
Interaction et recherche d'information : le projet RITEL

Sophie Rosset* — Olivier Galibert* — Gabriel Illouz**
Aurélien Max**

* *Groupe Traitement du Langage Parlé*
LIMSI-CNRS
{rosset,galibert}@limsi.fr

** *Groupe Langues, Information et Représentations*
LIMSI-CNRS et Université Paris Sud 11
{gabrieli,amax}@limsi.fr

RÉSUMÉ. L'objectif du projet RITEL est de réaliser un système de dialogue homme-machine permettant à un utilisateur de poser oralement des questions, et de dialoguer avec un système de recherche d'information généraliste. Dans cet article, nous présentons la plateforme actuelle et plus particulièrement les modules d'analyse, d'extraction d'information et de génération des réponses. Nous présentons les résultats préliminaires obtenus par les différents modules de la plateforme.

ABSTRACT. The Ritel project aims at integrating spoken language dialog and open-domain information retrieval to allow a human to ask general questions and refine her search interactively. In this paper, we describe the current platform and more specifically the analysis, information extraction and natural language generation modules. We present some preliminary results of the different modules.

MOTS-CLÉS : interaction homme-machine, recherche d'information, extraction d'information, génération en langue.

KEYWORDS: human-machine communication, information extraction, natural language generation.

1. Introduction

Le terme *système de dialogue* indique généralement un système permettant une interaction entre un humain et un système dans un cadre restreint (Glass *et al.*, 2000). Depuis quelques années cependant, la notion de système de dialogue impliquant un champ de compétences plus vaste commence à émerger, notamment dans le cadre des travaux sur les systèmes de Questions-Réponses, et en particulier dans le cadre des campagnes d'évaluation des systèmes interactifs de Questions-Réponses (Over, 1999) aux Etats-Unis, (Kato *et al.*, 2005) au Japon ou encore (Oard *et al.*, 2001) en Europe.

D'un point de vue général, nous pouvons dire qu'un dialogue est une suite d'échanges entre interlocuteurs dans un contexte donné. Un système de dialogue homme-machine interprète les requêtes de l'utilisateur en fonction de la tâche à accomplir, de l'historique du dialogue et du comportement de l'utilisateur. Son objectif est de donner à l'utilisateur les informations recherchées tout en assurant une interaction efficace et naturelle. Actuellement, la plupart des systèmes de dialogue (oral) homme-machine portent sur des domaines limités, par exemple les demandes d'information sur les horaires de trains (Lamel *et al.*, 2000), d'avions (Walker *et al.*, 2002), de bus (Raux *et al.*, 2005) ou l'information touristique (Bonneau-Maynard *et al.*, May 2006)). Ces projets ont donné lieu à des évaluations et ont permis d'asseoir la faisabilité de tels systèmes et ont notamment proposé des modèles de gestion dynamique du dialogue et de génération adaptée (Lamel *et al.*, 2000). Même si ce qui est entendu par interaction naturelle est très variable d'un système à un autre (Villaneau, 2003), la plupart de ces systèmes permettent une interaction (orale) relativement naturelle : l'utilisateur peut à tout moment changer d'avis et revenir sur des choix exprimés, interrompre la réponse du système en prenant la parole, et le système peut lui aussi changer de stratégie d'interaction en fonction des réactions de l'utilisateur.

Un système de dialogue utilise des sources de connaissances diverses et complexes : connaissances acoustiques, phonétiques, lexicales, morphologiques, syntaxiques et sémantiques, pragmatiques, ainsi que des connaissances sur le dialogue, sur la tâche à réaliser et sur l'interlocuteur. À chaque type de connaissance peut correspondre un ou plusieurs modules spécialisés réalisant un traitement spécifique : un système de reconnaissance de la parole (connaissances acoustiques, phonétiques, lexicales voire morpho-syntaxiques), un analyseur syntaxico-sémantique, un gestionnaire du dialogue (connaissances pragmatiques, sur la tâche, sur l'utilisateur, etc.).

Le terme *gestion de dialogue* recouvre différentes acceptions. Il peut inclure la planification et la résolution de problème (par exemple dans (Allen *et al.*, 1995)). Il peut également concerner la gestion des énoncés du système, de l'interaction et des successions d'échanges dans le dialogue (par exemple dans (Zue *et al.*, 2000; Lamel *et al.*, 2000)). Son rôle est de permettre une interaction naturelle entre l'utilisateur et le système. Le projet RITEL que nous présentons dans cet article est de ce dernier type.

En recherche d'information et en extraction d'information, des progrès sont apparus à la suite de campagnes d'évaluation, par exemple la campagne américaine TREC depuis 1998 (Voorhees, 2005), la campagne européenne CLEF depuis 2003 (Vallin *et*

al., 2005) ou encore la campagne nationale Equer/Technolangue (Ayache *et al.*, 2006). Les systèmes de Questions-Réponses évalués sont limités : ils prennent en entrée une question et fournissent en retour une seule réponse, constituée uniquement de l'élément supposément recherché. Depuis 1999, des tentatives ont été faites pour permettre des questions enchaînées, portant sur un même thème (Over, 1999; Hersh *et al.*, 2000). Ici, la notion d'interaction concerne en fait l'enchaînement de questions simples et essentiellement la résolution d'anaphores. Dans le cadre des évaluations NTCIR-QAC (évaluation de systèmes de Questions-Réponses en japonais), une tâche portant sur la recherche d'information *via* une interaction a également été proposée (Kato *et al.*, 2005). Toutefois, là aussi, il s'agit essentiellement de résolution d'anaphores. Dans ces deux cas, il ne s'agit pas de dialogues, car il n'y a pas d'interaction ni de négociations possibles (Kato *et al.*, 2004). En dehors des expériences menées sur des tâches strictement Questions-Réponses avec des questions et des réponses factoides, nous pouvons citer les travaux réalisés dans le cadre du projet AQUAINT (TREC, n.d.a). Ce projet a connu deux phases. La première visait à favoriser des recherches sur le thème de la recherche d'information dans des données de sources multiples (vidéo, parole, texte) et la deuxième phase, qui a débuté en 2004, a pour objet des recherches sur la synthèse d'informations à partir de sources multiples en vue de faciliter le travail d'analystes humains. Dans le cadre de ce projet, nous pouvons citer plus précisément pour les aspects qui nous intéressent les travaux de (Small *et al.*, 2003). Dans le système HITIQA, la notion de dialogue est clairement inscrite. Mais il s'agit de dialogue écrit avec un retour d'information visuel pouvant être complexe. Deux projets intègrent également les aspects liés à l'interaction orale ou à tout le moins la formulation orale de questions : le projet VAQA et le projet IMIX. VAQA (Harabagiu *et al.*, 2002) est un système de Questions-Réponses activé oralement. Il n'inclut toutefois pas réellement d'interaction. IMIX (den Os *et al.*, 2005) intègre, lui, complètement la notion de dialogue et d'interaction. Le démonstrateur vise à une certaine multimodalité et intègre en particulier la reconnaissance de la parole. Toutefois, la recherche d'information porte sur un domaine spécifique, le domaine médical.

Le projet RITEL a pour objectif d'intégrer des capacités conversationnelles et orales à des systèmes de recherche d'information et plus particulièrement de Questions-Réponses. Le dialogue oral joue un rôle essentiel dans notre approche de la tâche de Questions-Réponses, car il permet de décomposer de manière naturelle des requêtes complexes en requêtes simples à formuler pour l'utilisateur. L'exploitation d'un historique permet en outre de rendre implicites des thèmes et entités saillantes. Enfin, le système peut se comporter de manière proactive pour suggérer des voies de recherche à l'utilisateur. À terme, nous souhaiterions que le système puisse répondre à des questions telle que : *J'aimerais savoir de quand date la première version du fi lm où un grand euh singe grimpe sur une tour... aux États-Unis.*

Nous présentons dans cet article les objectifs généraux du projet RITEL et un état des lieux de la plateforme actuelle. Nous commençons par présenter rapidement les objectifs et l'architecture générale de la plateforme, qui est fortement dépendante des objectifs (section 2). Le système de reconnaissance de la parole est également présenté dans la section 3. Puis nous décrivons plus en détails l'analyse effectuée sur les énon-

cés des utilisateurs et des documents et la recherche d'information (sections 4 et 5). La génération en langue et la gestion de l'interaction sont présentées dans la section 6. Enfin, nous terminons par la présentation des perspectives ouvertes par un tel projet.

2. La plateforme RITEL

2.1. Objectif

L'objectif du projet RITEL est de réaliser un système de dialogue homme-machine permettant à un utilisateur de dialoguer oralement avec un système de recherche d'information généraliste (par exemple, chercher sur l'Internet *Qui est le Président du Sénat ?* ou encore *Quelle a été l'évolution du prix de l'essence ces dix dernières années ?*) et d'affiner ou préciser sa demande de manière interactive et dynamique. Ce projet aborde donc un certain nombre de points nouveaux :

- questions orales en domaine ouvert ;
- dialogue et raffinement interactif ;
- recherche à partir de questions orales dans des documents écrits et retour oral ;
- synthèse d'informations multi-sources.

L'objectif principal de ce projet est d'obtenir un système qui réagit perceptuellement de façon instantanée, de façon à pouvoir assurer un aspect naturel à l'interaction. Tous ces aspects sont soumis à deux contraintes : une contrainte de vitesse de réactivité forte à tous les niveaux et un vocabulaire potentiellement illimité. Les différents modules qui ont été implémentés ont été conçus pour répondre à cette contrainte de vitesse. Les systèmes existant actuellement ne permettent pas en effet de répondre à cette contrainte. Nous avons développé une première plateforme qui répondait aux contraintes de vitesse mais n'incluait pas de recherche d'information générale (Galibert *et al.*, 2005). Une infrastructure permettant également de gérer ce type d'interaction a été développée. Nous voulions pouvoir rapidement collecter un corpus aussi réaliste que possible de requêtes utilisateurs en interaction, des études ayant par ailleurs montré que l'acquisition de corpus par l'utilisation d'un Magicien d'Oz ne permettaient pas de collecter des données de qualité suffisante pour l'apprentissage des modèles de langage et des modèles acoustiques utilisés par le système de reconnaissance de la parole (Salter *et al.*, 1996). Ce corpus (Rosset *et al.*, 2006) a servi de base pour le développement du système de reconnaissance de la parole et des modules d'analyse et de recherche d'information présentés dans cet article.

2.2. Architecture

L'architecture du système est modulaire et distribuée, comme le montre la figure 2.2. Ces deux aspects permettent, entre autre, d'utiliser la plateforme dans dif-

férents modes : dialogue, Questions-Réponses, analyse, etc. Ceci permet de faciliter l'évaluation des différents modules indépendamment de l'interaction.

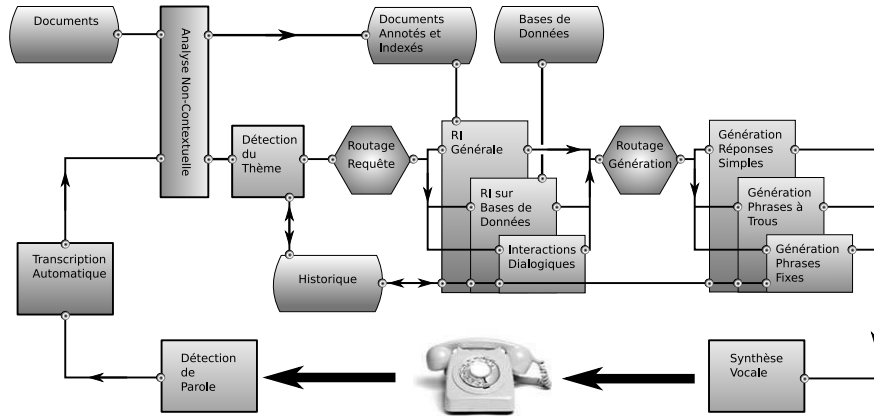


Figure 1. Architecture de la plateforme RITEL

Le signal audio provenant de la ligne téléphonique est segmenté en parole/non parole. Les segments de parole sont transcrits automatiquement par le système de reconnaissance. Les énoncés obtenus sont transmis à l'analyse non contextuelle qui en extrait les informations utiles. À partir de ce résultat la requête est dirigée vers un sous-module de traitement approprié, recherche d'information générale ou spécifique, ou gestion d'interactions dialogiques, qui produit une trame sémantique envoyée à la génération en langue. Celle-ci renvoie une ou plusieurs phrases qui sont synthétisées pour envoi à l'utilisateur. Les aspects inhabituels de cette architecture sont la présence d'une analyse commune pour les énoncés utilisateurs et les documents, le routage des énoncés vers des sous-modules de traitement de types différents, et l'absence de gestionnaire de dialogue centralisé. En effet, contrairement à la tradition (Zue *et al.*, 2000), nous considérons la gestion du dialogue comme devant être distribuée entre les différents composants du système global, chacun apportant sa part. De notre point de vue, la gestion du dialogue se rapporte à la compréhension en contexte de dialogue, la gestion de l'historique du dialogue, la recherche d'information et la génération de la réponse en langue.

Les principaux composants que nous avons implémentés sont le *système de reconnaissance de la parole*, l'*analyse non contextuelle*, la *recherche d'information* et la *génération en langue des réponses*. Le système de synthèse vocale est commercial (Sayso, de la société Acapela). Ces composants sont ouverts les uns aux autres et communiquent librement.

3. Détection et reconnaissance de la parole

La détection de parole utilise notre segmenteur de conversations téléphoniques. Le système tourne en continu. Des mesures ont été effectuées indiquant que le système doit tourner en un maximum de 0,9RT (RT = Real Time ; temps d'exécution = $0,9 \times$ temps de parole) pour obtenir des réactions immédiates du système.

Ce système de transcription a été développé à partir du corpus de quatre heures de parole utilisateur collecté avec la première version de la plateforme (Galibert *et al.*, 2005). Il est constitué d'un décodeur monopasse utilisant de gros modèles acoustiques monophones non-contextuels et un modèle de langage trigramme suivi d'un *rescoring* en consensus avec un modèle quadrigamme. Les modèles de langage ont été obtenus par interpolation de modèles de diverses sources sur le corpus de développement (le corpus d'entraînement RITEL (Rosset *et al.*, 2006), textes de journaux de 1988 à nos jours, transcriptions d'émissions d'information radio- et télé-diffusées et collections de questions venant de quizz). La vitesse actuelle est de 0,5RT pour un taux d'erreur un peu élevé d'environ 28%. Ceci est bien plus élevé que l'état de l'art en transcription du français qui est de 10,3% pour un système 10RT pour des émissions d'information radio-diffusées (Gauvain *et al.*, 2005). Les causes sont multiples :

- très faible quantité de données d'entraînement du bon type. Cela représente probablement une augmentation du taux d'erreur de 5 à 10% absolus ;
- système *streamé*, i.e. qui tourne en même temps que la personne parle. L'impact mesuré sur le taux d'erreur est d'environ 2-3% absolus ;
- système travaillant sur du signal téléphonique, qui est notoirement plus difficile à traiter que des émissions enregistrées ;
- système sans adaptation au locuteur. Dans un système 10RT tri-passe avec adaptation entre chaque passe comme notre système de reconnaissance pour des émissions radio- et télé-diffusées, le taux d'erreur passe à 10,3% ;
- système très rapide, un système identique mais sans contrainte de temps est meilleur de 5% absolus.

Notre expérience sur les systèmes de dialogue en domaine fermé nous a montré qu'obtenir de bons résultats nécessitait un taux d'erreur d'au maximum 20% (Lamel *et al.*, 2000). Ce taux d'erreur sur les mots, mesure classiquement utilisée en reconnaissance de la parole, n'est probablement pas le plus pertinent pour mesurer l'efficacité du système de reconnaissance. Toutefois, des mesures précédentes (Rosset, 2000) ont montré une corrélation entre le taux d'erreur sur les mots et le taux d'erreur sur des entités nécessaires à la compréhension de la question de l'utilisateur. Ces précédents travaux portaient sur un système de dialogue oral en domaine limité (information sur des trajets en train) dans lequel la part des entités nommées était très importante (plus de 80 % du lexique) et qui, du point de vue du modèle de langage, étaient équiprobables. D'autres études ont montré (Miller *et al.*, 2000) que pour une détection fine des entités nommées, il existe une corrélation forte entre la F-mesure et le taux d'erreur sur les mots. Toutefois, dans le cadre d'une interaction, on peut espérer que, comme

en domaine restreint, le dialogue permette de réduire l'impact des erreurs de reconnaissance. Dans le but d'atteindre moins de 20 % d'erreurs, nous tentons d'augmenter la quantité de données de façon significative. De plus, nous travaillons à l'utilisation de techniques d'adaptation dynamique à la voix en cours de conversation (méthodes VTLN ou CMLLR) ainsi qu'à l'amélioration des modèles acoustiques en utilisant les techniques habituellement utilisées dans nos systèmes de transcription d'émissions radio- et télé-diffusées comme l'utilisation de modèles discriminants (MMIE par exemple). Enfin, nous travaillons également sur la sélection et/ou l'adaptation dynamique des modèles de langage en fonction de l'état du dialogue.

4. Analyse des énoncés utilisateurs et des documents : une analyse commune

Nous avons voulu, autant que possible, utiliser exactement les mêmes analyses (et analyseurs) pour les énoncés des utilisateurs et pour les documents dans lesquels la recherche d'information est effectuée. Cela doit permettre, selon nous, de récupérer des erreurs : une erreur d'analyse commise dans un énoncé le sera probablement aussi dans les documents. Cette analyse est dite *non contextuelle* car elle n'utilise aucune information sur le dialogue, et qu'elle ne va pas au-delà de l'unité phrase dans les documents textuels. L'objectif de cette analyse est de détecter l'*information pertinente* dans un énoncé ou une question d'utilisateur (écrit ou oral) ou dans du texte (écrit ou transcrit de l'oral). La figure 2 montre un exemple de ce que nous appelons information pertinente.

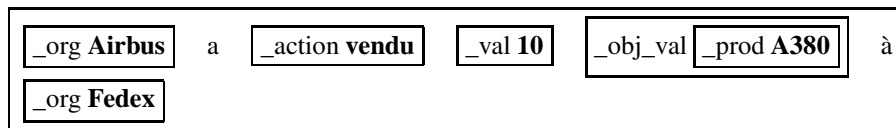


Figure 2. Exemple d'information pertinente

Cette information pertinente recouvre différentes catégories : entités nommées, entités linguistiques (verbes, prépositions, substantifs, etc.), entités spécifiques (entités qui se rapportent à un domaine spécifique, comme les maladies, des mesures, etc.) ou encore entités étendues (citations, titres, etc.). Ce concept d'information pertinente concerne également les frontières de ces groupes de mots. Nous considérons également que tous les mots doivent entrer dans une catégorie car tous portent potentiellement une information utile. L'approche choisie est d'annoter la suite de mots la plus longue, non typée, ayant un sens unique et cohérent, autrement dit des entités complexes mais que nous ne savons pas typer. Cette définition se rapproche fortement de celle donnée par (Abney, 1995) pour la notion de *chunk*.

Enfin, le système doit être utilisé dans le cadre d'une interaction orale pour de la recherche d'information. Un système de ce type est soumis à deux contraintes majeures : il doit pouvoir être utilisé sur des données écrites et orales (textes et requêtes) et il doit être aussi rapide que possible.

4.1. Définition des entités

4.1.1. Les entités classiques

Les entités nommées désignent classiquement des noms de lieux, de personnes et d'organisations. Les entités numériques (souvent associées aux entités nommées) représentent des dates ou des unités monétaires. Ces entités nommées sont perçues comme des éléments majeurs pour la compréhension de texte et de documents et donc pour la recherche d'information. Les conférences MUC (Kaufman, 1998) ont mis en avant plusieurs tâches génériques dont l'analyse en entités nommées. Cette analyse recouvre d'une part la détection des entités nommées (les passages contenant une entité nommée et ses frontières) et d'autre part le typage de cette entité. Ce typage est effectué d'après des ontologies définies préalablement. La définition de base la plus utilisée est héritée des conférences MUC. Elle comprend trois catégories : les expressions de noms propres (personnes, lieux organisations), les expressions temporelles (dates, heures) et les expressions numériques (valeurs monétaires, pourcentages). La hiérarchie définie est présentée dans (Grishman, 1995). Ces définitions sont parfois étendues à l'intérieur d'une catégorie, comme par exemple les organisations afin d'en affiner le contour référentiel, telles celles adoptées dans le cadre de la campagne d'évaluation ACE (ACE, 2000). Afin de couvrir de nouveaux besoins ou de nouvelles tâches, ces définitions et hiérarchies sont étendues. Par exemple, Sekine propose jusqu'à 200 éléments (Sekine, 2004).

Dans le cadre de la plateforme RITEL, nous avons besoin d'une analyse allant au-delà des entités nommées. Nous avons intégré des définitions d'entités proches de celles proposées par (Sekine, 2004) et dans (ACE, 2000), et nous avons inclus d'autres niveaux d'analyses. Quelques exemples des types d'entités retenus sont présentés dans le tableau 1. Ces différentes entités sont utilisées comme clefs de recherche pour la réponse ou la catégorisation de la question.

4.1.2. Entités nommées précises et non précises

Nous avons étendu la définition d'entité nommée à la définition suivante : *expression décrivant un élément spécifique que d'un type donné*. Le *type* désigne ici les catégories habituelles d'entités nommées. Par exemple, *président des Etats-Unis en 2006* est une entité nommée de type *personne* selon notre définition. De plus, nous utilisons une description hiérarchique des entités. Ainsi l'entité *personne président des Etats-Unis en 2006* contient plusieurs entités : une entité de type *fonction* avec *président* comme valeur, une entité de type *organisation* et une autre de type *pays* avec *Etats-Unis* comme valeur et enfin une entité *année* avec *2006* comme valeur. Une entité de type *pays* peut être reconnue comme *organisation* ou *localisation* mais dans ce contexte, il s'agira d'une *organisation*.

D'autre part, nous avons défini les entités dites non précises, qui décrivent un ensemble d'éléments d'un type donné à cardinalité non définie. Par exemple, *le festival de Cannes* est un événement non défini et est annoté comme *Eve*, par opposition à *eve* pour un événement précis comme *le festival de Cannes de 2006*.

| | |
|--------------------------------|---|
| Entités nommées | <_org> NIST </> <_eve> festival de Cannes de 2006 </> qui a dit <_cit> veni vidi vici </> |
| Entités non précises | <_Eve> festival de Cannes </> le <_Pers> president </> a déclaré ... |
| Entités étendues multi-niveaux | Fonctions, titres (président, professeur, évêque...) couleurs, animaux... |
| Super classes hiérarchiques | évêque → fonction religieuse → fonction |
| Marqueurs thématiques | Je m' intéresse aux <_litterature> romans </> de ... qui a gagné le <_sport> Mondial </> de 1998 |
| Marqueurs interrogatifs | <_Qqui> qui </> a écrit ce livre <_Qmesure> de combien </> d' heures dure ... |
| Marqueurs d'interaction | <_DA_close> au-revoir </> <_DA_yes> oui s'il vous plaît </> |
| Mots composés | les <_NN> logiciels de base de données </> sont ... les <_NN> élections multi-raciales </> |
| Chunks verbaux | il a <_action> gagné</> ... ils <_action> prendront part à </> ... |
| Entités linguistiques | <_adj_comp> le plus gros </> exportateur... cela se produit <_adv> souvent </> quand ... |

Tableau 1. Différents types d'entités (les exemples sont, ici, annotés au plus large)

4.1.3. Entités spécifiques et étendues

Nous avons aussi étendu les types possibles suivant trois directions principales :

- couverture des besoins fins en analyse des sous-systèmes de recherche d'information spécifique ;
- couverture aussi large que possible des objets nommables présents dans le corpus d'apprentissage (animaux, couleurs, etc.) ;
- détection des thèmes (littérature, géographie, etc.).

Nous avons utilisé pour cela le corpus d'apprentissage de RITEL qui avait été annoté en entités et thèmes (Rosset *et al.*, 2006). Le fonctionnement est similaire à celui des entités nommées, avec distinction précise/non précise au besoin et hiérarchisation.

4.1.4. Marqueurs de question et d'interaction

Dans le cadre des systèmes de Recherche d'Information en langue le typage des questions permet de déterminer le type de la réponse attendue. Cela nécessite une détection et classification des marqueurs d'interrogation tels que *qui*, *que*, *quoi*. De même, un système de dialogue, pour permettre un échange naturel, doit être en mesure

de détecter les éléments de gestion de l'interaction tels que *oui*, *non*, *au revoir* ou *pouvez-vous répéter* ? Nous les avons donc intégrés dans l'analyse non contextuelle.

4.1.5. *Chunks*

Les mots non couverts par les définitions précédentes possèdent potentiellement une information utile, ce qui nécessite donc de les annoter. Ils sont regroupés en blocs les plus longs possibles de catégories comparables (groupe nominal, groupe verbal, etc.) typés avec des catégories syntaxiques grossières (nom composé, verbe, adjectif, etc.). Ces groupes sont des éléments de bas niveau et ne sont donc pas récursifs. Comme dit précédemment, cette définition se rapproche fortement de celle donnée par (Abney, 1995) pour la notion de *chunk*.

4.2. *Description du système d'analyse non contextuelle*

Ainsi que nous l'avons dit, notre objectif est de réaliser un système d'analyse multi-niveaux rapide. Ce système doit être également capable de traiter des énoncés interrogatifs et du texte ou du discours. Il doit également être utilisable pour des données orales transcrites manuellement ou automatiquement et des données écrites. Les entités que nous avons définies se rapportent à au moins deux niveaux d'analyses classiquement considérées : analyse syntaxique (les chunks non typés) et entités nommées. En ce qui concerne les entités nommées classiques, plusieurs méthodes d'analyse ont été proposées. Les approches statistiques, telles que décrites dans (Bikel *et al.*, 1997) à base de Modèles de Markov Cachés ou dans (Isozaki *et al.*, 2002) qui utilisent un classifieur à vecteurs support ou encore (Surdeanu *et al.*, 2005) qui utilisent des techniques d'apprentissage, ont l'inconvénient de nécessiter des corpus préalablement annotés manuellement et de taille conséquente. Les systèmes de détection d'entités nommées qui reposent sur des approches linguistiques se fondent entre autre sur un étiquetage morpho-syntaxique ou une analyse syntaxique préalable (voir par exemple (Wolinski *et al.*, 1995)). Les analyseurs syntaxiques sont le plus souvent adaptés à du texte écrit et non à une production orale qui comporte des hésitations, des reprises, des reformulations ainsi que des erreurs (utilisation d'un système de reconnaissance de la parole). Du fait de ces contraintes (traitement de l'oral, vitesse de traitement) utiliser des analyses syntaxiques paraissait difficile. D'autre part, ne disposant pas de corpus de taille suffisante contenant l'annotation des différentes entités définies, les méthodes strictement statistiques n'étaient pas directement et simplement utilisables. Nous avons opté pour l'utilisation d'expressions régulières, nous rapprochant en cela de (Sekine, 2004) qui par ailleurs cherche à détecter des entités similaires aux nôtres.

4.2.1. *Moteur d'expressions régulières de mots*

Le système actuel est fondé sur l'écriture de règles. Celui-ci doit pouvoir utiliser des listes pour les détections initiales, des définitions de contextes locaux et permettre des catégorisations simples. Compte tenu de ces contraintes, nous avons opté pour un

système d'expressions régulières de mots (ERM). Ce moteur ERM a été implémenté avec certaines caractéristiques utiles pour le traitement de la langue. Il inclut :

- assertions positives et négatives avant et arrière (*positive and negative lookaheads*);
- quantifieurs gourmands et non-gourmands (*shy and greedy groupings*);
- classes nommées et macros;
- stratégies pour définir l'ordre d'application des règles;
- substitutions arborescentes et recherches dans des sous-arbres;
- annotation externe de mots (pour les classes morpho-syntaxiques);
- catégorisations internes des mots (nombre, acronyme, nom propre, etc.).

Voici un exemple de règles d'expressions régulières de mots :

```
_loc : ((=& prep_loc) %caps | %pays | %ville);
```

où `_loc` est le nom de la règle (l'entité nommée), `&prep_loc` fait référence à une macro qui décrit des prépositions ou groupes prépositionnels désignant un lieu, `%caps` la classe des mots commençant par une majuscule et `%pays` et `%ville` les classes des mots appartenant à un dictionnaire de pays et de villes (des ressources propres).

Le moteur fonctionne sur les principes classiques de retour sur trace (backtracking) avec une masse de précalculs sur les règles pour assurer de bonnes performances. Le résultat de ces précalculs peut être sauvegardé sous une forme compilée permettant ainsi d'assurer des temps de chargement minimes.

4.2.2. Méthodologie

Nous pouvons distinguer dans les entités que nous cherchons à repérer trois catégories : les entités nommées, étendues et non-spécifiques ; les marqueurs thématiques, de questions et d'interaction ; et enfin les chunks non typés.

– **Entités nommées, étendues et non spécifiques** : des listes sont utilisées pour la détection initiale. Ces listes contiennent des noms propres, des noms de villes, de pays, etc. Des contextes caractéristiques sont aussi détectés. Une catégorisation morpho-syntaxique (nom, verbe, adjectif, etc.) et sémantique (verbe de mouvement, déclaratif, etc.) assez grossière est également effectuée et est utilisée pour aider au typage des entités nommées. L'étiquetage morpho-syntaxique n'est pas désambiguïsé, l'étiquetage d'un mot reprend donc toutes les étiquettes possibles. Par exemple, la suite de mots *il est entré* est annotée :

```
il|_PRO est|_V|_N|_A entré|_V|_K
```

où `_PRO` = pronom, `_V` = verbe, `_N` = substantif, `_A` = adjectif, `_K` = participe passé

Nous utilisons pour cela le dictionnaire DELAS (Courtois *et al.*, 1990). La désambiguïssation est effectuée via une analyse légère¹ ascendante. Toutefois, certaines

1. Autrement dit qui ne repose sur aucune analyse syntaxique ou morphosyntaxique.

décisions sont prises *a priori* d'après la fréquence de l'association mot-catégorie. Par exemple, un mot qui peut être un pronom comme dans l'exemple ci-dessus sera annoté en tant que pronom. Par contre, toujours dans l'exemple ci-dessus, annoter *est* en auxiliaire et *entré* en verbe est dépendant d'une règle contextuelle (si V2 suit V1 qui précède PRO alors V2 est un verbe et V1 un auxiliaire). De plus, des combinaisons de contextes et de patrons aident à la construction des hiérarchies.

– **Marqueurs** : certains marqueurs thématiques sont directement liés à la présence d'entités étendues ou de super-classes (*je m'intéresse à la littérature*). D'autres sont le résultat de cooccurrences de super-classes (ex. : *_fonction + _pays* → géopolitique). Pour les marqueurs interrogatifs et interactionnels, des listes sont utilisées (ex. : *quoi, comment, quand*, ainsi que certaines structures prépositionnelles ou verbales (ex. : *combien + heures* → *_Qmesure_phys*)).

– **Chunks** : le regroupement des mots en chunks est fondé sur l'utilisation de patrons morpho-syntaxiques, en appliquant les plus probables d'abord et en utilisant les entités comme contextes et frontières.

L'analyse elle-même est multipasse et chaque analyse s'appuie sur les analyses précédentes qu'elle peut éventuellement affiner ou modifier. Différentes listes sont utilisées. Elles contiennent environ 2600 noms propres, 500 noms de pays, 185 000 noms de ville, 300 noms de langue, etc. L'analyse est effectuée en une quarantaine d'étapes et prend 4 ms par énoncé². La figure 3 montre un exemple d'analyse non contextuelle.

```

Enoncé : le président Bush a rencontré le premier ministre israélien après les der-
nières élections palestiniennes
<_det> le </> <_fonc> <_fonction_publicque> président </></><_pers> Bush</>
<_aux> a </> <_action> rencontré </>
<_det> le </>
<_pers_fonct>
  <_fonc> <_fonction_publicque> premier ministre </> </>
  <_org> <_pays> <Israël> israélien </> </> </> </>
<_prep> après </>
<_range_objet> <_det> les </> dernières </>
<_Eve>
  <_Eve>élections</><_org><_pays><Palestine>palestiniennes</></></></>

```

Figure 3. Exemple de résultat d'une analyse non contextuelle

2. Vitesse suffisante pour l'analyse de la question mais clairement insuffisante pour des documents arrivant en flux continu du Web.

4.3. Résultats préliminaires

Nous avons effectué une évaluation sur les entités nommées et étendues (les seules ayant été formellement définies). Nous avons utilisé pour cela différents corpus, décrits dans le tableau 2 :

- requêtes orales (Sq) : énoncés utilisateurs collectés avec la première plateforme RITEL, transcrits manuellement et orthographiquement;
- questions écrites (Wq) : questions de l'évaluation CLEF'04 ;
- émissions d'information radio- et télé-diffusées (BN) : documentaires et informations en français transcrits manuellement et orthographiquement;
- journaux (NP) : Le Monde et ATS 1994-1995.

Ces corpus ont été annotés manuellement en fonction des définitions des entités que nous traitons. Ils ont servi de référence pour cette évaluation. Les entités évaluées

| Categorie | # doc. | # mots | # Entités |
|-----------|--------|--------|-----------|
| Sq | 840 | 10k | 1102 |
| Wq | 200 | 1,5k | 331 |
| BN | 7887 | 88k | 5898 |
| NP | 1000 | 22k | 2485 |

Tableau 2. *Caractéristiques des corpus de test*

étaient les suivantes : loc, org, pers, date, amount, val, age, couleur, fonction, meteo, money, mesure_phys, org_gvt, eve, prix et museum. Les entités des rangs inférieurs (par exemple, ville, pays, province, fleuve pour loc) ont également été évaluées. Dans le cas d'un groupe hiérarchique l'entité la plus large a été évaluée.

Le tableau 3 présente les résultats obtenus sur les différents corpus. Ce système obtient une F-mesure allant de 0,82 sur le corpus BN à 0,88 sur le corpus Sq. La plupart des systèmes de détection en entités nommées obtiennent une F-mesure autour de 0,9 (Poibeau, 2005) sur des données journalistiques et dans le cadre d'une définition simple des entités nommées. (Favre *et al.*, 2005) rapportent une F-mesure entre 0,84 et 0,74 sur un corpus d'émissions d'information radio- et télé-diffusées de la campagne d'évaluation ESTER (Gravier *et al.*, 2004). Sur des transcriptions manuelles comme celles de notre corpus BN, leur meilleur système obtient une F-mesure de 0,84. Par ailleurs, (Surdeanu *et al.*, 2005) rapportent une F-mesure sur des corpus de type conversationnel (en anglais) de 0,75. Comparer nos résultats avec ceux-ci est relativement peu aisé car les définitions des entités diffèrent. Néanmoins, on peut probablement affirmer que sur des données journalistiques textuelles (comme notre corpus NP) notre système est légèrement moins bon que ceux décrits dans la littérature. Par contre, il est à hauteur de l'état de l'art pour les corpus oraux.

Les analyses sont les mêmes pour les énoncés utilisateurs et pour les documents dans lesquels la recherche est effectuée. Ce faisant, si une erreur est produite d'un côté

elle a toutes les chances d'être également produite de l'autre. Ceci permet de limiter l'impact des erreurs d'analyse sur le résultat de la recherche. Nous avons mené des expériences sur la désambiguïsation de certaines catégories de noms propres pouvant apparaître dans des entités nommées de catégories différentes. Pour cela nous avons utilisé une approche de type *Memory Based Learning* qui nous a permis d'améliorer la F-mesure sur les entités de type *_loc* (ville, pays, etc.) et *_pers* de 12.5%. Cette approche était utilisée sur les résultats des analyses fournies par le module NCA. Ces résultats étant concluants, nous allons intégrer l'approche présentée dans le module d'analyse.

| Mesure | Sq | Wq | BN | NP |
|--------|-------|-------|-------|-------|
| Prec. | 90,3% | 85,5% | 83,5% | 87,1% |
| Rap. | 86,2% | 83,4% | 81,3% | 86,5% |
| F-mes. | 0,882 | 0,845 | 0,824 | 0,868 |

Tableau 3. Résultats pour la détection des entités

5. Routage et recherche d'information

Comme le montre la figure 2.2, il n'existe pas de gestionnaire de dialogue en tant que tel dans la plateforme. La gestion du dialogue est distribuée à travers l'ensemble des composants. L'approche choisie pour le développement de la plateforme est une approche totalement intégrée. La *détection de changement de thème*, le *routage de la question* et le *routage pour la génération en langue* peuvent tous être considérés comme des composants de la gestion de dialogue.

L'idée générale de la détection de changement de thème est de répondre à la question suivante : *Voulons-nous compléter la requête courante avec les éléments provenant des échanges précédents ?* Ce module utilise les marqueurs thématiques détectés lors de l'analyse non-contextuelle. S'il y a un changement de thème, l'historique est ignoré.

Après cette détection de changement de thème, l'énoncé de l'utilisateur est transmis au module approprié par routage. Le module qui prend en charge ce routage s'appuie sur le résultat de l'analyse NCA et les marqueurs d'interaction. Ce module transmet l'énoncé analysé à l'un des trois modules suivants :

- recherche d'information spécifique (sur bases de données)³ ;
- recherche d'information générale (cf. section 5.1) ;
- interaction strictement dialogique ou énoncé non classifiable (cf. section 6).

Pour chacun de ces modules, une complétion de l'énoncé en fonction de l'historique est effectuée. Le principe actuel de cette complétion est relativement simple :

3. Nous ne présentons pas dans cet article le module de recherche d'informations spécifiques. Il a été décrit dans (Galibert *et al.*, 2005) et se rapproche très fortement de ce qui est habituellement fait dans les systèmes de dialogue oral homme-machine pour des domaines limités.

une nouvelle entité remplace l'entité de niveau équivalent dans l'historique. La figure 4 illustre cette complétion. Cette complétion s'appuie fortement sur les entités et hiérarchies définies pour l'analyse.

| |
|---|
| <p><i>Complétion sur énoncé utilisateur</i></p> <p>U : quel est le président de la France S : Jacques Chirac U : et sa capitale schéma après complétion : <_Qcapitale> capitale </>, <_pays> France </></p> <hr/> <p><i>Complétion sur réponse système</i></p> <p>U : de quel pays Paris est la capitale S : France me semble être la réponse. J'écoute votre question U : qui est son premier ministre schéma après complétion : <_fonctions> 1er ministre </>, <_pays> France </></p> |
|---|

Figure 4. Exemples de complétion

5.1. Recherche d'information générale

Le module de *recherche d'information générale* correspond à un système classique de Questions-Réponses. Ce module prend en charge la recherche d'information dans des documents généraux comme les articles de journaux, des documents du web, des transcriptions d'émissions radio- et télé-diffusées. Les documents sont normalisés puis analysés par le module d'analyse non-contextuelle décrit dans la section 4. Les paires (type, valeur) sont indexées par un indexeur spécialisé afin de permettre une recherche d'information rapide. D'autre part, l'indexation des documents est faite au niveau des phrases de façon à permettre une recherche dans des passages courts.

L'approche choisie peut être comparée à une approche de type *sacs de mots*, à ceci près que ce ne sont pas des mots mais des mots typés ou des groupes de mots typés. L'index est construit au niveau des paires (type, valeur) et chacune de ces paires constitue une unité de recherche potentielle. Ceci est proche de ce qui est décrit dans (Kamps *et al.*, 2003) pour la recherche d'information dans des documents structurés de type XML. La recherche d'information générale s'effectue en plusieurs étapes. Tout d'abord, à partir de l'entrée (analyse de l'énoncé utilisateur éventuellement complétée par l'historique), l'extraction du type de réponse attendu (i.e. l'entité recherchée) est effectuée. Ensuite, une requête est générée et envoyée au moteur qui interroge l'index des documents et retourne les passages contenant potentiellement la réponse. Enfin, l'extraction proprement dite de la réponse est effectuée. Si la requête initiale ne renvoie aucun passage pertinent, alors des techniques de repli (*backoff*) sont utilisées et une ou plusieurs nouvelles requêtes sont générées.

5.1.1. Détection du type de la réponse

La détection du type de la réponse est effectuée en utilisant les marqueurs de questions et les cooccurrences d'entités nommées, étendues et non-spécifiques. Par exemple, une requête comportant le marqueur interrogatif `_Qqui` (comme dans *Qui a vendu Manhattan ?*) déclenche une recherche sur `_pers`, `_pers_def` ou `_org`. Si le marqueur interrogatif non spécifique `_Qquel` apparaît avec le marqueur `_fonction` (*Quel est le nom du pape ?*), alors c'est une entité de type `_pers` qui est recherchée. Actuellement nous avons défini 13 routages possibles selon le type de la question et surtout le type le plus probable de la réponse. Dans l'absolu, nous avons autant de différents types de réponses que de types d'entités analysées et annotées par le module NCA (soit 182 catégories), et l'analyse NCA fournit 24 types de question possibles. Nous sommes ici bien loin des 166 types de question de (Blaudez *et al.*, 2005) ou des 86 types de (Laurent *et al.*, 2005).

5.1.2. Génération des requêtes et extraction de passages

Les requêtes contiennent l'ensemble des paires (type, valeur) que doivent *a priori* contenir les passages recherchés. Ces paires sont celles présentes dans l'énoncé. Si une requête ne ramène pas de passage, alors une nouvelle requête procédant d'un agrandissement de la fenêtre de recherche puis d'un relâchement de contraintes est appliquée. Le relâchement de contraintes porte sur les paires (type,valeur) de l'énoncé qui constituent les clefs de recherche. Il comprend la possibilité d'un changement de type (une valeur associée au type `_org` peut être recherchée avec le type `_loc`), la possibilité de faire des recherches sur des entités incluses ou inclusives (par exemple Bush → Georges Bush) et l'abandon de certaines entités. Ces abandons dépendent de l'importance relative des différentes entités présentes dans les corpus (par exemple, une entité nommée est plus importante qu'un verbe). Ce relâchement de contraintes est appliqué tant qu'aucune réponse candidate n'est trouvée. La figure 5 illustre les différents relâchements de contraintes appliqués sur une requête. Les passages contenant les éléments de la requête sont ainsi récupérés. Selon la taille de la fenêtre de recherche utilisée, un passage contient de une à plusieurs phrases. Pour le moment, la fenêtre maximale contient trois phrases.

| |
|---|
| <p>énoncé : de quel groupe politique Charles Millon est-il le président requête 1 : <code>_Org = groupe politique ; _pers = Charles Millon ; _fonc = président</code> relâchement 1 : <code>_Org = * ; _pers = Charles Millon ; _fonc = président</code> relâchement 2 : <code>_pers = Charles Millon ; _fonc = président</code> relâchement 3 : <code>_pers = Millon ; _fonc = président</code> relâchement 4 : <code>_pers = Charles Millon</code></p> |
|---|

Figure 5. Exemple de relâchements de contraintes

5.1.3. Extraction des réponses candidates

La recherche et l'extraction des réponses candidates dans les passages s'effectuent en fonction du type de la réponse attendue. Toute entité qui correspond à ce type

est considérée comme candidate. La sélection du type du candidat obéit également à une logique de repli. Si on prend l'exemple donné dans la figure 5, le type de réponse attendu est une entité nommée de type `_org`. Toutefois, une entité d'un type différent peut être recherchée si aucune organisation n'est trouvée. Par exemple une organisation probable (`_org_prob`) ou un acronyme (`_acronym`) peuvent être considérés comme des candidats acceptables. La sélection de la réponse est effectuée à partir d'une classification simple des candidats fondée sur le nombre d'occurrences de la valeur du candidat dans la liste des réponses possibles. Actuellement, seule cette information est utilisée. Nous considérons qu'elle permet de donner une bonne appréhension de la confiance que le système peut avoir dans sa réponse. Cette information est utilisée pour le retour de l'information à l'utilisateur.

5.2. Discussion

Le système présenté est un système minimal servant de point de départ pour nos recherches futures. Ce système remplit la contrainte majeure de vitesse que nous avons. En effet, pour analyser la question, identifier et catégoriser le type de réponse attendue, générer la requête, extraire les passages et les réponses possibles, classer les réponses puis sélectionner une réponse, le système met au total environ 0,1s.⁴

Les premiers résultats obtenus sur le corpus d'évaluation de la campagne CLEF'05 montrent une Réciproque du Rang Moyen (*Mean Reciprocal Rank*, MRR)⁵ d'environ 45% sur l'ensemble des 200 questions et que 33% des réponses classées en premier sont justes. On peut comparer ces résultats à ceux obtenus par les participants à la tâche monolingue de CLEF'05 (Vallin *et al.*, 2005). Le premier système avait obtenu 64% de réponses correctes, les deux suivants 35% et 34.5% et ensuite les sept suivants entre 14% et 23%. Le MRR obtenu sur les questions strictement factoides est d'environ 60%. L'analyse des premiers résultats du système de recherche d'information générale met en évidence plusieurs aspects intéressants de notre point de vue.

Tout d'abord, les recherches qui s'appuient sur les entités hiérarchiques sont les plus efficaces pour deux raisons. La première est liée à l'approche choisie pour l'extraction de la réponse et la seconde à la recherche elle-même.

En ce qui concerne l'extraction de la réponse, nous avons dans un premier temps opté pour l'extraction d'une seule paire (type, valeur), cette paire constituant la totalité de la réponse. Il est évident que ce choix pose un problème car nombre de réponses sont constituées de plusieurs éléments. Toutefois, la représentation hiérarchique choisie ainsi que les chunks, dont certains peuvent être longs, permettent de trouver certaines de ces réponses. Ainsi, les définitions de personnes ou leur activité sont trouvées. Par

4. Le corpus CLEF'05 est traité en 20 s. sur un PC dual xeon EM64T à 3.2 Ghz et 6 Go de RAM.

5. Ce critère tient compte du rang de la première bonne réponse trouvée. Une bonne réponse trouvée au premier rang "vaut" 1, au deuxième rang 1/2, au troisième rang 1/3 etc. Une bonne réponse trouvée plusieurs fois n'est comptée qu'une fois (TREC, n.d.b).

exemple la question *Qui est Goodwill Zwelithini ?* reçoit la bonne réponse *roi des Zoulous*, cette réponse étant représentée par :

<_pers_fonct> <_fonctions> <_fonction> **roi** </> </> **des** <_org_prob> <_np> **Zou-**
lous </> </> </>

Pour ce qui est de la recherche d'information, nous voyons au moins deux avantages :

1) Les entités hiérarchiques permettent de s'appuyer sur des types de patrons, qui peuvent en partie être vus comme des patrons syntaxiques (ex. :<pers_fonct> <fonction/> de la <org><pays/> </> </>).

2) Les entités hiérarchiques permettent *via* l'utilisation de stratégies de repli de dégrouper des entités. Par exemple, une recherche portant sur *le festival de Cannes de 2006* analysée en <_eve><_Eve><_Eve> festival </> de <_ville> Cannes </> </> de <_date> 2006 </_date> </> peut porter dans un premier temps sur l'entité complète (donc la paire : <_eve> = le festival de Cannes de 2006) puis, au fur et à mesure de la descente dans les stratégies de repli, sur les différentes sous-entités qui la composent de manière indépendante :

- paires : <_Eve> = festival de Cannes, <_date> = 2006

- paires : <_Eve> = festival, <_ville> = Cannes, <_date> = 2006

Par contre, cette approche par sacs de mots typés limite la remontée des bonnes réponses puisqu'aucune information positionnelle n'est utilisée, à l'inverse de ce qui se produit lors de l'utilisation de patrons syntaxiques. Ainsi, si deux réponses candidates proviennent du même passage et que la deuxième constitue la bonne réponse, elle apparaît au mieux en seconde position. Si en plus la première apparaît dans d'autres passages alors la deuxième sera classée encore plus bas par le classifieur, celui-ci n'utilisant que la fréquence absolue des candidats. Une des pistes possibles pour pallier ce genre de problème est d'utiliser l'information positionnelle. Cette information pourrait être utilisée pour, par exemple, pondérer la fréquence des réponses trouvées.

Un autre aspect sur lequel nous allons axer nos prochains travaux est l'apprentissage automatique des sacs de mots typés et leur pondération en fonction de la qualité des réponses candidates associées. Cela suppose l'utilisation de données d'apprentissage pour la constitution des sacs de mots typés et d'un corpus de développement pour leur pondération, de façon à valider les sacs sur des données différentes et ainsi éviter les risques liés à un sur-apprentissage. Ceci devrait nous permettre de limiter une des faiblesses de notre approche actuelle liée à l'utilisation de techniques de repli. Si nous sommes capables de pondérer les différentes formes de requêtes générées à partir d'une question et surtout de sélectionner les plus pertinentes, alors nous pourrions pour chaque question utiliser plusieurs requêtes au lieu d'une seule et récupérer davantage de réponses (pondérées). Actuellement, compte tenu de la contrainte de temps et de la combinatoire possible pour chaque requête (une requête sur laquelle on applique toutes les techniques de repli génère plusieurs centaines de requêtes), il est nécessaire d'utiliser des techniques de repli pour limiter la quantité de requêtes effectuées.

Enfin il est évident que cette approche souffre d'une grande faiblesse : elle ne tient absolument pas compte des variantes, qu'il s'agisse des variantes sémantiques ou des dérivations morphologiques.

6. Interaction et génération en langue des énoncés du système

Le système doit être capable de produire deux types d'énoncés : il doit pouvoir gérer les interactions purement dialogiques, concernant la gestion générale de l'interaction (demandes de répétition, relances, formalités d'ouverture et de fermeture de dialogue, etc.), et il doit être capable de formuler des réponses pour les questions qu'il reçoit. Les énoncés de l'utilisateur qui n'entrent pas dans une de ces catégories produisent une réaction de non compréhension du système.

Le caractère naturel des énoncés produits par le système est particulièrement important pour favoriser une interaction naturelle avec l'utilisateur. Cela implique notamment d'avoir recours aux répétitions pour réaliser des confirmations implicites, et d'indiquer la confiance du système dans ses réponses par le biais de formulations appropriées. Il apparaît donc essentiel d'établir une interaction étroite entre le composant de génération et les composants d'extraction d'information et de dialogue.

6.1. Énoncés du système pour l'interaction dialogique

La nature du projet RITEL impose que le système soit en toutes circonstances capable de produire des énoncés pour communiquer avec l'utilisateur, ne serait-ce que pour assurer la fonction phatique (entretien du canal de communication). Ainsi, dans des cas où le système a terminé son tour de parole précédent et que l'utilisateur ne s'exprime plus pendant un certain temps, nos expériences ont montré qu'il est nécessaire d'indiquer la disponibilité ainsi que la réactivité du système à l'utilisateur par le biais de phrases telles que *Oui ?* ou *Je ne vous entends plus. Avez-vous d'autres questions que vous souhaiteriez poser ?* De même, il apparaît important d'inviter l'utilisateur à formuler d'autres questions lorsqu'une réponse vient d'être donnée.

Les interactions dialogiques peuvent être traitées de façon générale par le biais de patrons de texte préparés (*canned text* (Reiter *et al.*, 2000)), qui peuvent contenir des variables et des appels de fonctions et appeler récursivement d'autres patrons. L'utilisation d'une approche par jetons sur une liste chaînée avec bouclage permet de visiter plusieurs paraphrases pour un même but communicatif et ainsi d'éviter des répétitions trop rapprochées dans le cadre d'un dialogue. Outre des énoncés pour garder ouvert le canal de communication (ex : *Oui, je vous écoute*), les buts communicatifs pour lesquels le système peut formuler des énoncés recouvrent principalement : salutation de l'utilisateur (ex : *Bienvenue sur le système Ritel*) ; demandes de répétition (ex : *Pouvez-vous répéter votre question ?*) ; demandes de reformulation/précision (ex : *Vous voulez connaître la capitale de quel pays ?*) ; demandes de nouvelle question (ex : *Autre chose ?*) ; indications de non-compréhension (ex : *Je ne parviens pas*

à vous comprendre); demandes de patience (ex : *Je fais l'analyse des 337 documents que j'ai trouvés et je vous donne la réponse*); propositions d'aide (ex : *Souhaitez-vous écouter le guide ?*).

6.2. Énoncés du système pour le retour d'information

La génération de réponses naturelles n'a pas attiré un intérêt significatif dans le cadre des systèmes de Questions-Réponses jusqu'à récemment, cet aspect n'entrant pas en jeu dans les métriques d'évaluation traditionnelles de ces systèmes. Or, dans le contexte d'un système tel que RITEL il est indispensable que l'utilisateur soit convaincu du caractère *coopératif* du système au travers des formulations qu'il produit. Les approches les plus prometteuses à notre sens pour permettre cela incluent notamment l'aide à la construction et au raffinement de la recherche, la justification des réponses et la suggestion de voies de recherche (Benamara, 2004), ou encore l'enrichissement des réponses, lorsque cela est approprié, par des éléments de réponses ou des parties de documents (Marsi *et al.*, 2005; Bosma, 2005).

Le système doit être capable de produire des énoncés de façon robuste, quels que soient l'entrée du système et le résultat de son analyse. Le composant de génération essaie donc tout d'abord des techniques plus complexes, et a recours à plusieurs techniques de repli. La version actuelle du système peut donner de une à plusieurs réponses en les énumérant, et formule des réponses selon une échelle de confiance à cinq niveaux (de *très certain* à *très incertain*).

Une réponse naturelle reprend souvent la formulation de la question, ne serait-ce que pour effectuer une confirmation implicite. Or il est difficile d'effectuer une analyse fine des transcriptions produites par le composant de reconnaissance de la parole du fait des erreurs qu'elles contiennent.⁶ Un niveau intermédiaire de réponse consiste à utiliser le résultat de l'analyse des questions (type de la réponse et clefs de recherche), et d'associer à des patrons d'analyse des patrons de génération, ce qui permet de couvrir rapidement des types de questions récurrentes (ex : *&prefi xe_confiance(\$score) la ville de \$ville se trouve &en_pays(\$pays)*). Cette approche peut néanmoins produire des erreurs si l'analyse de la question est elle-même erronée, mais elle donne au système global un fonctionnement cohérent.

En cas de non application d'une technique plus complexe, le système produit une réponse qui n'utilise que les réponses extraites et leur score. Les cinq niveaux de confiance retenus sont associés à des patrons (ex : de *C'est \$reponse.* à *J'ai trouvé \$reponse comme réponse, mais je n'en suis pas du tout certain.*).

Enfin, il peut être important de commencer à donner une réponse (en apparence) alors que la recherche d'information n'est pas terminée. Pour cela, le composant de génération peut essayer de produire des préfixes de réponse et ainsi laisser d'im-

6. Nous prévoyons cependant d'évaluer la qualité des questions transcrites afin d'en extraire, lorsque c'est possible, des fragments réutilisables à l'aide d'une analyse syntaxique robuste.

portantes secondes supplémentaires pour la recherche d'information. Par exemple, le patron *Et bien, d'après les \$nombre_docs documents que j'ai trouvés, &patron_ville_dans_pays({ville1 => \$ville})* permet de produire un préfixe tel que *Et bien, d'après les 404 documents que j'ai trouvés, la ville de Barcelone est &en_pays(\$pays)*. Le composant de synthèse de la parole peut alors prononcer l'énoncé jusqu'à la partie non instanciée, puis le résultat de l'évaluation de cette partie lorsque la réponse a finalement été extraite (*en Espagne*, pour cet exemple). L'utilisation de préfixes de réponse permet également de réaliser des confirmations implicites, qui permettent à l'utilisateur d'interrompre le système s'il réalise que celui-ci ne répond pas à la bonne question.

7. Perspectives : interaction et recherche d'information avancées

Le système doit être capable d'aider l'utilisateur dans la construction de sa requête, ce qui implique de pouvoir poser des questions discriminantes lorsque le composant d'extraction des réponses signale des ambiguïtés. Il faut cependant que le système puisse déterminer si une hypothèse est plus vraisemblable que toutes les autres, et indiquer dans sa formulation de la réponse de laquelle il s'agit. Par exemple, une référence à la *Tour Eiffel* sera ambiguë au sens d'une recherche d'information sur Internet, mais il est néanmoins possible de décider que la Tour Eiffel de Paris est vraisemblablement celle dont parle l'utilisateur, et de reprendre cet élément dans la formulation de la réponse pour permettre à l'utilisateur de corriger : *La hauteur de la Tour Eiffel de Paris est de 300 mètres*.

L'acceptabilité du système passera notamment par sa capacité à indiquer sa confiance dans les réponses qu'il produit. Certaines heuristiques d'extraction d'information font que des informations sûres peuvent être clairement identifiées comme telles, comme dans *D'après le site Internet de la Maison Blanche, le Président Américain en 1944 était Franklin D. Roosevelt*.⁷ En cas de réelle incertitude du système, il faut indiquer que d'autres solutions existent et éventuellement les donner à la demande de l'utilisateur, ex : *Je crois que la chanson Yesterday est interprétée par les Beatles, mais il existe de nombreuses autres réponses. Voulez-vous les connaître ?*

Pour une même question, des réponses à des granularités différentes peuvent exister. Il faut donc que le système puisse décider de la granularité de la réponse à fournir, par exemple en utilisant la plus précise trouvée (ex : *Henri IV a été assassiné le 14 mai 1610* plutôt que *en 1610*). Dans le cas où les réponses possibles sont identifiées comme étant de types différents, une question du système demandant le type d'information recherchée peut être formulée.

Les réponses *complétives* visent à apporter des informations complémentaires qui n'ont pas été explicitement demandées par l'utilisateur mais qui sont susceptibles de l'intéresser. Ces éléments peuvent être sélectionnés sur la base de leur cooccurrence

7. Un de nos projets futurs vise à nommer automatiquement des pages web pour utiliser ces dénominations dans ce type de réponses.

avec l'élément recherché et de leur type. Par exemple, une réponse à la question *Quand a été assassiné Henri IV ?* telle que *Henri IV a été poignardé le 14 mai 1610* apporte une information supplémentaire sur le type d'assassinat par le biais de l'utilisation de l'hyperonyme *poignarder* du verbe *assassiner*. Cependant, il est important de donner les éléments demandés le plus rapidement possible, afin de permettre éventuellement à l'utilisateur d'interrompre le système. Il est bien évidemment important que ces informations supplémentaires n'apparaissent pas déjà dans l'historique des recherches.

Enfin, formuler des questions *suggestives* consiste à identifier des patrons répétés de questions et faire des invites permettant de regrouper des questions liées (ex : U : *Qui est le président du Mexique ? ... De la Finlande ? ...* S : *Pour quels autres pays souhaiteriez-vous connaître les présidents ?*) ou à permettre d'explorer des thèmes lorsqu'il semble hésiter (ex : U : *Qui est le président français ? ... Le premier ministre ? ... (pause)* S : *Voudriez-vous d'autres informations sur le système politique français ?*).

8. Conclusion

Dans cet article, nous avons présenté la plateforme RITEL. L'objectif est une recherche d'information interactive, avec deux modalités d'entrée : l'écrit et l'oral. La principale contrainte réside dans la vitesse nécessaire pour fournir à un utilisateur des réponses, ou à défaut une réaction. La plateforme actuelle répond à ce premier besoin. Le système de reconnaissance tourne à 0,5 fois temps réel et la recherche d'information ne prend que quelques millisecondes. Ceci permet à l'utilisateur d'avoir une réponse/réaction immédiate de la part du système.

La majeure partie du travail présenté dans cet article, dont l'analyse non contextuelle et la recherche d'information, est fondée sur l'analyse de corpus. L'analyse non contextuelle vise à la détection d'entités spécifiques et hiérarchiques le cas échéant ainsi qu'à la détection de chunks généralistes (noms composés, verbes, adjectifs simples ou composés, etc.). Actuellement, aucune dépendance entre ces chunks n'est calculée sauf pour les entités hiérarchiques qui sont également des chunks mais typés. Cette analyse est une analyse légère et de haut niveau. Elle utilise un moteur d'expressions régulières de mots. Elle permet d'analyser aussi bien des énoncés et questions écrits et oraux que des documents écrits ou transcrits de l'oral. Nos mesures initiales ont permis d'asseoir cette approche. Les premiers résultats en terme de détection d'entités nommées et étendues montrent une F-mesure allant de 0,82 sur des données radio- et télé-diffusées transcrites manuellement à 0,88 pour des énoncés oraux (collectés avec la première version de la plateforme RITEL).

La recherche d'information est effectuée soit par l'intermédiaire d'un module de recherche d'information générale, soit par l'utilisation de systèmes dédiés. Le module de recherche d'information générale permet une recherche d'information dans de larges collections de documents. Celle-ci est essentiellement fondée sur l'analyse non contextuelle présentée. Les résultats préliminaires obtenus sur le corpus de l'évaluation CLEF'05 ont montré un MRR d'environ 45% et un taux de réponses justes en

premier rang de près de 33%. Les questions strictement factuelles obtiennent un MRR d'environ 60%.

Enfin, nous avons implémenté une première version du composant de génération des réponses, et nous avons identifié plusieurs voies d'amélioration pour l'interaction et la recherche d'information, notamment la génération de réponses complétives supposées devancer des besoins d'information de l'utilisateur tout en répondant aux questions posées. La figure 6 donne un exemple de dialogue que nous obtenons actuellement.

| |
|--|
| <p>S : <i>Salut, c'est Ritel. Que voulez-vous ?</i> U : quelle est le premier ministre de la France S : <i>La réponse est Balladur. Avez-vous une autre question ?</i> U : sa capitale S : <i>La réponse est Paris. Une autre question ?</i> U : celle de l' Italie S : <i>Rome me semble être la réponse. Souhaitez-vous me poser une autre question ?</i> U : son premier ministre S : <i>C'est sûrement Silvio Berlusconi. Une autre question ?</i> U : euh oui S : <i>Oui, que voulez-vous ?</i> U : euh rien merci S : <i>Au-revoir et merci d'avoir utilisé Ritel.</i></p> |
|--|

Figure 6. Exemple de dialogue

9. Bibliographie

- Abney S., « Chunks and Dependencies : Bringing Processing Evidence to Bear on Syntax », *Computational Linguistics and the Foundations of Linguistic Theory*, 1995.
- ACE, « Entity Detection and Tracking, Phase 1, ACE Pilot Study. Task Definition », <http://www.nist.gov/speech/tests/ace/phase1/doc/summary-v01.htm>, 2000.
- Allen J., al., « The TRAINS project : A Case Study in Defining a Conversationnal Planning Agent », *Journal of Experimental and Theoretical AI*, 1995.
- Ayache C., Grau B., Vilnat A., « Evaluation of question-answering systems : The French EQueR-EVALDA Evaluation Campaign », *In Proceedings of LREC'06*, Genoa - Italy, 2006.
- Benamara F., WebCoop : Un système de réponses coopératives sur le Web, Thèse de doctorat, Université Paul Sabatier, Toulouse, 2004.
- Bikel D., Miller S., Schwartz R., Weischedel R., « Nymble : a high-performance learning name-finder », *In Proceedings of ANLP'97*, 1997.
- Blaudez E., Crestan E., de Loupy C., « SQuAR : Prototypé de moteur de questions réponses », *In Proceedings TALN'05*, Dourdan, France, 2005.
- Bonneau-Maynard H., Ayache C., Bechet F., Denis A., Kuhn A., Lefevre F., Mostefa D., Qui-gnard M., Rosset S., Servan C., Villaneau J., « Results of the French Evalda-Media evaluation campaign for literal understanding », *In LREC'06*, Genoa, Italy, May 2006.

- Bosma W., « Extending Answers using Discourse Structure », in *RANLP Workshop on Crossing Barriers in Text Summarization Research*, Borovets, Bulgaria, 2005.
- Courtois B., Silberztein M., « Les dictionnaires électroniques du français », *Langue Française*, 1990.
- den Os E., Boves L., « Interactivity and Multimodality in the IMIX Demonstrator », *Proceedings of IEEE International Conference on Multimedia & Expo*, Amsterdam, 2005.
- Favre B., Bechet F., Nocéra P., « Robust Named Entity Extraction from Large Spoken Archives », *Proceedings of HLT-EMNLP'05*, Vancouver, Canada, p. 491-498, October, 2005.
- Galibert O., Illouz G., Rosset S., « Ritel : An Open-Domain, Human-Computer Dialog System », in *InterSpeech'05*, Lisbon, Portugal, 2005.
- Gauvain J.-L., Adda G., Adda-Decker M., Allauzen A., Gendner V., Lamel L., Schwenk H., « Where Are We in Transcribing French Broadcast News ? », *InterSpeech*, Lisbon, 2005.
- Glass J. R., Polifroni J., Seneff S., Zue V., « Data collection and performance evaluation of spoken dialogue systems : the MIT experience », in *ICSLP'00*, Pékin, Chine, 2000.
- Gravier G., Bonastre J., Geoffrois E., Galliano S., McTait K., Choukri K., « ESTER, une campagne d'évaluation des systèmes d'indexation automatique d'émissions radiophoniques en français », in *Proceedings of JEP'04*, Fèz, Maroc, 2004.
- Grishman R., « Where's the syntax ? The NYU MUX-6 system », in *Proc. MUC-6*, San Francisco, 1995.
- Harabagiu S., Moldovan D., Picone J., « Open-domain voice-activated question answering », *Proceedings of COLING*, p. 1-7, 2002.
- Hersh W., Over P., « TREC-8 Interactive Track Report », in *Voorhees and Buckland*, 2000.
- Isozaki H., Kazawa H., « Efficient Support Vector Classifiers for Named Entity Recognition », *Proceedings of COLING*, 2002.
- Kamps J., Marx M., de Rijke M., Sigurbjörnsson B., « XML retrieval : What to retrieve ? », *Proceedings of ACM-SIGIR'03*, p. 409-410, 2003.
- Kato T., Fukumoto J., Masui F., « An overview of NTCIR-5 QAC3 », *Proceedings of NTCIR-5 Workshop Meeting*, Tokyo, Japan, 2005.
- Kato T., Fukumoto J., Masui F., Kando N., « Handling Information Access Dialogue through QA Technologies - A novel challenge for open-domain question answering », in S. Harabagiu, F. Lacatusu (eds), *HLT-NAACL 2004 : Workshop on Pragmatics of Question Answering*, Boston, EU, p. 70-77, 2004.
- Kaufman M. (ed.), *Proceedings of the Seventh Message Understanding Conference*, 1998.
- Lamel L., Rosset S., Gauvain J., Bennacef S., Garnier-Rizet M., Prouts B., « The LIMSI ARISE System », *Speech Communication*, vol. 31, n° 4, p. 339-354, 2000.
- Laurent D., Séguéla P., Niège S., « Cross Lingual Question Answering using Qristal for CLEF 2005 », *Working Notes for the CLEF 2005 Workshop*, Vienna, Austria, 2005.
- Marsi E., Krahmer E., « Explorations in Sentence Fusion », in *Proceedings of the 10th European Workshop on Natural Language Generation*, Aberdeen, Scotland, 2005.
- Miller D., Boisen S., Schwartz R., Stone R., Weischedel R., « Named entity extraction from noisy input : Speech and OCR », *Proceedings of ANLP'00*, San Francisco, EU, 2000.
- Oard D., Gonzalo J., « The CLEF 2001 Interactive Track », *Working Notes for the CLEF 2001 Workshop*, Darmstadt, Germany, 2001.

- Over P., « TREC-7 Interactive Track Report », *In Voorhees and Buckland*, 1999.
- Poibeau T., « Sur le statut référentiel des entités nommées », *In Proceedings TALN'05*, Dourdan, France, 2005.
- Raux A., Langner B., Bohus D., Black A. W., Eskenazi M., « Let's Go Public ! Taking a Spoken Dialog System to the RealWorld », *in InterSpeech'05*, Lisbon, Portugal, 2005.
- Reiter E., Dale R., *Building Natural Language Generation Systems*, Cambridge University Press, Cambridge, 2000.
- Rosset S., Stratégies et gestionnaire de dialogue pour des systèmes d'interrogation de bases de données à reconnaissance vocale, Thèse de doctorat, Université Paris Sud, Orsay, 2000.
- Rosset S., Petel S., « The Ritel Corpus - An annotated Human-Machine open-domain question answering spoken dialog corpus », *In Proceedings of LREC'06*, Genoa, Italy, 2006.
- Salter A. L. I., Temem J., Bernard F., Rosset S., Bennacef S., Lamel L., « Data collection for the Mask kiosk : WOz vs prototype system », *in ICSLP'96*, Philadelphia, USA, 1996.
- Sekine S., « Definition, dictionaries and tagger of Extended Named Entity hierarchy », *in LREC'04*, Lisbon, Portugal, 2004.
- Small S., Shimizu N., Strzalkowski T., Ting L., « HITIQA : A Data Driven Approach to Interactive Question Answering : A Preliminary Report », *In Proceedings of AAAI Spring Symposium on Advances in Question Answering*, Stanford, EU, 2003.
- Surdeanu M., Turmo J., Comelles E., « Named Entity Recognition from spontaneous Open-Domain Speech », *in InterSpeech'05*, Lisbon, Portugal, 2005.
- TREC, « AQUAINT - ARDA », , <http://informedia.cs.cmu.edu/aquaint/index.html>, n.d.a.
- TREC, « Text REtrieval Conference », , <http://trec.nist.gov>, n.d.b.
- Vallin A., Giampiccolo D., Aunimo L., Ayache C., Osenova P., Peñas A., de Rijke M., Sacaleanu B., Santos D., Sutcliffe R., « Overview of the CLEF 2005 Multilingual Question Answering Track », *Working Notes for the CLEF 2005 Workshop*, Vienna, Austria, 2005.
- Villaneau J., Contribution au traitement syntactico-pragmatique de la langue naturelle parlée : approche logique pour la compréhension de la parole, PhD thesis, Université de Bretagne Sud, France, 2003.
- Voorhees E. M., « The Fourteenth Text REtrieval Conference Proceedings (TREC 2005) », *In Voorhees and Buckland*, 2005.
- Walker M., Rudnicky A., Aberdeen J., Bratt E., Garofolo J., Hastie H., Le A., Pellom B., Potamianos A., Passonneau R., Prasad R., Roukos S., Sanders G., Seneff S., Stallard D., « DARPA Communicator Evaluation : Progress from 2000 to 2001 », *in ICSLP'02*, Denver, EU, 2002.
- Wolinski F., Vichot F., Dillet B., « Automatic Processing of Proper Names in Texts », *In Proceedings of EACL'95*, 1995.
- Zue V., Glass J., « Conversational Interfaces : Advances and Challenges », *Proceedings of the IEEE, Special Issue on Spoken Language Processing*, 2000.