le repérage de la LCC permet de relever le score du verbe « succeed » de 0,26 à 0,38.

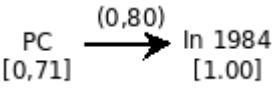
La version Multi+F+LSO, qui présente un repérage plus strict des LCC, ne permet pas de repérer la chaîne « succeeded Ferdinand Marcos as president », car le terme Ferdinand est absent de la chaîne, or le module LSO interdit la suppression de termes. Par conséquent, la justification produite par cette version module ne diffère pas de celle de la version Multi+F.

4.3.3 Variations sémantiques

La reconnaissance de variations sémantiques permet de faire remonter des bonnes réponses. Comparons les versions du programme Multi+Fastr0, qui n'accepte pas de variation sémantique, et Multi+Fastr. Pour la question 1549 : « What year was the PC invented? » on obtient :

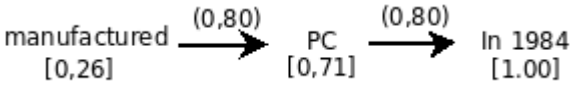
1549 : What year was the **PC invented**?

Version=Multi+Fastr0, Rang=15
R : The first PC manufactured by IBM in 1981 was named the IBM PC , and in the consumer world , **PC** has also come to mean ` ` not Macintosh . ' ' Apple Computer introduced its Macintosh line ***in 1984*** .



J :
Probabilité totale : 0,00570

Version=Multi+Fastr, Rang=4
R : The first PC **manufactured**{invented} by IBM [REP in 1981] was named the IBM PC , and in the consumer world , **PC** has also come to mean ` ` not Macintosh . ' ' Apple Computer introduced its Macintosh line ***in 1984*** .



J :
Probabilité totale : 0,118

La reconnaissance d'un terme supplémentaire a permis de faire remonter le passage contenant la réponse du rang 16 au rang 4 en substituant à la pénalité pour terme absent (10^{-2}) un coût d'appariement de 0,2079 constitué d'une variation sémantique(*invent* → *manufacture* : 0,2599) et du coût du lien syntagmatique entre « manufactured » et « PC ».

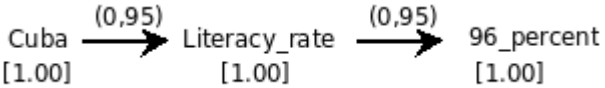
On notera que la réponse choisie (1984) n'est pas la bonne. Ceci ne remet pas en cause le classement du passage réponse, puisque cette entité nommée a permis les mêmes mêmes calculs que la vraie réponse (1981). Une telle erreur aurait pu être évitée de deux façons. La première solution consiste à corriger la segmentation en phrases de la collection initiale qui ne tient pas compte de la particularité anglaise consistant à placer la ponctuation finale à l'intérieur des guillemets. L'autre solution consisterait à modifier le module d'appariement pour rendre compte de la différence de probabilité de rattachement de deux termes d'une phrase en fonction de leur distance dans la phrase.

4.3.4 Réponses monophrases et multiphrases

Notre système permet de ramener des passages justifiant entièrement la réponse, rapportant plusieurs phrases si cela est nécessaire. Citons quelques exemples de justifications monophrases obtenues.

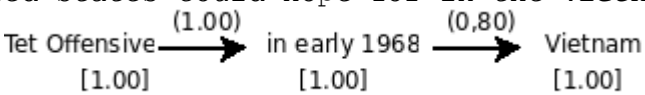
```
Q1569 : When did the Vietnam War end?
R : The Vietnam War ended in 1975 .
```

```
Q1532 : What is the literacy_rate in Cuba?
Version=Multiphrase+Fastr+LCCseuilordre, Rang=1
R : Cuba s literacy_rate , [REP 96 percent] , ranks third in Latin
America , behind Uruguay and Argentina , according to United
Nations statistics .
```




```
J :
Probabilité totale : 0,893
```

```
Q1555 : When was the Tet_offensive in Vietnam?
Version=Multiphrase+Fastr+LCCseuilordre, Rang=1
R : When CBS anchor Walter Cronkite aired a special on the
Tet_Offensive [REP in_early_1968] , he concluded that the best the
United States could hope for in the Vietnam War was a stalemate .
```

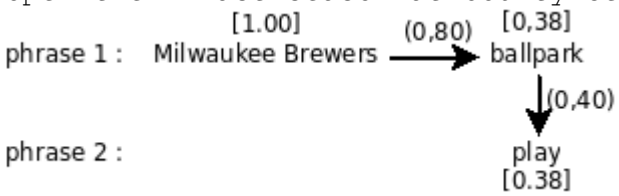



```
J :
Probabilité totale : 0,800
```

On notera que dans ce second exemple, l'éloignement du terme Vietnam donne à la justification un moins bon score que dans l'exemple précédent. Voici en comparaison quelques exemples de réponses multiphrases rapportées par le système.

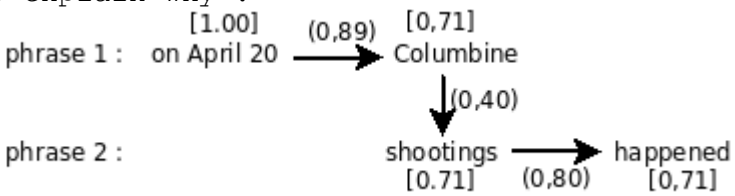
```
Q1524 : What is the name of the ballpark that the
Milwaukee_Brewers play at?
Version=Multi+F+LSO, Rang=1
R : MILWAUKEE ( AP ) -- Hall of Famer Warren Spahn will return
to the mound at [REP County Stadium] to throw out the ceremonial
```

first pitch during the **Milwaukee_Brewers** ' last opening day game at the **ballpark** .
The Brewers are scheduled to **play** the Chicago Cubs on April 16 to open their last season at County Stadium .



J :
Probabilité totale = 0,0470

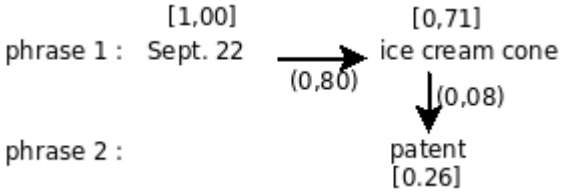
Q1407 : When did the **shootings** at **Columbine happen**?
Version=Multi+F+LSO, Rang=3
For **[REP on April 20] Columbine** was invaded not just by two troubled students , but also by nearly 1000 journalist from around the world , so many that they quickly swamped Denver s telephone lines .
In a drill made all too familiar by previous school **shootings** , the news media not only set about reporting what had **happened** , but also asking the victims , witnesses and their families to help explain why .



J :
Probabilité totale = 0,103

Les justifications rapportées peuvent éventuellement comporter une information reliée par thématisation, comme c'est le cas de « ice cream cone » dans le cas suivant, où les deux phrases de la justification sont distantes de 2 autres phrases :

Q1561 : When was the **first patent filed** on the **ice_cream_cone**?
Version=Multi+Fastr, Rang=3
R : **[REP Sept. 22]** : The birthday of the **ice_cream_cone** .
[...]
On *[REP Sept. 22 , 1903]* , he applied for a **patent** for a *cone* shaped mold that he made out of paper to hold his tasty delicacy .



J :
Probabilité totale : $1{,}184.10^{-6}$

Pour cet exemple, la justification obtenue semble trop petite du fait que les critères dédoublés entre la phrase 1 et la phrase 2 n'apparaissent que dans la première phrase. Il pourrait sembler plus naturel de faire apparaître les deux occurrences dans la justification. De plus, le choix de l'entité nommée « Sept. 22 » au lieu de « Sept. 22, 1903 » n'est pas le plus judicieux mais rien dans notre algorithme ne permet de trancher entre les deux.

4.3.5 Variations dans la taille des passages

Notre système a la particularité de chercher à extraire des passages contenant exactement la justification, sans phrases manquantes ni superflues. Nous avons évalué la qualité des justifications selon les notions d'exactitude et de correction, en nous inspirant de la validation des réponses courtes dans la campagne CLEF (cf section 1.2.2).

Pour ce faire, nous avons validé manuellement les justifications présentes dans les passages réponses rapportés par la version LCCseuilordre de notre système pour les 200 premières questions de la campagne TREC11. Nous avons annoté les passages justificatifs classés au pire au rang 10 et contenant le patron de la réponse. Le jeu d'étiquettes était le suivant :

- Correct (C) : Toutes les informations nécessaires sont présentes et le passage ne contient pas de phrase superflue.
- Non-exact (X) : Le passage justificatif est trop long ; il contient une phrase superflue.
- Missing (M) : Nous avons ajouté cette catégorie dans le cas où le passage réponse justifie presque la réponse. Ceci nous permet de faire l'économie de rechercher si l'information manquante est présente dans le document.
- Unsupported (U) : Le passage rapporté ne permet pas de justifier suffisamment la réponse
- Incorrect (I) : Le patron réponse a été reconnu à tort.

L'unité utilisée pour évaluer la longueur est celle de la phrase, ce qui fait, compte tenu de la longueur de certaines phrases journalistiques, que certains passages rapportés peuvent être bien trop étendus par rapport à l'endroit où se trouve l'information utile. Notre algorithme de sélection de passages prenant actuellement la phrase pour unité, nous avons choisi de peu sanctionner notre système dans le cas des informations superflues à l'intérieur d'une phrase participant à la justification, sauf dans le cas où l'information supplémentaire est d'une longueur gênante et n'apporte aucun éclaircissement à la justification. Notons également que la présence d'un peu plus de contexte peut aider le lecteur à vérifier que son interprétation de la réponse est correcte, facilitant ainsi la vérification que la réponse est pertinente.

C	**X**	**C+X**	**M**	**C+X+M**	**U**	**I**	**U+I**	**Total**
214	66	280	9	289	55	7	62	**353**
61%	19%	80%	2%	82%	16%	2%	18%	**100%**

Tab. 9: Validation des justifications pour 200 questions de TREC11

Ces données illustrent le fait que l'utilisation pour la validation automatique surévalue le nombre de bonnes réponses trouvées, 18% des passages réponses ont été considérés pertinents à tort par cette validation automatique (étiquettes U+I).

Si l'on se concentre uniquement sur les vraies justifications (cf tableau 10), on obtient qu'il y a 23% de justifications trop longues(X), 3% de justifications trop courtes(M) et 74% de justifications de taille convenable(C). Ce bon score s'explique en partie par le choix de la phrase comme unité minimale pour notre évaluation.

C	X	M	Total
214	66	9	379
74%	23%	3%	100%

Tab. 10: Proportion de justifications trop longues et insuffisantes

Nous remarquons que notre algorithme de sélection des justifications a tendance à générer des passages trop grands. Les erreurs les plus fréquentes provoquant des passages trop étendus sont :

- La reconnaissance à tort d'un terme dans cette phrase.
- La présence dans une phrase du type général, qui, bien que n'étant pas relié à la justification aide à rendre la sélection plus fiable, car le terme *confection* n'est pas reconnu :

```
Q1539 : What is Ronald Reagan's favorite candy?
In 1980 , presidential candidate Ronald Reagan gave the
company better marketing than money could buy by naming the
Jelly Belly his favorite confection .
By 1981 , the Herman Goelitz Candy Co. was backlogged 76
weeks in jellybean orders .
```

- Les erreurs de segmentation des phrases notamment en raison de l'usage américain de déplacer la ponctuation de fin de phrase à l'intérieur des guillemets lorsque la phrase finit par une citation.
- Les phrases trop longues qui traitent d'informations diverses, notamment les énumérations de faits séparés par des points-virgules

Enfin, nous avons mesuré la taille des passages justificatifs rapportés par notre programme (version LCCseuilordre) sur les passages jugés Correct, Non-exact et Missing. Nous obtenons les résultats suivants :

Nb phrases	1	2	3
Nb passages	183	85	21
pourcentage	63%	30%	7%

Tab. 11: Nombre de phrases par justifications

Nous avons également mesuré l'étendue des passages c'est-à-dire la distance entre la première et la dernière phrase sélectionnées. Nous obtenons les résultats suivants :

Étendue	1	2	3	4	5	6	7	8	9
Nb passages	183	61	23	7	5	2	5	2	1
pourcentage	63%	21%	8%	2%	2%	1%	2%	1%	0%

Tab. 12: Étendue des passages

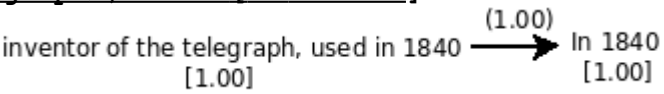
Nous en concluons que notre système permet de trouver l'étendue des justifications avec une assez bonne précision (74% à 77% de passages correctement délimités) et qu'il permet également de rapporter des justifications fortement étendues dans les passages, ce qui confirme les mesures sur corpus vues en section 3.5.2.

4.3.6 Remontée de documents grâce au LCC

L'introduction du module de détection des plus longues sous chaînes communes permet de détecter et de faire remonter des passages reformulés presque à l'identique. Pour illustrer ceci, nous comparons les sorties du programme utilisant le Noyau Multiphrase tout seul, avec celles du programme utilisant Noyau + LCC_seuil.

Le module LCC permet de repérer des réponses reprenant la formulation de la question à l'identique. Dans les exemples, les plus longues chaînes communes seront repérées par un soulignage. Les critères de la question qui composent la chaîne sont indiqués en gras.

```
Q1400 : When was the telegraph invented ?
Version=Multiphrase+Fastr+LCCseuil
Rang=1
R1 : Globe Wireless , a marine communications company , signed
off using the same line Samuel F.B. Morse , inventor of the
telegraph , used [REP in 1844] : ` ` What hath God wrought ?
```



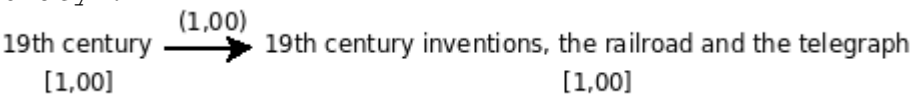

```
J :
Probabilité totale : 1
```

On notera que la réponse est dupliquée, apparaissant dans la plus longue chaîne commune et comme critère isolé. Il s'agit d'un léger bug qui n'a pas d'influence sur l'ordonnancement des passages réponses produits par le système. La justification devrait être composée d'un seul fragment de type LCC, de probabilité 1, couvrant tous les critères de la question. On notera également qu'une probabilité de 1 semble surévaluée, du fait que le module de LCC accepte l'inclusion de mots pleins, ici « used », qui peuvent altérer le sens de la chaîne, comme dans le cas suivant, où une LCC serait repérée à tort :

```
April 2nd, 1872. Samuel FB Morse, American painter and inventor
of the telegraph (born in 1791) dies on April 02, 1872[14]
```

La réponse au rang 2 est inexacte car trop imprécise. Cependant, elle est totalement justifiée :

```
Q1400 : When was the telegraph invented ?
Version=Multiphrase+Fastr+LCCseuil
Rang=2
R2 : To a great degree , modern politics is still shaped by a
couple of [REP 19th century] inventions , the railroad and the
telegraph , which allowed candidates to vanquish the challenges
of distance and appeal to voters with a revolutionary
immediacy .
```




```
J :
Probabilité totale : 1
```

Si l'on supprime le module LCC_seuil, la réponse la mieux classée est toujours R1, mais elle descend en 4ème position. La réponse R2 reste en 2ème position. La structure de la justification obtenue pour R1 est alors la suivante :

```
Q1400 : When was the telegraph invented ?
```

14 *Trouvé sur www.inhistorytoday.com/192494*

R1 : Globe Wireless , a marine communications company , signed off using the same line Samuel F.B. Morse , **inventor** of the **telegraph** , used **[REP in 1844]** : ` ` What hath God wrought ?
Version=Multiphrase+Fastr
Rang = 4

J :
Probabilité totale : 0,456

D'autres passages apparaissent alors comme mieux classés, car ils contiennent tous les termes de la question bien que ceux-ci soient plus dispersés, comme :

R'1 : **In 1867** , Edward A. Calahan **invented** the stock ticker , which outdid the **telegraph** as a mechanism for transmitting stock price information automatically and continuously .

R'3 : **In 1824** , when the Marquis de Lafayette visited the church , he remarked , ` ` Yes , that is the man I know and more like him than any other portrait . ' ' Washington , 67 , died Dec. 14 , 1799 , some 45 years before the **invention** of the **telegraph** .

4.4 Validation des modules et résultats

Nous avons testé les résultats des variantes de notre système avec les données de la campagne d'évaluation TREC11. Nous utilisons ces résultats pour donner une idée de l'impact des différents modules rassemblés dans notre système.

4.4.1 Intégration du programme dans la chaîne

Dans les campagnes d'évaluation TREC, le but est de ramener un maximum de courts passages contenant la réponse et de classer ceux-ci en premier dans la liste de documents rapportés. Les résultats sont reportés au tableau 13. Comme nous l'avons vu au chapitre 1, la mesure utilisée dans TREC11 pour attribuer un score à cette tâche est le MRR (colonne 1), qui attribue pour chaque question un score de 1 si la première bonne réponse est trouvée en première position, de 1/2 en seconde, de 1/3 en troisième, etc. Le MRR est la moyenne de ces scores sur toutes les questions, il est donc compris entre 0 et 1.

Nous reprenons dans cette section les différentes combinaisons de modules présentées à la section 4.3 et les testons sur la campagne TREC11. Certains passages réponses étaient ramenés plusieurs fois par les systèmes en raison de la présence de doublons dans la collection de passages, comme expliqué à la section 1.2.3.1. Notons que ce phénomène n'est pas anecdotique puisqu'on trouve environ 7% de passages dupliqués dans les résultats des systèmes. Ceux-ci sont gênants pour le calcul du MRR puisqu'ils détériorent les scores obtenus et provoquent des variations de rang excessives selon qu'un passage vient se classer au dessus ou en dessous d'un lot de passages dupliqués. Pour cette raison, nous avons supprimé les doublons parmi les passages réponses rapportés.

Dans le tableau des résultats nous adjoignons à ce score le nombre de questions qui possèdent une réponse au premier rang (colonne 2), dans les cinq premières places (colonne 3), dans les dix premières (colonne 4).

Système	MRR	1ère place	<=5e place	<=10e place
QALC	0,369	**139**	238	272
Mono + F	0,347	128	227	253
Multi + Fastr0	**0,381**	**138**	**253**	**284**
Multi + F	**0,384**	**141**	**258**	**281**
Multi + LCC	0,257	88	182	209
Multi + F + LS	0,375	133	**259**	**284**
Multi + F + LSO	0,369	129	**255**	**285**

Tab. 13: Bilan général des différents systèmes proposés

Nous commenterons en détail ce tableau dans la suite de cette section.

Afin de mieux voir quels systèmes sont plus semblables, nous avons mesuré pour chaque couple de systèmes le nombre de questions pour lesquelles les deux classements présentent une différence importante du rang de la première réponse.

Nous avons jugé qu'une augmentation de rang était importante si, étant donnés les rangs de deux classements r_1 et r_2 tels que r_2 est meilleur que r_1 ($r_2 < r_1$), on a $r_1 - r_2 \geq 0{,}7 \, . \, r_2$, soit encore $r_1 / r_2 \geq 1{,}7$. Symétriquement, une diminution importante sera prise en compte si $r_2 / r_1 \geq 1{,}7$. Avec cette formule, les plus petites variations de rang prises en compte sont : (2→1, 4→2, 6→3, 7→4, 9→5 ..., 100→58,...)

Voici les variations trouvées au format : nombre de variations (+nombre de phrases mieux classées dans le classement de la ligne par rapport à celui de la colonne, -nombre de phrases moins bien classées) :

	QALC	Fastr0	F	F + LS	F + LSO	Mono F
Fastr0	202(+125,-77)					
F	200 (+126,-74)	50(+27,-23)				
F+LS	209(+126,-83)	101(+49,-52)	57(+26,-31)			
F+LSO	203(+118,-85)	100(+46,-54)	51(+20,-31)	47(+20,-27)		
Mono+F	189(+96,-93)	111(+34,-77)	112(+29,-83)	140(+49,-91)	144(+53,-91)	
Mono+F+LS	205(+95,-110)	149(+48,-101)	146(+39,-107)	143(+45,-98)	111(+28,-83)	50(+18,-32)

Tab. 14: Différences entre les systèmes

On voit dans ce tableau que les systèmes les plus similaires sont les groupes {F+LSO,F+LS,F} avec au plus 57 différences importantes, les deux systèmes monophrases :{Mono+F, Mono+F+LS} avec 50 différences importantes, et {F, Fastr0} avec 50 différences importantes.

Le système QALC se démarque des autres par un grand nombre de changements dans le classement, ce qui se comprend puisque son architecture est différente. Le système qui lui est le plus proche est le système Monophrase + Fastr, ce qui semble naturel puisque c'est le système qui utilise les modules les plus similaires.

4.4.2 Comparaison au système QALC

Nous avons comparé nos systèmes à la chaîne QALC. Le système QALC utilise des modules assez similaires aux nôtres, notamment la même reconnaissance des entités nommées que celle de ReponseEN, la même détection des termes par Fastr, mais avec une méthode de pondération différente, et une mesure de densité semblable à QALC_TermLinker_Monophrase.

Le système QALC diffère principalement du nôtre par la façon d'intégrer des connaissances. Nous avons donc jugé utile de comparer les performances de notre système à celles de ce système déjà testé dans de nombreuses campagnes. Nous avons utilisé le système d'ordonnancement des réponses de QALC qui ne tient pas compte du LCC. La comparaison la plus juste est donc celle avec notre version Noyau Monophrase + Fastr.

La comparaison en termes de MRR montre que le système QALC donne des résultats meilleurs que la version Noyau Monophrase + Fastr. Toutefois, la version Multiphrase + Fastr se classe au dessus de QALC. Cela indique que la gestion de plusieurs phrases permet de rapporter des réponses supplémentaires.

Dans l'ensemble, les systèmes donnent des résultats comparables, ce qui est encourageant si l'on tient compte que les poids des modules doivent encore être réglés.

4.4.3 Apport du traitement multiphrase

Nous avons testé l'apport d'un module de liaison syntagmatique Multiphrase en comparant les résultats des modules QALC_TermLinker (monophrase) et QALC_TermLinker_Multiphrase pour différentes compositions de modules. Les compositions utilisées sont Noyau + Fastr et Noyau + Fastr + LCCseuil. Cette comparaison donne les résultats suivants :

Système	**MRR**	**1ère place**	**<=5e place**	**<=10e place**
Noyau Monophrase + Fastr	0,347	128	227	253
Noyau Multiphrase + Fastr	**0,384**	**141**	**258**	**281**
Noyau Monophrase + Fastr + LCCseuil	0,331	116	223	251
Noyau Multiphrase + Fastr + LCCseuil	**0,375**	**133**	**259**	**284**

Tab. 15: Comparaison entre systèmes monophrases et multiphrases

Ces mesures montrent que la gestion de plusieurs phrases par le lieur QALC_TermLinker_Multiphrase permet une amélioration relativement importante du MRR.

4.4.4 Apports des modules LCC_seuil et LCC_seuil_ordre

Nous avons testé deux variantes de LCC dans notre algorithme. L'une autorise n'importe quelle succession de termes (module LCC_seuil) et la seconde conserve l'ordre (module LCC_seuil_ordre) comme décrit en section 4.2.1.3. Comme nous l'avions expliqué, ces deux variantes ne traitent que les sous-chaînes contenant au moins 3 critères de la question, afin de corriger l'imprécision de la mesure dans le cas d'une question possédant peu de critères. Rappelons les résultats déjà présentés dans le tableau général :

Système	MRR	1ère place	<=5e place	<=10e place
Multi + F	**0,384**	**141**	**258**	**281**
Multi + F + LS	0,375	133	**259**	**284**
Multi + F + LSO	0,369	129	255	**285**

Tab. 16: Apports des modules de plus longue chaîne commune

Les variantes du module LCC testées détériorent les résultats du système en terme de MRR. Cependant, on note une très légère amélioration du nombre de phrases rapportées jusqu'à la position 10, ce qui indique que ces modules ont fait remonter quelques réponses qui avaient été mal classées précédemment. De plus, il faut noter que 78 bonnes réponses sont remontées, et seulement 31 sont descendues dans le classement. Il serait donc intéressant de poursuivre l'étude pour isoler les cas qui apportent une remontée et corriger la mesure. De plus il semble indispensable de revoir la pondération de ces modules qui ne provient pas actuellement d'une mesure de fréquences sur corpus.

4.4.5 Sélection des réponses courtes

En plus de la sélection de passages justificatifs, notre système permet de localiser la réponse à la question dans le cas où le type attendu de la réponse est une entité nommée. En effet, dans ce cas la réponse constitue un des critères de la question et par conséquent, un des nœuds de la structure de justification, comme nous l'avons vu en section 3.3.2.1.

Nous avons mesuré le MRR sur la sélection de réponses en nous limitant aux 344 questions de type entité nommée. Nous avons supprimé les doublons au niveau de la justification, mais pas au niveau des réponses courtes fournies. Par conséquent, une même réponse courte peut être proposée plusieurs fois si les justifications d'où elles proviennent sont différentes. Ceci tente de tenir compte du fait que même si le programme sélectionne des réponses courtes, il est obligé de fournir les justifications pour permettre à l'utilisateur de vérifier que le système n'a pas fait d'erreur.

Certaines questions posent problème car elles contiennent un terme qui est une entité nommée du même type que le type attendu de la réponse par exemple dans « Who is Tom Cruise married to? », la réponse est de type Personne, tout comme l'entité nommée « Tom Cruise ». Dans ce cas, c'est presque toujours l'entité nommée présente dans les termes de la question qui est sélectionnée comme réponse courte. Ceci détériore le score et nécessiterait un traitement adapté comme par exemple, l'interdiction de sélectionner une réponse qui fasse partie de la question. Pour donner une

mesure précise, nous avons décidé de tester également la sélection des réponses sur les questions ne présentant pas d'entité nommée du type attendu.

Les résultats sont les suivants :

	Questions EN	**Questions EN sélectionnées**
MRR	0,289	0,310

Tab. 17: Problème de sélection des réponses EN

Nous remarquons que dans le cas des entités nommées sélectionnées, le MRR se rapproche de celui de l'évaluation sur des réponses courtes, ce qui montre que la sélection par entités nommées est relativement sûre. Il serait utile d'étendre la sélection de la réponse aux questions n'attendant pas d'entité nommée, et notamment aux questions réclamant un type général.

4.5 Perspectives

Notre travail a permis de mettre en place un système de sélection des documents et d'extraction de la justification opérationnel. Ce système est à l'heure actuelle minimaliste quant au nombre de modules utilisés. Nous devons ajouter certains traitements langagiers rencontrés dans le corpus, notamment le traitement des variations sémantiques plus complexes, la résolution des anaphores, la recherche d'informations ciblées dans des documents externes. Un développement ultérieur du système consisterait à ajouter au système un module de vérification des relations syntagmatiques de la question.

Nous pensons également améliorer la pondération des modules. Voici quelques défauts de la pondération actuelle que nous souhaitons améliorer.

Les probabilités affichées par le programme sont parfois peu vraisemblables : Comme nous l'avons vu dans les exemples(section 4.3.6), certaines probabilités peuvent être surestimées. Nous avons repéré ce problème dans le cas de LCC portant une probabilité de 1 alors que l'algorithme de détection des paraphrases laisse à l'évidence la possibilité de mauvais rattachements..

Dans des cas fréquents, la probabilité totale obtenue pour un passage justifiant la réponse est très faible, celle-ci pouvant fréquemment se situer aux environs de 10^{-7}. On peut voir à cela plusieurs raisons :

- Poids approximatifs : Il est nécessaire de calculer les poids des différents types de variations, paraphrase et rattachement distant par des mesures sur corpus comme présenté en section 4.1.1.2.
- Pénalisation uniforme de l'absence de termes : Le système devrait pénaliser moins l'absence de certains termes, notamment en fonction de leur catégorie grammaticale. En effet, certaines catégories grammaticales comme les adverbes ou les adjectifs sont moins significatives que les noms ou les verbes. D'autre part, pour certaines catégories comme les verbes, la détection échoue fréquemment, soit à cause d'une variation sémantique non détectée, soit parce que cette détection nécessite une inférence.
- Problème lié à l'hypothèse d'indépendance des modules et de la fonction de probabilité qui ne tient pas compte que plusieurs informations peuvent se renforcer mutuellement. Deux voix sont à poursuivre pour ceci :

  1. Augmenter la fiabilité d'un appariement dans le cas où il est effectué de façon

indépendante en utilisant différentes connaissances.

2. Étudier s'il est possible d'augmenter la fiabilité de la reconnaissance d'un terme grâce à son contexte. Ce problème est partiellement traité par l'utilisation de LCC, qui permet d'augmenter le score d'un terme si celui-ci est contenu dans une succession de termes proche de la formulation de la question, mais n'est pas traitée de façon générale.

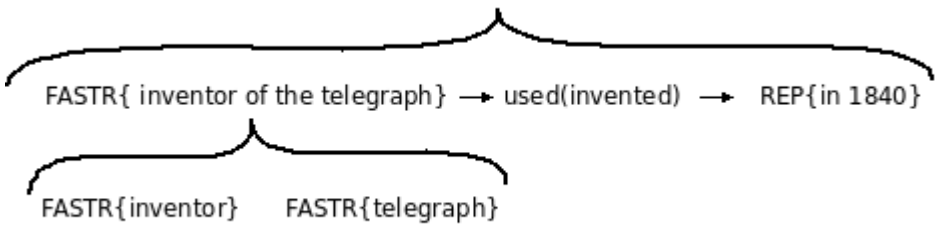
Le calcul des probabilités des différentes détections selon les phénomènes linguistiques qu'elles utilisent, par une mesure de fréquences, pose le problème du choix des éléments sur lesquels effectuer la mesure. Selon qu'on effectue celle-ci sur des éléments plus ou moins filtrés par un programme, on obtiendra une probabilité plus ou moins élevée. On peut envisager plusieurs façons de sélectionner les éléments utilisés pour cette mesure de fréquence :

- Taux d'éléments pertinents parmi les éléments justificatifs sélectionnés par le programme
- Taux d'éléments pertinents sur les passages réponses complets. Cela requiert de décider si l'on doit effectuer la mesure sur tous les documents ou sur le passages réponses portant la réponse uniquement. De plus, il est difficile de décider combien de passages réponses il faut utiliser pour obtenir la probabilité souhaitée.

Nous pensons mettre en œuvre la pondération des modules de façon à conserver les résultats de cette annotation dans le corpus. Cet enrichissement progressif du corpus devrait permettre une automatisation progressive de la tâche de pondération.

En ce qui concerne la présentation de la justification, il nous semble que la sélection de termes apporte une aide pertinente à la vérification de la justification. L'utilisation de modules permettant de repérer une longue chaîne de termes en une seule fois est toutefois problématique. En effet, si ce « pseudo-terme » est sélectionné, la structure de la justification obtenue est alors réduite à un graphe très simplifié pouvant éventuellement contenir un seul nœud représentant la totalité de la justification. Ce résultat a l'inconvénient de masquer les niveaux de granularité plus fine de la justification. Une solution simple de remédier à ce problème consisterait à fournir en sortie du système plusieurs niveaux de granularité. Par exemple :

Q1400 : When was the **telegraph invented** ?
R : .... **inventor** of the **telegraph,** used in **1840** ....



J :

4.6 Conclusion

Dans ce chapitre, nous avons décrit l'implémentation de notre système de questions-réponses fondé sur une formalisation de la justification. Nous avons décrit de nombreux modules collaborant dans une architecture fortement modulaire pour permettre de composer une représentation de la justification.

Ces modules, pour collaborer, ont besoin d'être calibrés sur une échelle commune dans lesquels ils doivent exprimer la valeur de fiabilité qu'ils attribuent aux appariements qu'ils effectuent. Nous avons choisi d'utiliser des probabilités comme échelle commune. Ces probabilités présentent l'avantage d'être mesurables sur corpus en comptant le nombre d'occurrences pertinentes du phénomène dans les justifications.

Nous avons ensuite montré le bon fonctionnement de notre programme qui permet d'améliorer légèrement le score du système de questions-réponses QALC et mis en évidence certaines de ses faiblesses.

Chapitre 5 : Conclusion générale

Bilan

Notre thèse, située dans le cadre de la recherche d'informations précises et en particulier des systèmes de questions-réponses, avait pour but d'expliciter la justification apportée par un passage répondant à une question.

Nous avons pour cela proposé un système d'extraction des justifications permettant de donner une représentation fine de la justification trouvée tout en améliorant les performances de l'étape de filtrage des passages candidats à fournir la bonne réponse.

Dans le premier chapitre, nous avons dressé l'état de l'art des systèmes de questions-réponses et décrit plus particulièrement le fonctionnement du système QALC. Nous nous sommes concentrés sur deux tâches liées à la vérification de la réponse : la validation des réponses fournies par un système de QR et la vérification de l'existence d'une implication entre deux énoncés.

Cette étude nous a permis de noter un écart important entre des méthodes fortement intégrées et structurées nécessitant d'importantes ressources linguistiques et des techniques fondées sur la réunion de traits hétérogènes mis en commun grâce à un système de décision utilisant l'apprentissage artificiel. Nous avons décidé de proposer une approche médiane, visant à conserver au système la capacité à produire une justification structurée, tout en rendant possible l'intégration d'une grande hétérogénéité de traitements linguistiques, de nature, de niveau de granularité et de fiabilité variés.

Dans le chapitre 2, nous avons décrit les ressources, notamment sémantiques, permettant de détecter les phénomènes de variation sémantique dans les documents candidats à apporter une réponse. Nous avons présenté l'utilisation de ressources sémantiques, morphologiques et pragmatiques afin de réduire la distance entre la question et la réponse. De plus nous avons souligné l'existence de phénomènes linguistiques permettant de lier entre elles plusieurs phrases d'un document, voire deux documents distincts.

Nous avons insisté sur la difficulté de trouver certaines variations notamment du fait que certaines variations ne sont pas traitées par les ressources connues ou que ces ressources ont une couverture limitée, notamment en français. Nous en avons déduit la nécessité de faire collaborer plusieurs techniques de détection de variations sémantiques locales ou de rattachement d'informations distantes dans le but d'augmenter les chances de trouver l'information.

Nous avons ensuite, dans le chapitre 3, présenté la construction d'un corpus de questions-réponses et l'étude des phénomènes linguistiques qui s'y présentent. Pour cela nous avons rassemblé un corpus de questions-réponses qui nous a permis d'expliciter la notion de justification, en

proposant une représentation formelle de celle-ci. Le corpus a été rassemblé par une technique de construction que nous avons conçue pour encourager sa richesse sémantique.

La représentation formelle des justifications que nous avions choisie a alors été utilisée pour annoter le corpus et étudier la configuration des justifications en termes de variation sémantique, d'étendue spatiale. Ce corpus annoté nous a permis de mettre en évidence la variété des types de variations sémantiques présents ainsi que de montrer que les justifications sont souvent étendues sur plusieurs phrases.

Dans notre étude, nous avons approximé la composante syntaxique des justifications en ayant recours à l'étiquetage de zones de justification continues contenant les éléments justificatifs liés par une relation syntaxique. La syntaxe pourrait faire l'objet d'un prolongement de l'étude de corpus afin de donner une idée de la variation des relations syntaxiques entre la question et la réponse. Il serait intéressant de compléter avec les liens syntaxiques les structures de justification déjà annotées. Il faut toutefois prendre en compte que l'annotation de relations syntaxiques requiert un travail important.

À partir de la modélisation des justifications, nous avons proposé un algorithme capable d'implémenter les différents phénomènes linguistiques mis en évidence. Ce programme devait d'une part améliorer les performances de l'étape de filtrage des passages candidats à fournir la bonne réponse et d'autre part permettre la mise en évidence de la structure de justification contenue dans ces passages.

Nous avons montré une amélioration de l'étape de filtrage de documents entre terme de MRR grâce au passage à des justifications multiphrases. L'utilisation de modules tels que LCC n'a pas pour l'instant donné de résultat probant. Nous pensons que ces modules requièrent une amélioration de leur pondération.

Informations locales, distantes, manquantes

Le système permet d'extraire des justifications illustrant les différents phénomènes linguistiques étudiés.

Nous avons montré que notre système est capable de bénéficier de la détection des variations sémantiques. Notre système implémente différents types de variations : morphologie dérivationnelle, synonymie, variations complexes de termes composés, paraphrases de fragments de phrases. Il met en évidence la possibilité d'intégrer des entités linguistiques à plusieurs niveaux de granularité (terme simple, terme composé, fragments d'énoncés).

Nous avons également démontré la capacité de notre système à rattacher des informations situées dans des phrases distinctes d'un même passage. Nous avons aussi pris en compte dans notre formalisme la possibilité de faire appel à des documents externes pour compléter les informations. Bien que cette fonction ne soit pas implémentée, l'architecture du système autorise cette intégration. Dans le cas où ces techniques ne permettraient pas de retrouver un des éléments de la justification, notre système est capable d'ignorer les termes de la question faisant défaut.

Les différentes capacités évoquées ci-dessus tentent de répondre au même but : retrouver une justification la plus complète possible de la réponse et la modéliser sous forme d'une structure de graphe.

Flexibilité du système

Nous avons également mis en évidence la flexibilité de notre système puisque les modules peuvent s'ajouter et s'enlever d'une façon naturelle. De plus, l'ajout ou la suppression d'un module

entraîne des transformations des justifications repérées. Nous avons montré notamment que le système peut préférer une occurrence d'un terme de la question à une autre selon les modules utilisés.

L'architecture proposée permet de facilement ajouter ou enlever un module. Nous avons illustré la variation des structures de justifications obtenues en fonction des modules utilisés. Nous avons également montré que l'apport de variations sémantiques de type synonymes améliore le fonctionnement du système.

Perspectives

Le programme réalisé constitue une architecture de base laissant entrevoir de nombreux développements.

Nous souhaitons y intégrer de nombreux traitements linguistiques, tels que le traitement de variations sémantiques plus complexes, la vérification des relations syntaxiques, la résolution des anaphores, la recherche d'informations ciblées dans des documents externes.

Nous souhaitons également mettre en place un système de pondération semi-automatique des modules qui utilisera le corpus de questions-réponses comme référence pour déterminer la pertinence de chaque appariement question-réponse. Les éléments non encore annotés seront ajoutés au corpus qui s'enrichira ainsi au fil des évaluations.

Enfin nous pensons améliorer certains aspects de notre programme. Nous souhaitons notamment affiner le score de pénalisation des termes absents. Nous souhaitons également tester la possibilité d'introduire plus de collaboration entre les modules, en permettant à des appariements d'un même terme obtenus de façon indépendante de se renforcer ou en permettant de renforcer la fiabilité estimée d'un terme en fonction de son contexte.

Bibliographie

[ADA06] Adams R. Textual Entailment Through Extended Lexical Overlap. In *The Second PASCAL Recognising Textual Entailment Challenge*. Bernardo Magnini, Ido Dagan(réd.), pp128-133, Italie, 2006.

[BAR05a] Barbier V. et Ligozat A.-L. A syntactic strategy for filtering sentences in a question answering system. In *Proceedings of RANLP*. Bulgaria, 2005.

[BAR05b] Barbier V., Grau B., Ligozat A-L., Robba I. et Vilnat A. Semantic Knowledge in Question-Answering Systems. In *Workshop KRAQ (Knowledge and Reasoning for answering question)*. pp49-52, IJCAI, 2005 .

[BAR05c] Barbier V. Quels types de connaissance sémantique pour Questions-Réponses? In *Actes de TALN-RECITAL* 2005. pp535-544, Dourdan, France, 2005.

[BARH06] Bar-Haim R., Dagan I., Dolan B., Ferro L., Giampiccolo D., Magnini B. and Szpektor I. The Second PASCAL Recognising Textual Entailment Challenge. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, Venice, Italy, 2006.

[BARZ97] R. Barzilay and M. Elhadad, 1997. Using lexical chains for text summarization. In Proceedings of the Intelligent Scalable Text Summarization Workshop (ISTS'97), ACL, Madrid, Spain.

[BAT08] Battistelli D., Couto J., Minel J.-L. et Schwer S.R. Représentation algébrique des expressions calendaires et vue calendaire d'un texte. In *Actes d e TALN 2008*, Avignon, 2008.

[BOU05] Bouma G., Fahmi I., Mur J., van Noord G., van der Pals L. et Tiedermann J. Linguistic knowledge and question answering. In *Traitement Automatique des Langues*. Vol. 46, n°3/2005, pp15-39. Hermès, Paris, 2005.

[BUD99] Budanitsky A. Lexical *Semantic Relatedness and its Application in Natural Language Processing*, Technical Report, CSRG390. University of Toronto, Canada,1999.

[CHA02] de Chalendar G., Dalmas T., Elkateb-Gara F., Ferret O., Grau B., Hurault-Plantet M., Illouz G., Monceau L., Robba I., Vilnat A. The question answering system QALC at LIMSI: experiments in using Web and WordNet. In *The Eleventh Text REtrieval Conference Proceedings (TREC 2002)*.

[CHU02] Chu-Carroll J, Prager J., Welty C., Czuba K. et Ferrucci D. A Multi-Strategy and Multi-Source Approach to Question Answering. In *The Eleventh Text REtrieval Conference Proceedings (TREC 2002)* .

[DER90] Derwester S., Dumais S. T., Furnas G. W., Landauer T. K. *Indexing by Latent*

Semantic Analysis. In Journal of the American Society for Information Science, 1990.

[ELA07] El Ayari, S. Evaluation transparente de systèmes de questions-réponses : application au focus In Actes de RECITAL 2007.

[FEL98] Fellbaum C. WordNet: An Electronic Lexical Database. Cambrige, MA: The MIT Press.

[FER98] Ferret O. ANTHAPSI, un système d'analyse thématique et d'apprentissage de connaissances pragmatiques fondé sur l'amorçage. Thèse de doctorat de l'Université Paris XI, Orsay.1998.

[FER01a] Ferret O., Grau B., Hurault-Plantet M., Illouz G., Jacquemin C. Document selection refinement based on linguistic features for QALC, a question answering system. In *Proceedings of RANLP 2001.* Tsigov Tchark, Bulgarie, 2001.

[FER01b] Ferret O., Grau B., Hurault-Plantet M., Illouz G., Monceau L., Robba I., Vilnat A. Finding an answer based on the recognition of the Question Focus. In *The Tenth Text REtrieval Conference Proceedings (TREC 2001)*, Gaithersburg, États-Unis, 2001.

[GIL05] Gillard L., Sitbon L., Bellot P., El-bèze M. Dernières évolutions de SQuaLIA, le système de Questions/Réponses du LIA. In *Traitement Automatique des Langues (TAL)*. vol. 46, n°3/2005, pp41-70. Hermès, Paris, 2005.

[GLI05] Glickman O. Dagan I. et Koppel M. Web Based Probabilistic Textual Entailment. In *Proceedings of the First PASCAL Workshop on Recognising Textual Entailment.* Southampton, Royaume-Uni, 2005.

[GLO07] Glöckner I. University of Hagen at CLEF 2007: Answer Validation Exercise. In *Working Notes for the CLEF 2007 Workshop.* Budapest, Hongrie, 2007.

[GRA08] Grappy A., Ligozat A.-L., Grau, B. Evaluation de la réponse d'un système de question-réponse et de sa justification. In *Actes de CORIA 2008*.

[GRAU08] Grau B., Vilnat A., Ayache C. EQueR : Évaluation de systèmes de question-réponse. In *L'évaluation des technologies de traitement de la langue: les campagnes Technolangue*. Chaudiron S., Choukri K.(réd.). Chapitre 6. Hermes, Paris, 2008.

[HAR00] Harabagiu S., Moldovan D., Paaca M., Mihalcea R., Surdeanu M., Bunescu Z., Girju R., Rus V., Morarescu P. FALCON: Boosting knowledge for answer engines. In *The Ninth Text REtrieval Conference Proceedings (TREC-9)*, pp479-488. 2000.

[HART04] Hartrumpf S. Question Answering using Sentence Parsing and Semantic Network Matching. In *Actes de CLEF 2004*.

[HART05] Hartrumpf S. University of Hagen at QA@CLEF 2005 : Extending Knowledge and Deepening Linguistic Processing for Question Answering. In *Actes de CLEF 2005.*

[HIC06] Hickl A., Williams J., Bensley J., Roberts K., Rink B. et Shi Y. Recognising Textual entailment with LCC's GROUNDHOG System. In *The Second PASCAL Recognising Textual Entailment Challenge*. Magnini B. et Dagan I.(réd.), pp80-85. 2006.

[HIR97] Hirst G. et St-Onge D. Lexical Chains as representation of context for the detection and correction of malapropisms. In *WordNet: An electronic lexical database and some of its applications*. Fellbaum C.(réd.), MIT Press, Cambridge, États-Unis,1997.

[HOVY01] Hovy E., Hermjakob U., Lin C.-Y. The Use of External Knowledge of Factoid QA. In *The Tenth Text REtrieval Conference Proceedings (TREC 2001), pp644-652.*

[JAC99] Jacquemin, C. Syntagmatic and paradigmatic representations of term variation. In

Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL'99), pp341-348. University of Maryland, 1999.

[JAC06] Jacquin C., Monceaux L. et Desmontil E. Systèmes de question-réponse et EuroWordNet. In *Actes de TALN 2006*, volume 1, pp 188-197. Louvain, 2006.

[JIJ05] Jijkoun V. et de Rijke M. Recognizing Textual Entailment Using Lexical Similarity. In *The First Challenge Workshop Recognizing Textual Entailment*. Southampton, Royaume-Uni, 2005.

[JOC03] Leidner J. L., Bos J., Dalmas T., Curran J. R., Clark S., Bannard C. J., Webber B. et Steedman M. Qed: The Edinburgh trec-2003 question answering system. In *The Twelth Text REtrieval Conference Proceedings (TREC 2003).*

[KAI05] Kaisser M. QuALiM at TREC2005: Web-Question Answering with FrameNet. In *The Fourteenth Text REtrieval Conference Proceedings (TREC 2005).*

[KAI07] Kaisser M., Webber B. Question Answering based on Semantic Roles**.** In T*he ACL 2007 Deep Linguistic Processing Workshop, ACL-DLP 2007.*

[KIN02] Kingsbury P. Adding Semantic Annotation to the Penn TreeBank. In *Proceedings of the Human Language Technology Conference*. 2002.

[KOU05] Kouylekov M. et Magnini B. Recognizing Textual Entailment with Tree Edit Distance Algorithms. In *Proceedings of the First PASCAL Workshop on Recognising Textual Entailment.* Southampton, Royaume-Uni, 2005.

[KOU06] Kouleykov M., MagniniB. The Tree Edit Distance for Recognizing Textual Entailment, Estimating the Cost of Insertion. In *The Second PASCAL Recognising Textual Entailment Challenge*. Bernardo Magnini, Ido Dagan(réd.), pp68-73, Italie, 2006.

[KOZ06] Kozareva Z., Montoyo A. MLEnt: The machine Learning Entailement System of the University of Alicante. In *The Second PASCAL Recognising Textual Entailment Challenge*. Bernardo Magnini, Ido Dagan(réd.), pp16-21, Italie, 2006.

[LAU05] Laurent D., Nègre S., Séguéla P. QRISTAL, le QR à l'épreuve du public. In *Traitement Automatique des Langues*. volume 46, n°3/2005, pp71-101. Hermes, Paris, 2005.

[LEH78] Lehnert W. *The Process of Question Answering: a Computer Simulation of Cognition.* Lawrence Erlbaum Associates, 1978.

[LIG06] Ligozat, A.-L. *Exploitation et fusion de connaissances locales pour la recherche d'informations précises*. Thèse de doctorat en informatique, Université Paris 11, 2006.

[LIN98] Lin D. An information-theoretic definition of similarity. In *Proceedings of International Conference on Machine Learning*.1998.

[LIN02] Lin C.-Y. The effectiveness of Dictionary and Web Based Answer Reranking.In *19th International Conference on Computational Linguistics (COLING2002).* Taipei, Taiwan 2002.

[MAG01] Magnini B., Negri M., Prevete R. et Tanev H. Multilingual Question/Answering: the DIOGENE System. In *Proceedings of the Tenth Text REtrieval Conference*, 2001.

[MOL99] Moldovan D., Harabagiu S. Mihalcea R. Goodrum R et Rus V. Lasso: A Tool for Surfing the Answer Net. In *The Eighth Text REtrieval Conference Proceedings (TREC-8)*, 1999.

[MOL02] Moldovan D., Harabagiu S., Girju R., Morarescu P., Lacatusu F., Novischi A., Badulescu A. et Bolohan O. LCC Tools for Question Answering. In *The Eleventh Text REtrieval Conference Proceedings (TREC 2002)*.

[MOL03] Dan Moldovan, Christine Clark, Sanda Harabagiu, Steve Maiorano. 2003. COGEX: A

Logic Prover for Question Answering. In Proceedings of the Human Language Technology Conference.

[MOL05] Harabagiu S., Moldovan D., Clark C., Bowden M., Hickl A. et Wang P. Employing Two Question Answering Systems in TREC 2005. In *The Fourteenth Text REtrieval Conference Proceedings (TREC 2005)*.

[NAR04] Narayanan S. et Harabagiu, S. Question answering based on semantic structures. In *20th international Conference on Computational Linguistics*. Genève, Suisse, 2004.

[NEW05] Newman E., Stokes N., Dunnion J. et Carthy J. UCD IIRC Approach to the Textual Entailment Challenge. In *The First Challenge Workshop Recognizing Textual Entailment*. Southampton, Royaume-Uni, 2005.

[PRA04] Pradhan S., Ward W., Hacioglu K., Martin J. et Jurafsky D. Shallow Semantic Parsing using Support Vector Machines. In *NAACL-HLT 2004*.

[RUP06] Ruppenhofer J., Ellsworth M., Petruck M. R. Johnson C. R. et Scheffczyk J. FrameNet II, Extended Theory and Practice. 2006. url : http://framenet.icsi.berkeley.edu/index.php?option=com_wrapper&Itemid=126

[RUS06] Russel S. et Norvig P. *Intelligence artificielle*, 2ème édition. PEARSON Education, 2006.

[SAL05] de Salvo Braz R., Girju R., Punyakanok V., Roth D. and Sammons M. *An Inference Model for Semantic Entailment in Natural Language.* In *First Challenge Workshop, Recognising Textual Entailement*, Southampton, U.K, 2005.

[SCH77] Schank R. et Abelson R. *Scripts, plans, goals and understanding : an inquiry into human knowledge structures.* Hillsdale, NJ: Lawrence Erlbaum Associates, 1977.

[SHI04] Shi L. and Mihalcea R. Semantic parsing using FrameNet and WordNet. In *Proceedings of the Human Language Technology Conference (HLT/NAACL 2004)*. Boston, États-Unis, 2004.

[SPA03] Spark Jones, K. Is question answering a reational task?. In *Questions and Answers: Theoretical and Applied Perspectives (Second CoLogNET-ElsNET Symposium)*. 2003.

[TAN05] Tanev H., Kouylekov M., Magnini B., Negri M. et Simov K. Exploiting Linguistic Indices and Syntactic Structures for Multilingual Question Answering: ITC-irst at CLEF 2005. In Working Notes for the CLEF 2005 Workshop.

[TATU06] Tatu M., Iles B., Slavick J., Novischi A. et Moldovan D. COGEX at the Second Recognizing Textual Entailment Challenge. In *The Second PASCAL Recognising Textual Entailment Challenge*. Bernardo Magnini, Ido Dagan(réd.), pp128-133, Italie, 2006.

[VOO02] Voorhees E.M., Overview of the Trec 2002 question answering Track, In *The Eleventh Text REtrieval Conference Proceedings (TREC 2002)*. Gaithesburg, États-Unis, 2002.

[VOS03] Vossen P. *EuroWordNet General Document*. University of Amsterdam, Novembre 2008.

[ZAN06] Zanzotto F.M., Moschitti A., Pennacchiotti M. et Pazienza M.T. Learning textual entailment from examples. In *The Second PASCAL Recognising Textual Entailment Challenge*. Bernardo Magnini, Ido Dagan(réd.), pp128-133, Italie, 2006.

[ZHA90] Zhang K. et Shasha D., Fast algorithm for the unit cost editing distance between trees. In *Journal of algorithms*, vol. 11, pp1245-1262, 1990.