

L'utilité d'un vérificateur de grammaire comme outil pédagogique:  
l'exemple de LINGER (Language INdependent Grammar Reporter)

by

RUTH MARY HANSON

B.A. (Hons), The University of British Columbia, 1997

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF  
THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF ARTS

in

THE FACULTY OF GRADUATE STUDIES

(Department of French, Hispanic & Italian Studies)

We accept this thesis as conforming  
to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

April 2001

© Ruth Mary Hanson, 2001

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of French, Hispanic & Italian Studies

The University of British Columbia  
Vancouver, Canada

Date April 27, 2001

## Abrégé

Sachant que l'idée de se servir d'un vérificateur de grammaire comme outil pédagogique dans le domaine de l'enseignement des langues secondes constitue un sujet de débat continu, nous abordons une étude d'un vérificateur de grammaire, LINGER (Language Independent Grammatical Error Reporter), conçu pour les apprenants de français, d'espagnol et d'allemand et réalisé à l'Université d'Exeter, les résultats de laquelle nous permettent de formuler des généralisations qui aident à comprendre si en théorie un vérificateur de grammaire est un outil pédagogique utile ou pas. En plus, nous recommandons ce qu'il faudrait inclure dans le vérificateur de grammaire idéal pour les apprenants.

Afin de fournir une idée de ce dont la technologie actuelle est capable dans le domaine de la correction automatique/assistée des langues et d'établir les mérites et les problèmes de l'usage d'un vérificateur de grammaire comme outil pédagogique, nous présentons une étude faite sur les vérificateurs de grammaire commerciaux, *Antidote '98* et *Le Correcteur 101 – Version 3.5 Professionnel*. Ensuite, pour pouvoir évaluer LINGER et faire des suggestions au sujet de l'architecture d'un système idéal, nous expliquons en détail les trois parties de LINGER (la grammaire, le dictionnaire et la coquille) et comment se fait l'analyse syntaxique. Avant de regarder de près les corrections de LINGER, nous explorons une définition théorique de la correction ainsi que la façon dont l'on apprend des langues secondes. La production écrite d'un groupe d'étudiants qui sont au niveau de la première année en français fournit les données avec lesquelles nous évaluons LINGER et nous nous servons d'un des vérificateurs de grammaire commerciaux avec ces mêmes données afin d'évaluer les critères de performance technique requis pour l'usage d'un tel logiciel par des apprenants.

A la lumière de tout ce que nous soulevons, nous concluons que LINGER n'est pas ni serait jamais un outil pédagogique dont des apprenants puissent se servir mais qu'en général un vérificateur de grammaire seraient un outil pédagogique utile pourvu qu'il soit conçu pour des apprenants de la même langue maternelle; qu'il incluse des explications grammaticales pour chaque correction qui aident l'apprenant à mieux comprendre la règle de grammaire en question; que son analyse soit juste avec un minimum de correction fautive, de manques d'identification de fautes et d'identification fautive de la source des fautes et qu'il fournisse une correction assistée, plutôt qu'une correction automatique.

## TABLE DES MATIÈRES

|   |     |
|---|-----|
| Abrégé.....   | ii  |
| Table des matières.....   | iii |
| Liste des figures.....  | iv  |
| Remerciements.....  | v   |
| Introduction.....   | 1   |
| Chapitre 1 : Un aperçu de la technologie actuelle.....                                    | 4   |
| Chapitre 2 : LINGER.....  | 12  |
| LE DICTIONNAIRE.....  | 13  |
| Les catégories grammaticales.....   | 13  |
| Les mots.....   | 14  |
| Les terminaisons.....   | 16  |
| LA COQUILLE.....  | 19  |
| La pré-analyse.....   | 19  |
| LA GRAMMAIRE.....   | 28  |
| La syntaxe forte.....   | 28  |
| La syntaxe faible.....  | 30  |
| Chapitre 3: La correction théorique et appliquée.....                                     | 52  |
| VERS UNE DÉFINITION THÉORIQUE DE LA CORRECTION.....                                       | 52  |
| La définition d'une faute.....  | 52  |
| Les critères d'une faute.....   | 53  |
| L'origine des fautes.....   | 55  |
| La correction.....  | 58  |
| ÉVALUATION DE LINGER.....   | 61  |
| La description du corpus.....   | 61  |
| L'analyse des corrections de LINGER.....  | 63  |
| ANTIDOTE 2000: ÉVALUATION DE LA TECHNOLOGIE ACTUELLE.....                                 | 67  |
| Les conclusions et les recommandations.....   | 71  |
| Appendice A.....  | 76  |
| LE DICTIONNAIRE DE LINGER.....  | 76  |
| Appendice B.....  | 84  |
| LA SYNTAXE FORTE DE LINGER.....   | 84  |
| Appendice C.....  | 88  |
| LA SYNTAXE FAIBLE ORGANISÉE PAR CATÉGORIE GRAMMATICALE.....                               | 88  |
| Appendice D.....  | 93  |
| LA CATÉGORISATION DES FAUTES DE L'ÉCHANTILLON DE PRODUCTION ÉCRITE DES<br>APPRENANTS..... | 93  |
| Les fautes stylistiques.....  | 93  |
| Les fautes sémantiques.....   | 93  |
| Les fautes morpho-syntaxiques.....  | 95  |
| Appendice E.....  | 102 |
| LE RENDEMENT DE LINGER.....   | 102 |
| Appendice F.....  | 129 |
| LA PERFORMANCE D'ANTIDOTE 2000.....   | 129 |
| Bibliographie.....  | 133 |

## Liste des figures

|   |    |
|---|----|
| Figure 1.1: La catégorisation des fautes.....   | 5  |
| Figure 1.2: La distribution de la correction effectuée par <i>Le Correcteur 101</i> ..... | 5  |
| Figure 1.3: Comparaison entre Antidote et le Correcteur .....                             | 8  |
| Figure 1.4: Usage du vocabulaire .....  | 9  |
| Figure 2.1 : L'interprétation de la structure générale des faits en <i>info/4</i> .....   | 14 |
| Figure 2.2 : Un fait <i>info/4</i> appliqué aux noms.....                                 | 14 |
| Figure 2.3: L'interprétation de la structure générale des faits en <i>word/5</i> .....    | 15 |
| Figure 2.4: Deux exemples possibles du fait <i>word/5</i> .....                           | 15 |
| Figure 2.5: Les deux structures générales du fait <i>ending_type/3</i> .....              | 16 |
| Figure 2.6: Deux exemples du fait <i>ending_type/3</i> .....                              | 17 |
| Figure 2.7: La structure du fait <i>actual_endings/2</i> et un exemple détaillé .....     | 18 |
| Figure 2.8: Le <i>gp_end</i> .....  | 18 |
| Figure 2.9: Exemples de l'organisation des informations grammaticales de la DCG.....      | 28 |
| Figure 2.10: Exemple d'une règle DCG et son interprétation.....                           | 29 |
| Figure 2.11: Exemple d'une vérification de la syntaxe faible .....                        | 30 |
| Figure 2.12: Le modèle des vérifications .....  | 30 |

## Remerciements

J'aimerais remercier Mary et John pour toujours s'intéresser à tout ce que je fais et Graham pour avoir écouté les mises à jour quotidiennes de mon progrès (même si ses oreilles ne fonctionnaient qu'à 75%!).

Je voudrais remercier aussi chacun des membres de mon comité: Chantal Phan pour avoir assuré que mon français était au point; Trude Heift de Simon Fraser University pour avoir accepté d'être examinatrice externe et surtout mon directeur de mémoire, Bill Winder, pour avoir été toujours disponible et prêt à m'encourager.

## **Introduction**

Le mariage de l'intelligence artificielle et de la linguistique a inspiré, parmi d'autres applications, des vérificateurs de grammaire qui sont censés analyser un texte écrit, en montrer les fautes morpho-syntaxiques et suggérer des réponses qui sont « bien construites » grammaticalement. Depuis à peu près dix ans, l'idée de se servir d'un vérificateur de grammaire comme outil pédagogique dans le domaine de l'enseignement des langues secondes constitue un sujet de débat continu. Malgré les controverses, on a réalisé peu d'études qui évaluent la performance d'un vérificateur de grammaire prototype dans un contexte pédagogique et la plupart de celles qui ont été réalisées « [...]semble remettre en question l'usage des vérificateurs de grammaire[...] » (Bolt, 26) dans l'enseignement des langues secondes. Donc, nous proposons une étude d'un vérificateur de grammaire, LINGER (Language Independent Grammatical Error Reporter), conçu pour les apprenants de français, d'espagnol et d'allemand et réalisé à l'Université d'Exeter. Puisque les vérificateurs de grammaire fonctionnent tous selon les mêmes critères de base (c'est-à-dire que chacun réalise une analyse syntaxique des phrases à partir d'une grammaire et d'un dictionnaire) et qu'ils partagent des limites semblables (par exemple, la correction fautive ou bien le manque de correction), nous maintenons que les résultats de notre étude d'un seul vérificateur de grammaire nous permettront de formuler des généralisations qui aideront à comprendre si en théorie un vérificateur de grammaire est un outil pédagogique utile ou pas. En plus, nous pourrions recommander ce qu'il faudrait inclure dans le vérificateur de grammaire idéal pour les apprenants.

A cette fin, nous commencerons par présenter une étude faite sur les vérificateurs de grammaire commerciaux, *Antidote '98* et *Le Correcteur 101 – Version 3.5 Professionnel*, ce

qui fournira une idée de ce dont la technologie actuelle est capable dans le domaine de la correction automatique/assistée des langues et aidera à établir les mérites et les problèmes de l'usage d'un vérificateur de grammaire comme outil pédagogique. Ensuite, afin de pouvoir évaluer LINGER et faire des suggestions au sujet de l'architecture d'un système idéal, nous expliquerons en détail les trois parties de LINGER (la grammaire, le dictionnaire et la coquille) et comment se fait l'analyse syntaxique. Avant de regarder de près les corrections de LINGER, nous explorerons une définition théorique de la correction ainsi que la façon dont l'on apprend des langues secondes. La production écrite d'un groupe d'étudiants qui sont au niveau de la première année en français fournira les données avec lesquelles nous pourrions évaluer LINGER et nous nous servirons d'un des vérificateurs de grammaire commerciaux avec ces mêmes données afin d'évaluer les critères de performance technique requis pour l'usage d'un tel logiciel par des apprenants. Nous précisons dès le départ que la correction ni sémantique ni anaphorique ne sera traitée, car la technologie actuelle est limitée à l'analyse morpho-syntaxique.

Nous pouvons imaginer toutes sortes de questions liées aux implications de se servir d'un vérificateur de grammaire comme outil pédagogique: Comment intégrer le vérificateur de grammaire dans le cours ? Est-ce qu'il ne devrait être disponible que dans des conditions restreintes pour des devoirs précis ou est-ce que les apprenants devraient avoir le droit de mettre un exemplaire du logiciel sur leur ordinateur à la maison et s'en servir comme ils veulent ? Quand est-ce que les apprenants devraient commencer à se servir d'un vérificateur de grammaire ? Est-ce que le vérificateur de grammaire devrait servir comme outil qui enseigne la grammaire ou qui la renforce ? Aucune de ces questions ne sera traitée ici parce qu'elles présupposent une réponse affirmative à l'hypothèse que nous essayons de confirmer ici.

Nous envisageons également des questions au sujet de la programmation optimum d'un vérificateur de grammaire conçu pour des apprenants: Comment programmer le vérificateur de grammaire selon les besoins des étudiants ? Est-ce qu'il y a un moyen de tenir compte des niveaux des fautes, par exemple, une faute de genre comparée à une faute d'accord ? Est-ce qu'il est possible de suivre le progrès de l'apprenant d'une séance à l'autre, modulant les explications grammaticales automatiquement au besoin ? Est-ce que le logiciel peut catégoriser et faire l'addition des fautes d'une séance et puis nommer les structures grammaticales que l'apprenant a besoin de réviser ? Comment est-ce que le vérificateur de grammaire devrait corriger la production écrite de l'apprenant, en fournissant la bonne réponse avec une explication des fautes ou en référant l'apprenant au chapitre pertinent d'un livre de grammaire ? Alors qu'elles sont toutes des questions très intéressantes, il y en a trop à traiter ici. Donc, nous limiterons nos commentaires à propos de la programmation du vérificateur de grammaire idéal aux traits déjà incorporés dans LINGER, *Antidote* et *Le Correcteur 101*.

## Chapitre 1 : Un aperçu de la technologie actuelle

La première version du *Correcteur 101* est apparue sur le marché en 1991 et Machina Sapiens continue à tirer profit des avancements technologiques pour perfectionner son vérificateur de grammaire. Depuis sa conception, *Le Correcteur 101* est devenue le standard auquel toute la compétition a été mesurée et aucun logiciel n'a rivalisé avec lui jusqu'à l'arrivée d'*Antidote* en 1996.

Puisque la grammaire des apprenants de français comme langue seconde exige beaucoup de correction et que *Le Correcteur 101* et *Antidote* font exactement cela, est-ce qu'un apprenant pourrait se servir d'un vérificateur de grammaire pour améliorer son français? Cette question a motivé deux études, dont le but était d'établir si, en effet, les vérificateurs de grammaire conçus pour des francophones arrivaient à gérer des textes écrits par des apprenants de français. Dans les deux études critiques que nous allons aborder, Jack Burston a évalué *Antidote '98* et Eugene Mogilevsky *Le Correcteur 101 – Version 3.5 Professionnel*.

Prenons, d'abord, *Le Correcteur 101*. Au début de l'étude, Mogilevski nous rappelle que «[...] no grammar checker is capable of semantic analysis [...]» (Mogilveski, 3). Pour cette raison-ci, il a exclu toutes les fautes liées à l'usage du vocabulaire, des déterminants, de "Ce/Il", des pronoms, des temps verbaux, des adverbes, des prépositions et des conjonctions. Les fautes qui restent sont résumées dans la Figure 1.1.

| Error Categories & Examples |                           |                                   |
|-----------------------------|---------------------------|-----------------------------------|
| Error Class                 | Error Type                | Examples                          |
| <b>Noun Group</b>           | noun gender               | <i>un</i> université              |
|                             | noun number               | <i>Les</i> politiques m'intéresse |
| <b>Adjectives:</b>          | gender agreement          | Une <i>joli</i> fille             |
|                             | number agreement          | Les gens <i>pressé</i>            |
|                             | form                      | Une histoire <i>oublier</i>       |
| <b>Verbal group:</b>        | conjugation               | J'ai <i>devé</i>                  |
|                             | mood                      | Il voulait que je <i>venais</i>   |
|                             | subject agreement/ person | Nous <i>connaissez</i>            |
|                             | subject agreement/ number | Ils <i>arrive</i>                 |
|                             | negation                  | Je veux jamais pas y aller        |
| <b>Orthography:</b>         | accent                    | <i>ec</i> rire                    |
|                             | character                 | raisonnable                       |
|                             | approximation             | particul <i>ia</i> ire            |
|                             | elision                   | que <i>il</i>                     |

Figure 1.1: La catégorisation des fautes (Mogilevski, 3)

La Figure 1.2 résume la performance du *Correcteur 101*<sup>1</sup> où la première colonne indique le nombre total de fautes dans le corpus corrigées par Mogilevski selon la classification de la Figure 1.1. Une cinquantaine de compositions écrites par des étudiants de deuxième année dans des conditions d'examen constitue le corpus.

| Corrected Errors |                        |                      |       |                            |            |                    |
|------------------|------------------------|----------------------|-------|----------------------------|------------|--------------------|
| N/Err            | Corrected <sup>2</sup> | Flagged <sup>3</sup> | Total | Misidentified <sup>4</sup> | Undetected | False <sup>5</sup> |
| 1262             | 74%                    | 17%                  | 91%   | 7%                         | 2%         | 49                 |

Figure 1.2: La distribution de la correction effectuée par *Le Correcteur 101*

<sup>1</sup> L'étude de Mogilevski compare deux versions du Correcteur 101. Nous ne nous intéressons qu'à la version 3.5, la plus récente, alors nous n'avons cité que les résultats des tests faits sur elle. (Mogilveski, 4)

<sup>2</sup> "Corrected errors included mistakes that were either replaced by correct forms or identified in such a way to make correction straightforward (e.g. clicking on the dictionary to find an unrecognizable word, making subject/verb agreement in accordance with the grammatical analysis given)." (Mogilveski, 4)

<sup>3</sup> "Flagged errors were those which, though marked, required the user to independently decide what corrective action to take." (Mogilveski, 4)

<sup>4</sup> "Misidentified errors were those in which the source of the mistake was wrongly attributed. Since these were misleading, they were not counted as successful detections." (Mogilveski, 4)

Les statistiques sont impressionnantes; le logiciel est arrivé à corriger 91% des fautes. En plus, quand on clique sur l'icône correspondante, *Le Correcteur 101* fournit des explications grammaticales ainsi que de la vérification lexicale, qui corrige les fautes orthographiques et typographiques, un dictionnaire, qui définit un mot sélectionné et donne quelques-unes de ses caractéristiques grammaticales, et un dictionnaire synonymique, qui aiderait l'apprenant à élargir son vocabulaire.

Malgré la correction efficace et les outils supplémentaires, Mogilevski souligne qu'il ne serait pas prudent de se servir du *Correcteur 101* comme outil pédagogique pour l'apprentissage du français langue seconde. Un des problèmes prioritaires est le manque de correction, ou au moins de détection, des structures qui sont grammaticalement possibles mais rarement appropriées, surtout dans le contexte de la production écrite des apprenants.

Mogilevsky donne comme exemple les expressions de quantité:

[A]t present C3.5 does not detect deviations from the base forms "beaucoup de", "pas de", "peu de" and "combien de". Use of the definite article with these expressions, though grammatically possible [par exemple, *J'aime beaucoup des sports joués au Canada.*], is nearly always semantically inappropriate in student compositions [par exemple, *\*J'aime beaucoup des sports différents.*] Because in some contexts "beaucoup des", etc. may be correct, it is of course important not to overcorrect by automatically treating these forms as errors. On the other hand, simply flagging them as possible problems would offer valuable information to students (7).

Il mentionne aussi l'usage des prépositions après des verbes transitifs. En raison de l'interférence de l'anglais, les apprenants ont tendance à ajouter des prépositions aux verbes comme *regarder*, *payer* et *écouter*. Comme l'exemple des expressions de quantité, le logiciel accepte ce genre de faute parce que dans certains contextes, la préposition est nécessaire pour changer le sens du verbe; par exemple, *Il ne regarde pas à la dépense. Il a payé pour tout le*

---

<sup>5</sup> "False corrections were those which were wrongly diagnosed as errors." (Mogilveski, 4)

monde. *J'ai fait écouter un disque à mon frère.* Dans le cas de l'apprenant, il est probable qu'il ajoute la préposition où elle est inappropriée sémantiquement et, puisque le logiciel n'est pas sensible au sens des phrases, cette faute de base ne sera pas corrigée.

La correction fautive pose un problème sérieux à l'usage du *Correcteur 101* par des apprenants aussi. Il est sûr que pour eux la correction fautive remettrait en question la fiabilité du logiciel et causerait de la confusion et de la frustration.

La deuxième étude, faite par Jack Burston de Monash University, compare *Le Correcteur* au vérificateur de grammaire concurrent: *Antidote*. Comme *Le Correcteur 101*, *Antidote* fournit des explications grammaticales et un dictionnaire ainsi qu'un conjugueur de 8000 verbes et un livre de grammaire qui traite plus de 250 thèmes, créé par *Druide Informatique* à la base du *Bon Usage*. En plus de vérifier la grammaire, la publicité prétend qu'*Antidote* soit sensible aux homonymes, aux paronymes, aux anglicismes et aux régionalismes (Burston, 1). Il suffit de cliquer sur l'icône appropriée afin de voir l'explication d'une faute, d'accepter une correction et modifier le texte original, d'ajouter un mot inconnu au dictionnaire personnel, de voir l'analyse syntaxique complète d'une phrase et de consulter le dictionnaire, le conjugueur de verbes ou le livre de grammaire.

Burston s'est servi du même corpus que celui de l'étude de Mogilevsky; c'est-à-dire la cinquantaine de compositions des étudiants en deuxième année. Nous combinons les résultats des deux études dans la Figure 1.3.

| Student Compositions |       |           |          |       |                 |               |            |
|----------------------|-------|-----------|----------|-------|-----------------|---------------|------------|
|                      | N/Err | Corrected | Detected | Total | False Corrected | Misidentified | Undetected |
| Correcteur           | 1262  | 74%       | 17%      | 91%   | 49 examples     | 7%            | 2%         |
| Antidote             | 1262  | 77%       | 5%       | 82%   | 28 examples     | 11%           | 7%         |

Figure 1.3: Comparaison entre Antidote et le Correcteur (Burston, 9)

*Antidote* ne réussit pas à détecter et à corriger autant de fautes que *Le Correcteur* (82% versus 91%) ni simplement à les détecter (5% chez *Antidote* versus 17%), mais il en corrige plus (77% comparé à 74% chez *Le Correcteur*). Il faut remarquer aussi qu'*Antidote* fait moins de corrections fautives (28 comparées à 49). Cependant, parmi les fautes qui n'étaient pas détectées du tout, *Le Correcteur* est plus performant (2% des fautes manquées versus 7% chez *Antidote*) ainsi que pour les cas où le logiciel attribue la faute à la mauvaise source (7% versus 11% chez *Antidote*). Burston souligne que les fautes d'orthographe posent des problèmes à *Antidote*, surtout:

its misidentification of some fairly obvious spelling errors (e.g. premiere, videttes, employés) and its failure to recognize spelling mistakes in sentence initial words, which in every case were incorrectly attributed to unknown proper nouns.

Il ajoute que:

Also when operating in grammar correction mode, *Antidote*, unlike *Le Correcteur*, does not offer lexical alternatives to misspelled words. It either automatically makes a correction or informs the user that it doesn't recognize the word (9).

Il est clair que la mauvaise identification et le manque de reconnaissance des fautes d'orthographe et d'alternatives aux mots mal écrits constituent des faiblesses sérieuses qui, entraîneraient chez l'apprenant de la frustration inutile et des orthographe fautives.

Malgré ces problèmes d'orthographe, *Antidote* se rachète à travers son avertissement de l'usage du vocabulaire. Dans cette étude, il a reconnu 236 fautes d'usage, dont les plus fréquentes sont résumées dans la Figure 1.4:

| Vocabulary Usage        |                     |                    |
|-------------------------|---------------------|--------------------|
| actuellement/réellement | tâche/tache         | programme/émission |
| anniversaire/fête       | peut-être/peut être | immigrant/émigrant |
| opportunité/occasion    | ces/ses             | original/originel  |
| nombre/chiffre          | anglais/Anglais     |                    |
| apporter/amener/emmener | français/Français   |                    |

Figure 1.4: Usage du vocabulaire (Burston, 10)

Quand le logiciel nous avertit d'un problème d'usage d'un mot, il fournit une courte définition du mot du texte et d'autres mots possibles. En lisant les définitions du mot qu'il a écrit et les autres possibilités proposées par le logiciel, l'apprenant peut choisir le mot qui correspond au contexte. En plus, si la courte définition n'est pas suffisante, l'apprenant peut consulter le dictionnaire, en cliquant sur l'icône appropriée. Comme l'indique Burston, pour des apprenants le désavantage de cette fonction est qu'*Antidote* risque de proposer des alternatives qui ne sont pas très intéressantes à l'apprenant et qui pourraient troubler son orthographe, par exemple, anglais/anglet, jeune/jeûne, matin/mâtin et ton/taon/thon (Burston, 10). Cependant, l'avantage est clair:

...used as a pedagogical device, lexical alerts can provide an interesting and contextually sensitive means of vocabulary acquisition (10).

S'inspirant des conclusions de Eugene Mogilevski et de Jack Burston présentées ci-dessus, nous résumons les avantages et les désavantages de se servir d'un vérificateur de grammaire comme outil pédagogique. La faiblesse fondamentale des vérificateurs de grammaire, et de n'importe quelle application de traitement de langue automatique, est que la technologie actuelle n'est pas capable de faire l'analyse sémantique; c'est-à-dire que la correction est

limitée aux fautes morpho-syntaxiques. Pourtant, comme le signale Burston, «[n]otwithstanding, as most French teachers are painfully aware, this [les fautes morpho-syntaxiques] is an area much in need of remediation in student essays. » (9).

Étant donné la performance d'*Antidote* et du *Correcteur 101* au niveau de la détection et de la correction des fautes, il est clair qu'enfin il existe une technologie qui fonctionne extrêmement bien. En plus de corriger, nous avons vu que les vérificateurs de grammaire fournissent d'autres outils, comme un dictionnaire, un dictionnaire synonymique, un conjugueur de verbes et un livre de grammaire qui incluent toutes les informations dont un apprenant aurait jamais besoin et qui l'aiderait à perfectionner son orthographe et la conjugaison ainsi qu'à améliorer le vocabulaire et la compréhension de la grammaire. Puisque tous ces outils se trouvent au prix d'un clic et les informations qui apparaissent sont reliées directement à la phrase en question, l'apprenant peut résoudre ses difficultés tout de suite, sans passer de temps à feuilleter des livres à la recherche d'une solution qui explique précisément sa faute particulière, une recherche qu'il risque fort d'abandonner ou, dans certains cas, de ne même pas entreprendre. Parmi les outils automatiques, les explications grammaticales aideraient l'apprenant à clarifier la grammaire, la vérification lexicale améliorerait son orthographe et lui ferait focaliser surtout sur les fautes d'orthographe répétées et les avertissements de l'usage du vocabulaire supprimeraient les anglicismes et raffinerait le sens et la justesse de l'usage des mots dans l'esprit de l'apprenant.

Les désavantages principaux des vérificateurs de grammaire comme *Antidote* et *Le Correcteur 101* sont triples: la correction fautive, le manque d'identification de fautes et l'identification fautive de la source des fautes. Quoique ces trois faiblesses soient problématiques, elles se remarquent dans la correction de l'écriture des apprenants

partiellement à cause du fait que *Le Correcteur 101* et *Antidote* sont conçus pour des francophones. Nous nous rappelons les exemples des expressions de quantité (*beaucoup de(s)*, etc.) et des verbes transitifs (*regarder (à)*, *écouter (à)*, *payer (pour)* ) soulignés par Mogilevsky qui montrent que les vérificateurs de grammaire faits pour des francophones accepteront tout ce qui est grammaticalement possible et, donc, ne reconnaîtront pas les constructions inappropriées qui se voient dans les compositions des apprenants. Logiquement, ce genre de faute représente au moins une partie du pourcentage des fautes manquées, qui est déjà très bas, dont la correction pourrait être améliorée, si le logiciel était programmé de façon à prévoir les fautes typiques des apprenants. Quant à l'identification fautive de la source des fautes, Burston attribue cela aussi au fait que *Le Correcteur 101* et *Antidote* sont conçus pour des francophones:

[I]n dealing with student compositions, grammar checkers are confronted with such a quantity of anomalies that it becomes difficult to determine the syntactic structure needed to correct even basic mistakes. Moreover, very often the nature of student errors defies all attempts at analysis. The following example illustrates the kind of sentences which are all too frequently encountered:

Il fallait que nous avons promis notre mère de ne pas se séparer.

Inevitably, such deviant forms produce incomplete analyses, and with only a partial result to go by, comprehensive and accurate error identification and remediation is simply not within the power of any native-speaker grammar checker to provide. (8)

Étant donné leur précision et leurs matériaux de référence additionnels, les vérificateurs de grammaire sont un outil pédagogique très prometteur. Cependant, il faut les modifier pour qu'ils puissent corriger les fautes morpho-syntaxiques pertinentes aux apprenants de français

## Chapitre 2 : LINGER

Reconnaissant la valeur et l'utilité potentielles d'un outil pédagogique automatique qui puisse analyser et corriger n'importe quelle phrase en langue naturelle et, en plus, expliquer ses corrections, les premières recherches au sujet des vérificateurs de grammaire ont commencé à l'Université d'Exeter en 1985. Les systèmes initiaux étaient « French Robust Grammar Checker » suivi de « French Grammar Analyser ». Pensant qu'il serait plus efficace de créer un système qui puisse gérer des données en plusieurs langues, notamment en allemand, en italien, en espagnol et en français, le « Language INdependent Error Reporter », ou LINGER, (Barchan) est né. Depuis le début des recherches, le but de tous les vérificateurs de grammaire réalisés à Exeter est de traiter

The analysis of the grammatical structure of a sentence and, where no correct structure can be found, to provide the learner with sufficient guidance to construct a grammatically acceptable sentence based largely on the learner's original input. (Bolt, 2)

LINGER est passé par deux versions; la première de 1986 jusqu'à 1988, dont nous nous servons ici, et la deuxième de 1988 jusqu'à 1992. Malgré son ancienneté, nous avons choisi d'utiliser LINGER comme exemple d'un vérificateur de grammaire correctif principalement parce que contrairement aux logiciels fabriqués par des entreprises privées, comme *Antidote* et *Le Correcteur 101*, la première version de LINGER est dans le domaine public. Cela nous a permis d'avoir accès au code, dont l'analyse est essentielle pour faire une étude comme la nôtre. De plus, au départ le fonctionnement de LINGER nous semblait prometteur, même si l'idée d'inclure quatre langues dans le même logiciel paraissait un peu ambitieuse.

Le système comprend trois parties : le dictionnaire, la coquille et la grammaire. Le but de ce chapitre est d'expliquer la fonction de chaque composante du système, tout en fournissant des exemples de sorte qu'il soit clair comment LINGER analyse et corrige des données. De plus, nous saupoudrons le chapitre de critiques là où cela nous semble pertinent.

### **Le dictionnaire**

Le dictionnaire contient tous les mots que LINGER connaît, ainsi que les informations qui précisent comment et dans quelles circonstances on peut modifier ces mots. Nous présentons la représentation de ces informations chez LINGER en trois parties: les catégories grammaticales, les mots et les terminaisons.

### **Les catégories grammaticales**

Avant les entrées des mots précis, LINGER inclut des faits qui généralisent les informations nécessaires pour toutes les catégories grammaticales. Chacun de ces faits a le prédicat *info* avec quatre arguments. La Figure 2.1 montre l'interprétation de leur structure générale (Uren, 19, 3.3.1).

info(<name>, [<individual information>], [<general information>], [<global information>],).

où:

<name> est le <nom> soit de la catégorie grammaticale (nom, verbe, etc.) soit de la catégorie de terminaisons verbales (er\_end pour les verbes en -er, re\_end pour les verbes en -re, etc.);

[<individual information>] est une liste de <contraintes précises> qui représentent tous les traits morphologiques possibles qu'une catégorie grammaticale puisse avoir (c'est-à-dire le nombre, le cas, la présence d'une apostrophe, le genre masculin ou féminin et les personnes);

[<general information>] indique une liste des <contraintes générales> de la catégorie grammaticale au niveau syntaxique (c'est-à-dire le genre, le nombre, la personne, le placement, la transitivité, le temps verbal, la présence d'une préposition après le verbe principal, le choix du verbe auxiliaire et la présence d'un verbe réfléchi) qui change selon le mot et l'usage ; et

[<global information>] est une liste de <traits> communs à tous les mots dans une catégorie grammaticale donnée. (Par exemple, tous les noms utilisés comme sujets de la phrase ont le trait d'être à la troisième personne.)

Figure 2.1 : L'interprétation de la structure générale des faits en *info/4*

Nous clarifions cette structure générale dans la Figure 2.2, en fournissant un exemple du dictionnaire qui s'applique à la catégorie des noms (cf. Appendice A, 2.0):

info(noun, [[plurality(s)], [plurality(p)]], [gender], [person(3)] ).

où:

<nom> est un nom;

[<contrainte précise>] est le nombre, soit singulier, soit pluriel;

[<contrainte générale>] est le genre du nom; et

[<contrainte générale>] est la personne, qui sera toujours la troisième quand un nom est le sujet de la phrase.

Figure 2.2 : Un fait *info/4* appliqué aux noms

### Les mots

Pour représenter les informations nécessaires au sujet d'un mot en particulier, le même principe que nous avons abordé en haut s'applique. Cette fois, le prédicat est *word* et il a cinq arguments, comme nous l'indiquons dans la Figure 2.3 (Uren, 3.3.2, 19-20):

word(<reference name>, [<word root>], <grammatical type>, [<word ending>] ou <word ending type>, [<general attribute>] ).

où:

<reference name> est le <lemme>. Cette forme n'est souvent pas requise parce que d'habitude elle est identique au [<word root>]. Si elle est nécessaire, normalement elle prend la forme du masculin singulier dans le cas des noms et des adjectifs ou bien l'infinitif dans le cas des verbes et elle est écrite entre guillemets.

[<word root>] est le <radical du mot>, un terme créé par les réalisateurs de LINGER qu'on ne devrait pas confondre avec la définition étymologique. Dans notre contexte, il est défini plutôt comme le fragment du mot auquel on peut systématiquement ajouter les terminaisons correspondantes. Elle est écrite entre guillemets.

<grammatical type> est la <catégorie grammaticale> du mot.

[<word ending>] est une <liste de terminaisons irrégulières> entre guillemets et

<word ending type> est une étiquette pour une <liste de terminaisons régulières> qui se situent ailleurs dans le dictionnaire.

[<general attribute>] est une liste de <traits généraux> qui sont significatifs pour le mot en question, comme le genre et le placement, par exemple.

Figure 2.3: L'interprétation de la structure générale des faits en *word/5*

Nous fournissons, dans la Figure 2.4, deux exemples qui montrent les deux applications possibles du fait *word/5*. La première montre le fait, *word*(<reference name>, [<word root>], <grammatical type>, [<word ending>], [<general attribute>] ) dans le cas du mot « doux » (cf. Appendice A, 5.0) et la deuxième pour le mot « grand » (cf. Appendice A, 5.0).

*word*("doux",["dou"], adjective,["x","x","ce","ces"],[post]).

| <lemme> | [<radical du mot>] | <catégorie grammaticale> | [<liste de terminaisons irrégulières>] | [<traits généraux>] |
|---------|--------------------|--------------------------|--|---------------------|
| "doux"  | ["dou"]            | adjective                | ["x","x","ce","ces"]                   | [post]              |

où:

- le <lemme > est "doux";
- le [<radical du mot>] est "dou" auquel on peut ajouter une terminaison convenable ;
- la <catégorie grammaticale> est adjectif ;
- la [<liste de terminaisons irrégulières>] sont les terminaisons précises pour l'adjectif "doux"; et
- les [<traits généraux>] est le placement de l'adjectif par rapport au nom.

*word*("",["grand"],adjective,gp\_end,[pre]).

| <lemme> | [<radical du mot>] | <catégorie grammaticale> | <liste de terminaisons régulières> | [<traits généraux>] |
|---------|--------------------|--------------------------|------------------------------------|---------------------|
| ""      | ["grand"]          | adjective                | gp_end                             | [pre]               |

où:

- l'entrée pour le <lemme > n'est pas précisée parce qu'on peut ajouter les terminaisons directement au [<radical du mot>] sans avoir besoin d'en « fabriquer » une.
- Le [<radical du mot>] est, donc, "grand";
- la <catégorie grammaticale> est adjectif ;
- la < liste de terminaisons régulières > est "gp\_end" qui représente les terminaisons régulières qui vont avec "grand" (cf. Figure 2.8); et
- [<traits généraux >] est le placement de l'adjectif par rapport au nom.

Figure 2.4: Deux exemples possibles du fait *word/5*

## Les terminaisons

Il y a deux faits qui traitent les terminaisons des mots dans le dictionnaire: *ending\_type/3* et *actual\_endings/2*. Nous présentons, d'abord, les deux structures générales du fait *ending\_type/3* dans la Figure 2.5 (Uren, 3.3.3, 20-21). La différence entre les deux structures est simplement qu'une traite les verbes irréguliers et l'autre les verbes réguliers.

`ending_type(<word ending type>, [<actual ending>] ou <actual ending type>,[<general attribute>]).`

où:

<word ending type> est le nom du <type de terminaison du mot> en question (par exemple, les terminaisons pour les verbes en *-er*).  
[<actual ending>] s'applique aux verbes irréguliers et liste la <conjugaison complète> entre guillemets pour un temps donné, tandis que pour <actual ending type> l'on voit plutôt une étiquette qui représente les <terminaisons du verbe> pour un temps donné qui est détaillée ailleurs dans le logiciel.  
[<general attribute>], ou bien (<traits généraux>), indique, normalement, le temps auquel les terminaisons appartiennent.

Figure 2.5: Les deux structures générales du fait *ending\_type/3*

Nous présentons deux exemples du fait *ending\_type/3* dans la Figure 2.6:

1. `ending_type(aller_end,["vais","vas","va","allons","allez","vont"],[present]).`

| <type de terminaison du mot> | [<conjugaison complète>]                    | [<traits généraux>] |
|------------------------------|---|---------------------|
| aller_end                    | ["vais","vas","va","allons","allez","vont"] | [present]           |

où:

le < type de terminaison du mot > indique que la conjugaison qui va suivre est du verbe "aller";  
la [<conjugaison complète>] est la conjugaison du verbe "aller"; et  
le [<trait général>] est le temps de la conjugaison du verbe "aller", détaillée en <conjugaison complète>].

2. `ending_type(er_end,er_end_pres,[present]).`

| <type de terminaison du mot> | <terminaisons du verbe > | [<traits généraux>] |
|------------------------------|--------------------------|---------------------|
| er_end                       | er_end_pres              | [present]           |

où:

le <type de terminaison du mot> indique que les terminaisons qui vont suivre correspondent aux verbes en *-er*;  
 les <terminaisons du verbe> sont l'étiquette *er\_end\_pres* qui représente les terminaisons pour les verbes en *-er* au présent qui se trouvent dans le fait *actual\_endings/2* (cf. Figure 2.6).  
 le [<trait général>] est le temps de la conjugaison des verbes réguliers qui correspond aux <terminaisons du verbe>.

Figure 2.6: Deux exemples du fait *ending\_type/3*

Au départ il nous semblait superflu de répéter [present] dans l'argument [<trait général>] quand l'étiquette *er\_end\_pres* indiquait déjà le temps. Pourtant, nous nous sommes rendu compte que selon la façon dont les réalisateurs de LINGER ont organisé le dictionnaire allemand de LINGER, ils avaient besoin de noter le temps verbal dans l'argument [<trait général>]. Alors, il fallait remplir l'espace de cet argument en français aussi afin que le processus d'unification réussisse.

Une autre chose qui nous paraît curieux est le choix de préciser la conjugaison d'un verbe comme « aller » à tous les temps verbaux puisque au conditionnel présent et au futur simple, par exemple, la conjugaison est tout à fait systématique. Cela est encore plus étonnant quand on sait des <terminaisons du verbe> comme *er\_end\_pres* qu'il est possible de détailler une série de terminaisons une fois, leur donner une étiquette et les appeler quand il est nécessaire.

Le fait *actual\_endings/2* possède la même terminologie que *ending\_type/3*, donc, nous citons et la structure (Uren, 3.3.3, 21) et un exemple détaillé dans la Figure 2.7:

*actual\_endings*(<actual ending type>, [<actual ending>]).

```
actual_endings(er_end_pres,["e","es","e","ons","ez","ent"]).
```

où, dans le cas de notre exemple:

<actual ending type>, (<terminaisons du verbe>), est l'étiquette *er\_end\_pres* que nous avons vue dans le fait *ending\_type(er\_end,er\_end\_pres,[present])*. qui guide le logiciel aux terminaisons.

Figure 2.7: La structure du fait *actual\_endings/2* et un exemple détaillé

L'usage du reste des étiquettes suit le même schéma que celui de *er\_end\_pres*, sauf *gp\_end* qui est un peu plus compliqué. Le but de l'étiquette *gp\_end* est d'éviter la duplication des terminaisons d'accord régulières. Elle ne représente rien au masculin singulier, "s" au masculin pluriel, "e" au féminin singulier, et "es" au féminin pluriel. Ces terminaisons s'appliquent aux adjectifs réguliers et aux participes passés. Nous présentons le format du *gp\_end* dans la Figure 2.8 (cf. Appendice A, 1.0):

```
info(gp_end,[[gender(m),plurality(s)],[gender(m),plurality(p)],[gender(f),  
plurality(s)], [gender(f), plurality(p)]],[[],[]]).  
ending_type(gp_end,["","s","e","es"],[]).
```

Figure 2.8: Le *gp\_end*

Selon le premier fait, les terminaisons d'accord régulières (*gp\_end*, ou "group endings") peuvent être au masculin ou au féminin, au singulier ou au pluriel. Les terminaisons précises, " " (masculin, singulier), "s" (masculin, pluriel), "e" (féminin, singulier), et "es" (féminin, pluriel) sont données dans le deuxième argument du *ending\_type/3*.

## **La coquille**

La coquille contient toutes les routines qui font fonctionner le logiciel. Le travail de la coquille qui nous intéresse est celui de l'analyse syntaxique. Elle se divise en deux : la pré-analyse et l'analyse, ce que nous traitons séparément dans les pages qui suivent.

### **La pré-analyse**

Barchan résume le but de la pré-analyse ainsi :

The task of the pre-parse phase is to extract words from the input and attempt to match them against the entries in the dictionary. If this succeeds it will also return information about the word in general and about its acceptable endings together with the individual information they convey in particular. [c'est-à-dire tout ce qui est inclus dans l'entrée du mot dans le dictionnaire][...] Regardless of the relative success or failure of this stage, i.e. how many words actually correspond to dictionary entries, it always succeeds and conveys the items of information it managed to gather to the subsequent parse phase. (Barchan, 59)

Il ajoute plus loin que ce processus n'établit pas les catégories grammaticales des mots à cause des limites que cela implique. Étant donné que l'analyse syntaxique de la phrase se fait à la base de l'ordre des mots de la phrase, il est possible que, dans des cas où la fonction du mot puisse ne pas être claire pour l'apprenant, certaines alternatives soient exclues. Par exemple, le nom « travail » se confond souvent avec le verbe « travailler » à la troisième personne du singulier. Alors, si le logiciel n'identifie le mot « travail » que comme nom et un apprenant essayait la phrase *\*Il travail.*, l'analyse échouerait parce que la machine n'aurait pas pris en compte que « travail » soit une conjugaison fautive du verbe «travailler».

Afin d'avoir un aperçu complet de la pré-analyse, nous l'expliquons en six parties, commençant par décrire comment LINGER établit les arbres syntaxiques et ensuite comment il peut changer les données s'il est impossible de créer un arbre syntaxique à partir de ce que l'apprenant a écrit.



Ensuite, LINGER cherche mot par mot parmi la liste de radicaux possibles créée pendant la pré-analyse, afin de voir lequel entre eux correspond à la catégorie grammaticale identifiée par la syntaxe forte. Il commence avec le premier mot dans la liste de radicaux qui représente les possibilités pour le premier mot de la phrase originale. Si la catégorie grammaticale du premier radical de la liste est pareille à celle identifiée par la syntaxe forte, LINGER enlève ce radical de la liste, la met dans l'arbre syntaxique et le processus se répète sur la liste suivante de radicaux de la phrase. Si le radical est de la mauvaise catégorie grammaticale (comme « travail » de notre exemple *\*Il travail.*), la routine échoue et LINGER cherchera une autre possibilité en faisant marche arrière.

Comme nous l'avons dit dans l'explication de la grammaire de LINGER, le logiciel permet une faute de syntaxe forte par phrase. Afin d'assurer que l'analyse continue malgré cette faute, Barchan a créé ce qu'il appelle une routine de récupération (« recovery routine », 75) qui modifie la phrase dans l'espoir que le système puisse corriger la structure fautive. Il y a cinq types de modifications possibles: (1) déplacer un mot; (2) ajouter un mot; (3) substituer un mot; (4) supprimer un mot et (5) supprimer un mot et en ajouter un autre.

#### *Déplacer un mot*

Prenant une version fautive de notre phrase comme exemple, *\*Je ne jamais habite à Langley.*, LINGER reconnaîtrait cette phrase comme fautive parce que la structure *syntagme verbal --> particule négative, verbe auxiliaire, participe passé, particule négative* n'existe pas dans la grammaire du logiciel. Cependant, puisque la construction d'un énoncé au négatif requiert deux particules et elle ne correspond pas à l'anglais, ni à d'autres langues européennes, les apprenants se trompent parfois au placement du « ne » et « pas ». Afin de repérer une telle faute de placement, LINGER essaie de déplacer chaque radical de la phrase

à son tour, procédant de la gauche de la phrase à la droite. Dans le cas de notre exemple, cela veut dire que LINGER commence avec « Je », le mettant après « ne », ensuite après « jamais » etc., et puis il fait de même avec « ne » et puis « jamais » jusqu'à ce qu'il arrive à l'ordre des mots correct : Je n'habite jamais à Langley. Satisfaisant à la règle *syntagme verbal --> particule négative, verbe auxiliaire, particule négative, participe passé*, l'analyse de la syntaxe forte est acceptée. Si dans la phrase, il ne s'agit pas d'une faute de placement, cet algorithme échoue et LINGER passe au suivant : ajouter un mot.

#### *Ajouter un mot*

Parmi les exemples de notre corpus, nous remarquons que les apprenants ont tendance à ne pas inclure les articles (*\*J'aime la chaleur quand je suis à la plage mais quand je suis marche sur (le) terrain de rugby je suis malheureux.*), les prépositions (*\*Et, si ton amie veut aller au Brésil, elle peut aller à Rio de Janeiro ou le (au) nord-ouest du pays, qu'il y a des plages merveilleuses.*), le pronom relatif « que » (*\*Je pense (que) le quiz électronique est une idée fantastique pour étudier.*) et même les verbes auxiliaires (*\*L'été dernier, je (ai) visitée les Philippines.*). Dans ces cas où il y a un mot qui manque, LINGER essaie de réparer la phrase en insérant chaque catégorie grammaticale à son tour dans les écarts de la phrase jusqu'à ce qu'il trouve celle reliée à une structure décrite dans la grammaire. Par exemple, s'il analysait *\*L'été dernier, je visitée les Philippines.*, ayant identifié « je » comme sujet du syntagme nominal et « visitée les Philippines » comme participe passé et complément d'objet direct du syntagme verbal, en théorie LINGER devrait commencer à essayer d'insérer des catégories grammaticales entre le sujet et le participe passé jusqu'à ce qu'il trouve la catégorie grammaticale « verbe auxiliaire » et que l'analyse réussisse.

### *Substituer un mot*

Si, en déplaçant ou en ajoutant des catégories grammaticales, l'analyse ne réussit toujours pas, LINGER essaie de substituer les catégories grammaticales les unes pour les autres. En ce qui concerne nos exemples, cet algorithme nous semble utile pour corriger des fautes de conjugaison où l'apprenant utilise un infinitif à la place d'un participe passé ou même au lieu de conjuguer le verbe. Pour un exemple tel que *\*Il étudier beaucoup.*, LINGER supprimerait chaque mot de la phrase à son tour, commençant avec « Il », et essaierait d'insérer chaque catégorie grammaticale (verbe auxiliaire, pronom réfléchi, etc.) jusqu'à ce qu'il en trouve une qui marche (verbe qui remplacerait infinitif).

### *Supprimer un mot*

Pour supprimer un mot, LINGER enlève le premier mot de la phrase et puis il essaie d'analyser la phrase sans ce mot. Pour une phrase comme *\*Je veux fait yoga quand je suis moins d'occupé.*, il enlèverait le « Je » d'abord et essaierait de faire l'analyse. Cela échouerait, bien sûr, alors il répète ce processus à partir du prochain mot de la phrase et ainsi de suite, jusqu'à ce qu'il arrive à enlever le « d' » et l'analyse réussit.

### *Supprimer un mot et ajouter un autre*

Cet algorithme permet de gérer deux fautes différentes dans la même phrase où il y a un mot manquant ainsi qu'un mot supplémentaire. Cela serait nécessaire pour corriger une phrase comme *\*J'aime la chaleur quand je suis à la plage mais quand je suis marche sur terrain de rugby je suis malheureux.* parce qu'afin de la rendre acceptable à la pré-analyse, il faudrait ajouter un article avant le mot « terrain » et supprimer le mot « suis ». Sans les deux opérations à cette phrase, elle ne pourrait pas passer à l'analyse parce que même après qu'on aurait corrigé une des fautes, l'autre empêcherait la pré-analyse de la phrase d'avoir lieu.

Ayant recueilli dans le dictionnaire tous les radicaux des mots possibles de la phrase ainsi que leurs informations grammaticales, LINGER les place dans une base de données et passe à l'analyse de la phrase.

### **L'analyse**

L'analyse se divise en deux étapes : (1) raffiner les arbres syntaxiques selon les règles de syntaxe faible et (2) choisir la phrase correcte s'il y en a plus qu'une.

#### *Raffiner les arbres syntaxiques selon les règles de syntaxe faible*

Les règles de syntaxe faible sont représentées sous la forme d'une série de vérifications qui établit la morphologie de la phrase à partir des catégories grammaticales et des radicaux des mots identifiés dans la pré-analyse. Les informations récoltées de la pré-analyse permettent aussi au logiciel de ne sélectionner que les vérifications pertinentes à la phrase sous analyse.

Les vérifications comprennent sept types qui s'effectuent dans l'ordre suivant: `change_word`, `select_on_first`, `should_be`, `concord`, `comment`, `reorder` et `spelling`. Nous ne décrivons que brièvement la fonction de chacun de ces types de vérifications, comme leur application aux règles de syntaxe faible que nous traitons plus loin dans ce chapitre servira comme explication plus détaillée.

#### *Change\_word*

Les vérifications du type *change\_word* sont conçues pour enlever un mot et le remplacer par un autre de la même catégorie grammaticale. Elles s'appliquent aux constructions où le

choix de catégorie grammaticale est figé, comme par exemple dans le cas du choix du verbe auxiliaire en conjuguant à un temps composé avec un verbe réfléchi.

### Select on first

Les buts des vérifications du type *select\_on\_first* sont de réduire le nombre de radicaux possibles pour une catégorie grammaticale précise selon le mot écrit par l'apprenant et de choisir une valeur implicite si le mot ne soit reconnaissable. Ce type de vérification est nécessaire pour des catégories grammaticales qui peuvent avoir plusieurs formes correctes avec lesquelles d'autres mots dans la phrase doivent s'accorder. Logiquement le système établit cette forme (soit le singulier, soit le pluriel par exemple) avant de permettre des vérifications du type *concord* de faire les accords.

### Should be

Dans cette vérification, il s'agit d'identifier les traits de certains mots et de les modifier selon des conditions précises. Au contraire des vérifications du type *select\_on\_first* qui utilisent le mot de l'apprenant, sauf s'il n'est pas reconnaissable, *should\_be* fait son choix selon les conditions grammaticales de la structure où le mot apparaît. Une fois son choix fait, toutes les autres possibilités sont effacées de la base de données. Des vérifications du type *should\_not\_be* existent aussi et, comme l'indique leur nom, elles trouvent les mots qui ne devraient pas faire partie d'une structure particulière et les effacent de la base de données.

### Concord

Puisque la langue française exige plusieurs genres d'accord (sujet-verbe, nom-adjectif, etc.), il n'est pas étonnant que les vérifications du type *concord* soient les plus nombreuses. Elles

traitent les accords en constatant la catégorie grammaticale à modifier (ex. le participe passé), la structure où la catégorie grammaticale se trouve (ex. la proposition principale), des conditions auxquelles la phrase doit satisfaire (ex. conjuguer avec l'auxiliaire *être*), les accords qui sont applicables (ex. le genre et le nombre) et la catégorie grammaticale avec laquelle les accords se font (ex. le sujet).

### Comment

Au lieu de simplement présenter la phrase corrigée avec une liste cryptique des modifications effectuées, LINGER est programmé afin d'inclure des commentaires. Les vérifications du type *comment* incluent les commentaires qui s'intéressent à certaines fautes ainsi qu'une série de contraintes qui rassurent que le commentaire ne se fait que dans les circonstances appropriées.

### Reorder

Étant donné que l'allemand est une des premières langues que LINGER a su gérer, il fallait beaucoup de vérifications qui traitaient l'ordre des mots. Tandis que cela ne pose pas trop de difficultés en français, les vérifications du type *reorder* sont quand même nécessaires pour l'ordre des adjectifs et des pronoms en français. Les mots à considérer sont arrangés dans l'ordre selon lequel ils devraient apparaître de gauche à droite dans la phrase (par exemple, priority (1,determiner,[]), priority (2, adjective, [position(pre)]), priority(3, noun, []), priority(4, adjective, [position(post)]).).

### Spelling

Puisque LINGER fonctionne selon un système de radicaux et de terminaisons, il n'y a pas d'algorithme qui traite l'orthographe. Cependant il faut un moyen de contrôler les élisions, alors LINGER contient les vérifications du type *spelling* qui ajoutent ou enlèvent une apostrophe selon la première lettre du mot qui suit.

### Choisir la phrase correcte

Étant donné que LINGER ne fournit qu'une phrase correcte à l'apprenant, si, à la fin de l'analyse il ne reste plus qu'un arbre syntaxique qui est considéré comme correct, LINGER doit choisir la phrase qu'il va présenter à l'apprenant. S'inspirant de l'idée que LINGER devrait fournir une correction la plus proche possible de la phrase originale, il fait son choix en comparant les possibilités à la phrase originale. D'abord, il sélectionne les mots qui sont pareils entre la phrase originale et les possibilités, et ensuite, parmi les mots qui restent, il calcule le nombre de différences et choisit la phrase qui a le moindre nombre de différences comparée à l'originale. Nous comprenons que pour avoir un système qui corrige des phrases en plusieurs langues, ce processus de sélection doit être suffisamment général pour qu'il puisse marcher indépendamment de la langue en laquelle l'apprenant écrit. Pourtant, à notre avis, une meilleure alternative serait de programmer le logiciel pour qu'il choisisse la phrase correcte selon le niveau de l'apprenant. Une telle approche requerrait une hiérarchie des fautes selon laquelle le logiciel choisirait la phrase qui correspond le mieux au niveau de l'utilisateur.

## **La grammaire**

La grammaire de LINGER se divise en deux parties, nommées par les réalisateurs, syntaxe forte (« strong syntax ») et syntaxe faible (« weak syntax ») (Barchan). Nous traitons les deux sortes de syntaxe séparément dans les sections suivantes.

### **La syntaxe forte**

Le terme « syntaxe forte » chez LINGER réfère à la syntaxe d'une langue donnée ; il s'agit d'établir un système de règles qui définisse les constructions syntaxiques possibles en français. La notation qui exprime ce système de règles en Prolog s'appelle grammaire à clauses définies (Pereira). Les DCGs (de l'anglais, « Definite Clause Grammar») appartiennent à la grammaire syntagmatique non contextuelle, une classe de formalismes grammaticaux basés sur l'idée que la syntaxe d'une langue peut être exprimée selon des règles syntagmatiques qui sont valables dans n'importe quel contexte. Les informations grammaticales des DCGs sont organisées selon un symbole prédicat, suivi de termes entre parenthèses. Voici trois exemples dans la Figure 2.9:

```
sentence(formed(sentence,[MC,SC]))  
main_clause(MC)  
sub_clause(SC)
```

Figure 2.9: Exemples de l'organisation des informations grammaticales de la DCG

Liés ensemble, ces trois termes forment la règle syntagmatique suivante de la Figure 2.10:

```
sentence(formed(sentence,[MC,SC])) --> main_clause(MC), sub_clause(SC).
```

Autrement dit, une phrase complète peut contenir une proposition principale et une proposition subordonnée.

Figure 2.10: Exemple d'une règle DCG et son interprétation

À partir des structures les plus générales,  $\text{sentence}(\text{formed}(\text{sentence}, [\text{MC}])) \rightarrow \text{main\_clause}(\text{MC})$ . (cf Appendice B, 1.1) et  $\text{sentence}(\text{formed}(\text{sentence}, [\text{MC}, \text{SC}])) \rightarrow \text{main\_clause}(\text{MC}) , \text{sub\_clause}(\text{SC})$ . (cf Appendice B, 1.2), les règles de la syntaxe forte définissent les arbres syntaxiques de la langue française. Il s'agit d'identifier les syntagmes (par exemple  $\text{main\_clause}(\text{formed}(\text{main\_clause}, [\text{N}, \text{V}])) \rightarrow \text{subj\_phr}(\text{N}) , \text{verb\_phr}(\text{V})$ . (Appendice B, 2.1)), et d'établir les contenus des syntagmes (par exemple  $\text{subj\_phr}(\text{formed}(\text{subj\_phr}, [\text{S}])) \rightarrow \text{noun\_pron\_phr}(\text{S})$ . (Appendice B, 4.1)) jusqu'à ce que les règles arrivent à détailler les catégories grammaticales.

Le but de la syntaxe forte de l'analyse est d'établir le schéma syntaxique de la phrase écrite par l'apprenant ; c'est-à-dire d'identifier la catégorie grammaticale à laquelle chaque mot de la phrase appartient et d'établir les radicaux des mots de la phrase. Pourvu que cela réussisse, l'analyse continue et la phrase partiellement analysée fait l'objet des contraintes syntaxiques faibles. Pour gérer des phrases qui ont une faute de syntaxe forte, les réalisateurs de LINGER ont cherché une façon de continuer l'analyse :

The approach adopted has the distinct advantage of allowing a complete parse tree (rather than unconnected partial parses) to be generated in spite of the error as well as reporting the error. It involves a simple but effective priority-based recovery algorithm which makes a 'best-guess' as to the most appropriate sufficient correction. (Barchan, 5, 2.1.2.1)

Si la phrase a plus d'une faute de syntaxe forte, alors, l'analyse échouera. Barchan et Wusteman défendent comme suit leur choix de limiter les fautes de syntaxe forte à une seule:

In accordance with Imlah and du Boulay's observation that there is a point beyond which a set of words becomes too garbled to make a

reasonable guess as to the intended structure and the students are unlikely to produce utterances which are garbled to such a degree, it was felt to be reasonable to limit the number of strong syntactic errors in a single sentence to one. (Barchan, 5, 2.1.2.1)

### **La syntaxe faible**

La syntaxe faible réfère à la morphologie de la phrase. Afin d'engendrer les formes correctes des mots, à partir des radicaux identifiés par les contraintes de syntaxe forte, LINGER fait une série de vérifications. Chacune vérifie certains traits morphologiques d'une catégorie grammaticale selon son placement à l'intérieur de la phrase. Nous présentons un exemple dans la Figure 2.11:

```
check([determiner,adjective], noun_phr, [],concord([gender,plurality],noun,[])).
```

Autrement dit, dans le syntagme nominal composé d'un déterminant, d'un adjectif et d'un nom, vérifiez que le déterminant et l'adjectif s'accordent en genre et en nombre avec le nom.

#### Figure 2.11: Exemple d'une vérification de la syntaxe faible

Uren et Yazdani fournissent le modèle, que nous reproduisons dans la Figure 2.12, selon lequel toutes les vérifications sont conçues.

```
où check([<structure>], <scope>, [<precondition>], <requirement>).
```

[<structure>] sont les catégories grammaticales à vérifier (de notre exemple, le déterminant et le nom);

<scope> est la structure dans laquelle la vérification se fait (de notre exemple, le syntagme nominal) ;

[<precondition>] est une liste de conditions auxquelles la phrase doit satisfaire avant que la vérification ait lieu ; et

<requirement> est l'action que la vérification s'effectue sur la [<structure>] afin de changer sa morphologie. (3.2, 16)

#### Figure 2.12: Le modèle des vérifications

Puisque les vérifications s'occupent de beaucoup des corrections, il serait utile de les analyser dans le détail afin de mieux comprendre comment LINGER corrige. À cette fin, nous avons

organisé les vérifications selon leur catégorie grammaticale dans l'appendice C. Bien qu'il soit parfois difficile de reconnaître le modèle des vérifications que nous avons illustré dans la Figure 2.12, à cause des nombreuses conditions et des prédicats variés, il reste quand même cohérent.

### *Les pronoms*

Nous traitons tout d'abord les vérifications des pronoms. Leur fonction est, premièrement, de vérifier le choix de pronom dans plusieurs constructions par élimination de ceux dans la base de données qui ne sont pas possibles selon la structure de la phrase et, deuxièmement, d'établir l'ordre selon lequel les pronoms devraient apparaître dans la phrase. Le prédicat *select\_on\_first([case(acc)])* de la vérification 1.3 indique que la vérification des pronoms commence à partir des pronoms objet direct dans une phrase où il n'y a pas d'objet direct ni indirect. Cela exclut les pronoms nominatifs de la vérification, ce qui est juste parce qu'elle traite le contenu du syntagme verbal.

La vérification 1.1 précise que les pronoms qui font partie du syntagme nominal devraient être au nominatif selon la règle *should\_be([case(nom)])*, ce qui assure que s'il y a des pronoms d'un autre cas dans le syntagme nominal, ils seront exclus de la phrase. Inversement, la vérification 1.2, qui s'occupe des cas où le pronom fait partie du syntagme verbal, précise que les pronoms ne devraient pas être au nominatif selon la règle *should\_not\_be([case(nom)])*, ce qui assure que s'il y a des pronoms nominatifs dans le syntagme, ils seront exclus de la phrase. Dans le dictionnaire, chaque pronom est marqué nominatif, accusatif ou datif pour correspondre à ces vérifications. Le prédicat *should\_not\_be* prévoit des fautes dans l'usage des pronoms d'objet direct et indirect aussi. Elles interdisent de mettre un pronom d'objet direct dans une phrase où il y a déjà un

complément d'objet direct (cf Appendice C, 1.5). La même contrainte est maintenue pour les compléments d'objet indirect (cf Appendice C, 1.6). Pour établir qu'il y a déjà un complément d'objet direct dans le syntagme verbal, la vérification 1.5 utilise la condition *exists(dir\_obj\_phr)*. L'action de la vérification est d'assurer qu'aucun pronom dans la phrase écrite par l'apprenant n'est à l'accusatif. LINGER exprime cela par *should\_not\_be([case(acc)])*. La vérification 1.6 suit ce même schéma pour les pronoms d'objet indirect, sauf qu'elle substitue *dir\_obj\_phr* et *case(acc)* pour *indir\_obj\_phr* et *case(dat)*.

La vérification 1.7, qui s'effectue une fois que le système a établi le cas des pronoms de la phrase, utilise le prédicat *reorder* et chacun de ses arguments représente un groupe de pronoms, mis en priorité selon l'ordre que les pronoms objet doivent suivre. La vérification 1.7 commence au deuxième groupe qui inclut les pronoms *me, te, se, nous, vous, le, la, les*. Le troisième groupe comprend les pronoms *lui, leur*, le quatrième, *y* et le cinquième, *en*. Cet ordre nous semble problématique parce que les pronoms *me, te, se, nous* et *vous* ne sont pas séparés de *le, la* et *les*. Alors nous avons essayé la phrase *\*Il le me donne*. où l'ordre des pronoms est renversé. Cette expérience a produit la réponse suivante de LINGER :

Parse tree selected:

```

sentence
  main_clause
    subj_phr
      noun_pron_phr
        pronoun : [il,1]
      verb_phr
        pos_or_neg_verb_cons
        pos_verb_cons
        rp_pron_list
          pron_list
            pron_list2
              pronoun : [le,2]

```

pron\_list2  
pronoun : [me,3]  
pron\_list2  
governing\_verb  
verb\_or\_aux\_verb  
verb : [donne,4]  
dir\_obj\_phr  
indir\_obj\_phr

Your sentence was:

--- il le me donne . ---

I think the correct version of your sentence should read:

>>> il le me donne . <<<<

Nous n'étions pas étonnée de voir que LINGER n'avait pas corrigé la phrase. Cependant la présence d'un tel défaut dans le dictionnaire était assez surprenante et devrait être rectifiée, surtout parce que les apprenants ont tendance de mélanger l'ordre des pronoms.

L'autre vérification qui a le prédicat *reorder* contrôle l'ordre des mots dans un syntagme nominal; c'est-à-dire déterminant, adjectif, nom, adjectif. Elle inclut aussi les traits *position(pre)* and *position(post)* pour distinguer les adjectifs qui viennent avant le nom de ceux qui viennent après. Voici le schéma de cette vérification :

`reorder(beginning,noun_phr,[priority(1,determiner,[]),priority(2,adjective,[position(pre)]),priority(3,noun,[]),priority(4,adjective,[position(post)])])`.

(cf. 1.8)

Le dictionnaire contient les informations de placement pour chaque adjectif.

### *Le verbe principal, le verbe auxiliaire et les pronoms réfléchis*

En général, LINGER traite le verbe principal, le verbe auxiliaire et les pronoms réfléchis dans la même vérification parce qu'ils s'accordent tous avec le sujet de la phrase. Ce

manque de correction semble être une omission assez grave mais relativement facile à modifier. L'action des vérifications, exprimée par la règle *concord*([*person,plurality*],...) suivie d'un type de sujet, est de faire l'accord en nombre et en personne du verbe principal, du verbe auxiliaire et des pronoms réfléchis avec le sujet. Les différents types de sujets, précisés comme arguments du prédicat *concord*, et le placement de chacun à l'intérieur de la phrase, identifié par la structure où les verbes ou les pronoms apparaissent, incluent six cas différents: là où dans une proposition principale le sujet est un nom;

check([*verb,aux\_verb,reflexive\_pronoun*], **main\_clause**, [],  
concord([*person,plurality*],**subj\_phr(noun)**,[])).  
(cf 2.1)

dans une proposition subordonnée le sujet est un nom;

check([*verb,aux\_verb,reflexive\_pronoun*], **sub\_clause**, [],  
concord([*person,plurality*],**subj\_phr(noun)**,[])).  
(cf 2.2)

dans une proposition principale le sujet est un pronom;

check([*verb,aux\_verb,reflexive\_pronoun*], **main\_clause**, [],  
concord([*person,plurality*],**subj\_phr(pronoun)**,[])).  
(cf 2.3)

dans une proposition subordonnée le sujet est un pronom;

check([*verb,aux\_verb,reflexive\_pronoun*], **sub\_clause**, [],  
concord([*person,plurality*],**subj\_phr(pronoun)**,[])).  
(cf 2.4)

dans une proposition relative introduite par *qui* le sujet est le complément d'objet direct de la proposition principale;

check([*sub\_clause(verb),sub\_clause(aux\_verb),sub\_clause(reflexive\_pronoun)*], *sentence*,  
[**exists(relative\_subject)**],  
concord([*person,plurality*],**main\_clause(dir\_obj\_phr(noun))**,[])).  
(cf 2.5)

et dans une proposition relative introduite par *qui* le sujet est le pronom d'objet direct de la proposition principale.

check([sub\_clause(verb),sub\_clause(aux\_verb),sub\_clause(reflexive\_pronoun)], sentence,  
[exists(relative\_subject)],  
concord([person,plurality],main\_clause(pron\_list(pronoun)),[attribute(case(acc))])).  
(cf 2.6)

Nous ajoutons que ce n'est que les vérifications 2.5 et 2.6 qui ont besoin de conditions. Logiquement, la phrase doit satisfaire à la condition *[exists(relative\_subject)]* pour exiger un accord entre le complément d'objet direct de la proposition principale et le verbe principal, l'auxiliaire ou les pronoms réfléchis de la proposition relative.

Nous pouvons facilement imaginer des phrases qui correspondraient à l'accord que la vérification 2.5 est censé faire (par exemple l'accord entre *l'homme* et *va* dans la phrase suivante: *Je vois l'homme qui va au marché.*). Pourtant nous avons du mal à identifier des cas qui s'appliqueraient à la vérification 2.6 où l'accord se fait entre un pronom d'objet direct et le verbe de la proposition relative, sauf *Je le vois qui va au marché.*

Ces vérifications ne tiennent pas compte des phrases où l'antécédent est le sujet de la proposition principale, comme dans la phrase « Le chat qui voit le chien court. » Nous avons essayé cette phrase et LINGER l'a corrigée de la façon suivante :

Trying for a legal parse

Trying recovery level 1 (word in wrong place -- move)  
Trying a deletion

Recovery level 1 succeeded

Parse tree selected:

sentence  
  main\_clause  
    subj\_phr  
      noun\_pron\_phr

```

noun_phr
  determiner : [le,1]
  adj_list
  noun : [chat,2]
  adj_list
verb_phr
  pos_or_neg_verb_cons
  pos_verb_cons
  rp_pron_list
  pron_list
  pron_list2
  governing_verb
  verb_or_aux_verb
  verb : [voit,3]
dir_obj_phr
  noun_phr
  determiner : [le,4]
  adj_list
  noun : [chien,5]
  adj_list
indir_obj_phr
sub_clause
  relative_subject : [qui,6]
  verb_phr
  pos_or_neg_verb_cons
  pos_verb_cons
  rp_pron_list
  pron_list
  pron_list2
  governing_verb
  verb_or_aux_verb
  verb : [court,7]
  dir_obj_phr
  indir_obj_phr

```

Your sentence was:

--- le chat qui voit le chien court . ---

I think the correct version of your sentence should read:

>>> le chat voit le chien qui court . <<<<

\*\*\*\* Comments \*\*\*\*

relative\_subject "qui" was in the wrong place and has been moved

\*\*\*\* End Comments \*\*\*\*

Cette analyse est logique parce que la grammaire n'inclut pas de règles de syntaxe forte qui traite des propositions principales qui incluent des propositions relatives. Voilà une des nombreuses limites de LINGER par rapport à la complexité des phrases qu'il est capable de gérer.

En plus de vérifier l'accord des verbes et des pronoms réfléchis avec le sujet approprié, LINGER doit avoir un moyen de vérifier la conjugaison soit du verbe principal, soit du verbe auxiliaire. Cela est la tâche de la vérification 3.1 :

```
check([verb,aux_verb], sentence, [], select_on_first([tense(present)]))
```

Nous retrouvons dans cette vérification le prédicat *select\_on\_first* qui remplit deux fonctions. D'abord, il essaie de lier la forme du verbe dans la phrase avec une forme loisible selon le contenu du dictionnaire et, deuxièmement, s'il n'arrive pas à reconnaître le temps du verbe, il fixe le présent comme valeur implicite. Malheureusement, puisque LINGER cherche la plus proche alternative à ce qui est écrit par l'apprenant, il ne reconnaît pas dans un exemple comme *\*Je parlé.* que l'apprenant a simplement oublié le verbe auxiliaire. Alors, au lieu de rajouter un auxiliaire, il fournit une version vue comme la plus proche possible à l'originale selon sa programmation et corrige la phrase ainsi :

Parse tree selected:

```
sentence
  main_clause
    subj_phr
      noun_pron_phr
        pronoun : [je,1]
      verb_phr
        pos_or_neg_verb_cons
        pos_verb_cons
        rp_pron_list
```

pron\_list  
pron\_list2  
governing\_verb  
verb\_or\_aux\_verb  
verb : [parle1,2]  
dir\_obj\_phr  
indir\_obj\_phr

Your sentence was:

--- je parle1 .<sup>6</sup> ---

I think the correct version of your sentence should read:

>>> je parle . <<<<

\*\*\*\* Comments \*\*\*\*

verb "parle1" changed to "parle" :

[transitivity(i),follow\_prepos([]),can\_be\_reflex(y),person(1),plurality(s),tense(present)]

\*\*\*\* End Comments \*\*\*\*

Cela est vrai pour tous les verbes, sauf « voir » et « aller » parce que le participe passé de ces verbes ne ressemble pas du tout à la première personne du singulier du présent.

### *La transitivité d'avoir et d'être*

Etant donné que les verbes *avoir* et *être* peuvent constituer soit le verbe principal soit le verbe auxiliaire d'une phrase, il faut préciser à l'ordinateur de quel usage il s'agit dans la phrase donnée. Comme règle générale, LINGER attribue le trait *transitivité* à *avoir* et à *être*, (cf Appendice A, Section /\* Auxiliary Verbs \*/) puisqu'ils sont transitifs quand l'on les utilise comme des verbes principaux (ex. *J'ai un livre. Je suis étudiant.*). Sachant que LINGER se positionne par défaut sur *avoir* et *être* comme verbes principaux, quand on les utilise dans une phrase comme auxiliaire, il faut que l'ordinateur en tienne compte parce que sinon il s'attendra à un complément d'objet direct après *avoir* ou *être* et l'analyse et la correction

---

<sup>6</sup> Les numéros inclus dans les phrases traitées par LINGER indiquent les accents. 1 représente l'accent aigu, 2 l'accent grave, 3 la cédille et 4 l'accent circonflexe.

échoueront. Donc, le but de toutes les vérifications qui traitent les verbes auxiliaires est de rendre l'usage d'*être* ou d'*avoir* intransitif dans le calcul du logiciel. Puisque la transitivité est le trait syntaxique qui distingue l'usage d'*avoir* et d'*être*, LINGER se sert de la règle *change\_word("avoir",aux\_verb,ignore([transitivity]))* (cf 4.1) ou bien *change\_word("être",aux\_verb,ignore([transitivity]))* (cf 4.2, 4.3, 4.4, 4.5) pour dire à l'ordinateur que dans ce cas l'usage d'*avoir* ou d'*être* n'est pas la valeur implicite. Nous interprétons ces règles de la façon suivante :

« Changez le mot du verbe 'avoir' (ou bien 'être') en verbe auxiliaire, sans tenir compte de sa transitivité. »

Chacune s'applique à une phrase affirmative ou négative et précise les conditions auxquelles la phrase doit satisfaire pour activer le changement d'usage. Au lieu d'avoir une série de vérifications dont le but est de changer *avoir* et *être* à des verbes intransitifs, à notre avis, il vaut mieux inclure deux entrées dans le dictionnaire pour chaque verbe, l'une qui indique qu'ils sont transitifs et l'autre qu'ils sont intransitifs. De cette manière, l'analyse des temps composés réussira sans la complication de ces vérifications supplémentaires.

La vérification 4.1 comprend les conditions *[attribute(conjug\_with(avoir),past\_participle),empty(reflexive\_pronoun)]*; c'est-à-dire que la phrase doit contenir une forme conjuguée d'*avoir* et un participe passé et, en plus, afin d'assurer que l'auxiliaire est bien *avoir*, qu'il doit être sans pronom réfléchi. La vérification 4.2 est pareille sauf qu'elle traite l'auxiliaire *être*. La présence d'un verbe réfléchi, et donc l'auxiliaire *être* aussi, est indiquée par *exists(reflexive\_pronoun)* dans la vérification 4.3. Les vérifications 4.4 et 4.5 s'occupent de l'auxiliaire *être* dans le contexte d'un attribut (ex. *Je suis belle.*), la première dans le contexte d'une proposition principale et la seconde dans la proposition subordonnée. Selon la grammaire traditionnelle, on ne considère pas le verbe

*être* dans un syntagme adjectival comme verbe auxiliaire. Cependant, puisque la transitivité exige un complément d'objet direct, LINGER interprète le verbe *être* comme verbe auxiliaire dans le contexte d'un syntagme adjectival de sorte que l'analyse réussisse.

Puisque autrement LINGER traite le verbe *être* comme transitif, il ne peut pas gérer des propositions prépositionnelles. Par exemple, il répondra à une phrase comme *Je suis à la maison.* en disant que le verbe *être* est transitif et que par conséquent il s'attend à avoir un complément d'objet direct.

### *Les syntagmes adjectivaux*

Nous rencontrons les syntagmes adjectivaux plus loin dans les vérifications 5.1-5.4 dans l'appendice C, où LINGER doit vérifier l'accord en genre et en nombre entre le complément, qui est un adjectif à chaque fois, et le sujet. Les quatre vérifications comprennent deux catégories de structures : celle où le sujet avec lequel le syntagme adjectival doit s'accorder fait partie de la proposition principale et celle où le sujet est le complément d'objet direct de la proposition principale et le syntagme adjectival avec lequel il doit s'accorder fait partie de la proposition relative.

L'action d'accorder le syntagme adjectival avec le sujet suit un schéma qui est très semblable à celui que nous avons déjà vu pour les verbes principaux, les verbes auxiliaires et les pronoms réfléchis ; c'est-à-dire que chacune des quatre vérifications contient la règle *concord([gender,plurality],...)*, indiquant que l'accord se fait en genre et en nombre, suivie d'un type de sujet. Les seules conditions qui existent dans cette catégorie se trouvent dans les vérifications 5.3 et 5.4 et elles traitent la présence d'une proposition relative introduite par *qui* de la même façon que pour les vérifications 2.5 et 2.6.

Nous retrouvons le troisième argument, *[attribute(case(acc))]*, de la règle *concord* dans la vérification 5.4 qui précise que le pronom qui s'accorde avec l'adjectif est au cas accusatif. Nous n'arrivons pas à imaginer beaucoup de phrases qui aient un pronom objet direct comme sujet d'un syntagme adjectival introduite par *qui*.

### *L'accord du participe passé*

Les 21 vérifications des participes passés traitent l'accord. Les douze premières ont à voir avec l'usage d'*être* comme auxiliaire, les trois qui suivent s'intéressent à l'auxiliaire *avoir* précédé par un complément d'objet direct et les six dernières s'appliquent aux cas où l'auxiliaire est *avoir* tout simplement.

Nous divisons les douze vérifications d'accord du participe passé avec l'auxiliaire *être* en deux groupes : celles qui se réfèrent aux verbes transitifs et celles qui portent sur les verbes réfléchis. Le but de toutes est de vérifier l'accord en genre et en nombre du participe passé, ce qui est dénoté par *concord([gender,plurality],...)*. Parmi les vérifications 6.1-6.6, chacune a les conditions (*conjug\_with(e4tre),part\_participle*) (cf. 6.1, 6.2, 6.5, 6.6), ou bien (*conjug\_with(e4tre),sub\_clause(past\_participle)*) (cf 6.3, 6.4) pour assurer qu'il y ait une raison de faire un accord. Les variétés de structures dans lesquelles l'accord peut s'effectuer sont les suivantes: là où le participe passé est dans la proposition principale:

```
check([past_participle], main_clause, [attribute(conjug_with(e4tre),past_participle)],
concord([gender,plurality],subj_phr(noun),[])).
```

(cf 6.1)

ou bien dans la proposition subordonnée:

check([past\_participle], **sub\_clause**, [attribute(conjug\_with(e4tre),past\_participle)],  
concord([gender,plurality],**subj\_phr(noun)**,[])).  
(cf 6.2)

et le sujet accompagnant est un nom dans les deux cas:

1) où le participe passé est dans la proposition principale;  
check([past\_participle], **main\_clause**, [attribute(conjug\_with(e4tre),past\_participle)],  
concord([gender,plurality],**subj\_phr(pronoun)**,[])).  
(cf 6.5)

2) ou bien dans la proposition subordonnée;

check([past\_participle], **sub\_clause**, [attribute(conjug\_with(e4tre),past\_participle)],  
concord([gender,plurality],**subj\_phr(pronoun)**,[])).  
(cf 6.6)

et le sujet accompagnant est un pronom dans les deux cas

et 1) où le participe passé est dans la proposition relative introduite par *qui* et le sujet accompagnant est un nom;

check([**sub\_clause(past\_participle)**], sentence,  
[exists(**relative\_subject**),attribute(conjug\_with(e4tre),sub\_clause(past\_participle))],  
concord([gender,plurality],**main\_clause(dir\_obj\_phr(noun))**,[])).  
(cf 6.3)

2) ou bien un pronom;

check([**sub\_clause(past\_participle)**], sentence,  
[exists(**relative\_subject**),attribute(conjug\_with(e4tre),sub\_clause(past\_participle))],  
concord([gender,plurality],**main\_clause(pron\_list(pronoun))**,[attribute(case(acc))])),  
(cf 6.4)

qui est le complément d'objet direct de la proposition principale.

Puisque les vérifications 6.6-6.12 sont parallèles à celles que nous venons d'expliquer dans le sens qu'elles s'appliquent aux mêmes structures et elles utilisent l'auxiliaire *être*, nous ne soulignons que les particularités qui nous semblent importantes. Pour commencer, chacune contient la condition *exists(reflexive\_pronoun)* ou bien *exists(sub\_clause(reflexive\_pronoun))*. Puisque souvent un énoncé réfléchi n'a pas de complément d'objet direct, les vérifications 6.7-6.12 contiennent la condition *empty(dir\_obj\_phr)*. Finalement, pour assurer que la phrase n'ait pas de pronom d'objet direct, ces vérifications incluent la condition *not\_attribute(case(acc), pron\_list(pronoun))*.

Les vérifications 6.13-6.15 décrivent trois structures où le participe passé s'accorde en genre et nombre avec le complément d'objet direct. Elles incluent les cas où le participe passé est dans la proposition principale, la proposition subordonnée et la proposition relative introduite par *que*. La première condition de chacune est *attribute(conjug\_with(avoir),past\_participle)* (cf 6.13 et 6.14), ou bien *attribute(conjug\_with(avoir), sub\_clause(past\_participle))* (cf 6.15). En plus, les trois incluent *empty(reflexive\_pronoun)*, ou bien, *empty(sub\_clause(reflexive\_pronoun))* pour indiquer que si la phrase contient *me, te, nous* ou *vous*, ils sont plutôt des pronoms d'objet indirects (ex. *Il me les a donnés.*) ou bien d'objet direct (ex. *Il nous a abandonnés.*). Le reste des conditions dans la vérification 6.15 sont assez faciles à suivre. Elle précise que la phrase doit contenir le pronom relatif *que* (cf *exists(relative\_object\_que)*) et qu'il y a un complément d'objet direct dans la proposition principale (cf *exists(main\_clause(dir\_obj\_phr))*). L'action de la vérification 6.15 est de faire l'accord du participe passé en genre et en nombre avec ce complément d'objet direct qui est un nom (cf *concord([gender,plurality], main\_clause(dir\_obj\_phr(noun)),[])*). Les vérifications 6.13 et 6.14 sont très semblables entre elles, sauf que pour la 6.13 le participe passé se trouve dans la proposition principale et pour la 6.14 le participe passé se trouve dans la proposition subordonnée. Ainsi que les conditions *attribute(conjug\_with(avoir),past\_participle),empty(reflexive\_pronoun)...*, ces deux vérifications ajoutent que le syntagme verbal doit contenir un pronom et que le participe passé doit porter un accord (cf 6.13/6.14 *attribute(pp\_agree(y),pos\_or\_neg\_verb\_cons(pron\_list(pronoun)))*).

L'action de la 6.13 et la 6.14 est de vérifier que l'accord du participe passé, qui était déjà présent dans la phrase écrite par l'apprenant, correspond en genre et en nombre avec le pronom dans le syntagme verbal (cf 6.13/6.14 *concord([gender,plurality], pos\_or\_neg\_verb\_cons(pron\_list(pronoun)), [attribute(pp\_agree(y))])*). Étant donné qu'il est tout à fait possible qu'il y ait un pronom d'objet indirect ainsi qu'un pronom d'objet direct dans la phrase, il est impossible d'imposer que les pronoms de la phrase aient l'attribut «accusatif» (*attribute([case(acc)])*) parce que cela exclurait la possibilité d'avoir des pronoms d'objet indirect. En disant plutôt *pos\_or\_neg\_verb\_cons(pron\_list(pronoun))* cette vérification inclut les pronoms *me, te, le, la, les, nous, vous, lui, leur, y* et *en* (cf. Appendice A, /\* Pronouns \*/). Le désavantage de cette méthode est qu'elle ne fournit pas un moyen selon lequel LINGER peut distinguer entre les sens des pronoms *me, te, nous* et *vous*. Donc, quand on essaie un exemple tel que *\*Il nous l'a donnés.*, LINGER l'accepte comme correcte, parce qu'il ne reconnaît pas que le *nous* est un pronom d'objet indirect et ne devrait pas s'accorder avec le participe passé. Par contre, dans des phrases où il n'y a qu'un pronom, LINGER la corrige sans problème parce que la transitivité est incluse comme trait du verbe principal dans le dictionnaire.

La dernière condition à souligner dans les vérifications 6.13 et 6.14 est *empty(infinitive\_phr)*. Elle a à voir avec le placement des pronoms par rapport à l'accord. Si la phrase contient un infinitif, le pronom objet le précède (ex. *J'avais espéré la voir.*) ; ce n'est que dans le cas où il n'y a pas d'infinitif que le pronom se place avant le verbe conjugué. Donc, pour avoir un accord entre un pronom objet direct et un participe passé, la phrase ne peut pas contenir d'infinitif.

La fonction des vérifications 6.16 et 6.17 semble être de vérifier que dans des cas où le verbe est réfléchi et il y a un complément d'objet direct, le participe passé n'a pas d'accord (ex. *Elle s'est lavé les mains.*). LINGER est capable de corriger les fautes d'accord de ce type mais nous ne voyons pas comment, puisqu'une des conditions de ces vérifications est que le verbe auxiliaire soit « avoir ». Cela est encore plus énigmatique quand on sait que la vérification 4.3 rendra l'auxiliaire *être* avant que les vérifications 6.16 et 6.17 aient lieu.

Les vérifications 6.18-6.21 assure que le participe passé n'a pas d'accord selon quatre cas à travers la règle *should\_be([gender(m),plurality(s)])*. Par conséquent, 6.18 et 6.19 ont les conditions *[attribute(conjug\_with(avoir), past\_participle), empty(reflexive\_pronoun), not\_attribute(pp\_agree(y), pos\_or\_neg\_verb\_cons(pron\_list(pronoun)))]* pour préciser que le verbe doit être conjugué avec *avoir* suivi d'un participe passé et qu'elle n'aura pas de pronom réfléchi ni de pronom objet direct qui affecterait l'accord du participe passé. Puisque la vérification 6.19 traite les constructions où le participe passé est dans la proposition subordonnée, elle a un trait supplémentaire qui est *empty(relative\_object\_que)*. Cela enlève les instances d'accord entre l'antécédent du pronom relatif et le participe passé. Comparant la structure que les vérifications 6.20 et 6.21 représentent à celle des 6.18 et 6.19, la seule différence est que les premières ont le trait additionnel d'un infinitif. Elles sont un exemple du phénomène que nous avons exploré aux vérifications 6.13 et 6.14 à l'inverse. En incluant la règle *exists(infinitive\_phr)*, on évite toute complication d'accord entre le participe passé et un pronom et, donc, malgré l'absence de référence à des pronoms, ces vérifications devraient être capables de les gérer, si la phrase en contenait un.

### *Les commentaires*

À la différence des vérifications que nous avons regardées jusqu'ici, celles qui se réfèrent au verbe principal sont plutôt des corrections. Chacune s'applique à une construction grammaticalement impossible et fournit un commentaire qui explique la faute. Pourtant, ces corrections suivent le même schéma que les vérifications, sauf la quatrième étape où, au lieu de préciser le trait morphologique à être vérifié, il y a un commentaire particulier. Pour commencer, nous prenons comme exemples les quatre premières qui traitent des fautes de transitivité. 7.1 et 7.2 traitent des constructions où le verbe principal est intransitif (cf 7.1/7.2 *attribute(transitivity(i),governing\_verb)*) mais la phrase contient un complément d'objet direct dans la forme soit d'un nom (cf 7.1), soit d'un pronom (cf 7.2). Basé sur la présence de la condition *exists(dir\_obj\_phr)* (cf 7.1) ou bien, *attribute(case(acc), pos\_or\_neg\_verb\_cons(pronoun))* (cf 7.2) dans une phrase où le verbe principal est intransitif, LINGER engendre le commentaire "Verb is intransitive so does not expect an object". Nous ajoutons ici qu'il n'y a pas de prévision dans LINGER pour les verbes comme « sortir » ou « descendre » qui peuvent être soit transitifs, soit intransitifs selon l'usage. Les corrections 7.3 et 7.4, par contre, s'occupent des verbes transitifs dans des phrases où le complément est absent. Puisque le commentaire de cette correction "Verb is transitive so does expect an object" (cf 7.3/7.4) indique à l'apprenant qu'il aurait dû mettre un complément, les conditions doivent comprendre tous les compléments possibles qui n'étaient pas dans la phrase originale pour que le commentaire soit approprié. Parmi les verbes dans le dictionnaire qui sont marqués transitifs, il y en a trois (*décider*, *finir* et *manger*) qui ne sont pas obligés d'avoir un complément d'objet direct. Cela est une autre chose dont LINGER ne tient pas compte. Nous avons essayé la phrase *Je mange*. et tandis qu'il n'a rien ajouté à la

phrase, il a fait le commentaire "*manger*" : *Verb is transitive so does expect an object*, ce qui n'est pas forcément vrai selon l'usage du verbe. Afin de résoudre ce problème, nous suggérons que les verbes comme décider, finir et manger aient deux entrées lexicales, l'une qui indique qu'ils sont transitifs et l'autre qu'ils sont intransitifs. Une telle addition rectifierait des corrections fautives potentielles.

La correction 7.5 est conçue pour répondre à une phrase où l'apprenant a ajouté un pronom réfléchi à un verbe qui ne peut jamais avoir cet usage. Pour vérifier que le verbe que l'apprenant a utilisé ne peut pas être réfléchi, la correction inclut la condition *can\_be\_reflex(n)*. Cet attribut fait partie des traits qui décrivent chaque verbe dans le dictionnaire, alors pour savoir si le verbe en question peut être réfléchi ou pas, il faut que *can\_be\_reflex(n)* relie les informations sur le verbe dans le dictionnaire, sinon la correction échouera.

La correction 7.6 s'occupe des verbes qui ne peuvent pas être suivis d'un infinitif. Elle montre une condition que nous n'avons pas vue jusqu'ici : *follow\_prepos([],governing\_verb)*. Pour savoir ce que cela veut dire, il faut revenir au dictionnaire où nous découvrons que les informations entre parenthèses comprennent trois types. Prenant le verbe *décider* comme exemple du premier type, nous voyons que les informations qui correspondent à *follow\_prepos/1* sont la préposition *de* (cf Appendice A, /\* Verbs \*/) qui doit précéder l'infinitif, si la phrase en contient un. Les parenthèses peuvent inclure des guillemets aussi, comme dans le cas du verbe *aller* (cf Appendice A, /\* Verbs \*/) qui indiquent qu'un infinitif peut suivre le verbe principal, mais sans préposition. Finalement, nous arrivons au type de la correction 7.6, illustré dans le dictionnaire par le verbe *manger* (cf Appendice A, /\* Verbs \*/): les parenthèses vides. Elles veulent dire que ni un infinitif ni une préposition ne puissent

suivre ce verbe particulier. Nous avons remarqué que les informations pour quelques-uns des verbes sont incomplètes. Par exemple, selon le dictionnaire, *arriver* et *faire* ne peuvent être suivis ni d'un infinitif ni d'une préposition. Cependant cela ne tient pas compte des constructions *arriver + à + infinitif* (ex. *J'arrive à comprendre le français.*) ni *faire + infinitif + complément d'objet direct* (ex. *Je fais construire une maison.*). La correction 7.7, qui corrige les fautes de préposition avant un infinitif, utilise (*follow\_prepos([ ]),governing\_verb*) aussi mais dans ce cas il est introduit par le prédicat *not\_attribute* pour exprimer qu'il est possible que le verbe principal soit suivi d'un infinitif ou d'une préposition. Nous rencontrons aussi la règle *not\_one\_of(follow\_prepos,verb\_preposition,governing\_verb)* qui explique que la préposition qui suit le verbe principal dans un syntagme verbal de la forme *pos\_or\_neg\_verb\_cons(V)* , *verb\_preposition(P)* , *infinitive\_phr(I)* (cf Appendice B, 5.1) dans la phrase écrite par l'apprenant ne correspond pas à la préposition attribuée à ce verbe dans le dictionnaire. Cela assure que la correction ne s'effectue que sur cette construction, ce qui enlève la possibilité de la correction fautive d'une préposition qui n'existe pas avant un infinitif mais qui est obligatoire avant un complément de lieu et vice versa ex. *Je vais manger.* (*aller* sans préposition + infinitif) mais *Je vais à la discothèque.* (*aller* avec la préposition *à* + complément de lieu). Nous notons ici que Barchan admet que LINGER ne peut pas traiter les contractions comme « du », « au » etc. parce qu'il suppose que chaque mot appartienne à une seule catégorie grammaticale. Il suggère d'ajouter une fonction qui divise la contraction avant l'analyse syntaxique et de la reconstruire avant de présenter les résultats de la séance. Cependant il admet que cette idée pourrait poser des problèmes pour la contraction « du » parce qu'elle risquerait de compliquer l'analyse non-contextuelle par confusion avec « dû » (123).

Les corrections 7.8-7.10 traitent des phrases hypothétiques et potentielles en *si*. La première explique des fautes de temps dans la proposition relative.

Utilisant les contraintes,

```
[exists(relative_object_si),  
not_attribute(tense(present),  
sub_clause(governing_verb)),  
not_attribute(tense(imperfect),  
sub_clause(governing_verb))]
```

la correction assure que la phrase écrite par l'apprenant se construit avec *si*, et que le temps du verbe dans la proposition relative n'est ni le présent, ni l'imparfait, ce qui constitue la faute. Ayant satisfait à ces conditions, LINGER engendre le commentaire "'si' can only be followed by either the present or imperfect tenses". Ce commentaire n'est pas tout à fait vrai parce que le plus-que-parfait est possible aussi ainsi que plusieurs autres temps (par exemple, Je ne sais pas s'il viendra (le futur), Je ne savais pas s'il serait là (le conditionnel)).

Les corrections 7.9 et 7.10 s'orientent plutôt vers la proposition principale. La 7.9 s'applique à la faute de ne pas mettre le verbe dans la proposition principale au futur simple quand la proposition relative s'écrit au présent. Pour établir les contraintes de la proposition relative, la correction inclut les informations suivantes:

```
exists(relative_object_si),attribute(tense(present),sub_clause(governing_verb))
```

pour indiquer que la proposition relative commence avec *si* et le verbe s'écrit au présent. Elle précise aussi l'argument *not\_attribute(tense(future),main\_clause(governing\_verb))* afin de dire que le verbe principal ne s'écrit pas au futur simple. Le résultat de tout cela est le commentaire "If 'si' is followed by the present tense, the main clause should be in the

future". La correction 7.10 est exactement pareille sauf qu'elle s'applique aux phrases où le verbe de la proposition relative est à l'imparfait et le verbe principal devrait être au conditionnel présent.

### *Les noms*

Il n'y a qu'une vérification qui traite les noms (cf Appendice C, 8.1). Comme les autres vérifications de ce type que nous avons vues, *select\_on\_first* identifie si chaque nom dans la phrase est pluriel ou singulier et puis elle élimine les autres possibilités dans la base de données qui ne correspondent pas à ce nombre. Par exemple, si le nom était « chiens » la forme singulière « chien » serait enlevée de la base de données comme possibilité. La valeur implicite du *select\_on\_first* pour les noms est la forme plurielle et elle est réservée aux phrases où LINGER ne trouve pas le mot qui correspond à ce que l'apprenant a écrit.

Barchan justifie le choix du pluriel ainsi :

...[I]f a noun has an incorrect ending it is more likely that this is an incorrect plural form than a singular one - in which case a version such as « bateaus » will be corrected to « bateaux », as would presumably have been intended. (1987, 104)

### *L'orthographe*

Quant à l'orthographe, les vérifications 9.1-9.3 ne traitent que l'usage des apostrophes. Introduit par le prédicat *change\_spelling*, la 9.1 indique que si la première lettre du mot suivant commence avec une voyelle, il faut une apostrophe, tandis que la 9.2 décrit le cas opposé de ne pas mettre une apostrophe où le mot suivant commence avec une consonne.

En ce qui concerne d'autres aspects de l'orthographe, il n'y a pas d'algorithme qui traite ce genre de faute parce que les réalisateurs prétendent que:

...[T]here would be a strong danger of treating a perfectly correct but unknown word as a misspelling of one or more words which do exist in the dictionary. (Barchan, 5, 2.1.3)

Afin de gérer des fautes d'orthographe LINGER profite des radicaux des mots (<word roots>) que nous avons vus. Pourvu que le radical du mot soit dans le dictionnaire, le logiciel traite la faute d'orthographe comme si le mot avait la terminaison fautive. Si le radical du mot n'est pas dans le dictionnaire, LINGER devine qu'il appartient à une classe grammaticale « ouverte »; c'est-à-dire les noms, les verbes, les adjectifs, mais non pas les prépositions, les déterminants ni les conjonctions.

Grâce à cet aperçu de son architecture, nous voyons déjà que LINGER est assez limité. Les règles de syntaxe forte et faible ne semblent tenir compte que d'un petit morceau de grammaire de base, la façon de faire les analyses, surtout la pré-analyse, paraît inefficace et les commentaires sont la seule prévision pédagogique du système. Pourtant, pourvu que la structure et le vocabulaire des phrases soient incluses dans la grammaire et dans le dictionnaire, LINGER devrait être capable de les corriger. Nous consacrerons donc le chapitre qui suit à des tests qui révéleront à quel point LINGER pourrait corriger des phrases fautives.

## Chapitre 3: La correction théorique et appliquée

### ***Vers une définition théorique de la correction***

Dans son article « Idiosyncratic Dialects and Error Analysis », Stephen Corder définit la correction comme une reconstruction.

A reconstructed sentence is, roughly speaking, what a native-speaker of the target language would have said to express *that* meaning in *that* context, i. e. it is a translation equivalent. (Corder, 155)

Il est implicite dans cette définition que la nécessité de reconstruire une phrase provient du fait qu'elle est fautive (ou bien « idiosyncratic» selon la terminologie de Corder) mais, qu'est-ce que cela veut dire ? Quels sont les critères d'une faute ? Est-ce qu'on peut toujours identifier les fautes ? Pourquoi est-ce que les fautes se font et comment est-ce qu'elles nous aident à étendre la définition de la correction de Corder ? Quel est le rapport entre les théories de l'apprentissage des langues secondes, de l'analyse des fautes et de la correction présentées dans ce chapitre et celles que les réalisateurs de LINGER soutiennent ? Le but de la première partie de ce chapitre sera de proposer des réponses à ces questions.

### **La définition d'une faute**

Dans son étude d'un corpus parlé de quatre apprenants d'anglais, Paul Lennon cite deux notions d'une faute avant d'établir sa propre définition :

...[A]pparently clear-cut definitions of L2 error in spoken English such as, 'the use of a linguistic item in a way which, according to fluent users of the language, indicates faulty or incomplete learning' (Chun *et al.* 1982:538) or: 'An error occurs where the speaker fails to follow the pattern or manner of speech of educated people in English speaking countries today' (Liski and Puntanen 1983:227) are by no means unambiguous. A more cautious definition will be employed in this study, namely:

a linguistic form or combination of forms which, in the same context and under similar conditions of production, would in all likelihood, not be produced by the speakers' native speaker counterparts.  
(Lennon, 182)

Les trois définitions suggèrent que les francophones reconnaissent la justesse d'un énoncé selon une série de normes linguistiques. Grâce à leur formation et à leur familiarité avec la langue, ils savent, jusqu'à un certain point au moins, ce qui se dit et ce qui ne se dit pas en général. Il est clair que parmi la gamme de fautes possibles qui pourraient briser les normes linguistiques d'une langue, il y a deux niveaux: les phrases qui ne rentrent pas dans le cadre des règles morpho-syntaxiques de la langue (appelées des phrases « overtly idiosyncratic» par S.P. Corder (1971) 155) et celles qui, malgré leur justesse grammaticale, ne se disent pas et, donc, ne sont pas acceptables (appelées des phrases « covertly idiosyncratic» par S.P. Corder (1971) 155).

### **Les critères d'une faute**

James Carl identifie la grammaticalité et l'acceptabilité comme les deux critères auxquels la phrase doit satisfaire pour être correcte. Il décrit la grammaticalité en détail dans les remarques suivantes :

It is the grammar (not you or I) who decides whether something said by a learner is grammatical. The grammar bases decisions about the grammaticality of contentious cases on principles extracted from observing clear cases. Appeal to grammaticality is an attempt to be objective, to take decisions such as whether some bit of language is erroneous or not out of the orbit of human whim. So, if we can point to a bit of language and say that there are no circumstances where this could ever be said in this way, we are dealing with ungrammaticality. The reason for this is that ungrammaticality is context-free. Thus a judgment of the grammaticality of a sentence does not appeal to context. As we have seen, there is *no* context where an ungrammatical bit of language could be right. If, on the other hand, we can agree that it might be said in some context, it is well-formed."  
(James, 65)

James définit l'acceptabilité de la façon suivante :

'An acceptable utterance is one that has been, or might be, produced by a native speaker in some appropriate context and is, or would be, accepted by other native speakers as belonging to the language in question.' (Lyons, 1968: 137) [...] To decide on the acceptability of a piece of language we refer not to rules, but to contexts, trying to contextualize the utterance in question. (James, 67)

Afin d'illustrer cet argument, James propose l'exemple suivant :

*Pele (the Brazilian footballer) wore a green dress.* Yes, if he were taking part in the Rio carnival celebrations. But the learner who said this was referring to the context where Pele was playing football, and by 'dress' meant 'shirt' or 'strip'. So this is an error of acceptability. (James, 67)

Comme nous le voyons dans cet exemple, une phrase qui satisfait au critère de grammaticalité n'est pas forcément acceptable.

Au contraire des définitions précises de James qui semblent suggérer qu'il est facile d'établir la grammaticalité et l'acceptabilité d'une phrase, Lyons affirme que:

Since the sentences of the language being described are so numerous [...], one cannot hope to decide for every sentence generated by the grammar that it is definitely acceptable or unacceptable. In fact, one does not have to go very far with the grammatical description of any language before one finds disagreement among native speakers about the acceptability of sentences generated by rules tentatively established by the grammarian. (Lyons, 153)

Rejetant partiellement la généralisation que les personnes de langue maternelle savent intuitivement la structure grammaticale de leur langue, Lyons maintient que:

It is with respect to the less general principles that alternative grammars might differ in their characterization of sentences as grammatical and ungrammatical. And the 'intuitions' of linguists and speakers of the language tend to be unreliable and inconsistent at this point anyway. (154)

A notre avis, il existe des cas extrêmes où on ne peut plus se servir des intuitions des personnes de langue maternelle pour savoir si une phrase est acceptable ou pas parce qu'elles n'arrivent pas toujours à un consensus. Pourtant, puisqu'on ne sait pas, dans ces circonstances, si la phrase est acceptable ou pas, il n'y a aucun moyen de la corriger, alors il nous suffit de reconnaître qu'une faute, et par extension une correction, n'est pas toujours possible.

### **L'origine des fautes**

Maintenant que nous savons qu'une faute est l'élément dans un énoncé qui brise les normes linguistiques d'une langue et que, dans la plupart des cas, la grammaticalité et l'acceptabilité sont les deux critères qui déterminent les normes linguistiques, explorons les théories qui expliquent pourquoi les fautes se font. On peut classer les fautes en deux catégories : les fautes d'interférence qui proviennent de l'« interlangue » (Selinker, 71), ou bien du «transitional dialect » (Corder, 151), et des « intralingual and developmental errors » (Richards, 173) qui ont leur origine dans tout ce qui n'est pas lié à l'interlangue.<sup>7</sup>

### *L'interlangue*

Corder définit ce qu'il appelle « transitional dialect » de la façon suivante :

It has a grammar, and is, in principle, describable in terms of a set of rules, some sub-set of which is a sub-set of rules of the target social dialect. His dialect [celui de l'apprenant] is unstable [...] and is not, so far as we know a 'langue' in that its conventions are not shared by a social group, [...] and lastly, many of its sentences present problems of interpretation to any native speaker of the target dialect. Selinker (1969) has proposed the name *interlanguage* for this class of idiosyncratic dialects, implying thereby that it is a dialect whose rules share characteristics of two social dialects or languages, whether the

---

<sup>7</sup> Nous reconnaissons que les fautes imprévisibles dues à la fatigue, à la typographie etc. constituent également une catégorie. Pourtant, nous n'allons pas la traiter ici.

languages themselves share rules or not. [...] An alternative name might be *transitional dialect*, emphasising the unstable nature of such dialects. (Corder, 151)

Par définition, alors, la présence de l'interlangue signifie que:

a large number, but by no means all, of his errors [celles de l'apprenant], are related to the systems of his mother-tongue. These are ascribed to interference from the habits of the mother-tongue, as it is sometimes expressed. (Corder, 168)

En plus,

it is certainly the case that teachers work on the assumption that a group of learners having the same mother tongue and having had the same experience of learning the second language speak more or less the same interlanguage at any point in their learning career, and that what differences there are can be ascribed to individual variation in intelligence, motivation and perhaps attitude. This belief is inherent in the notion of 'teaching a class' as opposed to an 'individual', and indeed, it is difficult to see how one could proceed otherwise. (Corder, 154)

Ces remarques expliquent pourquoi, dans une classe de français composée d'anglophones qui ont passé par le même système d'éducation, les apprenants font beaucoup de fautes identiques. Il est clair à partir d'exemples fréquents comme *\*Je suis vingt ans.* et *\*Le temps est beau.* que l'interlangue des apprenants les mène à appliquer l'orthographe, la syntaxe et la morphologie de l'anglais à leurs énoncés en français.

#### *Les fautes intralingues et développementales*

Dans cette dernière citation, Corder attribue les fautes qui ne sont pas justifiées par l'interlangue aux variations d'intelligence, de motivation et d'attitude. Ces variations expliquent la présence des fautes que Richards identifie comme « intralingual and

developmental errors» dans son article « A Non-Contrastive Approach to Error Analysis». Il définit ces genres de fautes ainsi :

[I]ntralingual errors are those which reflect the general characteristics of rule learning, such as faulty generalization, incomplete application of rules, and failure to learn conditions under which rules apply. Developmental errors illustrate the learner attempting to build up hypotheses about the English language from his limited experience of it in the classroom or textbook. (174)

Afin de clarifier ces types de fautes, nous fournirons une définition et un exemple de chacun. Citant Jakobovits, Richards constate que le mot généralisation veut dire « the use of previously available strategies in new situations....In second language learning...some of these strategies will prove helpful in organising the facts about the second language, but others, perhaps due to superficial similarities, will be misleading and inapplicable » (174). Prenons comme exemple la conjugaison. À cause de la similarité entre infinitifs, les apprenants essaient souvent d'appliquer les règles de conjugaisons d'un verbe régulier à un verbe irrégulier. Par exemple, en vue de la ressemblance entre « rendre » (régulier) et «prendre » (irrégulier), les apprenants généralisent les règles de conjugaison, inventant *\*nous prendons*.

Le terme « incomplete application of the rules » s'applique aux fautes où les apprenants se trompent parce qu'ils ne comprennent pas complètement les règles. Quand ils apprennent comment former l'impératif avec des pronoms d'objet direct/indirect puis étudient l'impératif avec les verbes pronominaux, les apprenants ont tendance à rajouter un pronom réfléchi à l'impératif où aucun pronom n'est requis. Ils écriraient, par exemple, *\*Mangez-vous !* où ils veulent dire *Mangez !* Cela indique qu'ils n'arrivent pas à distinguer entre les pronoms d'objet direct/indirect et les pronoms réfléchis.

Le dernier type de faute intralingue, « failure to learn conditions under which rules apply », réfère à « the application of rules to contexts where they do not apply » (175). Un exemple de cela est l'accord du participe passé des verbes pronominaux. En généralisant que le participe passé s'accorde avec le pronom réfléchi, souvent les apprenants font l'accord même si le pronom réfléchi représente un complément d'objet indirect, par exemple *\*Elles se sont parlées*.

Les fautes développementales se produisent quand les apprenants formulent des hypothèses linguistiques fautives au sujet de la langue cible selon des règles qu'ils ont déjà apprises. Par exemple, ayant appris que les verbes intransitifs qui, normalement, expriment une action de mouvement, se conjuguent aux temps composés avec l'auxiliaire *être*, ils font l'hypothèse que cela est vrai pour tous les verbes qui ont ces traits. Donc, quand ils conjuguent un verbe comme *marcher*, qui ne correspond pas à cette règle, ils se trompent, écrivant quelque chose du genre *\*il est marché*.

### **La correction**

À la lumière des remarques que nous avons soulignées concernant la définition et l'origine des fautes, revenons à la définition de la correction. Nous reconnaissons que la notion de la correction comprend deux éléments liés : un processus et un résultat. Dans la définition que nous avons citée au début du chapitre, il s'agit d'un résultat ; la correction est le changement qu'on fait à un énoncé pour qu'il se conforme aux normes linguistiques de grammaticalité et d'acceptabilité établies par des locuteurs dont c'est la langue maternelle. Par contre, le processus de corriger exige, d'abord, qu'on identifie la faute et ensuite qu'on la transforme.

La façon de faire cela peut varier selon le but de la correction. James identifie trois types de correction ainsi que le but de chacun:

1. Informing the learners that there is an error, and leaving them to discover it and repair it themselves. I would call this sort of intervention **feedback**, commonly defined as giving knowledge of 'results'....
2. Providing treatment or information that leads to the revision and correction of the specific instance of error (the error **token**) without aiming to prevent the same error from recurring later. In addition to indicating that the present attempt is wrong, the corrector can specify how and where, suggest an alternative, give a hint...It is aimed at **product enhancement**. This I would call **correction** proper.
3. Providing learners with information that allows them to revise or reject the wrong rule they were operating with when they produced the error token. The result will be to induce learners to revise their mental representation of the rule, so that this error **type** does not recur....Since its purpose is primarily to improve the processes for all future productions rather than merely 'cosmetically' to improve the present product, I would call this **remediation**. (236-7).

Nous sommes d'accord avec les deux derniers types de correction mais nous rejetons le premier parce qu'il ne satisfait qu'à la première étape du processus. Le deuxième type correspond à notre définition de la correction et il est utile à l'apprenant parce qu'il montre ce que l'apprenant doit changer pour avoir un énoncé correct, mais le troisième type représente, à notre avis, la façon la plus efficace de corriger. Derrière l'analyse des fautes existe une théorie de l'apprentissage des langues secondes que Corder explique ainsi :

[L]anguage learning is some sort of data-processing and hypothesis-forming activity of a cognitive sort. According to this view his idiosyncratic sentences are signs of false hypotheses, which, when more data is available and processed, either by direct observation or by statements by the teacher, i. e. corrections and examples, enable the learner to formulate a hypothesis more in accordance with the facts of the target language.[...] The 'correction' of error provides precisely the sort of negative evidence which is necessary to discovery of the correct concept or rule. (Corder, 159)

Donc, que ce soit des fautes qui proviennent de l'interférence de la langue maternelle, des généralisations fautives, d'une compréhension soit incomplète soit trop vague ou d'une

supposition inapplicable, « remediation », comme l'appelle James, est la meilleure forme de correction parce qu'elle essaie de guérir l'hypothèse fautive concernant la représentation de la grammaire de la langue cible dans l'esprit de l'apprenant, ce qui avancera son apprentissage de la langue.

Les détails disponibles concernant les théories sur lesquelles LINGER est basée sont très limités. Nous savons du but d'un vérificateur de grammaire cité dans le chapitre deux que LINGER est censé « to provide the learner with sufficient guidance to construct a grammatically acceptable sentence based largely on the learner's original input. » et que LINGER fait cela en fournissant une version correcte de la phrase originale de l'apprenant ainsi que des remarques pertinentes qui expliquent les modifications qu'il a faites. En plus, s'accordant avec l'objectif que « the system should be easily configured for a particular natural language by an expert in that language but not in computer science » (Barchan, 4), un des traits désirés du système original était que « the grammar writer may include comments to be issued if certain expected mistakes are found in the input. » (6). À en juger par ces commentaires, on dirait que certains aspects de LINGER ont leurs racines dans la théorie que nous avons explorée. D'abord, le fait que LINGER explique les corrections qu'il fait semble indiquer que les réalisateurs du logiciel reconnaissent l'importance de la réparation dans la correction. En plus, l'usage des phrases écrites par des apprenants aborde les fautes intralingues et développementales, puisque le message qui explique les modifications faites à la phrase originale traite directement une faute particulière à cet apprenant. Enfin, l'idée de formuler le système de sorte qu'un professeur puisse rajouter des commentaires concernant les fautes attendues semble tenir compte du fait qu'il existe des fautes d'interférence ainsi que de quelques fautes intralingues et développementales qui sont communes à tous les apprenants de même langue maternelle. Alors, à partir des buts de LINGER, nous voyons

quelques applications des théories autour de la correction, de l'interlangue et de l'analyse des fautes. En vue de cela, l'utilité de LINGER comme outil pédagogique paraît être relativement prometteuse, mais afin de justifier cette affirmation, nous consacrerons la suite du chapitre à une étude de sa performance.

## ***Évaluation de LINGER***

### **La description du corpus**

Pour savoir ce que LINGER est capable d'analyser, nous nous servons d'un corpus d'exemples écrits par quinze étudiants de français au niveau du 110 à l'Université de Colombie-Britannique. Ils sont pris d'un babillard électronique par lequel les étudiants se sont écrit six fois par semaine pendant une période de quatre semaines entre la mi-septembre et fin octobre 2000. Parmi les deux mille quatre cents phrases au total, nous voulions étudier un échantillon de cent, alors nous avons numéroté toutes les phrases et fait la sélection avec une fonction dans Excel qui choisit des numéros automatiquement au hasard. Ensuite, nous avons identifié celles qui contiennent des fautes morpho-syntaxiques, sémantiques ou stylistiques (cf. Appendice D), afin d'en savoir la distribution. Cependant, il fallait limiter les tests aux fautes morpho-syntaxiques.

Nous avons trouvé que parmi les cent exemples, il y en avait 77 qui contenaient au moins une faute. Nous avons calculé un total de 159 fautes, 43 sémantiques (27%), 3 stylistiques (2%) et 113 morpho-syntaxiques (71%).

Nous avons identifié les seize catégories de fautes morpho-syntaxiques suivantes: l'accord (nom-adjectif), la conjugaison, les prépositions, le genre, les accents, les articles,

l'orthographe, les majuscules, les élisions, l'ordre des mots, la négation, le comparatif, les expressions de quantité, les noms adjectifs et l'absence du « que » dans une proposition subordonnée. La moitié de ces fautes sont celles de la conjugaison (20%), du choix de préposition (19%) et de l'accord (11%). Les fautes d'accents (9%), d'orthographe (9%), de majuscules (5%), de genre (5%) et d'élisions (5%) se font avec à peu près la même fréquence et ensemble elles constituent 33% du total. La suite des catégories des fautes comprend très peu de fautes et ensemble elles font les 13% qui manquent du total.

Nous avons dit dans le chapitre 2 que la grammaire et le dictionnaire de LINGER étaient limités aux constructions et aux catégories grammaticales suivantes:

- les accords (sujet-verbe, sujet-pronom réfléchi, sujet-adjectif, déterminant-nom-adjectif, pronoms-participes passés)
- les propositions relatives en « que » et en « qui »
- le placement et l'ordre des pronoms objet direct et indirect
- l'usage des prépositions avant un infinitif
- deux types de phrases hypothétiques en « si »
- les élisions
- l'orthographe
- les pronoms sujets, objet direct et indirect et réfléchis
- les déterminants
- une dizaine d'adjectifs
- une vingtaine de noms
- quatre adverbes
- une vingtaine de verbes et
- cinq prépositions

Face à notre corpus, nous nous sommes rendu compte que la grammaire LINGER serait capable de gérer seulement 21 des 77 exemples fautifs. Cependant, une analyse des corrections de LINGER de ces 21 exemples valait toujours la peine parce qu'elle montrerait les faiblesses du système, ce qui nous permettrait de généraliser quelques-unes des difficultés de construire un vérificateur de grammaire conçu pour des apprenants. Puisqu'il aurait fallu ajouter la plupart des mots de ces 21 exemples au dictionnaire simplement pour faire l'analyse, nous avons décidé de substituer au mot inconnu un mot de la même catégorie grammaticale déjà inclus dans le dictionnaire.

### **L'analyse des corrections de LINGER**

LINGER a réussi à corriger 9 des 21 phrases que nous avons testées. Les fautes corrigées incluent presque toutes celles d'accord entre le nom et le déterminant/l'adjectif, celles de genre et d'élosion, quelques-unes de conjugaison et de l'ordre des particules dans une négation. Examinons de près dix des cas problématiques qui nous semblent les plus pertinents.

Le genre et les élosions n'ont pas posés de problèmes à LINGER et les accords à l'intérieur du syntagme nominal ont réussi dans presque tous les cas. Cela n'est pas surprenant, puisque les règles sont faciles à formaliser. Comme nous l'avons vu au chapitre 2, LINGER établit le nombre du nom avant de traiter l'accord. Cela semble logique et marche pour la plupart des phrases, sauf celles comme notre exemple *\*La maison sont petites*. où, étant donné que le verbe ainsi que l'adjectif sont au pluriel, il est plus probable que le nom devrait être au pluriel aussi. Bien que ce genre de faute soit plutôt rare, il existe, alors un vérificateur de grammaire

devrait pouvoir la corriger d'une façon appropriée. Donc, à notre avis, afin de gérer les fautes de ce type, il vaut mieux avoir une fonction soit qui demande à l'apprenant le nombre du nom soit qui présume le nombre du nom sur la base des mots qui l'entourent.

Quoique LINGER arrive à corriger des fautes de conjugaison très simples au présent comme *\*Je voit*, *\*Il parler*, *\*Je me laver* et *\*La fille finir*, il a du mal à trouver une correction juste quand la faute est plus grave. Nous présentons ci-dessous des exemples où la conjugaison est plus problématique qu'une simple faute de terminaison ainsi que les corrections proposées par LINGER:

|                                 |     |  |
|---------------------------------|-----|--|
| <i>*Je mangée la pomme.</i>     | --> | <i>Je mange la pomme.</i>                                |
| <i>*Je alle à la maison.</i>    | --> | <i>Je &lt;alle&gt; à la maison.</i>                      |
| <i>*J'ai faire le tour.</i>     | --> | <i>J'ai &lt;adverb&gt; le tour.</i>                      |
| <i>*J'ai parler à la fille.</i> | --> | <i>J'ai &lt;relative_subject&gt; parlent à la fille.</i> |

En cherchant une explication pour ces corrections, nous nous sommes rendu compte que la structure du dictionnaire et les restrictions des règles de syntaxe forte constituaient les raisons fondamentales pour lesquelles LINGER n'arrive pas à gérer très bien la conjugaison. Dans le premier exemple, LINGER reconnaît *mangée* comme verbe principal, non pas comme participe passé. Après avoir reconnu que « Je » constitue le syntagme nominal, LINGER passe au mot suivant et, puisque ce n'est pas un verbe auxiliaire, il le traite comme verbe principal, satisfaisant à la règle de syntaxe forte: `verb_phr --> pos_or_neg_verb_cons(V), dir_obj_phr(DO)`. Ensuite, puisqu'il ne reconnaît pas le « ée » comme terminaison possible, il applique la valeur implicite: le présent.

Bien que LINGER n'ait pas corrigé *\*Je mangée la pomme*. comme il fallait, au moins il a reconnu le verbe et a fourni une réponse possible. Dans notre deuxième exemple, malgré le fait que *aller* apparaît dans le dictionnaire, LINGER ne reconnaît pas ce que l'apprenant a

écrit à cause de la représentation des verbes irréguliers dans le dictionnaire. Tandis que pour le verbe *manger* le dictionnaire inclut le radical « mang » à laquelle il ajoute des terminaisons, il n'y a pas de radical pour le verbe *aller* parce qu'au présent la conjugaison est très irrégulière. Donc, LINGER est obligé de décrire la conjugaison complète, au lieu d'avoir un radical et des terminaisons. C'est-à-dire que la conjugaison faite par l'apprenant dans ces cas doit être très proche de la forme correcte, sinon, comme dans le cas de notre exemple, LINGER ne la reconnaîtra pas. Il nous semble étonnant de présumer que les apprenants ne feront que des fautes mineures en conjuguant des verbes irréguliers.

Concernant la faute de mettre un infinitif à la place d'un participe passé, nous voyons à partir de nos deux derniers exemples que LINGER ne sait pas corriger ce genre de faute. Nous identifions les restrictions des règles de syntaxe forte comme explication. Supposant que «ai» soit le verbe principal et "le tour" le complément d'objet direct du syntagme verbal de la phrase *\*J'ai faire le tour.*, LINGER substitue "faire", le seul mot de la phrase qui ne satisfait pas à la règle de syntaxe forte qui correspond le mieux aux données, (c'est-à-dire syntagme verbal --> pos\_or\_neg\_verb\_cons(V), adverb(A), dir\_obj\_phr(DO), indir\_obj\_phr(IO)), par un adverbe. En plus, alors que la structure pos\_or\_neg\_verb\_cons(V) inclut un verbe auxiliaire plus un participe passé comme possibilité, puisque le verbe « faire » n'a pas de radical dans le dictionnaire, LINGER ne le reconnaît pas comme verbe et, donc, il rejette les constructions qui contiennent un participe passé dans la pré-analyse. Par contre, dans l'autre exemple, LINGER reconnaît *parler* comme verbe parce qu'il a un radical dans le dictionnaire, mais puisqu'il n'est pas un participe passé, LINGER l'interprète comme verbe principal de la proposition subordonnée et rajoute le pronom relatif « qui » à la phrase comme correction. L'addition du « qui » nous semblait très problématique, puisque

l'exemple ne contenait pas d'antécédent. Cela souligne un autre défaut dans la syntaxe forte de LINGER, ce que nous aborderons ici à l'aide des deux exemples suivants:

*\*Je vais a la maison.* --> *Je vais <relative\_subject> a la maison.*  
*\*Je suis marche.* --> *Je suis <relative\_subject> marche.*

Dans ces deux exemples, LINGER reconnaît deux verbes principaux, l'un de la proposition principale et l'autre de la proposition relative. Pourtant, il n'y a pas d'antécédent dans ces phrases. Le problème est que la syntaxe forte n'exige pas qu'il y ait un antécédent avant les pronoms relatifs. Elle précise simplement que la proposition relative doit être précédée d'une proposition principale. Cependant, étant donné qu'une proposition principale peut exister sans complément d'objet direct, LINGER accepte *Je vais* et interprète *a la maison* comme proposition relative qui manque le pronom relatif. Une telle omission nous semble très problématique parce qu'elle mène à des corrections fautives. Curieusement, dans des cas où le verbe de la proposition principale est transitif, comme *être* dans notre exemple *\*Je suis marche.*, LINGER inclut la remarque « Verb is transitive so does expect an object. » et, pourtant, il n'ajoute pas de complément dans l'analyse. Étant donné que la transitivité des verbes est un trait indiscutable dans la plupart des cas, à notre avis, il serait beaucoup plus utile d'insérer ou d'enlever les compléments selon le verbe avant d'établir la structure complète de la phrase parce que la présence d'un complément pourrait changer la suite de l'analyse. En plus, les apprenants anglophones ne se trompent pas en général au niveau de la transitivité parce que normalement elle est pareille en anglais et en français, par conséquent, ce genre de commentaire nous semble inutile.

Le dernier exemple que nous voulons traiter est le suivant:

*\*L'homme heureux --> l'homme heureux <verb>.*

Tandis que la correction que LINGER a fournie est tout à fait possible, elle n'est pas la seule réponse. Comme nous l'avons vu au chapitre 2, LINGER possède une fonction qui choisit un seul arbre syntaxique quand il y'en a plusieurs qui sont possibles. Cette approche nous semble trop serrée. S'il y a plus qu'une analyse possible, il devrait être la responsabilité de l'apprenant de choisir celle qui correspond le mieux à ses propres intentions.

Vu que LINGER est un prototype réalisé il y a quinze ans, il n'est pas étonnant que sa performance soit très limitée. Cependant, depuis cette époque, on s'attendrait à ce que la technologie dans le domaine du traitement des langues naturelles se soit beaucoup développée et améliorée. Donc, afin de découvrir ce que la technologie actuelle est capable de gérer, nous avons testé Antidote 2000, la version la plus récente de ce vérificateur de grammaire mentionné dans le chapitre 1, avec les 77 phrases fautives prises de notre échantillon.

### ***Antidote 2000: Évaluation de la technologie actuelle***

Des 77 phrases, Antidote n'a pu construire que 52 arbres syntaxiques (68%) parce que 25 des phrases contenaient des fautes syntaxiques qui les ont rendues inanalysables. Parmi les 52 phrases analysées, il y avait 66 fautes morpho-syntaxiques, 42 desquelles (64%) Antidote a réussi à corriger. Ces statistiques semblaient assez basses, surtout quand nous les avons comparées aux résultats des études que nous avons vus au chapitre 1. Par conséquent, nous avons regardé de près les phrases problématiques afin de voir s'il y avait un moyen d'expliquer pourquoi Antidote 2000 n'avait pas réussi à analyser plus des phrases.

Nous avons trouvé deux phrases où Antidote n'a pas détecté de fautes et cinq phrases qu'il a mal corrigées. Nous avons marqué toutes les fautes qu'Antidote aurait dû corriger en caractères gras et souligné toutes les fautes qu'il a réussi à corriger. Voici les phrases:

Phrases originales

Versions "corrigées" par Antidote

Il est tres difficile quand Je vais à l'université et je vais à travail dans un jour.

Il est très difficile quand je vais à l'université et je vais à travail dans un jour.

Mais je suis préjugé parce-que mes parents son roumain et ma mère a fait les gymnastiques beaucoup quand elle été étudiante.

Mais je suis préjugé parce-que mes parents son roumain et ma mère a fait les gymnastiques beaucoup quand elle été étudiante.

**Il étudier** beaucoup.

**Ils étudient** beaucoup.

Calgary est une petite université et le recrutement est plus bon que l'Université de colombie-britannique.

Calgary est une petite université et le recrutement est meilleur que l'université de Colombie britannique.

Cette anneé, je habite à Totem Park, mais ma famille habité dans Surrey.

Cette année, j'habite à Totem Park, mais ma famille habitée dans Surrey.

J'ai essayé au jouer le guitare mais c'est difficile.

J'ai essayé à la jouée la guitare mais c'est difficile.

Ils sont à la même école mais ils ne recontrent pas pendant le jour parce que ma mère commence à sept heures et elle est fini à midi ou 13 heures.

Ils sont à la même école mais ils ne remontrent pas pendant le jour parce que ma mère commence à sept heures et elle est finie à midi où à 13 heurte.

Les raisons pour lesquelles Antidote n'a pas pu corriger ces phrases ne nous sont pas évidentes. Nous supposons que tous les logiciels ont des petits problèmes techniques et qu'en vue de la variété de fautes potentielles qu'un vérificateur de grammaire est censé pouvoir gérer, aucun logiciel ne sera capable de les reconnaître et corriger toutes.

Concernant le reste des phrases, nous avons remarqué certaines tendances. D'abord, il nous semble qu'Antidote ne reconnaît pas les fautes de prépositions dans le contexte des mois et

des saisons. Il y avait quatre occurrences de ce genre de faute et Antidote n'en a corrigé aucune. Les phrases étaient *Je n'aime pas la pluie quand c'est froid dans l'automne ou l'hiver.*, *Dans* décembre quand il pleut j'irai snowboarding à Whistler et *J'habite en Californie dans l'été*. Deuxièmement, Antidote n'a pas fait de lien entre la préposition *pour* et le temps verbal de la phrase. Il n'a pas corrigé *J'ai joué au baseball pour 10 ans.* ni *J'ai habité dans la même maison pour toute ma vie.* ni *Je suis bien, mais je suis fatigué parce qu'hier après l'école j'ai travaillé pour huit heures.* Troisièmement, à partir des corrections fautives des phrases suivantes, nous avons déduit qu'Antidote établit l'analyse du syntagme nominal à partir des déterminants, ce qui a posé des problèmes dans la correction des trois phrases suivantes:

Phrases originales

Versions "corrigées" par Antidote

Ma mère est chrétien et **ma père** et juif!

Ma mère est chrétienne et **ma pairs** et juifs!

**Mon sports favori** sont le baseball, le football et l'hockey.

**Mon sport favori** est le baseball, le football et le hockey.

Je ne peux pas marcher dans un avion parce que **l'allée** sont très petites.

Je ne peux pas marcher dans un avion parce que **l'allée** est très petite.

Dernièrement, nous avons remarqué qu'Antidote n'ajoute pas de deuxième préposition où il faut, comme nous le voyons des exemples suivants:

*J'ai vécu en Tulameen et Vancouver.*

*Et, si ton amie veut aller au Brésil, elle peut aller à Rio de Janeiro ou le nord-ouest du pays, qu'il y a des plages merveilleuses.*

Nous résumons ici les fautes syntaxiques qui ont empêché l'analyse de 25 phrases. Elles incluent:

- le choix de prépositions dans le contexte des mois et des saisons
- l'usage de la préposition *pour* avec le temps verbal convenable

- le choix de déterminant et
- la répétition des prépositions avant un deuxième complément

Ces fautes tiennent compte de presque toutes les structures qu'Antidote a mal corrigées ou bien n'a pas corrigées du tout. Nous avons remarqué que c'étaient toutes des structures qu'un francophone formulerait sans doute correctement et, donc, le logiciel ne les prévoit pas. Par contre, nous savons qu'Antidote est capable de prévoir ou d'avertir des fautes pertinentes aux francophones. Par exemple, il signalera une alerte qui corrige et explique la confusion entre l'infinitif et le participe passé, ce qui est une faute typique chez les francophones. En plus, comme nous l'avons vu au chapitre 1, Antidote signale des problèmes lexicaux aussi. Parmi nos exemples, il a corrigé un calque (*prendre un cours d'anglais*), deux anglicismes (*quiz, originer*), un paronyme (*roumain/roman*), une confusion (*bibliographie/bibliographiques*) un faux ami (*date/sortie*), et des homophones (*son/sont, a/à*), ce qui est très utile pour l'apprenant. Les alertes sont aussi utiles pour des mots dont la majuscule change selon le contexte. Par exemple, il nous a averti que le mot *université* peut être soit avec majuscule, soit au avec minuscule dépendamment de son usage comme nom d'un établissement précis. Toutes ces alertes apparaissent chaque fois que le mot est utilisé et elles incluent des explications à partir desquelles l'utilisateur décide ce qui devrait être corrigé. À cause des limites sémantiques du logiciel, cette flexibilité nous semble essentielle.

Au niveau technique, nous concluons d'après cet aperçu d'Antidote que les vérificateurs de grammaire actuels sont très performants et que les problèmes de correction posés par nos exemples sont dus au fait qu'Antidote est conçu pour des francophones et non pas pour des apprenants de français comme langue seconde. En plus, Antidote fait la preuve qu'il est possible de prévoir des fautes morpho-syntaxiques et lexicales ainsi que des explications.

## Les conclusions et les recommandations

Ayant disséqué LINGER, nous avons une image assez claire des faiblesses et des forces du logiciel. Parmi les fautes morpho-syntaxiques que LINGER arrive à corriger sans trop de difficulté, nous identifions les élisions, le genre, les accords à l'intérieur du syntagme nominal, l'ordre des mots dans des constructions négatives et les conjugaisons simples. Les limites linguistiques de LINGER incluent une grammaire qui n'est pas suffisamment complexe pour gérer des propositions subordonnées, des propositions relatives à l'intérieur d'une proposition principale, des antécédents, des phrases hypothétiques en « si » au plus-que-parfait, ni l'euphonie des adjectifs possessifs et un dictionnaire qui est incomplet dans son traitement des prépositions. Cependant, à notre avis, on pourrait rectifier tous ces problèmes en étendant la quantité de règles de syntaxe forte, en ajoutant des vérifications et en raffinant les informations contenues dans le dictionnaire. Par contre, à la base des critiques de LINGER, nous trouvons que malgré les efforts qui ont été faits pour résoudre cette liste de limites qui ont leur origine dans la simplicité du logiciel, il reste des problèmes fondamentaux. Comme nous l'avons vu de l'analyse de LINGER, la structure du dictionnaire empêche la correction des verbes irréguliers dans des cas où la conjugaison de l'apprenant ne ressemble pas suffisamment à la forme correcte. Le degré de ressemblance requis est difficile à déchiffrer. Ayant fait quelques expériences, tout ce que nous pouvons dire est que pour le verbe *aller*, il faut qu'au moins les deux premières lettres du verbe correspondent à la forme correcte. Par exemple, *\*je va* sera corrigé *je vais*. Pour le verbe *faire*, au moins les trois premières lettres de la conjugaison au singulier sont nécessaires et quatre au pluriel. C'est-à-dire que *\*je fai*, ainsi que *\*nous fais*, seront corrigés. À notre avis, l'idée de construire le dictionnaire selon un système de radicaux et de terminaisons est le fond du problème. Tandis que cela ne pose pas trop de problèmes pour les noms et les

adjectifs, comme nous l'avons expliqué au chapitre 3, pour les verbes irréguliers il n'y a pas d'autre moyen que de fournir une conjugaison complète du verbe, ce qui veut dire que la conjugaison de l'apprenant doit être très proche à la forme correcte. Cependant, même dans notre petit échantillon, les fautes de conjugaison des verbes irréguliers tiennent compte de la moitié (11 sur 22) des fautes totales de conjugaison, donc, il nous semble clair qu'il faudrait changer la structure totale du dictionnaire de sorte qu'il puisse gérer des fautes de conjugaison des verbes irréguliers.

Malheureusement, à partir de ce que nous avons découvert à propos de la syntaxe forte, réviser le dictionnaire ne suffirait pas à guérir les problèmes de conjugaison. Comme nous l'avons vu dans l'exemple *\*J'ai faire le tour.* au chapitre 3, LINGER essaie de cadrer la phrase de l'apprenant dans la règle de syntaxe forte qui semble convenir le mieux. Le problème avec cette approche, au niveau de la conjugaison des temps composés au moins, est qu'elle exige que pour avoir une analyse juste, la conjugaison du verbe soit plus ou moins correcte; c'est-à-dire qu'il faut avoir quelque chose qui ressemble à un verbe auxiliaire et un participe passé bien construit. À la lumière de notre corpus, une telle présomption nous semble assez naïve.

Comme nous avons appris grâce à l'exemple *\*Je mangée la pomme.*, ce que LINGER a corrigé comme *Je mange la pomme.*, les règles de syntaxe forte ne sont pas sensibles à la possibilité que l'apprenant ait oublié le verbe auxiliaire, malgré le fait qu'il a inclus un participe passé dans sa phrase.

En ce qui concerne l'aspect pédagogique de LINGER, l'idée des réalisateurs de LINGER de produire un outil pédagogique qui puisse corriger des phrases écrites par des apprenants ainsi

que de présenter ce qu'il a modifié et pourquoi nous semble extrêmement utile. Pourtant, à notre avis, LINGER n'atteint ces buts ni dans ses corrections ni dans ses commentaires. Il est clair, d'après les difficultés qu'il a à corriger les fautes de conjugaison, que l'on n'a pas du tout tenu compte de la façon dont les apprenants se trompent en conjuguant. Quant à la liste de modifications effectuées à la phrase et les explications des fautes, telles que "voir" : *Verb is transitive so does expect an object* et verb "parle1" changed to "parle" : *[transitivity(i),follow\_prepos([]),can\_be\_reflex(y),person(3),plurality(s),tense(present)*, elles sont soit trop évidentes, soit trop cryptiques pour être utiles.

Malgré le fait qu'il n'est pas ni serait jamais un outil pédagogique dont des apprenants puissent se servir, d'après ce que nous avons observé de l'architecture et de la performance de LINGER, nous pouvons apprécier les complexités de construire un vérificateur de grammaire et proposer une série de recommandations pour un vérificateur de grammaire correctif idéal.

Commençant avec la façon dont l'analyse se fait, au lieu de cadrer la phrase écrite par l'apprenant selon la règle de syntaxe forte qui semble convenir le mieux aux données, il vaut mieux établir la structure de la phrase à partir d'une série de routines mises en priorité selon les fautes typiques des apprenants. Par exemple, à la base des nombreuses fautes de conjugaison dans notre échantillon, on dirait que la réparation des mauvaises conjugaisons devrait être une des premières choses à traiter dans l'analyse.

Nous ne sommes pas d'accord avec l'idée de concevoir un vérificateur de grammaire qui ne fournisse qu'une analyse, surtout quand il y en a plusieurs possibles, parce que le logiciel risque de choisir celle qui ne correspond pas à l'intention de l'apprenant. En plus, des cas

ambigus qui pourraient être résolus en consultant l'apprenant sont ignorés par un logiciel comme LINGER qui engendre une seule version corrigée selon ses propres critères. Il serait préférable d'avoir des alertes qui détectent des fautes possibles, tout en expliquant pourquoi il y a peut-être une faute et qui corrigent des fautes indiscutables, tout en rappelant à l'apprenant la règle de grammaire qu'il n'a pas respectée.

En évaluant Antidote 2000, nous avons découvert qu'il existe une technologie très performante et qui, selon un système d'alertes, ne fait pas de présomptions dans ses corrections et qui fournit des explications ciblées à l'utilisateur. En plus, nous avons remarqué qu'il est possible de prévoir des fautes, comme Antidote le fait pour certaines fautes typiques des francophones, notamment la confusion entre le participe passé et l'infinitif des verbes en *-er*. Sachant que les apprenants formulent une « interlangue » ou bien un « transitional dialect » qui est une série de règles qui décrivent la langue cible à partir de ce qu'ils savent déjà de leur langue maternelle et que la compétence linguistique s'améliore en perfectionnant la compréhension des règles de la langue chez l'apprenant, à notre avis, les explications des alertes serviraient comme un moyen de clarifier les règles mal comprises de la part de l'apprenant. Malheureusement, sans une programmation conçue pour prévoir les fautes des apprenants dans la création de l'arbre syntaxique, un vérificateur de grammaire comme Antidote ne traite pas les fautes d'interférence et dans plusieurs cas il n'arrive même pas à analyser la phrase, donc il n'est que moyennement utile à un apprenant du français. Quant aux explications des alertes, il faudrait les rendre appropriées à des apprenants, comme elles ne sont que des messages qui rappellent un utilisateur francophone d'une règle de grammaire oubliée. Il serait nécessaire d'inclure des explications écrites d'une manière que l'apprenant comprenne pourquoi il s'est trompé. En plus, il reste le problème de la correction fautive, le manque d'identification de fautes et l'identification fautive de la

source des fautes. Alors qu'en théorie un vérificateur de grammaire conçu pour des apprenants devrait réduire ces problèmes, on ne pourra jamais prévoir tout ce qu'il faut pour éviter ce genre de problème dans un logiciel qui est censé gérer n'importe quelle phrase qu'un apprenant veut vérifier. Alors, peut-être que la seule façon de minimiser la correction fautive, le manque d'identification de fautes et l'identification fautive de la source des fautes serait de limiter la grammaire et les thèmes que le logiciel doit traiter.

D'après tout ce que nous avons soulevé, nous tenons encore à notre hypothèse qu'un vérificateur de grammaire serait un outil pédagogique utile. Pourtant, avant d'en fournir un aux apprenants, il faudrait qu'il satisfasse aux conditions suivantes: il doit être conçu pour des apprenants de la même langue maternelle, ce qui veut dire que l'analyse des phrases procéderait à la base des types de fautes que font les apprenants; il doit inclure des explications grammaticales pour chaque correction qui aident l'apprenant à mieux comprendre la règle de grammaire en question; son analyse doit être juste avec un minimum de correction fautive, de manques d'identification de fautes et d'identification fautive de la source des fautes et il doit fournir une correction assistée, plutôt qu'une correction automatique dans le sens où il laisserait à l'apprenant la responsabilité d'accepter ou de rejeter une correction.

## Appendice A

### Le dictionnaire de LINGER

#### 1.0 /\* STANDARD ENDINGS \*/

```
info(gp_end,[[gender(m),plurality(s)],[gender(m),plurality(p)],[gender(f),plurality(s)],[gender(f),plurality(p)]],[[]]).
```

```
ending_type(gp_end,["","s","e","es"],[]).
```

#### 2.0 /\* NOUNS \*/

```
info(noun,[[plurality(s)],[plurality(p)]],[gender],[person(3)]).
```

```
word("","["bateau"],noun,["","x"],[m]).
word("","["banque"],noun,["","s"],[f]).
word("","["chat"],noun,["","s"],[m]).
word("","["chien"],noun,["","s"],[m]).
word("","["e1glise"],noun,["","s"],[f]).
word("","["femme"],noun,["","s"],[f]).
word("","["fille"],noun,["","s"],[f]).
word("","["garc5on"],noun,["","s"],[m]).
word("","["homme"],noun,["","s"],[m]).
word("","["livre"],noun,["","s"],[m]).
word("","["main"],noun,["","s"],[f]).
word("","["maison"],noun,["","s"],[f]).
word("","["me2re"],noun,["","s"],[f]).
word("","["mat"],noun,["","s"],[m]).
word("","["mensonge"],noun,["","s"],[f]).
word("","["pe2re"],noun,["","s"],[m]).
word("","["plume"],noun,["","s"],[f]).
word("","["pomme"],noun,["","s"],[f]).
word("","["tante"],noun,["","s"],[f]).
word("","["tour"],noun,["","s"],[m]).
word("","["vaisselle"],noun,["","s"],[f]).
```

#### 3.0 /\* PRONOUNS \*/

```
info(pronoun,[
[case(nom),apostrophe(n),order(1)],[case(nom),apostrophe(y),order(1)],
[case(acc),apostrophe(n),pp_agree(y),order(2)],[case(acc),apostrophe(y),pp_agree(y),order(2)
],
[case(dat),apostrophe(n),order(3)],[case(dat),apostrophe(y),order(3)],
[case(dat),order(4)],
[case(acc),order(5)]
],[person,plurality,gender],[[]]).
```

word("je",[""],pronoun,["je","j","me","m","me","m","",""],[1,s,any]).  
 word("tu",[""],pronoun,["tu","tu","te","t","te","t","",""],[2,s,any]).  
 word("il",[""],pronoun,["il","il","le","l","lui","lui","",""],[3,s,m]).  
 word("elle",[""],pronoun,["elle","elle","la","l","lui","lui","",""],[3,s,f]).  
 word("nous",[""],pronoun,["nous","","nous","","nous","",""],[1,p,any]).  
 word("vous",[""],pronoun,["vous","","vous","","vous","",""],[2,p,any]).  
 word("ils",[""],pronoun,["ils","","les","","leur","",""],[3,p,m]).  
 word("elles",[""],pronoun,["elles","","les","","leur","",""],[3,p,f]).  
 word("y",[""],pronoun,["","","","","y",""],[3,any,any]).  
 word("en",[""],pronoun,["","","","","en"],[3,any,any]).

#### 4.0 /\* REFLEXIVE PRONOUNS \*/

info(reflexive\_pronoun,[[apostrophe(n)],[apostrophe(y)]],[person,plurality],[]).

word("me",[""],reflexive\_pronoun,["me","m"],[1,s]).  
 word("te",[""],reflexive\_pronoun,["te","t"],[2,s]).  
 word("se",[""],reflexive\_pronoun,["se","s"],[3,s]).  
 word("nous",[""],reflexive\_pronoun,["nous",""],[1,p]).  
 word("vous",[""],reflexive\_pronoun,["vous",""],[2,p]).  
 word("se",[""],reflexive\_pronoun,["se","s"],[3,p]).

#### 5.0 /\* ADJECTIVES \*/

info(adjective,[[gender(m),plurality(s)],[gender(m),plurality(p)],[gender(f),plurality(s)],[gender(f),plurality(p)]],[position],[]).

word("","grand",adjective,gp\_end,[pre]).  
 word("","joli",adjective,gp\_end,[pre]).  
 word("","nouv",adjective,["eau","eaux","elle","elles"],[pre]).  
 word("","petit",adjective,gp\_end,[pre]).

word("","blanc",adjective,["","s","he","hes"],[post]).  
 word("doux",["dou"],adjective,["x","x","ce","ces"],[post]).  
 word("","heureu",adjective,["x","x","se","ses"],[post]).  
 word("","intelligent",adjective,gp\_end,[post]).  
 word("","laid",adjective,gp\_end,[post]).  
 word("","lent",adjective,gp\_end,[post]).  
 word("","rapide",adjective,["","s","s"],[post]).  
 word("","stupide",adjective,["","s","s"],[post]).  
 word("","vert",adjective,gp\_end,[post]).

#### 6.0 /\* ADVERBS \*/

info(adverb,[],[]).

word("","constamment",adverb,[],[]).  
 word("","lentement",adverb,[],[]).

word("",["rapidement"],adverb,[],[]).  
word("",["doucement"],adverb,[],[]).

#### 7.0 /\* DETERMINERS \*/

info(determiner,[[gender(m),plurality(s),apostrophe(n)],[gender(m),plurality(s),apostrophe(y)],[gender(m),plurality(p)],[gender(f),plurality(s),apostrophe(n)],[gender(f),plurality(s),apostrophe(y)],[gender(f),plurality(p)]]],[]).

word("ce",[""],determiner,["ce","cet","ces","cette","cette","ces"],[]).  
word("le",[""],determiner,["le","l","les","la","l","les"],[]).  
word("un",[""],determiner,["un","","des","une","","des"],[]).

word("mon",[""],determiner,["mon","","mes","ma","","mes"],[]).  
word("ton",[""],determiner,["ton","","tes","ta","","tes"],[]).  
word("son",[""],determiner,["son","","ses","sa","","ses"],[]).  
word("notre",[""],determiner,["notre","","nos","notre","","nos"],[]).  
word("votre",[""],determiner,["votre","","vos","votre","","vos"],[]).  
word("leur",[""],determiner,["leur","","leurs","leur","","leurs"],[]).

#### 8.0 /\* VERBS \*/

info(verb,[type],[transitivity, follow\_prepos, can\_be\_reflex],[]).

word("aimer",["aim"],verb,er\_end,[t,["a2",""],y]).  
word("aller",[""],verb,aller\_end,[i,[""],n]).  
word("arriver",["arriv"],verb,er\_end,[i,[],n]).  
word("chanter",["chant"],verb,er\_end,[i,[],n]).  
word("commencer",["commenc"],verb,c\_er\_end,[t,["a2","par"],n]).  
word("delcider",["delcid"],verb,er\_end,[t,["de"],n]).  
word("donner",["donn"],verb,er\_end,[t,[],y]).  
word("faire",[""],verb,faire\_end,[t,[],n]).  
word("laver",["lav"],verb,er\_end,[t,[],y]).  
word("manger",["mang"],verb,g\_er\_end,[t,[],n]).  
word("marcher",["march"],verb,er\_end,[i,[],n]).  
word("parler",["parl"],verb,er\_end,[i,[],y]).  
word("voir",[""],verb,voir\_end,[t,[""],y]).  
word("voyager",["voyag"],verb,g\_er\_end,[i,[],n]).

word("courir",["cour"],verb,ir\_end\_11,[i,[],n]).  
word("finir",["fin"],verb,ir\_end,[t,["de"],n]).  
word("sortir",["sor"],verb,ir\_end\_16,[i,[],n]).

word("entendre",["entend"],verb,re\_end,[t,[""],n]).

#### 9.0 /\* VERB ENDINGS \*/

info(er\_end,[[person(1),plurality(s)],[person(2),plurality(s)],[person(3),plurality(s)]],

[person(1),plurality(p)],[person(2),plurality(p)],[person(3),plurality(p)],[tense],[[]].

actual\_endings(er\_end\_pres,["e","es","e","ons","ez","ent"]).  
actual\_endings(er\_end\_imp,["ais","ais","ait","ions","iez","aient"]).  
actual\_endings(er\_end\_fut,["erai","eras","era","erons","erez","eront"]).  
actual\_endings(er\_end\_cond,["erais","erais","erait","erions","eriez","eraient"]).

ending\_type(er\_end,er\_end\_pres,[present]).  
ending\_type(er\_end,er\_end\_imp,[imperfect]).  
ending\_type(er\_end,er\_end\_fut,[future]).  
ending\_type(er\_end,er\_end\_cond,[conditional]).

info(g\_er\_end,[[person(1),plurality(s)],[person(2),plurality(s)],[person(3),plurality(s)],  
[person(1),plurality(p)],[person(2),plurality(p)],[person(3),plurality(p)]],[tense],[[]].

actual\_endings(g\_er\_end\_pres,["e","es","e","eons","ez","ent"]).  
actual\_endings(g\_er\_end\_imp,["eais","eais","eait","ions","iez","eaient"]).

ending\_type(g\_er\_end,g\_er\_end\_pres,[present]).  
ending\_type(g\_er\_end,g\_er\_end\_imp,[imperfect]).  
ending\_type(g\_er\_end,er\_end\_fut,[future]).  
ending\_type(g\_er\_end,er\_end\_cond,[conditional]).

info(c\_er\_end,[[person(1),plurality(s)],[person(2),plurality(s)],[person(3),plurality(s)],  
[person(1),plurality(p)],[person(2),plurality(p)],[person(3),plurality(p)]],[tense],[[]].

actual\_endings(c\_er\_end\_pres,["e","es","e","3ons","ez","ent"]).  
actual\_endings(c\_er\_end\_imp,["3ais","3ais","3ait","ions","iez","3aient"]).

ending\_type(c\_er\_end,c\_er\_end\_pres,[present]).  
ending\_type(c\_er\_end,c\_er\_end\_imp,[imperfect]).  
ending\_type(c\_er\_end,er\_end\_fut,[future]).  
ending\_type(c\_er\_end,er\_end\_cond,[conditional]).

info(ir\_end,[[person(1),plurality(s)],[person(2),plurality(s)],[person(3),plurality(s)],  
[person(1),plurality(p)],[person(2),plurality(p)],[person(3),plurality(p)]],[tense],[[]].

actual\_endings(ir\_end\_pres,["is","is","it","issons","issez","issent"]).  
actual\_endings(ir\_end\_imp,["issais","issais","issait","issions","issiez","issaient"]).  
actual\_endings(ir\_end\_fut,["irai","iras","ira","irons","irez","iront"]).  
actual\_endings(ir\_end\_cond,["irais","irais","irait","irions","iriez","iraient"]).

ending\_type(ir\_end,ir\_end\_pres,[present]).  
ending\_type(ir\_end,ir\_end\_imp,[imperfect]).  
ending\_type(ir\_end,ir\_end\_fut,[future]).  
ending\_type(ir\_end,ir\_end\_cond,[conditional]).

info(ir\_end\_11,[[person(1),plurality(s)],[person(2),plurality(s)],[person(3),plurality(s)],  
[person(1),plurality(p)],[person(2),plurality(p)],[person(3),plurality(p)]],[tense],[[]].

actual\_endings(ir\_end\_11\_pres,["s","s","t","ons","ez","ent"]).  
actual\_endings(ir\_end\_11\_fut,["rrai","rras","rra","rrons","rrez","rront"]).  
actual\_endings(ir\_end\_11\_cond,["rrais","rrais","rrait","rrions","rriez","rraient"]).

ending\_type(ir\_end\_11,ir\_end\_11\_pres,[present]).  
ending\_type(ir\_end\_11,er\_end\_imp,[imperfect]).  
ending\_type(ir\_end\_11,ir\_end\_11\_fut,[future]).  
ending\_type(ir\_end\_11,ir\_end\_11\_cond,[conditional]).

info(ir\_end\_16,[[person(1),plurality(s)],[person(2),plurality(s)],[person(3),plurality(s)],  
[person(1),plurality(p)],[person(2),plurality(p)],[person(3),plurality(p)]],[tense],[ ]).

actual\_endings(ir\_end\_16\_pres,["s","s","t","tons","tez","tent"]).  
actual\_endings(ir\_end\_16\_imp,["tais","tais","tait","tions","tiez","taient"]).  
actual\_endings(ir\_end\_16\_fut,["tirai","tiras","tira","tiron","tirez","tiron"]).  
actual\_endings(ir\_end\_16\_cond,["tirais","tirais","tirait","tirions","tiriez","tiraient"]).

ending\_type(ir\_end\_16,ir\_end\_16\_pres,[present]).  
ending\_type(ir\_end\_16,ir\_end\_16\_imp,[imperfect]).  
ending\_type(ir\_end\_16,ir\_end\_16\_fut,[future]).  
ending\_type(ir\_end\_16,ir\_end\_16\_cond,[conditional]).

info(re\_end,[[person(1),plurality(s)],[person(2),plurality(s)],[person(3),plurality(s)],  
[person(1),plurality(p)],[person(2),plurality(p)],[person(3),plurality(p)]],[tense],[ ]).

actual\_endings(re\_end\_pres,["s","s","","ons","ez","ent"]).  
actual\_endings(re\_end\_fut,["rai","ras","ra","rons","rez","ront"]).  
actual\_endings(re\_end\_cond,["rais","rais","rait","rions","riez","raient"]).

ending\_type(re\_end,re\_end\_pres,[present]).  
ending\_type(re\_end,er\_end\_imp,[imperfect]).  
ending\_type(re\_end,re\_end\_fut,[future]).  
ending\_type(re\_end,re\_end\_cond,[conditional]).

info(aller\_end,[[person(1),plurality(s)],[person(2),plurality(s)],[person(3),plurality(s)],  
[person(1),plurality(p)],[person(2),plurality(p)],[person(3),plurality(p)]],[tense],[ ]).

ending\_type(aller\_end,["vais","vas","va","allons","allez","vont"],[present]).  
ending\_type(aller\_end,["allais","allais","allait","allions","alliez","allaient"],[imperfect]).  
ending\_type(aller\_end,["irai","iras","ira","irons","irez","iront"],[future]).  
ending\_type(aller\_end,["irais","irais","irait","irions","iriez","iraient"],[conditional]).

info(faire\_end,[[person(1),plurality(s)],[person(2),plurality(s)],[person(3),plurality(s)],  
[person(1),plurality(p)],[person(2),plurality(p)],[person(3),plurality(p)]],[tense],[ ]).

ending\_type(faire\_end,["fais","fais","fait","faisons","faites","font"],[present]).  
ending\_type(faire\_end,["faisais","faisais","faisait","faisions","faisiez","faisaient"],[imperfect]).

ending\_type(faire\_end,["ferai","feras","fera","ferons","ferez","feront"],[future]).  
ending\_type(faire\_end,["ferais","ferais","ferait","ferions","feriez","feraient"],[conditional]).

info(voir\_end,[[person(1),plurality(s)],[person(2),plurality(s)],[person(3),plurality(s)],  
[person(1),plurality(p)],[person(2),plurality(p)],[person(3),plurality(p)],[tense],[ ]]).

ending\_type(voir\_end,["vois","vois","voit","voyons","voyez","voient"],[present]).  
ending\_type(voir\_end,["voyais","voyais","voyait","voyions","voyiez","voyaient"],[imperfect  
]).  
ending\_type(voir\_end,["verrai","verras","verra","verrons","verrez","verront"],[future]).  
ending\_type(voir\_end,["verrais","verrais","verrait","verrions","verriez","verraient"],[condition  
al]).

#### 10.0 /\* AUXILIARY VERBS \*/

info(aux\_verb,[  
[person(1),plurality(s)],[person(2),plurality(s)],[person(3),plurality(s)],[person(1),plurality(p)  
],[person(2),plurality(p)],[person(3),plurality(p)]  
],[tense,transitivity],[ ]]).

word("avoir",[""],aux\_verb,["ai","as","a","avons","avez","ont"],[present,t]).  
word("avoir",[""],aux\_verb,["avais","avais","avait","avons","aviez","avaient"],[imperfect,t]).  
word("avoir",[""],aux\_verb,["aurai","auras","aura","aurons","aurez","auront"],[future,t]).  
word("avoir",[""],aux\_verb,["aurais","aurais","aurait","aurions","auriez","auraient"],[condition  
al,t]).

word("e4tre",[""],aux\_verb,["suis","es","est","sommes","e4tes","sont"],[present,t]).  
word("e4tre",[""],aux\_verb,["etais","etais","etait","etions","etiez","etaient"],[imperfec  
t,t]).  
word("e4tre",[""],aux\_verb,["serai","seras","sera","serons","serez","seront"],[future,t]).  
word("e4tre",[""],aux\_verb,["serais","serais","serait","serions","seriez","seraient"],[condition  
al,t]).

#### 11.0 /\* PAST PARTICIPLES \*/

info(past\_participle,[[gender(m),plurality(s)],[gender(m),plurality(p)],[gender(f),plurality(s)],  
[gender(f),plurality(p)],[conjug\_with,transitivity,follow\_prepos,can\_be\_reflex],[ ]]).

word("",["aimel"],past\_participle,gp\_end,[avoir,t,["a2",""],y]).  
word("",["allel"],past\_participle,gp\_end,[e4tre,i,[""],n]).  
word("",["arrive1"],past\_participle,gp\_end,[e4tre,i,[],n]).  
word("",["chante1"],past\_participle,gp\_end,[avoir,i,[],n]).  
word("",["commence1"],past\_participle,gp\_end,[avoir,t,["a2","par"],n]).  
word("",["couru"],past\_participle,gp\_end,[avoir,i,[],n]).  
word("",["fait"],past\_participle,gp\_end,[avoir,t,[],n]).  
word("",["fini"],past\_participle,gp\_end,[avoir,t,["de"],n]).  
word("",["delcide1"],past\_participle,gp\_end,[avoir,t,["de"],n]).  
word("",["donnel"],past\_participle,gp\_end,[avoir,t,[],y]).  
word("",["entendu"],past\_participle,gp\_end,[avoir,t,[""],n]).  
word("",["lave1"],past\_participle,gp\_end,[avoir,t,[],y]).

```

word("",["mange1"],past_participle,gp_end,[avoir,t,[],n]).
word("",["marche1"],past_participle,gp_end,[avoir,i,[],n]).
word("",["parle1"],past_participle,gp_end,[avoir,i,[],y]).
/*word("",["parti"],past_participle,gp_end,[e4tre,i,[],n]). */
word("",["sorti"],past_participle,gp_end,[e4tre,i,[],n]).
word("",["voyage1"],past_participle,gp_end,[avoir,i,[],n]).
word("",["vu"],past_participle,gp_end,[avoir,t,[],n]).

```

## 12.0 /\* INFINITIVES \*/

```

info(infinitive,[[gender(m),plurality(s)],[gender(m),plurality(p)],[gender(f),plurality(s)],[gender(f),plurality(p)]],[transitivity,follow_prepos],[]).

```

```

word("",["aimer"],infinitive,[],[t,["a2", ""],y]).
word("",["aller"],infinitive,[],[i,[""],n]).
word("",["arriver"],infinitive,[],[i,[],n]).
word("",["chanter"],infinitive,[],[i,[],n]).
word("",["commencer"],infinitive,[],[t,["a2", "par"],n]).
word("",["courir"],infinitive,[],[i,[],n]).
word("",["decider"],infinitive,[],[t,["de"],n]).
word("",["donner"],infinitive,[],[t,[],y]).
word("",["entendre"],infinitive,[],[t,[""],n]).
word("",["faire"],infinitive,[],[t,[],n]).
word("",["finir"],infinitive,[],[t,["de"],n]).
word("",["laver"],infinitive,[],[t,[],y]).
word("",["manger"],infinitive,[],[t,[],n]).
word("",["marcher"],infinitive,[],[i,[],n]).
word("",["parler"],infinitive,[],[i,[],y]).
/*word("",["partir"],infinitive,[],[i,[],n]). */
word("",["sortir"],infinitive,[],[i,[],n]).
word("",["voyager"],infinitive,[],[i,[],n]).
word("",["voir"],infinitive,[],[t,[""],y]).

```

## 13.0 /\* NEGATIVE CONSTRUCTS \*/

```

info(neg_begin,[[apostrophe(n)],[apostrophe(y)]],[[],[]]).

```

```

word("ne",[""],neg_begin,["ne", "n"],[]).

```

```

info(neg_end,[[[]],[[],[]]).

```

```

word("pas",[""],neg_end,["pas"],[]).
word("jamais",[""],neg_end,["jamais"],[]).
word("rien",[""],neg_end,["rien"],[]).

```

## 14.0 /\* PREPOSITIONS \*/

```

info(preposition,[[[]],[[],[]]).

```

word("a2",[""],preposition,["a2"],[]).  
word("dans",[""],preposition,["dans"],[]).  
word("de",[""],preposition,["de"],[]).  
word("sur",[""],preposition,["sur"],[]).  
word("vers",[""],preposition,["vers"],[]).

#### 15.0 /\* RELATIVES \*/

info(relative\_subject,[],[],[]).

word("qui",[""],relative\_subject,["qui"],[]).

info(relative\_object\_que,[[apostrophe(n)],[apostrophe(y)]],[],[]).

word("que",[""],relative\_object\_que,["que","qu"],[]).

info(relative\_object\_si,[[apostrophe(n)],[apostrophe(y)]],[],[]).

word("si",[""],relative\_object\_si,["si","s"],[]).

#### 16.0 /\* GUESS CLASS \*/

open1(noun).

open1(adjective).

open1(adverb).

open1(verb).

open1(past\_participle).

open1(infinitive).

insert\_type(noun).

insert\_type(adjective).

insert\_type(verb).

insert\_type(adverb).

insert\_type(past\_participle).

insert\_type(infinitive).

insert\_type(determiner).

insert\_type(pronoun).

insert\_type(reflexive\_pronoun).

insert\_type(relative\_subject).

insert\_type(relative\_object\_que).

insert\_type(relative\_object\_si).

insert\_type(aux\_verb).

insert\_type(preposition).

insert\_type(neg\_begin).

insert\_type(neg\_end).

vowels("aeiouhy").

## Appendice B

### *La syntaxe forte de Linger*

1.0

1.1 sentence(formed(sentence,[MC])) --> main\_clause(MC).

1.2 sentence(formed(sentence,[MC,SC])) --> main\_clause(MC) , sub\_clause(SC).

2.0

2.1 main\_clause(formed(main\_clause,[N,V])) --> subj\_phr(N) , verb\_phr(V).

3.0

3.1 sub\_clause(formed(sub\_clause,[RS,V])) --> relative\_subject(RS) , verb\_phr(V).

3.2 sub\_clause(formed(sub\_clause,[RO,S,V])) --> relative\_object(RO) , subj\_phr(S) , verb\_phr(V).

4.0

4.1 subj\_phr(formed(subj\_phr,[S])) --> noun\_pron\_phr(S).

5.0

5.1 verb\_phr(formed(verb\_phr,[V,P,I])) --> pos\_or\_neg\_verb\_cons(V) , verb\_preposition(P) , infinitive\_phr(I).

5.2 verb\_phr(formed(verb\_phr,[V,A,DO,IO])) --> pos\_or\_neg\_verb\_cons(V) , adverb(A) , dir\_obj\_phr(DO) , indir\_obj\_phr(IO).

5.3 verb\_phr(formed(verb\_phr,[V,DO,IO])) --> pos\_or\_neg\_verb\_cons(V) , dir\_obj\_phr(DO) , indir\_obj\_phr(IO).

5.4 verb\_phr(formed(verb\_phr,[AV,C])) --> pos\_or\_neg\_aux\_verb\_cons(AV) , compliment\_phr(C).

6.0

6.1 noun\_pron\_phr(formed(noun\_pron\_phr,[N])) --> noun\_phr(N).

6.2 noun\_pron\_phr(formed(noun\_pron\_phr,[P])) --> pronoun(P).

7.0

7.1 pos\_or\_neg\_verb\_cons(formed(pos\_or\_neg\_verb\_cons,[V])) --> pos\_verb\_cons(V).

7.2 pos\_or\_neg\_verb\_cons(formed(pos\_or\_neg\_verb\_cons,[V])) --> neg\_verb\_cons(V).

8.0

8.1 verb\_preposition(formed(verb\_preposition,[P])) --> preposition(P).

8.2 verb\_preposition(formed(verb\_preposition,[ ])) --> [ ].

9.0

9.1 infinitive\_phr(formed(infinitive\_phr,[RP,I])) --> rp\_pron\_list(RP) , infinitive(I).

9.2 infinitive\_phr(formed(infinitive\_phr,[NB,NE,RP,I])) --> neg\_begin(NB) , neg\_end(NE) , rp\_pron\_list(RP) , infinitive(I).

10.0

10.1 dir\_obj\_phr(formed(dir\_obj\_phr,[DO])) --> noun\_phr(DO).  
10.2 dir\_obj\_phr(formed(dir\_obj\_phr,[])) --> [].

11.0  
11.1 indir\_obj\_phr(formed(indir\_obj\_phr,[P,IO])) --> preposition(P) , noun\_phr(IO).  
11.2 indir\_obj\_phr(formed(indir\_obj\_phr,[])) --> [].

12.0  
12.1 pos\_or\_neg\_aux\_verb\_cons(formed(pos\_or\_neg\_aux\_verb\_cons, [formed(governing\_verb,[V])])) --> aux\_verb(V).  
12.2 pos\_or\_neg\_aux\_verb\_cons(formed(pos\_or\_neg\_aux\_verb\_cons, [NB,formed(governing\_verb,[V]),NE])) --> neg\_begin(NB) , aux\_verb(V) , neg\_end(NE).

13.0  
13.1 compliment\_phr(formed(compliment\_phr,[C])) --> adjective(C).

14.0  
14.1 noun\_phr(formed(noun\_phr,[D,A1,N,A2])) --> determiner(D) , adj\_list(A1) , noun(N) , adj\_list(A2).

15.0  
15.1 pos\_verb\_cons(formed(pos\_verb\_cons,[PL,A,formed(governing\_verb,[PP])])) --> rp\_pron\_list(PL) , aux\_verb(A) , past\_participle(PP).  
15.2 pos\_verb\_cons(formed(pos\_verb\_cons,[PL,formed(governing\_verb,[V])])) --> rp\_pron\_list(PL) , verb\_or\_aux\_verb(V).

16.0  
16.1 neg\_verb\_cons(formed(neg\_verb\_cons, [NB,PR,A,NE,formed(governing\_verb,[PP])])) --> neg\_begin(NB) , rp\_pron\_list(PR) , aux\_verb(A) , neg\_end(NE) , past\_participle(PP).  
16.2 neg\_verb\_cons(formed(neg\_verb\_cons,[NB,PR,formed(governing\_verb,[V]),NE])) --> neg\_begin(NB) , rp\_pron\_list(PR) , verb\_or\_aux\_verb(V) , neg\_end(NE).

17.0  
17.1 adj\_list(formed(adj\_list,[A1,A2])) --> adjective(A1) , adj\_list(A2).  
17.2 adj\_list(formed(adj\_list,[])) --> [].

18.0  
18.1 rp\_pron\_list(formed(rp\_pron\_list,[RP,PL])) --> reflexive\_pronoun(RP) , pron\_list(PL).  
18.2 rp\_pron\_list(formed(rp\_pron\_list,[PL])) --> pron\_list(PL).

19.0  
19.1 pron\_list(formed(pron\_list,[PL])) --> pron\_list2(PL).  
19.2 pron\_list2(formed(pron\_list2,[P,PL])) --> pronoun(P) , pron\_list2(PL).  
19.3 pron\_list2(formed(pron\_list2,[])) --> [].

20.0  
20.1 verb\_or\_aux\_verb(formed(verb\_or\_aux\_verb,[V])) --> verb(V).  
20.2 verb\_or\_aux\_verb(formed(verb\_or\_aux\_verb,[AV])) --> aux\_verb(AV).

21.0  
21.1 relative\_object(formed(relative\_object,[R])) --> relative\_object\_que(R).  
21.2 relative\_object(formed(relative\_object,[R])) --> relative\_object\_si(R).

22.0  
22.1 adjective(formed(adjective,[A,Pos])) --> it\_is(adjective,A,Pos).

23.0  
23.1 adverb(formed(adverb,[A,Pos])) --> it\_is(adverb,A,Pos).

24.0  
24.1 aux\_verb(formed(aux\_verb,[AV,Pos])) --> it\_is(aux\_verb,AV,Pos).

25.0  
25.1 determiner(formed(determiner,[D,Pos])) --> it\_is(determiner,D,Pos).

26.0  
26.1 infinitive(formed(infinitive,[I,Pos])) --> it\_is(infinitive,I,Pos).

27.0  
27.1 neg\_begin(formed(neg\_begin,[NB,Pos])) --> it\_is(neg\_begin,NB,Pos).

28.0  
28.1 neg\_end(formed(neg\_end,[NE,Pos])) --> it\_is(neg\_end,NE,Pos).

29.0  
29.1 noun(formed(noun,[N,Pos])) --> it\_is(noun,N,Pos).

30.0  
30.1 past\_participle(formed(past\_participle,[PP,Pos])) --> it\_is(past\_participle,PP,Pos).

31.0  
31.1 preposition(formed(preposition,[P,Pos])) --> it\_is(preposition,P,Pos).

32.0  
32.1 pronoun(formed(pronoun,[P,Pos])) --> it\_is(pronoun,P,Pos).

33.0  
3.1 reflexive\_pronoun(formed(reflexive\_pronoun,[RP,Pos])) -->  
it\_is(reflexive\_pronoun,RP,Pos).

34.0  
34.1 relative\_object\_que(formed(relative\_object\_que,[R,Pos])) -->  
it\_is(relative\_object\_que,R,Pos).

35.0  
35.1 relative\_object\_si(formed(relative\_object\_si,[R,Pos])) -->  
it\_is(relative\_object\_si,R,Pos).

36.0

36.1 `relative_subject(formed(relative_subject,[R,Pos])) --> it_is(relative_subject,R,Pos).`

37.0

37.1 `verb(formed(verb,[NV,Pos])) --> it_is(verb,NV,Pos).`

## Appendice C

### La syntaxe faible organisée par catégorie grammaticale

|     |   |
|-----|---|
| 1.0 | Pronoms   |
| 1.1 | check([pronoun], subj_phr, [], should_be([case(nom)])).   |
| 1.2 | check([pronoun], verb_phr, [], should_not_be([case(nom)])).   |
| 1.3 | check([pronoun], verb_phr, [empty(dir_obj_phr),empty(indir_obj_phr)],<br>select_on_first([case(acc)])).   |
| 1.4 | check([pronoun], pron_list, [], select_on_first([plurality(s),gender(m)])).   |
| 1.5 | check([pronoun], verb_phr, [exists(dir_obj_phr)], should_not_be([case(acc)])).  |
| 1.6 | check([pronoun], verb_phr, [exists(indir_obj_phr)], should_not_be([case(dat)])).  |
| 1.7 | reorder(end,pron_list,[priority(2,pronoun,[order(2)]),priority(3,pronoun,[order(3)]),<br>priority(4,pronoun,[order(4)]),priority(5,pronoun,[order(5)])]).   |
| 1.8 | reorder(beginning,noun_phr,[priority(1,determiner,[]),priority(2,adjective,[position(pre<br>)]),priority(3,noun,[]),priority(4,adjective,[position(post)])]).   |
| 2.0 | Verbe principal, verbe auxiliaire et pronom réfléchi  |
| 2.1 | check([verb,aux_verb,reflexive_pronoun], main_clause, [],<br>concord([person,plurality],subj_phr(noun),[])).  |
| 2.2 | check([verb,aux_verb,reflexive_pronoun], sub_clause, [],<br>concord([person,plurality],subj_phr(noun),[])).   |
| 2.3 | check([verb,aux_verb,reflexive_pronoun], main_clause, [],<br>concord([person,plurality],subj_phr(pronoun),[])).   |
| 2.4 | check([verb,aux_verb,reflexive_pronoun], sub_clause, [],<br>concord([person,plurality],subj_phr(pronoun),[])).  |
| 2.5 | check([sub_clause(verb),sub_clause(aux_verb),sub_clause(reflexive_pronoun)],<br>sentence, [exists(relative_subject)],<br>concord([person,plurality],main_clause(dir_obj_phr(noun)),[])).                      |
| 2.6 | check([sub_clause(verb),sub_clause(aux_verb),sub_clause(reflexive_pronoun)],<br>sentence, [exists(relative_subject)],<br>concord([person,plurality],main_clause(pron_list(pronoun)),[attribute(case(acc))])). |
| 3.0 | Verbe principal, verbe auxiliaire   |
| 3.1 | check([verb,aux_verb], sentence, [], select_on_first([tense(present)])).  |
| 4.0 | Verbe auxiliaire  |

|     |   |
|-----|---|
| 4.1 | check([aux_verb], pos_or_neg_verb_cons, [attribute(conjug_with(avoir),past_participle),empty(reflexive_pronoun)], change_word("avoir",aux_verb,ignore([transitivity]))).  |
| 4.2 | check([aux_verb], pos_or_neg_verb_cons, [attribute(conjug_with(e4tre),past_participle)], change_word("e4tre",aux_verb,ignore([transitivity]))).   |
| 4.3 | check([aux_verb], pos_or_neg_verb_cons, [exists(reflexive_pronoun)], change_word("e4tre",aux_verb,ignore([transitivity]))).   |
| 4.4 | check([aux_verb], main_clause, [exists(compliment_phr)], change_word("e4tre",aux_verb,ignore([transitivity]))).   |
| 4.5 | check([aux_verb], sub_clause, [exists(compliment_phr)], change_word("e4tre",aux_verb,ignore([transitivity]))).  |
|     |   |
| 5.0 | Complément d'adjectif   |
|     |   |
| 5.1 | check([compliment_phr(adjective)], main_clause, [], concord([gender,plurality],subj_phr(noun),[])).   |
| 5.2 | check([compliment_phr(adjective)], main_clause, [], concord([gender,plurality],subj_phr(pronoun),[])).  |
| 5.3 | check([sub_clause(compliment_phr(adjective))], sentence,[exists(relative_subject)], concord([gender,plurality],main_clause(dir_obj_phr(noun),[])).  |
| 5.4 | check([sub_clause(compliment_phr(adjective))], sentence, exists(relative_subject), concord([gender,plurality],main_clause(pron_list(pronoun)),[attribute(case(acc))])).   |
|     |   |
| 6.0 | Le participe passé  |
|     |   |
| 6.1 | check([past_participle], main_clause, [attribute(conjug_with(e4tre),past_participle)], concord([gender,plurality],subj_phr(noun),[])).  |
| 6.2 | check([past_participle], sub_clause, [attribute(conjug_with(e4tre),past_participle)], concord([gender,plurality],subj_phr(noun),[])).   |
| 6.3 | check([sub_clause(past_participle)], sentence, [exists(relative_subject),attribute(conjug_with(e4tre),sub_clause(past_participle))], concord([gender,plurality],main_clause(dir_obj_phr(noun),[])).                       |
| 6.4 | check([sub_clause(past_participle)], sentence, [exists(relative_subject),attribute(conjug_with(e4tre),sub_clause(past_participle))], concord([gender,plurality],main_clause(pron_list(pronoun)),[attribute(case(acc))])). |
| 6.5 | check([past_participle], main_clause, [attribute(conjug_with(e4tre),past_participle)], concord([gender,plurality],subj_phr(pronoun),[])).   |
| 6.6 | check([past_participle], sub_clause, [attribute(conjug_with(e4tre),past_participle)], concord([gender,plurality],subj_phr(pronoun),[])).  |
| 6.7 | check([past_participle], main_clause, [exists(reflexive_pronoun),empty(dir_obj_phr),not_attribute(case(acc),pron_list(pronoun))], concord([gender,plurality],subj_phr(noun),[])).   |
| 6.8 | check([past_participle], sub_clause,  |

|          |   |
|----------|---|
|          | [exists(reflexive_pronoun),empty(dir_obj_phr),not_attribute(case(acc),pron_list(pronoun))],<br>concord([gender,plurality],subj_phr(noun),[])).  |
| 6.9      | check([past_participle], main_clause,<br>[exists(reflexive_pronoun),empty(dir_obj_phr),not_attribute(case(acc),pron_list(pronoun))],<br>concord([gender,plurality],subj_phr(pronoun),[])).  |
| 6.1<br>0 | check([sub_clause(past_participle)], sentence,<br>[exists(relative_subject),exists(sub_clause(reflexive_pronoun)),<br>empty(sub_clause(dir_obj_phr)),not_attribute(case(acc),<br>sub_clause(pron_list(pronoun)))],<br>concord([gender,plurality],main_clause(dir_obj_phr(noun)),[])).                                   |
| 6.1<br>1 | check([past_participle], sub_clause,<br>[exists(reflexive_pronoun),empty(dir_obj_phr),not_attribute(case(acc),<br>pron_list(pronoun))],<br>concord([gender,plurality],subj_phr(pronoun),[])).   |
| 6.1<br>2 | check([sub_clause(past_participle)], sentence,<br>[exists(relative_subject),exists(sub_clause(reflexive_pronoun)),<br>empty(sub_clause(dir_obj_phr)),not_attribute(case(acc),<br>sub_clause(pron_list(pronoun)))],<br>concord([gender,plurality],main_clause(pron_list(pronoun)),[attribute(case(acc))])).              |
| 6.1<br>3 | check([past_participle], main_clause,<br>[attribute(conjug_with(avoir),past_participle),empty(reflexive_pronoun),<br>attribute(pp_agree(y),pos_or_neg_verb_cons(pron_list(pronoun))),<br>empty(infinitive_phr)],<br>concord([gender,plurality],pos_or_neg_verb_cons(pron_list(pronoun)),<br>[attribute(pp_agree(y))])). |
| 6.1<br>4 | check([past_participle], sub_clause,<br>[attribute(conjug_with(avoir),past_participle),empty(reflexive_pronoun),<br>attribute(pp_agree(y),pos_or_neg_verb_cons(pron_list(pronoun))),<br>empty(infinitive_phr)],<br>concord([gender,plurality],pos_or_neg_verb_cons(pron_list(pronoun)),<br>[attribute(pp_agree(y))])).  |
| 6.1<br>5 | check([sub_clause(past_participle)], sentence,<br>[exists(relative_object_que),attribute(conjug_with(avoir),<br>sub_clause(past_participle)),empty(sub_clause(reflexive_pronoun)),<br>exists(main_clause(dir_obj_phr))],<br>concord([gender,plurality],main_clause(dir_obj_phr(noun)),[])).                             |
| 6.1<br>6 | check([past_participle], main_clause,<br>[attribute(conjug_with(avoir),past_participle),exists(reflexive_pronoun),<br>exists(dir_obj_phr)], should_be([gender(m),plurality(s)])).   |
| 6.1<br>7 | check([past_participle], sub_clause,<br>[attribute(conjug_with(avoir),past_participle),exists(reflexive_pronoun),<br>exists(dir_obj_phr)], should_be([gender(m),plurality(s)])).  |
| 6.1<br>8 | check([past_participle], main_clause,<br>[attribute(conjug_with(avoir),past_participle),empty(reflexive_pronoun),<br>not_attribute(pp_agree(y),pos_or_neg_verb_cons(pron_list(pronoun)))],<br>should_be([gender(m),plurality(s)])).   |

|          |  |
|----------|--|
| 6.1<br>9 | check([past_participle], sub_clause,<br>[empty(relative_object_que),attribute(conjug_with(avoir),past_participle),<br>empty(reflexive_pronoun),not_attribute(pp_agree(y),<br>pos_or_neg_verb_cons(pron_list(pronoun)))]), should_be([gender(m),plurality(s)]).   |
| 6.2<br>0 | check([past_participle], main_clause,<br>[attribute(conjug_with(avoir),past_participle),empty(reflexive_pronoun),<br>exists(infinitive_phr)], should_be([gender(m),plurality(s)]).   |
| 6.2<br>1 | check([past_participle], sub_clause,<br>[attribute(conjug_with(avoir),past_participle),empty(reflexive_pronoun),<br>exists(infinitive_phr)], should_be([gender(m),plurality(s)]).  |
|          |  |
|          |  |
| 7.0      | Verbe principal  |
|          |  |
| 7.1      | check([governing_verb], verb_phr,<br>[attribute(transitivity(i),governing_verb),exists(dir_obj_phr)],<br>comment([verb_phr,[],'Verb is intransitive so does not expect an object'])).  |
| 7.2      | check([governing_verb], verb_phr,<br>[attribute(transitivity(i),governing_verb)],<br>comment([verb_phr,[attribute(transitivity(i),governing_verb),attribute(case(acc),<br>pos_or_neg_verb_cons(pronoun))]],'Verb is intransitive so does not expect an object'])).   |
| 7.3      | check([governing_verb], main_clause,<br>[attribute(transitivity(t),governing_verb),empty(dir_obj_phr),empty(reflexive_pronoun)<br>,empty(infinitive_phr),empty(compliment_phr)],<br>comment([verb_phr,[attribute(transitivity(t),governing_verb),empty(dir_obj_phr),empt<br>y(reflexive_pronoun),not_attribute(case(acc),pos_or_neg_verb_cons(pronoun)),empty(<br>infinitive_phr),empty(compliment_phr)]], 'Verb is transitive so does expect an object'])).   |
| 7.4      | check([governing_verb], sentence,<br>[attribute(transitivity(t),sub_clause(governing_verb)),empty(sub_clause(dir_obj_phr)),e<br>mpty(sub_clause(reflexive_pronoun)),empty(sub_clause(infinitive_phr)),empty(sub_cl<br>ause(compliment_phr)),empty(relative_object)],<br>comment([verb_phr,[attribute(transitivity(t),sub_clause(governing_verb)),empty(sub_c<br>lause(dir_obj_phr)),empty(sub_clause(reflexive_pronoun)),not_attribute(case(acc),sub<br>_clause(pos_or_neg_verb_cons(pronoun))),empty(sub_clause(infinitive_phr)),empty(s<br>ub_clause(compliment_phr)),empty(relative_object)]], 'Verb is transitive so does expect<br>an object'])). |
| 7.5      | check([governing_verb], verb_phr,<br>[exists(reflexive_pronoun),attribute(can_be_reflex(n),governing_verb)],<br>comment([verb_phr,[],'This verb cannot be used reflexively'])).  |
| 7.6      | check([governing_verb], verb_phr,<br>[exists(infinitive_phr),attribute(follow_prepos([]),governing_verb)],<br>comment([verb_phr,[],'This verb cannot be followed by an infinitive phrase'])).  |
| 7.7      | check([governing_verb], verb_phr,<br>[exists(infinitive_phr),not_attribute(follow_prepos([]),governing_verb),not_one_of(foll<br>ow_prepos,verb_preposition,governing_verb)], comment([verb_phr,[],'Wrong<br>preposition used after this verb'])).  |
| 7.8      | check([sub_clause(governing_verb)], sentence,<br>[exists(relative_object_si)],   |

|          |  |
|----------|--|
|          | comment([sentence,[exists(relative_object_si),not_attribute(tense(present),sub_clause(governing_verb)),not_attribute(tense(imperfect),sub_clause(governing_verb))]],"si can only be followed by either the present or imperfect tenses')).   |
| 7.9      | check([main_clause(governing_verb)], sentence, [exists(relative_object_si)], comment([sentence,[exists(relative_object_si),attribute(tense(present),sub_clause(governing_verb)),not_attribute(tense(future),main_clause(governing_verb))]],'If "si" is followed by the present tense, the main clause should be in the future'))).               |
| 7.1<br>0 | check([main_clause(governing_verb)], sentence, [exists(relative_object_si)], comment([sentence,[exists(relative_object_si),attribute(tense(imperfect),sub_clause(governing_verb)),not_attribute(tense(conditional),main_clause(governing_verb))]],'If "si" is followed by the imperfect tense, the main clause should be in the conditional'))). |
|          |  |
| 8.0      | Le nom   |
|          |  |
| 8.1      | check([noun], noun_phr, [], select_on_first([plurality(p)]))).   |
|          |  |
| 9.0      | L'orthographe  |
|          |  |
| 9.1      | change_spelling([letter(vowel,first_letter,following_word)],[apostrophe(y)]).  |
| 9.2      | change_spelling([letter(consonant,first_letter,following_word)],[apostrophe(n)]).  |
| 9.3      | spelling_change_attribute(apostrophe).   |
|          |  |

## Appendice D

### **La catégorisation des fautes de l'échantillon de production écrite des apprenants**

(Le symbole « + » signifie qu'il y a une faute par phrase, « ++ » qu'il y en a deux par phrase et ainsi de suite. Le type de faute en question est celle qui est marquée dans la colonne directement à gauche.)

#### **Les fautes stylistiques**

|       |   |
|-------|---|
| Style | + |
|-------|---|

TOTAL: 3

| Count of Style |       |  |
|----------------|-------|--|
| No.            | Total | Exemples   |
| 81             | 1     | J'ai noté que vous avez un accent.   |
| 2440           | 1     | J'ai aussi un examen jeudi en anglais et j'espère que je vais faire bien!          |
| 2745           | 1     | Je pense que le film c'est intéressant, mais l'intrigue n'est pas très compliquer. |
| Grand Total    | 3     |  |

#### **Les fautes sémantiques**

NB: Nous avons inclus la catégorie « expressions figées » ici.

|            |     |
|------------|-----|
| Sémantique | +++ |
|------------|-----|

TOTAL: 33

| Count of Sémantique |       |  |
|---------------------|-------|--|
| No.                 | Total | Exemples   |
| 1078                | 1     | Je ne sais pas une personne qui ressemble J-Y Cousteau, je ne sais pas un homme qui est cela vieux.  |
| 1378                | 1     | Ils sont à la même école mais ils ne recountent pas pendant le jour parce que ma mère commence à sept heures et elle est fini à midi ou 13 heures. |
| Grand Total         | 2     |  |

|            |    |
|------------|----|
| Sémantique | ++ |
|------------|----|

|          |  |
|----------|--|
| Count of |  |
|----------|--|

| Sémantique  |       |   |
|-------------|-------|---|
| No.         | Total |   |
| 2957        | 1     | Je déteste les mid-terms, parce que ils sont très difficiles, et chaque question vaut beaucoup des marques. |
| 3064        | 1     | Il cache quand il les écoute.   |
| Grand Total | 2     |   |

|            |   |
|------------|---|
| Sémantique | + |
|------------|---|

| Count of Sémantique |       |   |
|---------------------|-------|---|
| No.                 | Total |   |
| 8                   | 1     | Tout de suite je ne connais pas qoui je ferai.  |
| 116                 | 1     | Le chaleur finir ma énergie et je devenir très fatigüe.   |
| 261                 | 1     | Pendant ma première année, je pris un cours dans l'anthropologie aussi.                           |
| 408                 | 1     | Je ne joue pas la guitare, mais j'aimerai apprendre.  |
| 894                 | 1     | Je suis bien, mais je suis fatigüe parce que hier après l'école j'ai travaillé pour huit heures.  |
| 1222                | 1     | Je suis très désolé, il est très dernier!   |
| 1231                | 1     | Je pense que j'ai faire le test très bien.  |
| 1234                | 1     | Mais, le question deux dans les communications était très difficile.                              |
| 1277                | 1     | Je suis toujours heureux quand je trouve quelqu'un qui a origine de Roumanie.                     |
| 1299                | 1     | Je n'aime pas aussi le rap, mais, j'aime presque toutes les genres de la musique.                 |
| 1390                | 1     | Je ne connais pas que je veux devenir.  |
| 1468                | 1     | Je pense les romans biographies sont moins intéressants que les romans d'aventure.                |
| 1499                | 1     | Quand il pleut, je me décourager.   |
| 1525                | 1     | Dans decembre quand il pleut j'irai snowboarding à Whistler.                                      |
| 1614                | 1     | Je suis payé 15\$-25\$ par le jeu.  |
| 1634                | 1     | Je suis très bien aussi.  |
| 1654                | 1     | Je suis aller regarder le pièce pour moi classe de l'anglais.                                     |
| 1843                | 1     | Je ne suis pas né à Prince George, mais je suis né à Langley, mais je ne jamais habite à Langley. |
| 1898                | 1     | C'est un bon ordinateur, mais, il est un ordinateur par son nature.                               |
| 2157                | 1     | Oui, c'est le temps pour le pleut, c'est l'octobre.   |
| 2184                | 1     | Ma mère est chrétien et ma père et juif!  |
| 2404                | 1     | Je alle au cinéma fois dans moment.   |
| 2460                | 1     | Le vendredi demain je vais aller à une fête d'hallowe'en avec mes amis.                           |
| Grand Total         | 23    |   |

### Les fautes morpho-syntaxiques

|             |   |
|-------------|---|
| Conjugaison | + |
|-------------|---|

TOTAL 22

| Count of Conjugaison |           |  |
|----------------------|-----------|--|
| No.                  | Total     |  |
| 41                   | 1         |  |
| 115                  | 1         |  |
| 263                  | 1         |  |
| 497                  | 1         |  |
| 644                  | 1         |  |
| 909                  | 1         |  |
| 1135                 | 1         |  |
| 1271                 | 1         |  |
| 1423                 | 1         |  |
| 1499                 | 1         |  |
| 1560                 | 1         |  |
| 1654                 | 1         |  |
| 2134                 | 1         |  |
| 2404                 | 1         |  |
| 2585                 | 1         |  |
| 2978                 | 1         |  |
| <b>Grand Total</b>   | <b>16</b> |  |

Et, si ton amie veux aller au Brésil, elle peut aller à Rio de Janeiro ou le nord-ouest du pays, qu'il y a des plages merveilleuses.

J'aime la chaleur quand je suis à la plage mais quand je suis marche sur terrain de rugby je suis malheureux.

L'été dernier, je visitée les Philippines.

Cette année, je habite à Totem Park, mais ma famille habitée dans Surrey.

Je veut parler un peu de stationnement à UBC.

J'ai étudié à l'université de Colombie-Britannique aussi.

Il étudié beaucoup.

Mes parents son roumaine et je parle la langue aussi (avec un accent fort)!

Je dormis beaucoup.

Quand il pleut, je me décourager.

La nourriture été incroyable (très bien:) et j'ai mangé beaucoup.

Je suis aller regarder le pièce pour moi classe de l'anglais.

Je vas être une sorcière.

Je alle au cinéma fois dans moment.

J'ai été Chrétien depuis 15 ans.

J'ai toujours détester les maths parce que je pense c'est difficile aussi.

|             |    |
|-------------|----|
| Conjugaison | ++ |
|-------------|----|

| Count of Conjugaison |          |  |
|----------------------|----------|--|
| No.                  | Total    |  |
| 116                  | 1        |  |
| 588                  | 1        |  |
| 1122                 | 1        |  |
| <b>Grand Total</b>   | <b>3</b> |  |

Le chaleur finir ma énergie et je devenir très fatiguer.

Mais je suis préjugé parce-que mes parents son roumain et ma mère a fait les gymnastiques beaucoup quand elle été étudiante.

Je veux fait yoga quand je suis moins d'occupé.

|             |   |
|-------------|---|
| Préposition | + |
|-------------|---|

TOTAL: 21

| Count of Préposition |       |  |
|----------------------|-------|--|
| No.                  | Total |  |

|                    |           |   |
|--------------------|-----------|---|
| 38                 | 1         | Je viens de Belo Horizonte, une ville qui est en l'intérieur du pays.   |
| 41                 | 1         | Et, si ton amie veux aller au Brésil, elle peut aller à Rio de Janeiro ou le nord-ouest du pays, qu'il y a des plages merveilleuses.              |
| 47                 | 1         | J'ai habité dans la même maison pour toute ma vie.  |
| 60                 | 1         | Je voudrais voyager à France.   |
| 261                | 1         | Pendant ma première année, je pris un cours dans l'anthropologie aussi.   |
| 298                | 1         | J'habite en Californie dans l'été.  |
| 400                | 1         | J'ai essayé au jouer le guitare mais c'est difficile.   |
| 497                | 1         | Cette année, je habite à Totem Park, mais ma famille habité dans Surrey.  |
| 551                | 1         | Calgary est une petite université et le recrutement est plus bon que l'université de Colombie-britannique.  |
| 894                | 1         | Je suis bien, mais je suis fatigué parce que hier après l'école j'ai travaillé pour huit heures.  |
| 1078               | 1         | Je ne sais pas une personne qui ressemble J-Y Cousteau, je ne sais pas un homme qui est cela vieux.   |
| 1378               | 1         | Ils sont à la même école mais ils ne reçoivent pas pendant le jour parce que ma mère commence à sept heures et elle est fini à midi ou 13 heures. |
| 1507               | 1         | Je n'aime pas la pluie quand c'est froid dans l'automne ou l'hiver.   |
| 1525               | 1         | Dans décembre quand il pleut j'irai snowboarding à Whistler.  |
| 1796               | 1         | J'ai joué au baseball pour 10 ans!  |
| 1857               | 1         | J'ai vécu en Tulameen et Vancouver.   |
| 2547               | 1         | Oui, j'ai regardé les Canucks sur la télévision.  |
| 3026               | 1         | Dans Prince George, il y a beaucoup de neige.   |
| <b>Grand Total</b> | <b>18</b> |   |

Préposition      +++

|                      |          |   |
|----------------------|----------|---|
| Count of Préposition |          |   |
| No.                  | Total    |   |
| 2801                 | 1        | Je suis allé à le Disneyland dans le California et le Euro-Disney dans le France aussi. |
| <b>Grand Total</b>   | <b>1</b> |   |

Accord      ++

TOTAL: 12

|                 |       |  |
|-----------------|-------|--|
| Count of Accord |       |  |
| No.             | Total |  |
| 3016            | 1     | Mon sports favori sont le baseball, le football et l'hockey. |

|             |   |  |
|-------------|---|--|
| 2184        | 1 | Ma mère est chrétien et ma père et juif! |
| Grand Total | 2 |  |

|        |   |
|--------|---|
| Accord | + |
|--------|---|

| Count of Accord |       |  |
|-----------------|-------|--|
| No.             | Total |  |
| 263             | 1     | L'été dernier, je visitée les Philippines.   |
| 636             | 1     | Je suis dans mon premier année de Université a l'Université de Colombie-Britannique.   |
| 706             | 1     | Ils sont irlandais et allemand.  |
| 1271            | 1     | Mes parents son roumaine et je parle la langue aussi (avec un accent fort)!  |
| 1854            | 1     | Moi favorit province est Colombie-Britannique.   |
| 2745            | 1     | Je pense que le film c'est intéressant, mais l'intrigue n'est pas très compliquer.   |
| 588             | 1     | Mais je suis préjugé parce-que mes parents son roumain et ma mère a fait les gymnastiques beaucoup quand elle été étudiante. |
| Grand Total     | 8     |  |

|        |   |
|--------|---|
| Accent | + |
|--------|---|

TOTAL: 10

| Count of Accent |       |  |
|-----------------|-------|--|
| No.             | Total |  |
| 497             | 1     | Cette année, je habite à Totem Park, mais ma famille habité dans Surrey.                         |
| 636             | 1     | Je suis dans mon premier année de Université a l'Université de Colombie-Britannique.             |
| 894             | 1     | Je suis bien, mais je suis fatigue parce que hier après l'école j'ai travaillé pour huit heures. |
| 1222            | 1     | Je suis très désolé, il est très dernier!  |
| 1231            | 1     | Je pense que j'ai faire le test très bien.   |
| 1249            | 1     | Je pense le quiz électronique est une idée fantastique pour etudier.                             |
| 1525            | 1     | Dans decembre quand il pleut j'irai snowboarding à Whistler.                                     |
| 1644            | 1     | Il est tres difficile quand Je vais à l'université et je vais à travail dans un jour.            |
| Grand Total     | 8     |  |

|        |    |
|--------|----|
| Accent | ++ |
|--------|----|

| Count of Accent |       |  |
|-----------------|-------|--|
| No.             | Total |  |

|             |   |   |
|-------------|---|---|
|             | 1 |   |
| 116         | 1 | Le chaleur finir ma énergie et je devenir très fatigüe. |
| Grand Total | 1 |   |

|             |   |
|-------------|---|
| Orthographe | + |
|-------------|---|

TOTAL: 10

| Count of Orthographe No. | Total |  |
|--------------------------|-------|--|
| 8                        | 1     | Tout de suite je ne connais pas qoui je ferai.   |
| 60                       | 1     | Je voudrais voyager à france.  |
| 239                      | 1     | J'ami regarder le sports.  |
| 551                      | 1     | Calgary est une petite université et le recrutement est plus bon que l'université de colombie-britannique. |
| 636                      | 1     | Je suis dans mon premier année de Université a l'Université de Colombie-Britannique.                       |
| 1078                     | 1     | Je ne sais pas une personne qui ressemble J-Y Cousteau, je ne sais pas un homme qui est cela vieux.        |
| 1854                     | 1     | Moi favorit province est Colombie-Britannique.   |
| 1903                     | 1     | Mais, si je vais en classe, je vais dormir en class et le professeur n'aime pas quand je dors en classe.   |
| 2134                     | 1     | Je vas être une sorcièree.   |
| 2801                     | 1     | Je suis allé à le Disneyland dans le California et le Euro-Disney dans le France aussi.                    |
| Grand Total              | 10    |  |

|           |   |
|-----------|---|
| Majuscule | + |
|-----------|---|

TOTAL: 6

| Count of Majuscule No. | Total |  |
|------------------------|-------|--|
| 116                    | 1     | Le chaleur finir ma énergie et je devenir très fatigüe.  |
| 551                    | 1     | Calgary est une petite université et le recrutement est plus bon que l'université de colombie-britannique. |
| 809                    | 1     | J'adore tokyo.   |
| 909                    | 1     | J'ai étudier à l'université de Colombie-Britannique aussi.   |
| 1644                   | 1     | Il est tres difficile quand Je vais à l'université et je vais à travail dans un jour.                      |
| 2585                   | 1     | J'ai été Chrétien depuis 15 ans.   |
| Grand Total            | 6     |  |

|       |   |
|-------|---|
| Genre | + |
|-------|---|

TOTAL: 5

| Count of Genre |       |  |
|----------------|-------|--|
| No.            | Total |  |
| 1234           | 1     | Mais, le question deux dans les communications était très difficile. |
| 1654           | 1     | Je suis aller regarder le pièce pour moi classe de l'anglais.        |
| 1854           | 1     | Moi favorit province est Colombie-Britannique.                       |
| 1898           | 1     | C'est un bon ordinateur, mais, il est un ordinateur par son nature.  |
| 2157           | 1     | Oui, c'est le temps pour le pleut, c'est l'octobre.                  |
| Grand Total    | 5     |  |

|   |   |
|---|---|
| Elision, trait-d'unions et contractions | + |
|---|---|

TOTAL: 6

| Count of Elision, trait-d'unions et contractions |       |  |
|--|-------|--|
| No.  | Total |  |
| 497  | 1     | Cette année, je habite à Totem Park, mais ma famille habité dans Surrey.   |
| 588  | 1     | Mais je suis préjugé parce-que mes parents son roumain et ma mère a fait les gymnastiques beaucoup quand elle été étudiante. |
| 894  | 1     | Je suis bien, mais je suis fatigue parce que hier après l'école j'ai travaillé pour huit heures.                             |
| 1092   | 1     | Je pense que ma professeure de français ressemble au Florence!   |
| 2957   | 1     | Je déteste les mid-terms, parce que ils sont très difficiles, et chaque question vaut beaucoup des marques.                  |
| 3016   | 1     | Mon sports favori sont le baseball, le football et l'hockey.   |
| Grand Total                                      | 6     |  |

|                |   |
|----------------|---|
| Ordre des mots | + |
|----------------|---|

TOTAL: 3

| Count of Ordre des mots |       |  |
|-------------------------|-------|--|
| No.                     | Total |  |
| 147                     | 1     | On seulement a besoin d'un pull-over pendant les jours de l'hiver. |
| 1843                    | 1     | Je ne suis pas né à Prince George, mais je suis né à               |

|             |   |  |
|-------------|---|--|
|             |   | Langley, mais je ne jamais habite à Langley.   |
| 1854        | 1 | Moi favorit province est Colombie-Britannique. |
| Grand Total | 3 |  |

|          |   |
|----------|---|
| Négation | + |
|----------|---|

TOTAL: 3

| Count of Négation |       |   |
|-------------------|-------|---|
| No.               | Total |   |
| 408               | 1     | Je ne joue pas la guitare, mais j'aimerais apprendre.                             |
| 1299              | 1     | Je n'aime pas aussi le rap, mais, j'aime presque toutes les genres de la musique. |
| 1688              | 1     | Je n'aime pas la date de l'examen français aussi.                                 |
| Grand Total       | 3     |   |

|                         |   |
|-------------------------|---|
| Expressions de quantité | + |
|-------------------------|---|

TOTAL: 2

| Count of Expressions de quantité |       |   |
|----------------------------------|-------|---|
| No.                              | Total |   |
| 644                              | 1     | Je veut parler un peu de stationnement à UBC.   |
| 2957                             | 1     | Je déteste les mid-terms, parce que ils sont très difficiles, et chaque question vaut beaucoup des marques. |
| Grand Total                      | 2     |   |

|            |   |
|------------|---|
| Comparatif | + |
|------------|---|

TOTAL: 1

| Count of Comparatif |       |  |
|---------------------|-------|--|
| No.                 | Total |  |
| 551                 | 1     | Calgary est une petite université et le recrutement est plus bon que l'université de colombie-britannique. |
| Grand Total         | 1     |  |

|                |   |
|----------------|---|
| Noms adjectifs | + |
|----------------|---|

TOTAL: 2

| Count of Noms adjectifs |       |
|-------------------------|-------|
| No.                     | Total |
|                         |       |

| No.                | Total    |   |
|--------------------|----------|---|
| 1654               | 1        | Je suis aller regarder le pièce pour moi classe de l'anglais. |
| 1688               | 1        | Je n'aime pas la date de l'examen français aussi.             |
| <b>Grand Total</b> | <b>2</b> |   |

|                  |   |
|------------------|---|
| absence du «que» | + |
|------------------|---|

TOTAL: 3

| Count of absence du «que» |          |  |
|---------------------------|----------|--|
| No.                       | Total    |  |
| 1249                      | 1        | Je pense le quiz électronique est une idée fantastique pour etudier.               |
| 1468                      | 1        | Je pense les romans biographies sont moins intéressants que les romans d'aventure. |
| 2978                      | 1        | J'ai toujours détester les maths parce que je pense c'est difficile aussi.         |
| <b>Grand Total</b>        | <b>3</b> |  |

## Appendice E

### Le rendement de LINGER

Voici les phrases originales que la grammaire de LINGER devrait être capable de gérer. (Dans quelques cas, il fallait ne prendre qu'un morceau de la phrase originale.) Les phrases testées montrent les modifications faites aux phrases pour les adapter à LINGER. Les fautes qu'on cherche à corriger sont expliquées au troisième colonne.

| No.  | Phrases originales  | Phrases testées                     | Fautes                                   |
|------|---|-------------------------------------|--|
| 1468 | Je pense les romans biographies sont moins intéressants que les romans d'aventure.                            | Je vois les livres sont verts.      | absence du que                           |
| 2153 | Aujourd'hui, le temps pluvieux.   | L'homme heureux.                    | absence du verbe                         |
| 3016 | Mon sports favori sont le baseball, le football et l'hockey.  | Mon chats intelligent sont heureux. | accord des adjectifs                     |
| 3043 | Je ne peux pas marcher dans un avion parce que l'allée sont très petites.                                     | La maison sont petites.             | accord des noms                          |
| 636  | Je suis dans mon premier année de Université a l'Université de Colombie-Britannique.                          | Je vais a la maison.                | confusion entre a (verbe et préposition) |
| 644  | Je veut parler un peu de stationnement à UBC.   | Je voit chanter la femme.           | conjugaison                              |
| 909  | J'ai étudier à l'université de Colombie-Britannique aussi.  | J'ai parler à la fille.             | conjugaison                              |
| 1135 | Il étudier beaucoup.  | Il parler constamment.              | conjugaison                              |
| 1231 | Je pense que j'ai faire le test très bien.  | J'ai faire le tour.                 | conjugaison                              |
| 1499 | Quand il pleut, je me décourager.   | Je me laver.                        | conjugaison                              |
| 2404 | Je alle au cinéma fois dans moment.   | Je alle à la maison.                | conjugaison                              |
| 115  | J'aime la chaleur quand je suis à la plage mais quand je suis marche sur terrain de rugby je suis malheureux. | Je suis marche.                     | conjugaison                              |
| 263  | L'été dernier, je visitée les Philippines.  | Je mangée la pomme.                 | conjugaison                              |
| 588  | Mais je suis préjugé parce-que mes parents son roumain  | Mes filles son jolie.               | conjugaison                              |

|      |   |                                |  |
|------|---|--------------------------------|--|
|      | et ma mère a fait les gymnastiques beaucoup quand elle été étudiante.                             |                                |  |
| 497  | Cette année, je habite à Totem Park, mais ma famille habité dans Surrey.                          | Ma femme parlé à sa tante.     | conjugaison et élisition                   |
| 116  | Le chaleur finir ma énergie et je devenir très fatigüe.   | Le fille finir sa église.      | conjugaison et genre                       |
| 2105 | Je suis une fanatique du chocolat.  | Je suis tante.                 | être sans article                          |
| 2134 | Je vas être une sorcièree.  | Je vas être tante.             | être sans article et conjugaison           |
| 408  | Je ne joue pas la guitare, mais j'aimerai apprendre.  | Je ne vois pas de pommes.      | l'article partitif dans une négation       |
| 1843 | Je ne suis pas né à Prince George, mais je suis né à Langley, mais je ne jamais habite à Langley. | Je ne jamais vais à sa maison. | l'ordre des mots dans une négation         |
| 2184 | Ma mère est chrétien et ma père et juif!  | Ma père et laid!               | les adjectifs possessifs et la conjugaison |

NO. 1468

Pre-parsing sentence

Trying for a legal parse

Trying recovery level 1 (word in wrong place -- move)

Trying a deletion

Trying recovery level 2 (word missing -- insertion)

Trying an insertion

Recovery level 2 succeeded

Parse tree selected:

```

sentence
  main_clause
    subj_phr
      noun_pron_phr
        pronoun : [je,1]
      verb_phr
        pos_or_neg_verb_cons

```

```

pos_verb_cons
rp_pron_list
pron_list
  pron_list2
governing_verb
  verb_or_aux_verb
    verb : [vois,2]
dir_obj_phr
noun_phr
  determiner : [les,3]
  adj_list
  noun : [livres,4]
  adj_list
indir_obj_phr
sub_clause
  relative_subject : [relative_subject,5]
  verb_phr
    pos_or_neg_aux_verb_cons
      governing_verb
        aux_verb : [sont,6]
    compliment_phr
      adjective : [verts,7]

```

Your sentence was:

--- je vois les livres sont verts . ---

I think the correct version of your sentence should read:

>>> je vois les livres <relative\_subject> sont verts . <<<

\*\*\*\* Comments \*\*\*\*

A word of type <relative\_subject> was missing and has been inserted

\*\*\*\* End Comments \*\*\*\*

NO. 2153

Pre-parsing sentence

Trying for a legal parse

Trying recovery level 1 (word in wrong place -- move)

Trying a deletion

Trying recovery level 2 (word missing -- insertion)  
Trying an insertion

Recovery level 2 succeeded

Parse tree selected:

```
sentence
  main_clause
    subj_phr
      noun_pron_phr
        noun_phr
          determiner : [l',1]
          adj_list
          noun : [homme,2]
          adj_list
            adjective : [heureux,3]
            adj_list
        verb_phr
          pos_or_neg_verb_cons
          pos_verb_cons
          rp_pron_list
            pron_list
            pron_list2
          governing_verb
            verb_or_aux_verb
            verb : [verb,4]
          dir_obj_phr
          indir_obj_phr
```

Your sentence was:

--- l' homme heureux . ---

I think the correct version of your sentence should read:

>>> l' homme heureux <verb> . <<<

\*\*\*\* Comments \*\*\*\*

A word of type <verb> was missing and has been inserted

\*\*\*\* End Comments \*\*\*\*

NO. 3016  
Pre-parsing sentence

Trying for a legal parse

Legal parse found

Parse tree selected:

```
sentence
  main_clause
    subj_phr
      noun_pron_phr
        noun_phr
          determiner : [mon,1]
          adj_list
          noun : [chats,2]
          adj_list
            adjective : [intelligent,3]
            adj_list
          verb_phr
            pos_or_neg_aux_verb_cons
            governing_verb
              aux_verb : [sont,4]
            compliment_phr
              adjective : [heureux,5]
```

Your sentence was:

--- mon chats intelligent sont heureux . ---

I think the correct version of your sentence should read:

>>> mes chats intelligents sont heureux . <<<

\*\*\*\* Comments \*\*\*\*

determiner "mon" changed to "mes" : [gender(m),plurality(p)]

adjective "intelligent" changed to "intelligents" : [position(post),gender(m),plurality(p)]

\*\*\*\* End Comments \*\*\*\*

NO. 3043

Pre-parsing sentence

Trying for a legal parse

Legal parse found

Parse tree selected:

```
sentence
  main_clause
    subj_phr
      noun_pron_phr
        noun_phr
          determiner : [la,1]
          adj_list
          noun : [maison,2]
          adj_list
        verb_phr
          pos_or_neg_aux_verb_cons
          governing_verb
          aux_verb : [sont,3]
          compliment_phr
            adjective : [petites,4]
```

Your sentence was:

--- la maison sont petites . ---

I think the correct version of your sentence should read:

>>> la maison est petite . <<<

\*\*\*\* Comments \*\*\*\*

aux\_verb "sont" changed to "est" : [tense(present),transitivity(t),person(3),plurality(s)]

adjective "petites" changed to "petite" : [position(pre),gender(f),plurality(s)]

\*\*\*\* End Comments \*\*\*\*

NO. 636

Pre-parsing sentence

Trying for a legal parse

Trying recovery level 1 (word in wrong place -- move)  
Trying a deletion

Trying recovery level 2 (word missing -- insertion)  
Trying an insertion

Recovery level 2 succeeded

Parse tree selected:

```
sentence
  main_clause
    subj_phr
      noun_pron_phr
        pronoun : [je,1]
      verb_phr
        pos_or_neg_verb_cons
          pos_verb_cons
            rp_pron_list
              pron_list
                pron_list2
            governing_verb
              verb_or_aux_verb
                verb : [vais,2]
          dir_obj_phr
            indir_obj_phr
        sub_clause
          relative_subject : [relative_subject,3]
          verb_phr
            pos_or_neg_verb_cons
              pos_verb_cons
                rp_pron_list
                  pron_list
                    pron_list2
                governing_verb
                  verb_or_aux_verb
                    aux_verb : [a,4]
              dir_obj_phr
                noun_phr
                  determiner : [la,5]
                  adj_list
                    noun : [maison,6]
                  adj_list
                indir_obj_phr
```

Your sentence was:

--- je vais a la maison . ---

I think the correct version of your sentence should read:

>>> je vais <relative\_subject> a la maison . <<<

\*\*\*\* Comments \*\*\*\*

A word of type <relative\_subject> was missing and has been inserted

\*\*\*\* End Comments \*\*\*\*

NO. 644

Pre-parsing sentence

Trying for a legal parse

Trying recovery level 1 (word in wrong place -- move)

Trying a deletion

Trying recovery level 2 (word missing -- insertion)

Trying an insertion

Recovery level 2 succeeded

Parse tree selected:

```
sentence
  main_clause
    subj_phr
      noun_pron_phr
        pronoun : [je,1]
      verb_phr
        pos_or_neg_verb_cons
        pos_verb_cons
        rp_pron_list
        pron_list
        pron_list2
        governing_verb
          verb_or_aux_verb
            verb : [voit,2]
        dir_obj_phr
        indir_obj_phr
    sub_clause
      relative_subject : [relative_subject,3]
      verb_phr
```

pos\_or\_neg\_verb\_cons  
pos\_verb\_cons  
rp\_pron\_list  
pron\_list  
pron\_list2  
governing\_verb  
verb\_or\_aux\_verb  
verb : [chanter,4]  
dir\_obj\_phr  
noun\_phr  
determiner : [la,5]  
adj\_list  
noun : [femme,6]  
adj\_list  
indir\_obj\_phr

Your sentence was:

--- je voit chanter la femme . ---

I think the correct version of your sentence should read:

>>> je vois <relative\_subject> chantent la femme . <<<<

\*\*\*\* Comments \*\*\*\*

verb "voit" changed to "vois" :

[transitivity(t),follow\_prepos([[]]),can\_be\_reflex(y),person(1),plurality(s),tense(present)]

A word of type <relative\_subject> was missing and has been inserted

verb "chanter" changed to "chantent" :

[transitivity(i),follow\_prepos([[]]),can\_be\_reflex(n),person(3),plurality(p),tense(present)]

"chanter" : Verb is intransitive so does not expect an object

"voir" : Verb is transitive so does expect an object

\*\*\*\* End Comments \*\*\*\*

NO. 909

Pre-parsing sentence

Trying for a legal parse

Trying recovery level 1 (word in wrong place -- move)

Trying a deletion

Trying recovery level 2 (word missing -- insertion)

Trying an insertion

Recovery level 2 succeeded

Parse tree selected:

```
sentence
  main_clause
    subj_phr
      noun_pron_phr
        pronoun : [j',1]
      verb_phr
        pos_or_neg_verb_cons
          pos_verb_cons
            rp_pron_list
              pron_list
                pron_list2
            governing_verb
              verb_or_aux_verb
                aux_verb : [ai,2]
          dir_obj_phr
          indir_obj_phr
        sub_clause
          relative_subject : [relative_subject,3]
          verb_phr
            pos_or_neg_verb_cons
              pos_verb_cons
                rp_pron_list
                  pron_list
                    pron_list2
                governing_verb
                  verb_or_aux_verb
                    verb : [parler,4]
            dir_obj_phr
            indir_obj_phr
              preposition : [a2,5]
            noun_phr
              determiner : [la,6]
              adj_list
              noun : [fille,7]
              adj_list
```

Your sentence was:

--- j' ai parler a2 la fille . ---

I think the correct version of your sentence should read:

>>> j' ai <relative\_subject> parlent a2 la fille . <<<

\*\*\*\* Comments \*\*\*\*

A word of type <relative\_subject> was missing and has been inserted

verb "parler" changed to "parlent" :

[transitivity(i),follow\_prepos([]),can\_be\_reflex(y),person(3),plurality(p),tense(present)]

"avoir" : Verb is transitive so does expect an object

\*\*\*\* End Comments \*\*\*\*

NO. 1135

Pre-parsing sentence

Trying for a legal parse

Legal parse found

Parse tree selected:

```
sentence
  main_clause
    subj_phr
      noun_pron_phr
        pronoun : [il,1]
      verb_phr
        pos_or_neg_verb_cons
        pos_verb_cons
        rp_pron_list
        pron_list
        pron_list2
        governing_verb
        verb_or_aux_verb
        verb : [parler,2]
        adverb : [constamment,3]
      dir_obj_phr
      indir_obj_phr
```

Your sentence was:

--- il parler constamment . ---

I think the correct version of your sentence should read:

>>> il parle constamment . <<<

\*\*\*\* Comments \*\*\*\*

verb "parler" changed to "parle" :

[transitivity(i),follow\_prepos([]),can\_be\_reflex(y),person(3),plurality(s),tense(present)]

\*\*\*\* End Comments \*\*\*\*

NO. 1231

Pre-parsing sentence

Trying for a legal parse

Trying recovery level 1 (word in wrong place -- move)

Trying a deletion

Trying recovery level 2 (word missing -- insertion)

Trying an insertion

Trying recovery level 3 (word is of wrong type -- substitution)

Trying a substitution

Recovery level 3 succeeded

Parse tree selected:

```
sentence
  main_clause
    subj_phr
      noun_pron_phr
        pronoun : [j',1]
      verb_phr
        pos_or_neg_verb_cons
        pos_verb_cons
        rp_pron_list
        pron_list
        pron_list2
        governing_verb
        verb_or_aux_verb
        aux_verb : [ai,2]
        adverb : [adverb,3]
```

dir\_obj\_phr  
noun\_phr  
determiner : [le,4]  
adj\_list  
noun : [tour,5]  
adj\_list  
indir\_obj\_phr

Your sentence was:

--- j' ai faire le tour . ---

I think the correct version of your sentence should read:

>>> j' ai <adverb> le tour . <<<

\*\*\*\* Comments \*\*\*\*

The word "faire" should not have been present and has been deleted

A word of type <adverb> was missing and has been inserted

\*\*\*\* End Comments \*\*\*\*

NO. 1499

Pre-parsing sentence

Trying for a legal parse

Legal parse found

Legal parse found

Parse tree selected:

sentence  
main\_clause  
subj\_phr  
noun\_pron\_phr  
pronoun : [je,1]  
verb\_phr  
pos\_or\_neg\_verb\_cons  
pos\_verb\_cons  
rp\_pron\_list  
pron\_list  
pron\_list2

pronoun : [me,2]  
pron\_list2  
governing\_verb  
verb\_or\_aux\_verb  
verb : [laver,3]  
dir\_obj\_phr  
indir\_obj\_phr

Your sentence was:

--- je me laver . ---

I think the correct version of your sentence should read:

>>> je me lave . <<<

\*\*\*\* Comments \*\*\*\*

verb "laver" changed to "lave" :

[transitivity(t),follow\_prepos([]),can\_be\_reflex(y),person(1),plurality(s),tense(present)]

\*\*\*\* End Comments \*\*\*\*

NO. 2404

Pre-parsing sentence

Trying for a legal parse

Legal parse found

Parse tree selected:

sentence  
main\_clause  
subj\_phr  
noun\_pron\_phr  
pronoun : [je,1]  
verb\_phr  
pos\_or\_neg\_verb\_cons  
pos\_verb\_cons  
rp\_pron\_list  
pron\_list  
pron\_list2  
governing\_verb

verb\_or\_aux\_verb  
verb : [alle,2]  
dir\_obj\_phr  
indir\_obj\_phr  
preposition : [a2,3]  
noun\_phr  
determiner : [la,4]  
adj\_list  
noun : [maison,5]  
adj\_list

Your sentence was:

--- je alle a2 la maison . ---

I think the correct version of your sentence should read:

>>> je <alle> a2 la maison . <<<

\*\*\*\* Comments \*\*\*\*

"alle" is unknown and has been guessed to be of type verb

\*\*\*\* End Comments \*\*\*\*

NO. 115

Pre-parsing sentence

Trying for a legal parse

Trying recovery level 1 (word in wrong place -- move)

Trying a deletion

Trying recovery level 2 (word missing -- insertion)

Trying an insertion

Recovery level 2 succeeded

Parse tree selected:

sentence  
main\_clause  
subj\_phr  
noun\_pron\_phr  
pronoun : [je,1]  
verb\_phr

```

pos_or_neg_verb_cons
pos_verb_cons
rp_pron_list
pron_list
pron_list2
governing_verb
verb_or_aux_verb
aux_verb : [suis,2]
dir_obj_phr
indir_obj_phr
sub_clause
relative_subject : [relative_subject,3]
verb_phr
pos_or_neg_verb_cons
pos_verb_cons
rp_pron_list
pron_list
pron_list2
governing_verb
verb_or_aux_verb
verb : [marche,4]
dir_obj_phr
indir_obj_phr

```

Your sentence was:

--- je suis marche . ---

I think the correct version of your sentence should read:

>>> je suis <relative\_subject> marche . <<<<

\*\*\*\* Comments \*\*\*\*

A word of type <relative\_subject> was missing and has been inserted

"e4tre" : Verb is transitive so does expect an object

\*\*\*\* End Comments \*\*\*\*

NO. 263

Parse tree selected:

```

sentence
main_clause
subj_phr
noun_pron_phr

```

pronoun : [je,1]  
verb\_phr  
pos\_or\_neg\_verb\_cons  
pos\_verb\_cons  
rp\_pron\_list  
pron\_list  
pron\_list2  
governing\_verb  
verb\_or\_aux\_verb  
verb : [mange1e,2]  
dir\_obj\_phr  
noun\_phr  
determiner : [la,3]  
adj\_list  
noun : [pomme,4]  
adj\_list  
indir\_obj\_phr

Your sentence was:

--- je mange le la pomme . ---

I think the correct version of your sentence should read:

>>> je mange la pomme . <<<

\*\*\*\* Comments \*\*\*\*

verb "mange1e" changed to "mange" :

[transitivity(t),follow\_prepos([]),can\_be\_reflex(n),person(1),plurality(s),tense(present  
]

\*\*\*\* End Comments \*\*\*\*

NO. 588

Pre-parsing sentence

Trying for a legal parse

Trying recovery level 1 (word in wrong place -- move)

Trying a deletion

Trying recovery level 2 (word missing -- insertion)

Trying an insertion

Trying recovery level 3 (word is of wrong type -- substitution)  
Trying a substitution

Recovery level 3 succeeded

Parse tree selected:

```
sentence
  main_clause
    subj_phr
      noun_pron_phr
        noun_phr
          determiner : [mes,1]
          adj_list
          noun : [filles,2]
          adj_list
        verb_phr
          pos_or_neg_aux_verb_cons
          governing_verb
          aux_verb : [aux_verb,3]
          compliment_phr
          adjective : [jolie,4]
```

Your sentence was:

--- mes filles son jolie . ---

I think the correct version of your sentence should read:

>>> mes filles sont jolies . <<<<

\*\*\*\* Comments \*\*\*\*

The word "son" should not have been present and has been deleted

aux\_verb "aux\_verb" changed to "sont" : [tense(present),transitivity(t),person(3),plurality(p)]

adjective "jolie" changed to "jolies" : [position(pre),gender(f),plurality(p)]

\*\*\*\* End Comments \*\*\*\*

NO. 1654

==>Je suis aller voir moi chat.

Pre-parsing sentence

Trying for a legal parse

Trying recovery level 1 (word in wrong place -- move)

Trying a deletion

Trying recovery level 2 (word missing -- insertion)

Trying an insertion

Trying recovery level 3 (word is of wrong type -- substitution)

Trying a substitution

Trying recovery level 4 (word should not be present -- deletion)

Trying a deletion

Trying recovery level 5 (remove one word & insert another -- deletion & insertion)

Trying a deletion

Trying an insertion

SORRY : Program unable to parse this sentence

NO. 497

Pre-parsing sentence

Trying for a legal parse

Legal parse found

Parse tree selected:

sentence

main\_clause

subj\_phr

noun\_pron\_phr

noun\_phr

determiner : [ma,1]

adj\_list

noun : [femme,2]

adj\_list

verb\_phr

pos\_or\_neg\_verb\_cons  
pos\_verb\_cons  
rp\_pron\_list  
pron\_list  
pron\_list2  
governing\_verb  
verb\_or\_aux\_verb  
verb : [parle1,3]  
dir\_obj\_phr  
indir\_obj\_phr  
preposition : [a2,4]  
noun\_phr  
determiner : [sa,5]  
adj\_list  
noun : [tante,6]  
adj\_list

Your sentence was:

--- ma femme parle1 a2 sa tante . ---

I think the correct version of your sentence should read:

>>> ma femme parle a2 sa tante . <<<

\*\*\*\* Comments \*\*\*\*

verb "parle1" changed to "parle" :

[transitivity(i),follow\_prepos([]),can\_be\_reflex(y),person(3),plurality(s),tense(present)]

\*\*\*\* End Comments \*\*\*\*

NO. 116

Pre-parsing sentence

Trying for a legal parse

Legal parse found

Parse tree selected:

sentence  
main\_clause  
subj\_phr

```

noun_pron_phr
  noun_phr
    determiner : [le,1]
    adj_list
    noun : [fille,2]
    adj_list
verb_phr
  pos_or_neg_verb_cons
  pos_verb_cons
  rp_pron_list
  pron_list
  pron_list2
  governing_verb
  verb_or_aux_verb
  verb : [finir,3]
dir_obj_phr
  noun_phr
    determiner : [sa,4]
    adj_list
    noun : [e1glise,5]
    adj_list
  indir_obj_phr

```

Your sentence was:

--- le fille finir sa e1glise . ---

I think the correct version of your sentence should read:

>>> la fille finit sa e1glise . <<<

\*\*\*\* Comments \*\*\*\*

determiner "le" changed to "la" : [gender(f),plurality(s),apostrophe(n)]

verb "finir" changed to "finit" :

[transitivity(t),follow\_prepos([[100,101]]),can\_be\_reflex(n),person(3),plurality(s),tense(resent)]

\*\*\*\* End Comments \*\*\*\*

NO. 2105

Pre-parsing sentence

Trying for a legal parse

Trying recovery level 1 (word in wrong place -- move)  
Trying a deletion

Trying recovery level 2 (word missing -- insertion)  
Trying an insertion

Recovery level 2 succeeded

Parse tree selected:

```
sentence
  main_clause
    subj_phr
      noun_pron_phr
        pronoun : [je,1]
      verb_phr
        pos_or_neg_verb_cons
        pos_verb_cons
        rp_pron_list
        pron_list
        pron_list2
        governing_verb
        verb_or_aux_verb
        aux_verb : [suis,2]
    dir_obj_phr
      noun_phr
        determiner : [determiner,3]
        adj_list
        noun : [tante,4]
        adj_list
    indir_obj_phr
```

Your sentence was:

--- je suis tante . ---

I think the correct version of your sentence should read:

>>> je suis <determiner> tante . <<<

\*\*\*\* Comments \*\*\*\*

A word of type <determiner> was missing and has been inserted

\*\*\*\* End Comments \*\*\*\*

NO. 2134

Pre-parsing sentence

Trying for a legal parse

Trying recovery level 1 (word in wrong place -- move)

Trying a deletion

Trying recovery level 2 (word missing -- insertion)

Trying an insertion

Recovery level 2 succeeded

Parse tree selected:

```
sentence
  main_clause
    subj_phr
      noun_pron_phr
        pronoun : [je,1]
      verb_phr
        pos_or_neg_verb_cons
          pos_verb_cons
            rp_pron_list
              pron_list
                pron_list2
                  governing_verb
                    verb_or_aux_verb
                      verb : [vas,2]
        dir_obj_phr
          noun_phr
            determiner : [determiner,3]
            adj_list
              adjective : [e4tre,4]
            adj_list
              noun : [tante,5]
            adj_list
          indir_obj_phr
```

Your sentence was:

--- je vas e4tre tante . ---

I think the correct version of your sentence should read:

>>> je vais <determiner> <e4tre> tante . <<<

\*\*\*\* Comments \*\*\*\*

verb "vas" changed to "vais" :

[transitivity(i),follow\_prepos([[ ]]),can\_be\_reflex(n),person(1),plurality(s),tense(present)]

A word of type <determiner> was missing and has been inserted

"e4tre" is unknown and has been guessed to be of type adjective

"aller" : Verb is intransitive so does not expect an object

\*\*\*\* End Comments \*\*\*\*

NO. 408

Pre-parsing sentence

Trying for a legal parse

Trying recovery level 1 (word in wrong place -- move)

Trying a deletion

Trying recovery level 2 (word missing -- insertion)

Trying an insertion

Recovery level 2 succeeded

Parse tree selected:

```
sentence
  main_clause
    subj_phr
      noun_pron_phr
        pronoun : [je,1]
      verb_phr
        pos_or_neg_verb_cons
          neg_verb_cons
            neg_begin : [ne,2]
          rp_pron_list
            pron_list
              pron_list2
            governing_verb
              verb_or_aux_verb
```

verb : [vois,3]  
neg\_end : [pas,4]  
dir\_obj\_phr  
indir\_obj\_phr  
preposition : [de,5]  
noun\_phr  
determiner : [determiner,6]  
adj\_list  
noun : [pommes,7]  
adj\_list

Your sentence was:

--- je ne vois pas de pommes . ---

I think the correct version of your sentence should read:

>>> je ne vois pas de <determiner> pommes . <<<

\*\*\*\* Comments \*\*\*\*

A word of type <determiner> was missing and has been inserted

"voir" : Verb is transitive so does expect an object

\*\*\*\* End Comments \*\*\*\*

NO. 1843

Pre-parsing sentence

Trying for a legal parse

Trying recovery level 1 (word in wrong place -- move)

Trying a deletion

Recovery level 1 succeeded

Parse tree selected:

sentence  
main\_clause  
subj\_phr  
noun\_pron\_phr  
pronoun : [je,1]

```

verb_phr
  pos_or_neg_verb_cons
  neg_verb_cons
  neg_begin : [ne,2]
  rp_pron_list
  pron_list
  pron_list2
  governing_verb
  verb_or_aux_verb
  verb : [vais,3]
  neg_end : [jamais,4]
  dir_obj_phr
  indir_obj_phr
  preposition : [a2,5]
  noun_phr
  determiner : [sa,6]
  adj_list
  noun : [maison,7]
  adj_list

```

Your sentence was:

--- je ne jamais vais a2 sa maison . ---

I think the correct version of your sentence should read:

>>> je ne vais jamais a2 sa maison . <<<

\*\*\*\* Comments \*\*\*\*

neg\_end "jamais" was in the wrong place and has been moved

\*\*\*\* End Comments \*\*\*\*

NO. 2184

Pre-parsing sentence

Trying for a legal parse

Trying recovery level 1 (word in wrong place -- move)

Trying a deletion

Recovery level 1 succeeded

Parse tree selected:

```
sentence
  main_clause
    subj_phr
      noun_pron_phr
        noun_phr
          determiner : [ma,1]
          adj_list
          noun : [pe2re,2]
          adj_list
            adjective : [laid,3]
            adj_list
          verb_phr
            pos_or_neg_verb_cons
            pos_verb_cons
            rp_pron_list
            pron_list
            pron_list2
            governing_verb
            verb_or_aux_verb
            verb : [et,4]
          dir_obj_phr
          indir_obj_phr
```

Your sentence was:

--- ma pe2re et laid . ---

I think the correct version of your sentence should read:

>>> mon pe2re laid <et> . <<<

\*\*\*\* Comments \*\*\*\*

determiner "ma" changed to "mon" : [gender(m),plurality(s),apostrophe(n)]

"et" is unknown and has been guessed to be of type verb

verb "et" was in the wrong place and has been moved

\*\*\*\* End Comments \*\*\*\*

## Appendice F

### La performance d'Antidote 2000

Nous n'avons compté que les fautes morfo-syntaxiques dans les totaux ci-dessous. Étant donné qu'Antidote acceptera n'importe quelle construction grammaticalement correcte même si la phrase n'a pas de sens, nous avons choisi d'enlever du total toutes les constructions qui n'étaient pas appropriées selon le sens de la phrase mais qui pourraient se dire dans un contexte différent.

Par exemple, la phrase « Je n'aime pas aussi le rap, mais, j'aime presque toutes les genres de la musique. » ne comptera qu'une faute, la faute d'accord entre « toutes » et « genres ». Pour les autres fautes (« aussi » avec une négation, déterminant avec « genres de »), il est possible de trouver un contexte (bien sûr très rare), où cette construction est possible (« aussi » comme adverbe de phrase équivalent à « donc »; « genres de la musique dont tu parles... »).

| Phrases analysées par Antidote  | Phrases corrigées par Antidote  | Nombre de fautes corrigées | Nombre total de fautes | Notes explicatives                 |
|---|---|----------------------------|------------------------|------------------------------------|
|   |   |                            |                        |                                    |
| Moi, je joue seulement le piano.  | Moi, je joue seulement le piano.  | --                         | --                     | expression figée                   |
| J'adore tokyo.  | J'adore Tokyo.  | 1                          | 1                      |                                    |
| J'ai étudier à l'université de Colombie-Britannique aussi.                        | J'ai étudié à l'Université de Colombie-Britannique aussi.                       | 2                          | 2                      |                                    |
| Je suis très désolé, il est très dernier!   | Je suis très désolé, il est très dernier!                                       | 1                          | 1                      |                                    |
| Mais, le question deux dans les communications était très difficile.              | Mais, la question deux dans les communications était très difficile.            | 1                          | 1                      |                                    |
| Je n'aime pas aussi le rap, mais, j'aime presque toutes les genres de la musique. | Je n'aime pas aussi le rap, mais, j'aime presque tous les genres de la musique. | 1                          | 1                      |                                    |
| Je dormis beaucoup.   | Je dormis beaucoup.   | --                         | --                     | sémantique                         |
| Je n'aime pas la pluie quand c'est froid dans l'automne ou l'hiver.               | Je n'aime pas la pluie quand c'est froid dans l'automne ou l'hiver.             | 0                          | 1                      | préposition de lieu                |
| Dans decembre quand il pleut j'irai snowboarding à Whistler.                      | Dans décembre quand il pleut j'irai snowboarding à Whistler.                    | 1                          | 2                      | préposition de temps avant un mois |
| Je suis très bien aussi.  | Je suis très bien aussi.  | --                         | --                     | sémantique                         |
| Il est tres difficile   | Il est très difficile quand   | 2                          | 3                      | contraction                        |

|  |   |    |    |  |
|--|---|----|----|--|
| quand Je vais à l'université et je vais à travail dans un jour.                                | je vais à l'université et je vais à travail dans un jour.                                     |    |    |  |
| J'ai joué au baseball pour 10 ans!   | J'ai joué au baseball pour 10 ans!  | 0  | 1  | préposition  |
| Je suis allé jouer le hockey.  | Je suis allé jouer le hockey.   | -- | -- | expression figée   |
| J'ai vécu en Tulameen et Vancouver.  | J'ai vécu en Tulameen et Vancouver.   | 1  | 2  | <i>Tulameen</i> ne pas reconnu comme ville mais il aurait été corrigé si le logiciel contenait ce mot dans son dictionnaire. Possible de le rajouter |
| C'est un bon ordinateur, mais, il est un ordinateur par son nature.                            | C'est un bon ordinateur, mais, il est un ordinateur par sa nature.                            | 1  | 1  |  |
| Je suis une fanatique du chocolat.   | Je suis une fanatique du chocolat.  | -- | -- | expression figée   |
| Aujourd'hui, le temps pluvieux.  | Aujourd'hui, le temps pluvieux.   | -- | -- | incompréhensible   |
| Ma mère est chrétien et ma père et juif!   | Ma mère est chrétienne et ma pairs et juifs !   | 1  | 2  | correction fautive   |
| J'ai aussi un examen jeudi en anglais et j'espère que je vais faire bien!                      | J'ai aussi un examen jeudi en anglais et j'espère que je vais faire bien!                     | -- | -- | stylistique  |
| Oui, j'ai regardé les Canucks sur la télévision.   | Oui, j'ai regardé les Canucks sur la télévision.  | -- | -- | sémantique   |
| J'ai été Chrétien depuis 15 ans.   | J'ai été chrétien depuis 15 ans.  | 1  | 2  | temps verbal   |
| Je pense que le film c'est intéressant, mais l'intrigue n'est pas très compliquer.             | Je pense que le film c'est intéressant, mais l'intrigue n'est pas très compliquée.            | 1  | 1  |  |
| Je suis allé à le Disneyland dans le California et le Euro-Disney dans le France aussi.        | Je suis allé au Disneyland dans le California et le Euro-Disney en France aussi.              | 5  | 5  | Disneyland, California et Euro-Disney n'étaient pas reconnus.  |
| Je déteste les mid-terms, parce que ils sont très difficiles, et chaque question vaut beaucoup | Je déteste les mid-terms, parce qu'ils sont très difficiles, et chaque question vaut beaucoup | 1  | 1  |  |

|  |  |    |    |  |
|--|--|----|----|--|
| des marques.   | des marques.   |    |    |  |
| Mon sports favori sont le baseball, le football et l'hockey.   | Mon sport favori est le baseball, le football et le hockey.  | 1  | 3  | correction fautive                     |
| Je ne peux pas marcher dans un avion parce que l'allée sont très petites.  | Je ne peux pas marcher dans un avion parce que l'allée est très petite.  | 0  | 2  | correction fautive                     |
| Il cache quand il les écoute.  | Il cache quand il les écoute.  | -- | -- | sémantique                             |
| Je viens de Belo Horizonte, une ville qui est en l'intérieur du pays.  | Je viens de Belo Horizonte, une ville qui est en l'intérieur du pays.  | 0  | 1  | manque de correction                   |
| Cette année, je habite à Totem Park, mais ma famille habité dans Surrey.   | Cette année, j'habite à Totem Park, mais ma famille habitée dans Surrey.   | 3  | 4  | correction fautive, Surrey pas reconnu |
| J'habite en Californie dans l'été.   | J'habite en Californie dans l'été.   | 0  | 1  | manque de correction                   |
| J'ai habité dans la même maison pour toute ma vie.   | J'ai habité dans la même maison pour toute ma vie.   | 0  | 1  | manque de correction                   |
| J'ai noté que vous avez un accent.   | J'ai noté que vous avez un accent.   | -- | -- | stylistique                            |
| Mais je suis préjugé parce-que mes parents son roumain et ma mère a fait les gymnastiques beaucoup quand elle été étudiante.                       | Mais je suis préjugé parce-que mes parents son roumain et ma mère a fait les gymnastiques beaucoup quand elle été étudiante.                       | 0  | 4  | manque de correction                   |
| J'ai essayé au jouer le guitare mais c'est difficile.  | J'ai essayé à la jouée la guitare mais c'est difficile.  | 1  | 2  | correction fautive                     |
| Ils sont à la même école mais ils ne recontrent pas pendant le jour parce que ma mère commence à sept heures et elle est fini à midi ou 13 heures. | Ils sont à la même école mais ils ne remontrent pas pendant le jour parce que ma mère commence à sept heures et elle est fini à midi où 13 heurte. | 1  | 1  | correction fautive                     |
| Il étudier beaucoup.   | Ils étudient beaucoup.   | 1  | 1  | correction fautive                     |
| Je voudrais voyager à france.  | Je voudrais voyager en France.   | 2  | 2  |  |
| Pendant ma première année, je pris un cours dans l'anthropologie aussi.  | Pendant ma première année, je pris un cours dans l'anthropologie aussi.  | -- | -- | sémantique                             |
| Mes parents son roumaine et je parle la langue aussi (avec un  | Mes parents sont roumains et je parle la langue aussi (avec un   | 2  | 2  |  |

|  |  |    |    |  |
|--|--|----|----|--|
| accent fort)!  | accent fort)!  |    |    |  |
| Je suis bien, mais je suis fatigué parce que hier après l'école j'ai travaillé pour huit heures.                                     | Je suis bien, mais je suis fatigué parce qu'hier après l'école j'ai travaillé pour huit heures.                                      | 2  | 3  | manque de correction                         |
| Je pense que ma professeure de français ressemble au Florence!   | Je pense que ma professeure de français ressemble à la Florence!   | -- | -- | sémantique                                   |
| Je veut parler un peu de stationnement à UBC.  | Je veux parler un peu de stationnement à UBC.  | 1  | 1  |  |
| Je suis dans mon premier année de Université a l'Université de Colombie-Britannique.   | Je suis dans ma première année d'Université à l'Université de Colombie-Britannique.  | 3  | 3  |  |
| Je n'aime pas la date de l'examen français aussi.  | Je n'aime pas la date de l'examen français aussi.  | -- | -- | sémantique                                   |
| Le vendredi demain je vais aller à une fête d'hallowe'en avec mes amis.  | Le vendredi demain je vais aller à une fête d'hallowe'en avec mes amis.  | -- | -- | sémantique                                   |
| Calgary est une petite université et le recrutement est plus bon que l'université de colombie-britannique.                           | Calgary est une petite université et le recrutement est meilleur que l'Université de Colombie britannique.                           | 2  | 4  | 1 manque de correction, 1 correction fautive |
| Tout de suite je ne connais pas qoui je ferai.   | Tout de suite je ne connais pas quoi je ferai.   | 1  | 1  |  |
| Je ne joue pas la guitare, mais j'aimerai apprendre.   | Je ne joue pas la guitare, mais j'aimerai apprendre.   | -- | -- |  |
| Ils sont irlandais et allemand.  | Ils sont irlandais et allemands.   | 1  | 1  |  |
| Et, si ton amie veux aller au Brésil, elle peut aller à Rio de Janeiro ou le nord-ouest du pays, qu'il y a des plages merveilleuses. | Et, si ton amie veut aller au Brésil, elle peut aller à Rio de Janeiro ou le nord-ouest du pays, qu'il y a des plages merveilleuses. | 1  | 2  | répétition de la préposition de lieu         |
|  | Total  | 42 | 66 | 64%  |

## **Bibliographie**

### Sources primaires

Linger (Language INdependent Grammatical Error Reporter). Logiciel. Exeter: Jonathan Barchan, 1988. IBM, Windows.

Antidote Version 5.1. Logiciel. Montréal: Druide Informatique, 2000. IBM, Windows.

### Sources secondaires

Barchan, J. « Language INdependent Grammatical Error Reporter. » M.Phil. Thesis. University of Exeter, 1987.

Barchan, J. & Wusteman, J. « A Prolog-Based Tool for Grammar Analysis of Western European Languages. » (1988): Digital Media Research Centre, University of the West of England, Bristol.

<<http://www.media.uwe.ac.uk/media/masoud/author>>

Bolt, P. & Yazdani, M. « The Evolution of a Grammar-Checking Program: LINGER to ISCA.» (1997): Digital Media Research Centre, University of the West of England, Bristol.

<<http://www.media.uwe.ac.uk/media/masoud/author>>

Burston, Jack. « Antidote '98. » CALICO Software Review (1999): 1-13.

<http://astro.temple.edu/~jburston/CALICO/review/antidote00.htm>.

Corder, S.P. "Idiosyncratic Dialects and Error Analysis." International Review of Applied Linguistics in Language Teaching 9.2 (1971): 147-159.

----- "The Significance of Learner's Errors." International Review of Applied Linguistics in Language Teaching 5.4 (1967): 161-170.

James, Carl. Errors in Language Learning and Use: Exploring Error Analysis. Applied Linguistics and Language Study. Ed. C.N. Candlin. London: Addison Wesley Longman Limited, 1998.

Lennon, Paul. "Error: Some Problems of Definition, Identification, and Distinction." Applied Linguistics 12.2 (1991): 180-196.

Lyons, J. Introduction to Theoretical Linguistics. London: Cambridge UP, 1968.

Mogilveski, Eugene. "Le Correcteur 101: A Comparative Evaluation of 2.2 versus 3.5 Pro." 1-8.

CALICO Software Review (1998):

<http://astro.temple.edu/~jburston/CALICO/review/c10135.htm>.

Pereira, F. & Warren, D. « Definite Clause Grammar for Language Analysis - A Survey of the Formalism and a Comparison with Augmented Transition Networks. » Artificial Intelligence 13 (1980): 231-278.

Selinker, L. « Language Transfer ». General Linguistics 9.1 (1969): 67-92

Richards, Jack C. « A Non-Contrastive Approach to Error Analysis. » Error Analysis: Perspective on Second Language Acquisition. Ed. Jack C. Richards. London: Longman Group Limited, 1974. 172-188.

Uren, Jo & Yazdani, Masoud. « Spanish LINGER. » (1998): Digital Media Research Centre, University of the West of England, Bristol.

<<http://www.media.uwe.ac.uk/media/masoud/author>>