

# Une grammaire multilingue partagée pour la traduction automatique de la parole

Pierrette Bouillon<sup>1</sup>, Manny Rayner<sup>1</sup>, Bruna Novellas<sup>1</sup>, Yukie Nakao<sup>2</sup>,  
Marianne Santaholma<sup>1</sup>, Marianne Starlander<sup>1</sup>, Nikos Chatzichrisafis<sup>1</sup>

<sup>1</sup>Université de Genève

pierrette.bouillon@issco.unige.ch, mrayner@riacs.edu, novella2@etu.unige.ch,  
{marianne.santaholma ; marianne.starlander}@eti.unige.ch,  
nikos.chatzichrisafis@vozzup.com

<sup>2</sup>National Institute for Communications Technology  
yukie-n@khn.nict.go.jp

## Résumé

Aujourd'hui, l'approche la plus courante en traitement de la parole consiste à combiner un reconnaiseur statistique avec un analyseur robuste. Pour beaucoup d'applications cependant, les reconnaiseurs linguistiques basés sur les grammaires offrent de nombreux avantages. Dans cet article, nous présentons une méthodologie et un ensemble de logiciels libres (appelé *Regulus*) pour dériver rapidement des reconnaiseurs linguistiquement motivés à partir d'une grammaire générale partagée pour le catalan et le français.

**Mots-clés** : traduction automatique de la parole, modélisation du langage, grammaire d'unification, reconnaissance linguistique, grammaires multilingues.

## Abstract

Today, the most common architecture for speech understanding consists of a combination of statistical recognition and robust semantic analysis. For many applications, however, grammar-based recognisers can offer significant advantages. In this paper, we present a methodology and an Open Source platform (*Regulus*), which together permit rapid derivation of linguistically motivated recognisers, in either language, from a bilingual grammar of Catalan and French.

**Keywords**: speech to speech translation, language modelling, unification grammar, grammar-based recognition, multilingual grammars.

## 1. Introduction

Aujourd'hui, l'approche la plus courante en traitement de la parole consiste à combiner un reconnaiseur statistique avec un analyseur robuste. Pour beaucoup d'applications cependant, les reconnaiseurs linguistiques basés sur les grammaires offrent de nombreux avantages. On sait qu'entraîner un reconnaiseur statistique requiert des données importantes, qui ne sont pas toujours disponibles au début du projet. Certaines comparaisons ont aussi montré que, pour les applications où la précision est plus importante que la robustesse, la reconnaissance linguistique tend à donner de meilleurs résultats que sa rivale (Knight *et al.*, 2001 ; Rayner *et al.*, 2005a). De plus, comme la grammaire développée pour la reconnaissance peut également être utilisée pour

14 BOUILLEON, RATNER, NOVELLAS, NAKAO, SANTAHOLMA, STARKLANDER, CHATZICHRISAMIS

l'analyse syntaxique, elle évite par là même le développement d'un analyseur robuste.

De nos jours, la plupart des plateformes vocales commerciales permettent à la fois la reconnaissance statistique et linguistique. Malgré les avantages mentionnés plus haut, la préférence va cependant toujours aux reconnaisseurs statistiques. La raison en est simple : les grammaires requises par l'approche linguistique doivent, aujourd'hui encore, être décrites dans le formalisme des grammaires indépendantes de contexte (CFG), voire même parfois dans des formalismes moins expressifs. Or personne n'ignore qu'avec ce type de langage, les grammaires importantes sont difficiles à développer. Même les petites grammaires *ad-hoc* pour des domaines limités deviennent vite redondantes et impossibles à maintenir, surtout si elles doivent couvrir des domaines liés.

Pour tirer parti de la précision des reconnaisseurs linguistiques, tout en évitant le formalisme CFG, différentes solutions peuvent être envisagées. La plus répandue consiste à développer des modèles du langage dans des formalismes grammaticaux plus puissants, typiquement celui des grammaires d'unification (GU), et à les compiler ensuite dans le formalisme CFG requis par les reconnaisseurs (Moore, 1998 ; Dowding *et al.*, 2001). Regulus est l'un des outils conçus dans ce but. Il s'agit d'une plateforme de logiciels libres (Regulus, 2005 ; Rayner *et al.*, 2006), développée dans le cadre du projet MedSLT, un système de traduction automatique de la parole dans le domaine médical, spécialisé pour le diagnostic d'urgence (Bouillon *et al.*, 2005).

Avec Regulus, nous voulons atteindre le plus haut niveau d'abstraction possible. Notre but est d'avoir une seule grammaire générale et motivée linguistiquement, qui pourra ensuite être réutilisée pour différents domaines et tâches, et partagée entre plusieurs langues. Pour un projet de traduction comme MedSLT, le gain est non négligeable. MedSLT utilise en effet un reconnaisseur et un système de traduction linguistiques. Il requiert ainsi une grammaire par sous-tâche (reconnaissance/analyse et génération) pour chaque sous-domaine du diagnostic (maux de tête, douleurs abdominales, douleurs thoraciques et maux de gorge) et, ceci, dans les six langues du système (français, catalan, finnois, anglais, japonais, espagnol). Sans Regulus, nous arriverions ainsi à un total de quarante-huit grammaires différentes !

Dans d'autres articles, nous avons déjà décrit comment dériver avec Regulus des reconnaisseurs linguistiques à partir des grammaires générales monolingues. Nous ne nous y attarderons donc plus ici. En résumé, nous partons d'une grammaire d'unification générale, motivée linguistiquement et de lexiques spécialisés avec le vocabulaire des sous-domaines du diagnostic pour chaque langue. Nous spécialisons ensuite cette grammaire d'unification générale dans des grammaires d'unification spécialisées par domaine avec des méthodes d'apprentissage basées sur la méthode de l'EBL (Explanation Based Learning ; (Rayner *et al.*, 2006), Chap. 10). Cette méthode est paramétrée : 1) par un corpus du domaine d'où le système va apprendre le vocabulaire et les types de phrases qui doivent figurer dans la grammaire spécialisée et 2) par des critères d'opérationnalité («operationality criteria») qui contrôlent la granularité de la grammaire spécialisée (à quelles généralisations la grammaire doit-elle aboutir à partir du corpus ?). Ces grammaires d'unification spécialisées sont ensuite compilées dans le format CFG, requis par notre plateforme vocale (Nuance, pour le projet medSLT). Grâce à cette méthodologie, nous combinons la précision et la performance des systèmes ad-hoc, développés pour un domaine spécifique, avec tous les avantages bien connus des grammaires générales : facilité de maintenance et de développement.

Dans cet article, nous montrons comment pousser encore plus loin le degré d'abstraction. Dans la lignée des travaux de (Bender *et al.*, 2002), nous proposons d'écrire une grammaire d'unification partagée pour différentes langues. La difficulté dans ce contexte particulier est que cette grammaire doit pouvoir être compilée efficacement dans des reconnaisseurs pour les différentes

langues. Pour cette première expérience, nous travaillons avec deux langues très proches (le catalan et le français), mais il ne s'agit ici que du préalable obligé d'une recherche plus ambitieuse qui consiste à construire une grammaire partagée pour toutes les langues du système. Dans la suite, nous présentons d'abord les spécificités des grammaires Regulus, par rapport à des grammaires conçues pour l'écrit. Nous décrivons ensuite la grammaire générale partagée pour le catalan et le français, puis nous montrons comment cette grammaire partagée est spécialisée pour les trois tâches de reconnaissance, d'analyse et de génération. Dans la dernière section, nous évaluons cette grammaire pour ces trois tâches.

## 2. Les grammaires d'unification Regulus : spécificités

Avec Regulus, nous voulons arriver à une seule grammaire d'unification générale, motivée linguistiquement, qui sera compilée pour la reconnaissance, l'analyse et la génération. Cette grammaire générale doit ainsi combiner différentes spécificités, propres à ces différentes tâches.

Du fait que cette grammaire doit pouvoir être transformée pour la reconnaissance dans des grammaires CFG, il faut tout d'abord que tous les traits aient un nombre fini de valeurs, le plus limité possible. En pratique, ceci signifie que les attributs ne peuvent pas prendre des valeurs complexes et que l'approche lexicaliste, défendue par LFG ou HPSG, devient difficile à mettre en oeuvre. Par exemple, la rection ne peut pas être traitée ici avec des schémas de règles généraux, comme dans HPSG. Il faut au contraire multiplier les règles syntagmatiques pour chaque type de verbes (intransitif, transitif, etc.). Bien que cette première contrainte conduise clairement à des grammaires moins élégantes et plus répétitives, ceci ne semble pas un frein au développement de grammaires de la complexité requise pour ce type d'application.

Comme il s'agit avant tout d'une grammaire pour la reconnaissance, elle doit aussi intégrer toutes les informations susceptibles d'améliorer ce processus. Par exemple, différentes évaluations ont montré que les restrictions de sélection ne peuvent pas être omises sans dégrader considérablement la reconnaissance (Rayner *et al.*, 2006). En pratique donc, toutes les grammaires Regulus générales contiennent plusieurs traits pour gérer ce type de contrainte. Par exemple, les noms sont typés sémantiquement ; les entrées verbales contiennent quant à elles des traits qui précisent le type de compléments, en fonction de leur rection, etc. Ces types, difficiles à définir de manière cohérente pour le vocabulaire général, ne posent normalement pas de problèmes pour les applications liées à la parole où le domaine est bien cerné et le vocabulaire assez limité. Ils n'ont d'ailleurs aucune influence sur la structure globale de la grammaire générale, puisqu'ils proviennent des lexiques spécialisés pour les différents domaines.

Dans Regulus, la reconnaissance et l'analyse (syntaxique et sémantique) sont réalisées par la plateforme Nuance, après conversion de la grammaire d'unification dans le format requis par Nuance. La représentation sémantique est donc, elle aussi, particulière à ce type d'applications. Nuance construit en effet la sémantique de la phrase de manière compositionnelle, par concaténation de la sémantique des mots ou des phrases. Dans notre cas, le résultat de l'analyse est le plus simple possible. Il s'agit d'une représentation plate, formée d'une liste de paires attributs-valeurs, avec un seul niveau d'enchâssement pour les phrases subordonnées (Rayner *et al.*, 2005b). Par exemple, *avez-vous mal à la tête quand vous buvez ?* sera représenté de la manière suivante :

```
[ [sc, quand], [clause, [[pronoun, vous], [voice, active],
                        [tense, present], [action, boire]]],
  [path_proc, avoir], [pronoun, vous], [symptom, mal],
```

[tense, present], [utterance\_type, sentence],  
[voice, active], [locative\_prep, à], [body\_part, tête]]

Ces représentations plates présentent différents avantages, en particulier elles facilitent le transfert vers l'interlangue ou à partir de celle-ci lors de la traduction (Rayner *et al.*, 2005b). Dans le même souci de simplicité, nous avons choisi de ne pas faire figurer les déterminants dans la représentation. Les évaluations ont en effet montré que ces derniers sont difficiles à reconnaître correctement et à traduire, une fois reconnus. Il en va de même pour l'information sur le nombre. Ce choix aura évidemment un impact sur la manière de dériver la grammaire de génération puisqu'elle devra être beaucoup plus contrainte que la grammaire générale dont elle est pourtant issue - notamment, elle devra associer à chaque nom le déterminant le plus approprié.

Dans l'approche défendue ici, la compilation de la grammaire de génération à partir de la grammaire générale se fait par une méthode similaire à celle de la « Semantic Head Driven Generation ». Comme nous venons de le suggérer, la reconnaissance et la génération ont cependant des exigences différentes. Pour la génération, la grammaire doit être très contrainte de manière à ne produire qu'un seul résultat pour chaque représentation ; pour la reconnaissance, nous voulons rester beaucoup plus souple puisque nous ne pouvons pas prédire la structure utilisée par l'utilisateur. Il y aurait évidemment différentes manières de poser la question plus haut : *Avez-vous ce mal à la tête quand vous buvez ? Avez-vous des maux de tête quand vous buvez ? Avez-vous vos maux de tête quand vous buvez ? Avez-vous mal de tête quand vous buvez ?*, etc. Pour répondre à ces exigences différentes, les grammaires générales Regulus décrivent donc toutes les tournures utiles pour notre domaine. Celles-ci seront ensuite spécialisées différemment pour les deux tâches de reconnaissance et de génération, grâce à des corpus et des règles de spécialisation différentes. Dans la suite, nous illustrons cette méthodologie avec la grammaire partagée pour le français et le catalan.

### 3. La grammaire partagée

La grammaire générale partagée a été conçue pour MedSLT. Elle décrit pour l'instant le français et le catalan et couvre les différents domaines du diagnostic vus plus haut. Elle comprend trois grandes parties : les règles syntagmatiques communes aux deux langues, plus deux ensembles de règles spécifiques.

Comme les deux langues sont très proches, les règles spécifiques ne sont pas très nombreuses. En français, elles sont au nombre de quatre et décrivent les questions oui-non, formées avec *est-ce que* (*Est-ce que vous avez mal à la tête ?*), par inversion du pronom sujet avec ajout d'un tiret (*avez-vous mal à la tête ?*) ou par inversion complexe (*La douleur est-elle latérale ? La douleur est-elle causée par le vin rouge ? En combien de temps la douleur irradie-t-elle la nuque ?*) ; la grammaire spécifique catalane, elle, contient six règles pour les phrases déclaratives et interrogatives sans sujet (*Té mal de cap ?*, « avez mal de tête ? »), ainsi que les phrases interrogatives sans inversion (*Quan el dolor apareix ?*, « quand la douleur apparaît ? ») ou avec inversion du nom ou du pronom sujet (*Quan apareix el dolor ?*, « quand apparaît la douleur ? » ; *Quan té vostè dolor ?*, « quand avez vous douleur »). Les questions sans inversion sont bien sûr aussi courantes en français dans la langue parlée (*Vous avez mal de tête le soir ?*), mais elles ne semblent pas utilisées par les médecins dans ce domaine - elles ne font donc pas encore partie de la grammaire commune.

La grammaire commune regroupe pour l'instant soixante-quatre règles syntagmatiques, paramétrables pour les deux langues. Le paramétrage se fait avec des macros quiinstancient une valeur

différente pour un attribut ou qui conduisent à un choix lexical spécifique pour les deux langues. Par exemple, dans la règle suivante :

```
vp_comps:[..., subcat=trans, ...] -->
  np:[@french_or_catalan(has_spec=yes, has_spec=_), ...].
```

la macro (caractérisée par le symbole @) rend compte du fait que, en français, le syntagme nominal objet des verbes transitifs requiert normalement un déterminant (sauf s'il s'agit d'un verbe support traité par une autre règle), alors qu'en catalan cette contrainte est moins forte (la valeur de l'attribut `had_spec=_` n'est donc pas spécifiée). De la même manière, la macro suivante spécifie que le pronom interrogatif sera réalisé différemment en français et en catalan :

```
wh_question:[...] -->
  @french_or_catalan('qu'est-ce', qui), què),
  vp:[inv=uninverted, ...].
```

Malgré ses spécificités pour la parole, la grammaire commune propose un traitement général des phénomènes les plus importants dans les langues romanes. Les clitiques ne sont pas traités avec des règles de mouvement, comme dans la plupart des grammaires (Rayner *et al.*, 2000). Comme nous ne pouvons pas avoir recours à des structures complexes, cette approche nous obligerait en effet à multiplier les traits, de manière à pouvoir faire passer les informations du constituant avec le verbe et le clitique à son *gap*. Pour éviter de trop complexifier la grammaire CFG, nous nous inspirons donc plutôt de (Miller et Sag, 1997) et nous proposons une approche lexicaliste où une règle de grammaire est utilisée comme une règle lexicale qui a ici la forme :

```
vbar:[ ] --> pronoun:[ ], vbar:[ ].
```

Par exemple, la règle suivante forme un `vbar` transitif à partir d'un pronom clitique et d'un `vbar` de type `ditransitif` (comme *donner* en français) :

```
vbar:[ subcat=trans ]
  -->
  pronoun:[ sem=Pron, pron_type=clitic, ... ],
  vbar:[ subcat=ditrans, ... ].
```

Ce `vbar` (par exemple, *vous donne*) pourra ensuite se combiner avec un objet direct, puis un sujet pour former une phrase complète.

Pour les mêmes raisons, l'inversion n'est pas non plus considérée comme un type de mouvement. À nouveau, nous exploitons le constituant `vbar` :

```
yn_question:[ ] --> vp:[ inv=inverted, ... ].

vp:[ inv=... ] --> vbar:[ inv=inverted\uninverted, ... ],
  vp_comps:[ ... ],
  optional_vp_postmods:[ ... ].
```

36 BOUILLEON, RATNER, NOVELLAS, NAKAO, SANTAHOLMA, STÅRLANDER, CHATZICHRISAMIS

Nous considérons qu'une question oui-non (`yn_question`) est formée d'un `vp` (inversé). Celui-ci est constitué d'un `vbar` et de son complément (`vp_comps`). Quand il est inversé, ce `vbar` pourra se réécrire de différentes manières, en fonction de la langue. En français, il peut s'agir d'un verbe suivi d'un tiret, d'un pronom et d'un éventuel adverbe :

```
vbar:[inv=inverted,...] -->
    verb:[], hyphen:[], pronoun:[], optional_adverb:[].
```

En catalan, nous aurons la variante suivante :

```
vbar:[inv=inverted,...] -->
    verb:[], (pronoun:[], np:[]), optional_adverb:[].
```

Cette approche ne complexifie pas la grammaire. Comme on le voit plus haut, la règle pour le `vp` s'applique en effet aux `vbar` inversé et non-inversé. De plus, elle permet de rendre facilement compte du fait que, en français, le `vbar` inversé peut être coordonné (*As-tu ou avez-vous déjà eu la grippe ?*).

Le traitement des questions *wh* est plus traditionnel que celui de l'inversion : il est impossible de les traiter sans au moins simuler le mouvement. Nous considérons donc que le pronom interrogatif est déplacé de sa position initiale (objet direct, PP, etc.), laissée vide, en position initiale (*quels médicaments<sub>i</sub> prenez-vous [i]*). Pour traiter le mouvement, nous utilisons le mécanisme standard du *gap threading*, proposé déjà par (Pereira, 1981). Le lien entre le constituant vide [*i*] et le constituant déplacé (*quels médicaments<sub>i</sub>* dans notre exemple) se fait avec les deux attributs `gapsin` et `gapsout`, qui apparaissent dans toutes les catégories impliquées par le mouvement. Par exemple, dans la règle suivante, ces attributs indiquent le `wh_np` n'est possible que si le `vp` contient un `np` vide (`gapsin=np_gap`).

```
wh_question:[ ] -->
    wh_np:[...],
    vp:[inv=inverted, gapsin=np_gap, gapsout=null].
```

Finalement, pour le passif, nous appliquons une règle lexicale. Celle-ci produit, à partir d'une entrée transitive au participe passé, une entrée passive avec la rection `passivised_trans` :

```
vbar:[voice=passive, subcat=passivised_trans, ....] -->
    verb:[vform=past_participle, subcat=trans, voice=active, ...],
    optional_adverb:[sem=Advp, position=post].
```

Ce `vbar` pourra ensuite se combiner avec un autre `vbar` (pour former un `vbar fini`) et un `VP_COMPS` (ici, un complément d'agent introduit par *par* ou *per*) pour former une phrase passive.

Dans la suite, nous voyons comment cette grammaire générale peut être spécialisée différemment pour la reconnaissance et la génération.

## 4. Spécialisation de la grammaire générale pour la reconnaissance et la génération

La spécialisation de la grammaire générale se fait automatiquement avec la méthode d'EBL citée plus haut. Pour arriver à des grammaires différentes pour la reconnaissance et la génération, nous faisons varier les deux paramètres vus plus haut : les corpus d'apprentissage et les critères d'opérationnalité. Les critères d'opérationnalité déterminent grâce à des règles la granularité de la grammaire spécialisée (en particulier l'ensemble des catégories possibles et comment elles se combinent) et précisent comment les règles de la grammaire d'unification générale doivent être restructurées (quels sont les niveaux d'enchâssement possibles dans la grammaire spécialisée ?). Les corpus donnent des exemples de mots et de types de structure qui doivent figurer dans la grammaire spécialisée.

Les deux versions de la grammaire spécialisée sont assez granulaires, incluent les mêmes catégories np, vp, pp et vbar et comprennent un vocabulaire d'approximativement mille formes fléchies. Certaines catégories disparaissent de la grammaire spécialisée comme celle pour les compléments des verbes (vp\_comps, dont nous avons parlé plus haut). Nous éliminons aussi les règles pour les constituants vides, utilisées notamment dans le traitement des questions *wh*. Celles-ci seront remplacées par les structures plates correspondantes, c'est-à-dire :

```
vp:[..., gaps_in=np_gap, gapsout=GOut, ...]-->
  vbar:[..., subcat=trans, ...],
  optional_pp:[].
```

De manière générale, il est intéressant de constater que le même niveau de granularité est utilisé pour les deux grammaires. Pour la génération, ce niveau suffit pour produire les bonnes structures, sans en surgénérer de mauvaises ; pour la reconnaissance, nous avons remarqué que, d'une part, les grammaires plus granulaires allourdissent le processus de décodage et que, d'autre part, les grammaires moins granulaires ne permettent plus assez de généralisations (Rayner *et al.*, 2006).

La principale différence entre les critères utilisés pour les deux grammaires concerne le traitement des NPs. Dans la grammaire pour la reconnaissance, nous généralisons, à partir du corpus, l'ensemble des déterminants et des noms. La grammaire spécialisée contient ensuite une règle qui forme de manière traditionnelle un NP à partir d'un déterminant et d'un nom. Celle-ci nous permet de reconnaître toutes les combinaisons des noms et des déterminants qui se trouvent dans le corpus, ce qui est bénéfique pour la reconnaissance. Pour la génération en revanche, les critères de granularité diffèrent : de manière à produire le déterminant plus approprié pour chaque nom sur base du corpus, nous apprenons des NP complets (lexicalisés), par exemple :

```
np --> vos maux de tête      np --> la douleur
```

Cette constatation nous amène à l'une des différences essentielles entre les corpus pour la génération et la reconnaissance ; le corpus de génération doit énumérer toutes les séquences de déterminants et de noms qui doivent pouvoir être générées. Par contre, le corpus de reconnaissance devra donner plus d'exemples puisqu'il doit couvrir toutes les variantes possibles au niveau du vocabulaire et des structures. Il contient, par exemple, des phrases interrogatives avec *est-ce que* (*Est-ce que vous avez mal sur le devant de la tête ?*), auxquelles on préférera

l'inversion lors de la génération (*Avez-vous mal sur le devant de la tête ?*). De même, ce corpus reprend aussi des phrases avec des déterminants élidés (*l', s'*, etc.) et des prépositions ou déterminants contractés (*au, aux, del*, etc.), qui pourront ainsi être reconnus par le système. Au niveau de la génération, ces différents phénomènes sont traités avec des règles orthographiques qui s'appliquent après la génération de la phrase. Cette approche nous évite de complexifier la grammaire générale avec les nombreux traits potentiellement nécessaires pour contraindre des phénomènes comme l'élision ou la contraction au niveau des règles elles-mêmes. Comment évaluer et comparer les deux grammaires spécialisées ?

## 5. Évaluation

Pour évaluer les grammaires de reconnaissance et de génération et les comparer entre elles, nous avons commencé par collecter des données en français et en catalan auprès de sujets qui jouaient le rôle du docteur. Ces données ont ensuite été divisées en trois groupes : (1) les questions qui font partie du corpus d'entraînement (*In training*), (2) les autres questions couvertes par la grammaire (*In coverage*) et (3) les questions non-couvertes par la grammaire (*Out of coverage*). Cette distinction entre données couvertes et non couvertes est importante dans le contexte de notre application où toute l'architecture est conçue pour diminuer le nombre de phrases non-couvertes par la grammaire, grâce à un système d'aide qui aide l'utilisateur à apprendre la couverture du système (Rayner *et al.*, 2005a). Pour chacun de ces groupes, nous avons ensuite mesuré le taux d'erreurs au niveau des mots (*WER*) et des phrases (*SER*). Comme ces mesures semblent parfois trompeuses (Wang *et al.*, 2003), nous avons aussi mesuré le nombre d'erreurs sémantiques (*SemER*), c'est-à-dire le pourcentage de phrases qui ne préservent pas le sens. Ce sont les phrases qui n'ont pas été envoyées à la traduction par les sujets lors de la collecte des données. Nous résumons les résultats dans la table 1.

	français				catalan			
	#Utts	WER	SER	SemER	#Utts	WER	SER	SemER
In training	86	5.9 %	29.1 %	14.0 %	303	5.6 %	27.1 %	5.3 %
In coverage	177	8.6 %	35.0 %	7.3 %	47	11.8 %	53.2 %	12.8 %
In coverage (all)	263	7.8 %	33.1 %	9.5 %	350	6.5 %	30.6 %	6.3 %
Out of coverage	148	37.4 %	100.0 %	68.9 %	180	60.2 %	96.1 %	81.2 %

Tableau 1. Reconnaissance avec la grammaire spécialisée pour la reconnaissance

À première vue, le *WER* pour les phrases couvertes par la grammaire *In coverage (all)* semble moins bon en français que pour le catalan et surtout pour l'anglais (5,7 %. Cf. (Bouillon *et al.*, 2005)). Ces chiffres doivent pourtant être relativisés. D'une part, le *SemER* est plus faible qu'en anglais, ce qui signifie que la plupart des erreurs sont minimales et n'entravent pas la traduction — il s'agit notamment d'erreurs liées à la reconnaissance des déterminants, beaucoup plus nombreux qu'en anglais. D'autre part, l'un des sujets catalans a lu les questions, ce qui fait que les données sont moins représentatives pour l'instant. Il est par ailleurs intéressant de noter que, pour le français, le pourcentage d'erreurs est plus faible au niveau du *SemER* pour les phrases qui ne figurent pas dans le corpus d'entraînement que pour les données d'entraînement elles-mêmes. Cette grande différence s'explique par le fait que les données d'entraînement contiennent plusieurs fois le verbe *eu* qui est mal reconnu pour l'instant. Les généralisations de la grammaire à partir des corpus semblent donc utiles et correctes et ceci dans les deux langues.

Pour mieux évaluer la qualité des grammaires spécialisées de reconnaissance, nous avons, dans un second temps, effectué la reconnaissance avec la grammaire de génération. Les résultats dans la table 2 confirment entièrement nos intuitions de départ. Du fait que la grammaire de génération est plus contrainte, le *WER* et le *SER* sont à première vue améliorés. Il faut cependant noter que le *SemER* ne diminue presque pas (9,5 % *versus* 8,2 %) : le pourcentage de phrases traduisibles reste donc quasi le même. Le gain est d'autant plus réduit pour l'utilisateur que cette faible amélioration du *SemER* est corrélée à une augmentation importante des phrases qui ne sont pas couvertes par la grammaire (215 *versus* 148), pour lesquelles le *SemER* est par définition mauvais<sup>1</sup>. C'est d'ailleurs précisément le type de questions que nous voulons éviter dans ce projet grâce au système d'aide, dont nous avons parlé plus haut. Nous arrivons donc à la conclusion finale que, dans le type d'application comme MedSLT, la diminution du *WER* et *SER* ne va pas de paire avec une amélioration globale du système. Il est, de plus, essentiel de pouvoir, grâce à une méthode comme la nôtre, dériver des grammaires différentes et les comparer entre elles pour arriver à la meilleure grammaire pour une application donnée.

	#Utts	WER	SER	SemER
In training	87	2.4 %	12.6 %	11.5 %
In coverage	109	4.0 %	13.8 %	5.5 %
In coverage (all)	196	3.2 %	13.3 %	8.2 %
Out of coverage	215	43.0 %	100.0 %	77.2 %

Tableau 2. Reconnaissance en français avec la grammaire spécialisée pour la génération

## 6. Conclusion

Dans cet article, nous avons présenté une méthode pour dériver facilement et rapidement des reconnaissances linguistiques et des générateurs à partir de grammaires générales partagées pour plusieurs langues. Nous avons montré que : (1) il est possible de développer des grammaires linguistiquement motivées pour la parole ; (2) ces grammaires peuvent être partagées entre plusieurs langues sans dégrader la performance et (3) il est important de pouvoir dériver des grammaires différentes pour pouvoir les utiliser pour plusieurs tâches et pour les comparer entre elles.

## Références

- BENDER E., FLICKINGER D. et OEPEN S. (2002). « The Grammar Matrix : An Open Source Starter-Kit for the Rapid Development of Cross-Linguistically Consistent Broad-Coverage Precision Grammars ». In *Proceedings of the 19th International Conference on Computational Linguistics*. Taipei, Taiwan.
- BOUILLON P., RAYNER M., CHATZICHRISAFIS N., HOCKEY B., SANTAHOLMA M., STARLANDER M., NAKAO Y., KANZAKI K. et ISAHARA H. (2005). « A Generic Multi-Lingual Open Source Platform for Limited-Domain Medical Speech Translation ». In *Proceedings of the 10th Conference of the European Association for Machine Translation (EAMT)*. Budapest, Hungary.

<sup>1</sup> Quelle que soit d'ailleurs la méthode de reconnaissance utilisée, linguistique ou statistique. Cf. (Bouillon *et al.*, 2005).

- DOWDING J., HOCKEY B., GAWRON J. et CULY C. (2001). « Practical Issues in Compiling Typed Unification Grammars for Speech Recognition ». In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*. Toulouse, France.
- KNIGHT S., GORRELL G., RAYNER M., MILWARD D., KOELING R. et LEWIN I. (2001). « Comparing grammar-based and robust approaches to speech understanding : a case study ». In *Proceedings of Eurospeech 2001*. Aalborg, Denmark, p. 1779-1782.
- MILLER P. H. et SAG I. (1997). « French Clitic Movement Without Clitics or Movement ». In *Natural Language and Linguistic Theory*, 15, 573-639.
- MOORE R. (1998). « Using Natural Language Knowledge Sources in Speech Recognition ». In *Proceedings of the NATO Advanced Studies Institute*.
- PEREIRA F. (1981). « Extraposition Grammars ». In *American Journal of Computational Linguistics*, 7, 243-256.
- RAYNER M., BOUILLON P., CHATZICHRISAFIS N., HOCKEY B., SANTAOLMA M., STARLANDER M., ISAHARA H., KANZAKI K. et NAKAO Y. (2005a). « A Methodology for Comparing Grammar-Based and Robust Approaches to Speech Understanding ». In *Proceedings of the 9th International Conference on Spoken Language Processing (ICSLP)*. Lisboa, Portugal.
- RAYNER M., BOUILLON P., SANTAOLMA M. et NAKAO Y. (2005b). « Representational and Architectural Issues in a Limited-Domain Medical Speech Translator ». In *Proceedings of TALN 2005*. Dourdan, France.
- M. Rayner, D. Carter, P. Bouillon, V. Digalakis et M. Wirén (éds.) (2000). *The Spoken Language Translator*. Cambridge University Press.
- RAYNER M., HOCKEY B. et BOUILLON P. (2006). *Putting Linguistics into Speech Recognition : The Regulus Grammar Compiler*. CSLI Press, Chicago.
- REGULUS (2005). <http://sourceforge.net/projects/regulus/>. As of 15 Dec 2005.
- WANG Y.-Y., ACERO A. et CHELBA C. (2003). « Is Word Error Rate a Good Indicator for Spoken Language Understanding Accuracy ». In *Proceedings of Eurospeech 2003*. Geneva, Switzerland, p. 609-612.