uOttawa

L'Université canadienne
Canada's university

FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTDOCTORALES

uOttawa

L'Université canadienne
Canada's university

FACULTY OF GRADUATE AND POSTDOCTORAL STUDIES

Matthieu Hermet

AUTEUR DE LA THÈSE / AUTHOR OF THESIS

Ph.D. (Computer Science)

GRADE / DEGREE

School of Information Technology and Engineering

FACULTÉ, ÉCOLE, DÉPARTEMENT / FACULTY, SCHOOL, DEPARTMENT

Automated Analysis of French-as-a-second-language Student's Free-text Answers for Computer Assisted Assessment

TITRE DE LA THÈSE / TITLE OF THESIS

S. Szpakowicz

DIRECTEUR (DIRECTRICE) DE LA THÈSE / THESIS SUPERVISOR

CO-DIRECTEUR (CO-DIRECTRICE) DE LA THÈSE / THESIS CO-SUPERVISOR

EXAMINATEURS (EXAMINATRICES) DE LA THÈSE / THESIS EXAMINERS

J-P. Corriveau

T. Heift

D. Inkpen

N. Japkowicz

Gary W. Slater

Le Doyen de la Faculté des études supérieures et postdoctorales / Dean of the Faculty of Graduate and Postdoctoral Studies

Automated Analysis of French-as-a-Second-Language Student's Free-Text Answers for Computer Assisted Assessment

by

Matthieu Hermet

Thesis submitted to the

Faculty of Graduate and Postdoctoral Studies

In partial fulfillment of the requirements

For the degree of Doctor of Philosophy in Computer Science

Ottawa-Carleton Institute for Computer Science

School of Information Technology and Engineering

University of Ottawa

Ottawa, Ontario, Canada

Your file Votre référence
ISBN: 978-0-494-61372-6
Our file Notre référence
ISBN: 978-0-494-61372-6

Abstract

This project is a proof-of-concept. It aims to demonstrate the feasibility of an approach to Computer-Assisted Assessment of free-text material in the domain of Computer-Assisted Language Learning (CALL). The underlying theory places the project within given pedagogic constraints, just as a Language Tutoring System should. The constraints call for the project to be addressed to the intermediate-advanced student of French-as-a-Second-Language, and be oriented toward the autonomous enhancement of text comprehension skills, following a previous project in CALL at the University of Ottawa, DidaLect. This type of learning activity requires a student to answer questions related to informative texts.

The goal of this work has been to build a framework to assess the correctness of the student's sentence's grammar on the one hand, and on the other hand to ensure that the answer's content matches the reference. In order to achieve this, the research has gone in two different directions.

First, we used Natural-Language-Processing (NLP) to find means of language error detection and correction (form), as well as means for semantic comparison (content). Content comparison amounts to performing deep analysis of the student's answer in order to guarantee that no material in the student's answer is irrelevant to the actual answer. We used a symbolic approach to the problem, because statistical methods can only provide approximations of content similarity, which is dangerous in a CALL context because the student can be error-prone. The fact that this work is the first of its kind, at least in what concerns Text Comprehension and French-as-a-Second-Language, has been another reason to opt for symbolic processing: in the absence of any comparable system, a symbolic approach might constitute a better baseline, to be challenged in the future by statistical methods. Finally, error detection is syntax-based in language technologies. As well, a symbolic approach permits to use the same structure for the assessment of form and content.

Second, we worked in the direction of didactics to frame the work within relevant theoretical grounds in terms of the relation from questions to text, especially in order to limit the impact of the knowledge gap between machine and humans (students) – a general consequence of the work-in-progress state of semantic analysis in NLP. The questions are controlled through a formal categorization that restricts the scope of the questions to answering material actually present in the text – no world-knowledge is *a priori* needed.

The system has been tested on a set of 273 student's answers gathered in class. The evaluation gave a result of 62% of answers correctly assessed as correct or incorrect. Due to the conservativeness of the system, precision on the assessment of correct answers is 100%, which satisfies the requirements of Second-Language Learning and CALL.

The implemented program is a contribution in itself because no comparable system exists, while there is a real demand from the world of CALL for such a tool. The project also makes a contribution to NLP through a new approach to paraphrase recognition. This uses previous work in computational linguistics (the Meaning-Text Theory) to provide a rule-based model of syntactic paraphrase which, although limited to syntax, actually constitutes the only generic model of paraphrase to our knowledge, and should be easily adapted to other languages.

Acknowledgment

Partial support for the project came from the Social Sciences and Humanities Research Council of Canada and from the Natural Sciences and Engineering Research Council of Canada. Also, I wish to thank Professor Lise Duquette from the Second Language Institute at the University of Ottawa for her help in annotating the students' answers, and especially Dr. Stan Spakowicz, the supervisor of this work, for his countless revisions and judicious advice.

Table of Contents

List of Tables and Figures

A Glossary of Terms

LT: Learning Technology – a generic term to describe computer realizations in the field of learning.

(I)TS/(I)LTS: (Intelligent) (Language) Tutoring Systems

CAL/CALL: Computer-Assisted (Language) Learning

CAA: Computer-Assisted Assessment

CBA: Computer-Based Assessment – somehow outdated. Encompasses CAA in the days of punch-cards.

SBA: Screen-Based Assessment – in opposition to other assessment tactics, such as punched-cards Optical Machine Reading assessment. Here, the user sits at and works with a computer.

CBT: Computer-Based Testing – A formal examination where computers replace paper-and-pencil.

VLE: Virtual Learning Environment – The digital environment which can distribute ITS and CAA systems, and branch them to other components, mainly communication devices downstream, to manage access rights, and authoring devices upstream to create the exercises.

Formative Assessment: A mode of assessment which aims at evaluating student's material with respect to a given paradigm of correctness and inform the student on the nature of his errors. This presupposes the possibility of retaking the exercises.

Summative Assessment: A mode of assessment which aims at marking student's material – typically in the context of a formal examination. No formal need to provide the student with more than his mark.

Diagnosis: evaluation rather than assessment – it encompasses the activity of evaluation of the correctness of language outside the domain of CAL. The goal is the same, but there is no need of feedback for user. But Diagnosis systems should come with strong error correcting solutions. Diagnosis can be applied to Machine Translation for instance (Chen & Tokuda, 1999)

Objective tutoring/assessment/questions(...): the order of exercise submission during a tutoring session is a function of their order of recording in system, and therefore also remains the same from one session to another. Static. This paradigm does not take adaptability into account, the student's performance has no impact on the succession of drills.

Adaptive tutoring/assessment/questions(...): the order of exercises submission during a tutoring session is a function of a student's results on previous exercises. Dynamic. Adaptability is thought to bring intelligence in ITS.

SLL/SLA: Second Language Learning/Acquisition – takes shape as **ESL** (English as a Second Language), **FSL** (French), and so on.

Open Items: the answers to exercises are in open-text (or free text) form, like a sentence or a full essay.

Closed Items: the answers to exercises are in closed-text form, like Multiple-Choice-Questions' slots or fill-in-the-blank response elements.

Error Analysis: qualitative analysis of the language errors occurring in a given textual content: usually a set of students' written or oral productions.

Error Diagnosis: an automatic procedure, or set of procedures, to detect, identify, and possibly correct the language errors present in a text or segment automatically processed.

Error Grammar: A resource categorizing error types and errors, to be integrated in an automatic procedure for diagnosis, whether through means of heuristics or parser-based methods.

Part 1.

Theory and Objectives

Chapter I. Introduction

This thesis presents a project in the domain of Computer-Assisted-Language-Learning (CALL). The goal of the project is to develop a proof of concept for the automatic assessment of students' free-text answers to text comprehension questions for Second-Language-Learning (SLL) purpose – specifically in the context of French-as-a-Second-Language (FSL). Simply put, the objective is to build a program to evaluate the correctness of students' answers to comprehension questions based on short texts. As we will see, this is entirely new, especially in the context of FSL, and the program should constitute a serious baseline for an activity considered by teachers and members of the CALL community as of primary interest to education.

Although the project is not intrinsically in the domain of Natural-Language-Processing (NLP), it is an NLP-based application in that the solutions employed to solve the problem all come from NLP. For this reason, the project may be better characterized as being centered on Intelligent-CALL, or Intelligent-Tutoring-Systems (ITS). While a good part of the work has had to do with the selection of the NLP means that best suits the task (see chapter IV and V), due to its highly inter-disciplinary nature, the project has also required an investigation from the side of didactics, particularly in what concerns Text Comprehension and question asking based on texts (chapter III). The program resulting from this work has been evaluated with specific FSL student material; it represents however a research study and is not intended for real-world deployment.

I.1. The Problem and Challenges

The kind of exercise for which this work intends to provide an assessment solution is rather simple and well understood from the side of education. A student is asked to read a text and has to answer questions based on it, as shown in the example below:

(1) REF: *Le nez est un organe très spécifique chez l'homme, et il nous inspire pas mal de méfiance. Y toucher dans le cadre d'une rhinoplastie, c'est s'attaquer au centre du visage et à l'un des fondements de l'identité. Bien souvent, ce type de demande masque des problèmes psychologiques graves, de l'ordre de la schizophrénie.*

(The nose is a very specific organ in men, and it arouses suspicion. To touch it through rhinoplasty is to tackle the middle of the face and one of the foundations of identity. Very often, this kind of demand conceals serious psychological problems, of the order of schizophrenia.)

Q: *Quel genre de problème une rhinoplastie peut-elle provoquer ?*

(What kind of problems can a rhinoplasty create ?)

S: *Des problèmes de l'ordre de la schizophrénie.*

(Problems of the order of schizophrenia)

The scientific problem in automation consists in providing the means to analyze the correctness of form and content, to deal with cases where the reformulation from reference to answer is more complex than in the example 1, which shows little reformulation from answer to reference. Form indicates the value of a sentence with respect to the grammar and orthography of the language. Content indicates the value of a sentence with respect to meaning: here, the answer to the question which comes as a referential segment in the original text. Furthermore, in what concerns form, incorrectness should be identified against a category of errors so as to be able to provide the learner with feedback, informative enough to help her correcting her input. This point is especially important in the context of SLL, where students have to master the specifics of a foreign language.

Natural Language Processing

In both aspects, analysis of form and content, NLP can provide solutions. Already existing solutions in NLP are however more advanced in what concerns the evaluation of content equivalence, or *reformulation*, than what concerns form. On the other hand, language error detection and correction is such a recent trend in NLP that it does not really have a well-understood acronym yet.

In NLP, the evaluation of content similarity is studied in various areas, more or less linked and aware of each other, as we will show in chapter IV. Summed-up, this problem can be dealt with through the comparison of semantics, or content strictly speaking – this is the object of semantic similarity. It can be otherwise treated as a sentence-level problem, which is the object of paraphrase detection. In practice, the techniques used in NLP by both approaches do not vary greatly, while the two notions are quite different from the perspective of linguistics. A characteristic of the way in which the problem of semantic equivalence is solved in semantic similarity or in paraphrase detection is *approximation*: in all cases two segments are compared by means of lexical overlap (even grammatical in some cases) more or less sophisticated, and the ratio of shared contents determines equivalence. We believe this statistical approach is too permissive in the context of SLL, where we cannot ignore part of an answer in the equivalence assessment process: this is just too dangerous when students have not yet mastered a second language. We will argue for that and show how a parser-based symbolic approach to the problem might be more careful in this context, as it draws on a deep analysis of the language material, necessary to the exercise. Ultimately, this has led us to design a reformulation assessment procedure based on a new paraphrase recognition method which borrows from existing linguistic theory, and which should be of interest to the whole NLP community.

Didactics

Just as NLP provides means to help solve the semantic equivalence problem, didactics should assist the question-making strategy, as text comprehension has been considered in education studies for decades. A problem in automatic free-text answers assessment concerns the impact of world-knowledge on the semantic deviance from reference to answer – depending on the scope of the question in relation to the text, a student can be led to express meaning drawn from his world-knowledge rather than from what is actually in the text. We will show how didactics can

help to shape a computerized application in text comprehension, as well as to bridge the knowledge gap between learner and machine.

DidaLect

This project originates from a previous joint project between University of Ottawa's Second Language Institute, School of Psychology and School of Information Technology and Engineering, called DidaLect (from *Didactique de la Lecture* – Balcom, Copeck & Szpakowicz, 2006). The project's objective was to design an autonomous adaptive tutorial aimed at supporting vocabulary acquisition and enhancing comprehension for learners of French as a Second Language. Briefly, the tutorial comprised sets of categorized texts upon which categorized multiple-choice questions were asked of students. Adaptability was encompassed through continuing evaluation of the student's answers, which would trigger display of texts of varying difficulty, according to the level of the student.

Before the DidaLect project was terminated this further project has been initiated to investigate the feasability of automatically analyzing and assessing in form and content free-text student's one-sentence inputs as answers to text comprehension questions. The motivation for this was to increase the variety of possible inputs, and so to increase the pedagogical value of the tutoring system, as well as to go into a research trend considered of direct interest by the communities of Education and CALL (see chapter II and III). Note that this project follows the inter-disciplinary nature of DidaLect, as it briefly showed in the trade-off summed-up above between didactics and NLP.

I.2. Characteristics of the Approach

In this section, some desirable characteristics of our solution are presented. On the one hand, these characteristics have to do with the kind of material necessary to provide answering references for the questions associated with the texts. On the other hand, they concern a realistic definition of a correct answer with respect to automation possibilities.

I.2.1. Pre-encoded material

It is fundamental to carefully evaluate the necessary information, or meta-information, that the computer has at its disposal to meet its assessment goal. This will lead to a natural selection of the computational means. Plus, there is a trade-off between the portability of a system – which asks for pre-encoded specific material to be minimal – and its performance – which asks for it to be maximal.

Our specific problem is to assess one-sentence free-text answers to categorized questions based on categorized texts, not just any question over any text. We will come back to the nature of those texts and questions in chapter III. But this means that the condition on the encoding of knowledge material asks the designer the following question: what amount and nature of information should appear in the questions' response that is not to be found in the reference text? And this nature and amount is what is to be placed within the knowledge base, making it formally: the set difference between information carried in texts and information needed to answer the given questions based on them.

Once this has been determined, a further question asks for the most economical means to encode the information in the system. If the information is known in advance, then is this amount limited or potentially infinite? Assessing essays, for instance, supposes world knowledge information to be potentially infinite and would call for ML techniques, to cover a maximum[1]. But if the amount of information necessary to "understand" a sentence is limited, then symbolic encoding of the information might be enough.

Note that this amount of information is what is needed to properly capture correctness in answers to a question. Thus there is no need to encode all that can constitute neither right or wrong material: all that is not an answer to the given question. Is what is needed to have a correct answer present in the current answer? What is not needed might as well be left out of any knowledge base, and as long as this superfluous material does not come into contradiction with

[1] This is theoretical. Actually, in practice, it's the contrary… Assessment of essays uses fewer resources than assessment of short answers. See chapter II.

what is correct, it is simply to be set aside (if *yes* is the correct answer to a question, ignore anything else that is to appear in the question as long as it is not *no*).

In our case, what is needed to capture correctness in an answer is limited, and known in advance. This is because the types of questions associated with the texts are limited and well constrained, as we shall see when defining types of texts and questions. Also, this necessary and sufficient amount of information is contained in the texts, at least in what concerns sense. Therefore, we have chosen to concentrate on NLP solutions to perform assessment (Borin 2002a, Nerbonne 2002). A knowledge base is to be used. It records rules and general facts of language, of generic nature, and not domain-oriented.

Current CAA systems usually require a lecturer to "author" exercises, possibly by creating model answers and highlighting the parts of interest to marking/assessment through structuring of these models. Therefore, the system works on the material encoded by lecturers to perform assessment of student material. That is for reality. In our perspective, the idea is to go as far as possible towards automation, but the fact that full automation might be out of reach would not rule out our procedure. Instead it would help to define the points in the procedure at which human expertise is required. Concretely, in the design of such a project, this implies making room for lecturers/trainees to structure the reference material.

I.2.2. Reformulation

CAA techniques have been around for a long time. Initially, they have been employed to assess binary questions which did not ask for any further world-knowledge in the validation process (think of punched-cards validated through Optical Machine Recognition). As the first successes drew the impulse towards greater assessment challenges, the discipline was soon confronted with a disturbing reality: simple techniques do work, but not always (simple techniques in assessing free-text material are illustrated for example by word-match or Finite State Automaton-like chains of lexical items slots). In particular, these simple solutions tend to work well for lazy students, but less well for both top and bottom-of-the-range students, for both the most audacious students and those who need attention the most. CALL and CAA certainly encourage lazy answering strategies.

In what deals with free-text answering in general in the fields of CALL and CAA, our approach is thought to serve the supplementary interest of tackling the issue of linguistic reformulation, in addition to the interest in topically assessing answers to given text-based questions for intermediary-advanced FSL students. For having now spent several years on this issue, this researcher has had the opportunity to see that existing CAA systems deal indirectly with the issue of linguistic reformulation – form is analyzed through sets of possible answers rather than through general language properties.

In our case, as long as the answer to a question is not metaphorical or elliptic, it should fall into the system's assessment capacity; and the system should always be able to come with decent evaluation as long as reformulation occurs at a syntactic level, and not semantic. The resources employed to detect reformulation should be general. The assumed independence of the system supposes no reference material related to the answers, and the assessment cannot be supported by question-specific semantic knowledge. We believe our symbolic approach is pertinent to capturing the specifics of linguistics here as corpus-based ML techniques would be to capture the specifics of domain semantics.

I.2.3. "Bad Faith" answers

Finally, we should say a word on "bad faith" answers. The discipline of CAA is so hazardous in dealing with a complex problem – literally, to understand human language – that it has formalized a notion otherwise very ill-suited to science: bad faith. Practically, students know very well the limits of computation, and computers are very easy to fool. This comes typically in the form of a facetious student who purposely produces a biased answer in order to push the program to its limits (Powers, 2001).

As language lecturers have observed this since long ago, they turned it into a notion very comforting to the art of CAA: that, conversely, students should be aware of the limits of the machine, and should therefore behave accordingly, and appropriately (Bull & McKenna, 2004, p95). This means that it is more-or-less admissible that CAA systems can be imperfect, and that, particularly in formative assessment, a certain degree of tolerance is expected on behalf of the student.

Concretely, such systems are granted the right to capitulate – a direct consequence of dealing with a complex issue under the fire of real-world application.

The possibility of bad faith answers, as well as the certainty of reformulated answers introduce a risk that automatic assessment could fail. Actually, in such applications results cannot be perfect. Various approaches can have diverging effects on the assessment: some can have a bias on correctness and others a bias on incorrectness. In a context such as French as a Second Language, or Second Language Learning in general, the bias should be on the assessment of correctness: to make sure that what is assessed as correct is really correct. There are various reasons for that and they will be presented and discussed.

I.3. Novelty and Contribution

As we briefly mentioned above, the project is placed in a quite precise area of research: that of assessment of free-text answers in a (Language) Tutoring System. This area is at the crossroads between Computer Science and Education (on the difficulty to establish standards, see Borin 2002b), between Computer Assisted Language Learning and Didactics, and between Natural Language Processing and Machine Learning. It also falls into Computer Assisted Assessment as applied in Intelligent Computer-Assisted-Language-Learning.

A consequence of this entanglement of disciplines and interests is that the specificity of the assessment task dictates the design of the system to perform it. Therefore, a new application of automatic assessment of learner material should produce novel solutions, at least different from what has been done in the field. Summative assessment of physics essays rarely calls for the same computational methods and resources as formative assessment of language skills – the resulting systems would be two instances of CAA or ICALL solutions, but would probably have few computational methods in common. Little exists for now in ICALL and this project is entirely new.

Being in Language Learning and in the assessment of semantic equivalence, the subject is however generic enough for a solution to be more than a case application and constitutes a contribution to CALL in general. To sum-up, the project described by this thesis is entirely new:

- In its scope:
    - questions to be answered for text comprehension
    - without training set of referential answers
- In its means:
    - using didactics principles to limit the answering semantic space
    - using a syntactical approach to assess reformulation

As well, the project makes a number of contributions:

- In NLP: use of a syntactic reformulation model based on linguistic theory to implement paraphrase recognition. It represents a symbolic approach to the problem, usually solved by a quantitative approach.
- In CALL:
    - The system itself represents a contribution because nothing equivalent exists.
    - The results of the study provide ground for further analysis of the problem raised by this type of exercise.

I.4. Overview of the Thesis

This thesis is divided into two parts. The two parts consist of chapters of unequal length – this is a consequence of the importance or simplicity of their contents in the project in general.

The first part presents the framework of the project, as well as related work. Chapter II makes a point on the domain of SLL in its relation to automation and presents some comparable projects or tools, as well as the target audience. In chapter III, we first sum-up the theoretical views around the question of language learning with respect to the didactics of text comprehension governing the project, and then we define the nature of correctness and incorrectness in this type of activity in contents and language, as well as what is to be expected concerning language

errors. Chapter IV presents an overview of the problem from the perspective of Computational Linguistics, as well as an overview of paraphrase as conceived by Computational and formal Linguistics and solutions in NLP. This chapter ends with a justification of our choice to use a syntactical approach to the problem of free-text answers assessment.

The second part deals with the implementation of the project. Chapter V presents the NLP solutions implemented in our framework, beginning with parsers we have been testing and the reasons of our choice and the post-processing applied to the output of the parser, necessary to its exploitation. The chapter follows with the lexical resources selected for this project and how they are used, with Error Diagnosis, in general and as applied in this project, and paraphrase model and rules to provide the system with paraphrase recognition capabilities. Chapter VI presents the results as well as the methodology of evaluation. Chapter VII presents the conclusion to this study.

Summing up, this project aims at implementing a French-as-a-Second-Language (FSL) system for the assessment of free-text answers to restricted questions, for which reference answers are only present in the displayed texts. The pre-encoded knowledge should be as limited as possible and carry generic language information only. In particular, the system should be independent of any training material, in the form of pre-existing model answers. The focus of the computation should be on the linguistic reformulation possibilities of each sentence; this to be able to deal with the answering material of the best students, as well as with the insufficiencies of weaker ones. The assessment procedure should favour the precision of correct answers over recall. It must include a strong language error correction component. Eventually, the project should also represent a solution to deal with the larger problem of linguistic reformulation in general. There are actually several projects going on in the field of CAA/CALL (see Chapter II).We know of at least two other CALL projects dedicated to the detection of syntactic errors in French tutorials, but these are now abandoned (FreeText, L'Haire & Vandeventer-Faltin, 2003a) or dormant (Burston 2001). We also know of projects designed for the evaluation of syntactic skills, independently of any specific content or subject. For instance, Giouroglou and colleagues propose a symbolic system designed for the mastering of English structural transformations, through user production of open answers, but this works on pre-defined sentences to be

transformed following deterministic patterns such as conversion from the active voice to the passive voice (Giouroglou, 2004). More ambitiously, the project Direkt Profil (Granfeldt & al., 2006) aims at evaluating FSL student's level of proficiency through input of free-text and evaluation of grammar quality and lexicon variety based on ML techniques, but this is similarly working outside of any subject or content specifics. These two last projects have more to do with the evaluation of user proficiency than with CAA, but they represent parent projects in the chain of procedures needed in CALL in general. Some of these projects and others are presented in more detail in chapter II.

Chapter II. Computer-Assisted-Language-Learning and Computer-Assisted Assessment

This chapter gives an overview of the problem with respect to CALL. First we present the domain in itself together with its parent domains, such as CAA and Intelligent-Tutoring-Systems. Then, in the second section, due to our special interest to CAA, we present our competition: the main systems that implements CAA solutions – within CALL, but also sometimes independently from it. The last section describes the audiences at which a free-text answers assessment tool is addressed and their degree of involvement.

II.1. Fields of Interest and Terminology

In this section, we give a review of the nomenclature used in the field of Learning Technology. The term *Learning Technology* is probably the most generic term to depict the domain in general. It has been adopted by the ADLI (Advanced Distributed Learning Initiative) of the US DoD (Department of Defence), which acts as a think-tank around the issue.

II.1.1. CALL

CALL – Computer-Assisted Language Learning (Hubbard & Levy, 2005, Chapelle, 2001 for a reference, and Davies, 2007, for examples of CALL activity through history) – is a concept covering a vast area of applications and technical realizations. Its object is to provide greater *accessibility* and *adaptability* in the field of language learning, whose subject ranges from

Second Language Learning (SLL) to domain lexicon acquisition, for instance in such fields as biology or finance (see Michaud & McCoy, 2000, on adaptability, or Michaud & al., 2001, on accessibility). Considering SLL, the research interest is to recover all matters of linguistics, ie. lexicology, as expressed by morphology or semantics, syntax, and even pragmatics.

A number of parallel concepts surround the CALL field of studies (for an overview of the domain, Hubbard 2005, or Sturgeon 2003) – typically, and up-to-date, an ITS (Intelligent Tutoring System), including a CAA component (Computer Assisted Assessment) embedded in a VLE (Virtual Learning Environment) all participate in the realization of an eLearning system. The latter concept, eLearning, encompasses a notion as vast as that of CALL – both can overlap depending on the exact application. Such tool that is directed at finance tutoring, for example (Vitanova, 2004, Boytcheva & al., 2001), and accessible on-line, is first understood as an eLearning System embedding a CALL module, for instance dedicated to the sub-task of domain lexicon acquisition, while a SLL tutoring tool accessible on-line and dedicated to the teaching of a foreign language is first understood as a CALL system realized as an eLearning system. That is the reality of the present nomenclature. Eventually, all concepts are subsumed by the notion of CAL – Computer-Assisted Learning, rarely used. Here, it is interesting to remark that this notion of CAL as a field of research has been at least partly replaced by the notion of CALL, due to the fact that the present trend of research is directed at embedding "intelligence" in the systems, through the means of NLP (Natural Language Processing) modules directed at the processing of language interaction between the student and the machine (Borin, 2002a, Nerbonne, 2003 – for applications to spoken language, see Hincks 2003); this interaction processing has in itself little to do with CALL, but as CALL is the natural recipient for the testing of such intelligent module (Nerbonne, 2002), it can be said that at least for the present time, CALL is the twin of all CAL systems in the surge of CAL to cope with artificial intelligence and NLP (Gamper 2002, in Nakov 2004 – Holland, Kaplan & Sams 1995). This notion of CAL has been taken over by that of Learning Technology.

II.1.1.1. ICALL

ICALL, or *Intelligent* CALL, brings error-specific (or adaptive) feedback capacities to CALL through the means of AI. The feedback should be tailored to the user's personal errors. As well,

the ordering of the exercises should adapt to the level of a student progress, following the principles of ITS (see section II.1.2 next). An application can be incorporated within a real-world ICALL system when this system has the means to send adequate feedback on the types of errors to be possibly committed in the context of this exercise. Nerbonne (2003) has pointed out the gap between the objectives of ICALL and the reality of processing capacities, especially in NLP. In the state of things, ICALL can barely do more than to correct material based on surface forms and exact match. The research objective in ICALL is therefore to increase response variation possibilities in the exercises – that is the number of possible answers to solve an exercise. A constraint to this is the limitation of available models of correctness, which puts a condition on a system's assessment capacities. Note however that ICALL is more involved in the qualitative evaluation of results than in marking learner's production, so that the field is open to some experiments and tolerates a degree of uncertainty in the performance.

The degree of response variation is also influenced by the assessment bias. In the current state of NLP research, choices have to be made as to whether to use solutions favouring the assessment of the greatest possible number of correct answers, at the cost of precision, or solutions to maximize the detection of incorrectness, and thus reduce recall of good answers in the assessment process. The goal of the tutoring activity should eventually decide on that.

This project proposes to test a compromise in the way of enhancing response variation (free-text) while reducing the space of possible answers. It is therefore involved in ICALL. The context of reading comprehension is adequate for this, as the knowledge space can be reduced to a manageable size with the help of some support resource to handle aspects of language reformulation, like knowledge bases or language models of paraphrase. Bailey (2008) has reached the same conclusion as we did at about the same time. Our approach is conservative, as we will explain, due to the fact that correctness should not be approximately evaluated in a context of Second-Language-Learning.

II.1.1.2. CALL and Software Engineering

The Software Engineering community has taken interest in CALL (see for instance Farmer & al., 2004). In a special issue of the CALL journal, S. Cushion (2006) takes on J. Colpaert (2004) to

advocate a shift in paradigm from the traditional SE "business" model to a new "research-based, research-oriented" model. The novelty of this model is that it distances itself from a strictly commercial view of software production, making space for a timely interaction between an evaluation phase (occurring at the time of the software's general use, separated from the testing phase) and the available technology, conditioned by the theory. The process of building software then appears as a concentric cycle of ever-increasing possibilities and efficiency around the notion of CALL, understood as an open frontier. Farmer and Gruba (2006) advocate an agile approach to SE's Model Development Architecture for CALL, which fits Cushion's views in that Cushion suggests a merge between the prototyping and development/implementation phase – parallel to the substitution of evaluation to testing. In all cases, the main characteristic of an SE approach to CALL is the emphasis on the involvement of the end-user. However, the demonstrations address a CALL researchers' and teachers' audience, and are meant to convince.

We will now examine briefly, field by field, the notions presented above for a clear understanding of the technology and concepts involved in CALL, or in which CALL can get involved.

II.1.2. I(L)TS

The design of computational systems to support foreign language instruction needs to be grounded in what we know about human learning, language processing, and human-computer interaction. Principles derived from these fields need to be tested and quantified in the context of specific tutoring systems.

MacWhinney (1995, p. 324)

Intelligent (Learning) Tutoring Systems have first to do with Tutoring Systems, which are perhaps as old as computer science itself (Page, 1966, Hartley & Sleeman, 1973). Initially, the system functions as a repository of information rather than a true pedagogical tool; and even

when the system offers a possibility for interaction through adaptability according to the quality of student input, the input material is either processed through a pre-defined answer model matching or straight validation by an instructor (see Valenti, 2003b, for an overview).

Therefore, "intelligence" is included to cope with the relativity of the notion of correctness – a notion particularly vivid in language matters; a tutoring tool concerned with the teaching of formal sciences could do without, except in the circumstance where "answering" is conveyed through means of language input, in which case validation has to pass the barrier of language "understanding", only achieved through means of Natural Language Processing. Thus the search for intelligence in tutoring systems is equivalent to linguistic analysis – the deeper, the more intelligent.

A second justification for working out Artificial Intelligence solutions in tutoring systems is related to the problem of feedback to the student. A frequent reproach addressed by the students to such tutoring systems is that when committing an error, the user is interested in knowing "why" rather than "how". Furthermore, such a wrong answer can be understood as partially or completely wrong – a typical case in language where an utterance can be completely off-topic or merely flawed with one or several circumstantial errors. Thus a concern of embedding intelligence is to enhance adaptability in the production of feedback as well as in the selection of exercises with respect to a student's progress (See Perèz, 2005).

This concern for adaptability, both for erroneous input tolerance and feedback production, directly arises from theories of learning, nurtured by research on psychology and cognition. These two latter subjects, as concerned with teaching and learning, state that learning has to do with understanding, which has itself to do with the construction of structured meaning (Kintsch & Van Dijk, 1978). Directly transposed to language understanding, comprehension is understood as a multi-level process, involving the proper mastering of the various levels of language: a micro-level linking morphosyntax to words, and a macro-level linking words to propositions (Barrière & Duquette, 2002). A further concern of cognition applied to language acquisition, and perhaps learning in general, deals with the nature of the users themselves, who all are different human beings, each one with their own background knowledge and level of competence. (for an overview of the question of CALL in ILTS, and ILTS in CALL, see Heift, 2001, 2003a, 2003b)

II.1.3. CAA

Computer-Assisted-Assessment (CAA) applies to so-called VLE (Virtual Learning Environments) and more specifically to tutoring systems. It originates from the notion, and practice, of CBT (Computer-Based Testing – such as having a student perform tests on slot cards to be assessed automatically through OMR – see Bull & McKenna, 2004, p. 104) as well as from the notion of assessment itself (see Roos, 2002 for an overview). It actually includes two different fields: one aiming at organizing a theory over the field, built upon educational and cognitive science and under the influence of educational researchers, and a second one directly focused on results coming from implemented experimental tools, which involves researchers from computer science, often seen under the prism of NLP and Machine Learning. Usually these two tendencies are merged together (see Rosé et al., 2003b, for a realization), especially as the impulse to create experimental tools historically comes from the educational side (Sturgeon, 2003). Today, one can observe an inverse tendency for CS researchers to come into the field because it is perceived as an interesting challenge and a natural by-product in the world of Information Technology (Chen & Tokuda, 1999, in the context of ILTS).

Briefly, theoretically speaking, the activity of learning, which puts conditions on CAA, is perceived as a development of certain cognitive abilities, independently of the studied subject: this taxonomy is organized according to two orders, a lower one including the activities of *Knowledge* [gathering], *Comprehension* and *Application*, and a higher one including *Analysis*, *Synthesis* and *Evaluation* (Fuller, 2002). The new trend in CAA, and perhaps what ultimately makes it interesting, is to tackle the higher-order assessment problems. This is new as traditionally these tasks were considered infeasible, until recent progress in computer power and computational solutions. Plus, there is also a commercial interest in the field (Valenti, 2003a).

Following the Journal of Information Technology Education (Valenti, 2003a), the precise topics of interest in the field are the following:

- Novel tools and approaches to Placement, Formative-assessment, Summative-assessment and self-assessment.
- Innovations in Assessment
- Assessment issues in VLEs

- Diagnostic testing
- Pedagogical issues in question design and content
- Impact on Institutional teaching and learning strategies
- Quality assurance issues
- Evaluation of Assessment Engines
- Assessment and standards

Obviously, the field holds ground to a gathering of concerns from various disciplines, ranging from didactics and university administration to IT research and software engineering (Valenti, 2003b).

Therefore, CAA's goal is to assess the capacity of a student to perform several cognitive tasks over a given informative chunk (bullets 1 and 2 above). In parallel to this, CAA is to be understood as performing over several *models*, which researchers from UCLES (Ahmed & Pollitt, 2002) identify as follows:

- The **Quality Model** or Judging model: measuring *how well* students perform in a given task. Typically, this involves confronting a student with questions of equal difficulty and see how well he answers.
- The **Difficulty Model** or Counting model: measuring *how difficult* a task a student can succeed in. Typically, this involves confronting a student with questions of diverging difficulties and see how much he can answer.
- The **Support Model**: measuring *how much help* a student needs to perform a given task. This involves interaction and is more ambitious, or computationally costly, than the two preceding – it can be added to either one of the quality or difficulty model, but implies a strong layer of analysis in order to detect the kind of help or support needed. Both analysis and feedback are typically dependent on the tasks and studied field.

That is the theory. Another trend in CAA simply consists in building tools of increasing complexity. It seems that to date, the degree of complexity is still quite low, when compared with other solutions reached by NLP or ML to tackle other problems. The key issue here is that CAA involves judgement – a highly subtle and subjective task – therefore, the idea is to first

define the theoretical boundaries of the problem, around cognition and didactics, before going on to sophisticated computational solutions.

In conclusion, CAA is a vast field that involves both judgement and evaluation – and CAA systems shouldn't be understood as means to simply mark essays or answers. A further field – and perhaps a field of application directly coming from CAA – is CAS, Computer-Assisted Scoring, which only encompasses the Quality Model (Yang & al., 2002, 2003), but whose systems present the advantage of performance formal evaluation. Needless to say, an ideal CAA tool should be able to perform over the three above models.

II.1.4. VLE

As we presented above tutoring and computer assisted assessment systems, we showed that these deal with the issue of adaptability in teaching and learning. VLEs, embedding the former systems, address the question of enabling accessibility and portability. Although any kind of CALL system implemented as an accessible and operative system can be understood as a VLE, the main interest of VLEs resides in their inclusion into means of diffusion, of which the most comprehensive is the Internet – and VLEs become themselves network environments, instantiating means of eLearning and of Learning Technology. This means that VLEs have more to do with communication systems and network administration on the one hand, and ergonomics on the other, than with CALL itself. Though such environments as CISCO, WebCT or HotPotatoes (see References) are all understood as eLearning environments, they actually have little to do with theories of learning, as applied to computerization or not. They simply function as repository of courseware, manually uploaded by teachers or administrators, and downloaded by users. But this is nevertheless what is understood for the present time by eLearning or Virtual Learning Environments. So that the whole point in relation to CALL is what to put inside these VLEs (Nakov, 2004, Davis & Davies, 2005).

However, at present, two notions central to the problem of CALL arise directly from the field of VLEs understood as eLearning platforms (and this is a point which makes clear the fact that VLEs are of little interest outside the reach of eLearning, except for ergonomy concerns).

The first one is the notion of authoring. Authoring is concerned with giving teachers the means to create exercises and tests – authoring tools include procedures and structures to encode reference material, and items of interest for the assessment (see section II.3). The second is a more vague subjective notion, though very pertinent to eLearning or CALL: that of dialogue (Aleven & al., 2001, Aleven & Rosé, 2004). Both are slightly related as typically, support for dialogue possibilities in VLEs is implemented through means of soft learning agents (Graesser & al. 2005) – a precise concept within the field of CS (Fenton-Kerr, 1998, Bogdan-Florin & Hunger, 2005). Dialogue has to be understood here as dialogue from 1/ user to instructor, 2/ user to user, 3/ user to system and 4/ instructor to system. Ideally, full support for dialogue should enable these four possibilities, in such a way that a user can interact and exchange information with other users of the same VLE, can exchange information with a instructor, can be granted support by the system, and that a instructor can interact with the system in order to obtain information on the users as individuals or as a community. Of these four categories of dialogue, only the third one (communication from user to system and from system to user) actually necessitates CALL dedicated tools, to perform language analysis – the other three are just means of communication and access to static pre-recorded or pre-computed information.

II.2. Tools for CAA and assessment in CALL

The assessment of "free-text" material is understood as dealing with two kinds of student's outputs: essays and free sentences. The first impulse dates from the mid-nineties (Burstein, 1996). The existing tools are quite simple and limited – but all are not experimental and some are already operating in the real world. Here, some systems dedicated to assessing free text material are presented, though all are in a more or less advanced state of completion (at the time of papers' publication), as they appear to be the most popular. In the field of CAA, the marking of essay is considered as the Holy Grail, therefore, the following systems are dedicated to the assessment of essays, with the exception of Automark.

Other systems are placed by their authors in the domain of CALL, even if they may refer to other projects in the field of CAA – we have seen in chapter II that these fields could largely overlap without explicit recognition of each other. These tend to focus on sentence assessment and none show as completed.

II.2.1. CAA Systems

As said, these tools address the issue of essay marking rather than that of sentences, or short texts – a simpler task as mentioned in section IV.4.1.1. Therefore they are at a more advanced stage of completion, as well as less documented scientifically since they often appear as commercial tools. In the last section, we present some tools or studies which still are in a state of research.

II.2.1.1. The Intelligent Essay Assessor (IEA) (Landauer & al., 2003a, 2003b)

This is proprietary, implemented by Knowledge Analysis Technologies (KAT) in 2003. It makes use of LSA (Latent Semantic Analysis) for scoring answers. This necessitates a reference training corpus over a given category of question/answers, which typically consists of reference answers to a given question, but which can also be a theoretical reference: in this case, essays are evaluated against one another, around a centroid evolving in the process – a scoring method referred to as *vicarious multi-human scoring*. LSA builds a two-dimensional matrix of word-document co-occurrence statistics, which are decomposed and normalized over dimensionality further enriched to include latent lexical relationships through decomposition/recomposition of the matrix space, using a matrix algebra technique known as SVD (Singular Value Decomposition). Students' answers are compared lexically to reference answers. IEA does not perform any syntactic analysis and can therefore be fooled easily, but it nevertheless aims at high-level assessment, scoring essays over a corpus of gold-standard essays or over essays which have been graded by humans. The aspects of writing it aims to evaluate are content, style and *mechanics*, which amounts in IEA state of the art to spell.

IEA authors claim a 85% reliability coefficient. This measures the agreement between IEA score and one of the experts, averaged over the total number of experts. These are results quite close to

E-rater. Now there are discrepancies in the results from domain to domain. IEA is thought to be particularly good on formal domains of knowledge characterized by enumeration, as biology. The authors modestly report that IEA cannot state whether a student made a specific point by means of "logic or persuasion". Now, a clear limit of IEA is its dependence on the contents of the reference material – in its operating mode, if comparison essays are insufficiently similar to the to-be-scored essay, or are too variable in their implications for content quality, the essay is passed to a human grader.

Another fundamental aspect of IEA is its tutoring quality, making it an ITS as much as a CAA system. As the system is not trained on just pre-marked essays, but also on manuals or textbooks, it can provide feedback to the student in the form of a link to appropriate material – the authors insist on enhancing IEA tutoring quality as a foundation of any future work. It is also equipped with plagiarism detection mechanisms, of which the authors say nothing.

II.2.1.2. The Essay-rater – or E-rater (Burstein & al., 2001, Attali & Burstein, 2006)

This encompasses a hybrid approach between semantic and syntactic analysis, statistical and symbolical approaches. It has been developed by the Education Testing Service (ETS). The system identifies syntactic structure and discourse features through shallow parsing. The syntactic and discourse components are analysed for themselves, and rated according to syntactic variety and discourse coherence. Content words are compared to a vector of reference weighted content words. Therefore, topic coherence, syntactic richness and lexical variety all work for higher student grades. E-Rater has the reputation of working well with simple tasks, possibly even of the higher order sort, and it has been used to rate the AWA (Analytical Writing Assessment) part of the GMAT (Business Schools admission test) as well as TWE essays (Test of Written English).

The scoring models have been defined based on a set of pre-marked essays – 400 for each of the two AWA essays. Features were defined and evaluated for mark prediction by means of linear regression. The features (about 100) include proxies, of which nothing is said, as well as linguistic features. An example of the linguistic features is a measure of syntactic sentences and

clauses' types. These features represent general qualities of *syntactic variety*, *topic content* and *organization of ideas*. Having refined the models, as subsets of the total number of features, by cross-validation, the authors claim an agreement of about 90% between expert's and automatic marking. The authors also claim portability from one domain to another, but this all depends on a training corpus.

II.2.1.3. The Conceptual-Rater – or C-Rater (Leacock, 2004)

C-Rater is a sister product of E-rater, developed at ETS. This is one of the projects closest to our own, as it aims to mark short answers, including answers to reading comprehension questions. A major difference is that it targets a K-12 audience of native users (primary and secondary school), and therefore is not concerned with SLA; the answers are marked without regard for the student's writing skills. Interestingly, it is the only comparable project which explicitly focuses on *paraphrase* – the program being presented as a *paraphrase recognizer*. Now, the paraphrase model is not really formal, and actually does not rely on linguistics, but rather on contextual reformulation. The paraphrase possibilities are framed within notions of syntactic variation, different inflections of a word, substitution of synonyms or similar terms, or the use of pronouns in the place of nouns – but these categories are not merged to define a model of paraphrase. Instead, a model is built for each question, using a set of training answers – between 100 and 200 depending on the question. It is at the level of this model that paraphrase possibilities are defined; so paraphrase recognition capacity depends on the variety of the examples within the training set.

C-Rater uses an authoring tool for the manual annotation of marking criteria. For each set of answers, a number of items of importance are identified, and for each training answer, the contents are associated with these items, by highlighting relevant parts. Tested on 19 reading comprehension and 5 algebra questions, C-Rater achieved a score of 85% agreement with the human annotators. Now, this test set is already a subset of more questions, from which questions yielding open answers have been discarded, as they «allow for metaphorical interpretations that cannot be fully catalogued». In other words, the authors discarded questions for which an efficient model could not be built with at most 200 training answers – which in turn means that the question classification in C-Rater is empirical.

II.2.1.4. Automark (Alridge & al., 2003)

This has been developed by a firm called Intelligent Assessment Technologies, but it is still well documented. It uses information extraction techniques to retrieve pieces of interest in a given document – namely, an answer to a question or a short essay. These pieces of interest consist of given syntactical categories expressed as lemmas. The same process is applied to a corpus of reference answers in order to build marking scheme templates, which are designed in the form of a syntactic tree with the nodes or leaves bearing appropriate lemmas for the question to be answered. Actually, one of the system flexibilities is that these templates can be composed in various ways, including an interactive way between ML models and expert validation, as handling of terminology is crucial to tackle reformulation problems. Plus, there is a scheme moderation module for experts/teachers to design helping schemes (a support model issue in ITS). Finally, an answer is parsed and checked against marking scheme templates for grading, which works through correspondence between a series of lemmas and phrases. This is a pattern-matching approach, and allows easily for having patterns/templates of various degrees of specificity.

II.2.1.5. Auto-marking (Sukkarieh & al. 2003)

This is worth mentioning as it is not yet proprietary – but it is intended to be used in real world for various Oxbridge tests and examinations, and is therefore quite ambitious. It resembles Automark (after which this system has been named, or so we suspect), in that it is pattern-matching. The templates are however more detailed as they are supposed to be able to cope with various relation orderings in the syntactic structure of the piece of text. This just means that the syntactic categories resemble more those of a PoS tagger than of a shallow parser. Moreover patterns and templates are more flexibly organized, as a template is understood as a series of patterns of varying forms and numbers. Here too, Information Extraction is applied to retrieve members of a fixed list of named entities and relations occurring in a given configuration.

II.2.1.6. Some more tools

The earliest (known) automatic assessment tool is the Project Essay Grader by veteran researcher E. B. Page (Page, 1994). It works on surface clues as is done for text complexity computation, but the obtained indices are put against reference measures computed on a training set – content is left unanalyzed. BETSY (Bayesian Essay Test Scoring System, Rudner & Liang 2002, following Larkey, 1998) uses a naïve Bayes classification technique to score essays on both style and content. SEAR (Schema Extract Analyse and Report – Christie, 1999) is still work in progress, but makes use of sophisticated calibration for the computation of weights on features: for a set of reference essays, the weights are evaluated as a compromise between system's judgement and human judgement. Pérez used a variant of the Machine-Translation quality BLEU metric (Papineni and al., 2002) to assess free-text answers to questions of comprehension in the domain of computer-science (Pérez & al., 2004). BLEU compares two sentences by counting the frequency of n-grams from candidate to reference, at various values of n – the resulting value is modified by a brevity factor to penalize shorter answers. The results are evaluated through correlation between automatic marking and teacher's marking. This approach exemplifies the quantitative assessment strategy and shows well its limits, as teachers mark with either 0 or 1, while the automatic marker has been shown empirically to mark between 0 and 0.7 in a very spread distribution – a problem being therefore to know how to convert this real-value marking to a binary evaluation.

II.2.2. Approaches to Assessment in CALL

Assessment of free-text answers in CALL poses another difficulty with respect to CAA: having to assess language grammaticality. Generally, CALL as applied to analysis of free-text falls into the domain of ICALL. The rise of ICALL systems is directly linked to the development of parsing. Early examples of ICALL systems include *Herr Kommissar* (DeSmedt, 1995) or MILT (Kaplan and al., 1998). Note that these two projects were not research projects, or at least weren't presented as such.

In *Herr Kommissar*, a learner is presented with a mystery to solve by asking questions. NLP technologies include a spellchecker and a predication-driven parser (modelling attachments of

complements to verbs). The Knowledge-Base implements a microworld by ways of an ontology and logic rules. But the fact that the KB is handcrafted prevents the portability of the approach. DeSmedt gives a figure of 91.6% of grammatical errors detected, but there is no information on the break-down of the error categories nor on the testing procedure. There is no more information on the questioning method and the evaluation of answer assessment.

MILT is another microworld approach to the assessment of free-text answers. Students are given an exercise that allows them to use language production to manipulate a graphics microworld. An authoring module enables an instructor to define tasks for the learner to accomplish using spoken commands, such as moving pieces of furniture within a defined space. The system analyzes the student entry to detect syntactic, semantic or factual errors. The student's input is then analysed with a LCS dictionary (Lexical Conceptual Structure) (Jackendoff, 1990) to be transformed into a LCS structure. This is finally mapped to the representation of the reference for assessment. LCS represents action through primitives grouped into types (for instance, HERE, GO, BE, THERE as primitives of the type LOCATION). Eventually, the system seems to be best used for the practice of pronunciation. No results are provided on performance.

We know at least two projects which address the problem of assessment in the context of ICALL and at the level of sentences or short text as answers to open questions. These are research projects; one does address the point of error correction and the other not. One is dedicated to ESL and the other to FSL.

II.2.2.1. FreeText

The FreeText project is closely related to our own, although its projected scope was much larger. The aim was to develop a CALL tutorial for the teaching of FSL. The proposed software should have included various types of exercises such as multiple choice questions and fill-in-the-blanks, and also open-ended questions on comprehension to be answered by free-text – in fact covering the whole line of drills format to address four types of learning activities following Chapelle (1998): comprehension, exploration, creation and manipulation of textual data. These tutorials would have been based on real texts very comparable to the ones used in DidaLect. A framework has been defined for the assessment of free-text answers, based on PseudoSemantic Structures

(L'Haire & Vandeventer-Faltin, 2003a). PseudoSemantic Structure represents sentences in a role predicate structure, where predicate is further characterized by features of voice, tense or aspects, among others, and roles by features of number and theta role (agent or patient). Both can receive *characteristics* in the form of adjectives, or adverbs. An answer must match this manually created representation in order to be assessed as correct.

It seems the project has not been completed (FreeText 2002). At least it appears that the proposed drills have never been implemented. However, this project drew considerable effort into setting a component for the detection and diagnosis of language errors, using techniques of grammatical constraints relaxation (L'Haire & Vandeventer-Faltin 2003b). This will be reviewed in chapter V.

The categories of error to be identified include some high-level ones as verbal complement subcategorization, voice, or homophony, which can be problematic in French. This component is tested on a small set of real learners' productions and yields a result of nearly 73% in error detection, but some error categories are excluded from the accounts, such as voice or verbal complements subcategorization – that is errors of high syntactic level, difficult to identify. By experience we know that students tend to make a lot of these errors in their productions, so it seems strange that none show in the study's statistics. However the system is quite good at detecting errors of homophony. This is done by mapping lexical suspects (causing a disturbance in the syntactic analysis) to a phonological representation, which is in turn used to gather a set of phonologically similar words – the sentence is analyzed again using these replacement words, and they are proposed as replacements when yielding a correct syntactic analysis. As it seems the effort has been put into the realization of an interface to display analyses and results – the error diagnosis module performs only 17.4% better than MS Word 2000 grammatical checker, largely due to its handling of homophonic errors. Eventually, it seems the error correction module did not address higher level syntactic errors and concentrated rather on the graphical display of identified errors.

II.2.2.2. Content Assessment Module

This project is perhaps the most closely related to our own of all the systems surveyed (Bailey & Meurers, 2007). Here, learner answers to questions are compared to a single target answer through means of shallow language processing strategies. The learner's answers are tokenized, lemmatized, tagged and chunked for Noun Phrases detection. Then lexical relations are detected using PMI-IR scoring technique (Turney, 2001) on the Exalead search engine. The semantic type of nouns is added using WordNet. Eventually, dependency parsing is applied using the Stanford parser. The information gathered in this way is then used to produce alignment of sentence's descriptors: tokens, NPs and dependencies. The main difference with our approach is that this information is then used as features to train a supervised memory-based classifier (TiMBL). The main features are Keyword overlap, aligned target tokens, aligned learner tokens, aligned target chunks, aligned learner chunks, aligned target triples, aligned learner triples and token matches.

Although CAM shows very good results, with a 88% accuracy in its best configuration, we believe these results are partly due to a bias of the approach towards *correctness*. On the one hand, using a tunable ML approach, the system is trained to detect correctness based on measures of distance between the quantities of the features – this overlap strategy has clearly an effect of error under-diagnosis passed a certain quantity of shared material between training and test answers. On the other hand, the training set includes 188 correct answers and 76 incorrect answers while the test set includes 187 correct answers against 37 incorrect answers. This should actually introduce a bias towards correctness, at least in the testing procedure. Should the system be trained to detect *incorrectness,* should the training and test sets be inversed, or ultimately the ratio of correct/incorrect responses be inversed in both sets, the results might be inferior. The authors also discard from training and test sets so-called *Alternate answers* which are purely semantic reformulations of the target response. This approach is nevertheless fruitful as it permits to sidestep the problem of partial semantic reformulations. As related to our project as it is, CAM is for English, and it shows major differences in that it follows an opposite strategy in assessment, as we will show in section X.1.2. Moreover, it does not address errors in form, as grammatical errors.

II.2.3. Relation to Question-Answering

Question-Answering (QA, see for instance Andrenucci, 2005, for an overview) calls for the evaluation of contents: a question is mapped to an answer based on surface clues, or features, enlarged or decomposed through means of NL or AI processing of possibly ever-increasing complexity (from synonymy to abduction, or co-reference resolution to entailment). The state-of-the-art in the field of Question-Answering is still imperfect, and the effort is still in a pioneering phase – but the means to solve the problem have been identified, at least partly.

Now, there are some differences between the free-text assessment and QA. We have identified at least three. The first difference is to be found between answer analysis in the context of an ITS and in that of QA. QA traces an answer from a question in a space of theoretically infinite size while we need to evaluate an answer from another answer: the reference in text, which can be unique. This also implies that we have no need for detecting the unknown: the answer to the question. The second difference is in the quality of language: QA has no responsibility in this respect and can posit that questions should be correct – the point of the answers correctness does not arise, as QA works on reference material. This means that QA can ignore the syntactic aspects of language evaluation. The last difference is that QA can make use of statistics to weight the distance between question and potential answers, and can therefore rely on stochastic methods of correspondence. This offers the advantage of *relativity* of correspondence between question and answer: select the closest. In free-text answering assessment, reference answer and candidate answer should match perfectly – or else we have no means to decide where to set a threshold on weighted partial correspondence.

A fundamental similarity, however, is that in both cases a degree of reformulation can bias the correspondence of contents. Reformulation takes place between question and answer in QA, and between answer and reference answer in free-text answering. Reformulation can come in the form of paraphrase, or in the form of a lexical subset of the reference answer.

A good illustration of the relation between QA and free-text answering, and of its limits, can be found in the QA Mitre Deep Read project (Hirschman & al., 1999, 2000).

II.2.3.1. The Deep Read Project

From its beginnings, the authors of this project have set their research work, dedicated to QA, in the line of free-text answering – admittedly to provide both a means of evaluation for the quality of QA retrieved answers as well as to set a foot in the domain of tutoring systems (Hirschman & al., 2000). The project seems to have been abandoned, but it gives a flavor of early solutions to address the problem of short-answer validation.

The Mitre team encountered a problem in what concerned the evaluation of their QA system: evaluation. In order to cope with it, they doubled the scope of their project, adding an answer analysis tool to provide evaluation of the QA system's answers. They identified two means for answer evaluation: 1/ building a classifier based on valid answers or 2/ producing answer keys for each question and use them as a reference. The second approach was retained for lack of a corpus of correct answers. The keys to be compared have been produced by hand, and the comparison method between the keys and answers works on two modes: answer correctness and lexical recall (answer correctness amounts to exact match). Lexical recall implied the definition of a recall value threshold, which has been empirically set. This evaluation is limited, and the authors acknowledged its limits: absence of measure for an answer's *conciseness* and *intelligibility*.

However simplistic, this project encompasses the first approach to our knowledge to short answer validation, or free-text assessment.

II.3. Students and Lecturers.

II.3.1. Lecturers and Authoring

A trend in CAA and CALL is to implement *authoring* tools. The reason is that CAA systems are never more efficient than when designed for very specific tasks, which necessitate shared components but rarely shared methodology. This has had the double effect of opening the field of CAA to numerous new assessment tasks with the possibility to create exercises of various

content, but also to standardize their types (see SCORM/ADL – Sharable Content Object Reference Model, by Advanced Distributed Learning, on the necessary components of an e-learning system, and their linkage) as well as the types of questions to be asked (see QTI – Questions and Tests Interoperability, on the type of admissible questions in CALL/CAL, and how to describe them technically : for example, multi-choice questions, lists of items to be ordered, fill-in-the-blanks, etc.).

Typically, an authoring tool includes NLP methods to analyze the language – a necessity whatever the subject of the exercises – as well as graphics analysis methods to make assessment of figures possible, which is a requisite in sciences. They also include data-mining and ML methods to detect and weigh item quantities in the assessed material. With such tools, it is possible to create drills for any subject that is part of an academic curriculum (see Qualrus for an example).

On the other hand, whatever the subject and the material to be assessed (drills or essays), the assessment methods all depend on the inclusion of substantial annotated reference material to train models of what is correct and what is not. In other words, these tools are ill-equipped to implement dynamic methods of judgement over an abstract model of correctness, as they work on exemplifications of correctness/incorrectness.

With pros and cons, these tools have nevertheless the advantage of putting back CAA into the hands of lecturers or trainers, who are entirely responsible for the kind and design of the drills. A trainer must provide any training material, making sure it is properly annotated with what is right and wrong, design the exercises, design the feedback model and feed the system with any further necessary resource. All this calls for good will towards CAA and familiarity with computers.

As it turns out, such authoring tools offer good solutions for the design of summative assessment tasks for large test submissions and crowded audiences. For example, implementations of drills for physics do actually target undergraduate entry exams into physics program, as well as special programs tests of prerequisites (Van Lehn & al., 2002 for an example). They are unsuited to the design of day-by-day formative assessment solutions.

Under this paradigm, the entire academic world is the standard (constant methods of testing students, and of assessing them). If we focus the assessment problem at a level closer to subject(s) standard (English, maths, history, etc.), using models of subject correctness (what is

correctness over defined topics or problems in Maths, English, etc.) rather than topical examples, then it is possible to come up with generic solutions that take some of the burden of design off the lecturers' / trainers' shoulders: that of having to provide reference answers. This should enable the provider to offer the students frequently renewed formative material.

So, there is a trade-off between the expected level of intervention on the lecturers' part in designing the drills and generality/specificity of the assessment tools. The greater the generality, the greater the re-usability (of software as a whole), but the greater the effort for the lecturer to produce drills – the lesser this effort of intervention, the lesser the generality, and so re-usability.

Another trade-off is between the expected level of intervention from the lecturers and the formative/summative orientation of the assessment tool. The lesser the lecturer's effort, the greater the formative capacity, as it becomes cheap for the lecturer to renew test material.

Here is not the place to discuss a third and fundamental trade-off: that between production cost of a CAA system and re-usability. We have no answer to balance this trade-off, and all we can state is that the CAA pack is led by an association between university management agencies and the industry, which favour large-scope CAA tools for evident economic reasons: implementation requires less field expertise and the tools so produced can target a broader audience, which justifies the purchase (Graesser and al., 2005).

From this perspective, the role of the lecturer has been (and remains) at the crossroads between machine and students, that is, between content and students. The antique relation between "master and pupil" is just about to turn into a triad involving master, pupil and a computer.

II.3.1.1. In the context of text-as-reference based open questions in SLL

What should the role of the lecturer be? Asking the lecturers themselves appears as something difficult to achieve: for this project, four teachers of French as a Second language have been contacted – only one replied. The reason for this defiance is natural and expected: non-technicians fear the machine, either because they perceive it as a threat to their position or knowledge, or they simply feel ill-at-ease with the manipulation of computers in general. This does not favour communication (see Coniam, 2004, for a discussion on the issue).

However, our pedagogical problem being so precise and restricted, it is feasible to endorse the role of a lecturer to see what can come out of his/her involvement. The automatic assessment task itself does not call for the lecturer's participation – although if the lecturer is to participate more actively in the definition of questions and putting them in correspondence with text reference segments, then the assessment modules should provide the methods to make the best of the lecturer's effort.

When creating a question, the lecturer can do a lot to 1/ relieve the system of some hazardous computations, and 2/ provide information about questions and references easing assessment, which otherwise would be out of the system's reach (so-called world knowledge). These two orientations are non-exhaustively examined in turn.

A. What can be done easily by a lecturer and not-so easily by a computer.

   - To perform Anaphora and Co-reference Resolution.

   Computerized solutions to the problem of defining co-reference in texts do exist and come with acceptable results (Mitkov, 2002, for an overview), including in the context of free-text evaluation (Pérez & al., 2005a). But they perform with varying accuracy according to the precise resolution task: intra-sentential pronominal anaphora is easier to solve than extra-sentential cataphora. A lecturer has to perform resolution anyway, mentally, in order to choose a text segment interesting to the task (would it be only when reading the text prior to creating the questions): that s/he simply produces an ordered list of the nouns for which the anaphoric lexemes stand is enough to solve the whole problem. There is not even a need to pair them with actual anaphoric element; the system can just replace anaphora by its referents orderly. This is only valid for the reference sentence, and does not help with solving anaphora in student's answers. But reference material is more prone than the single sentence student's answers to showing complex co-reference phenomena, such as inter-sentential.

   - To link sentences in the text.

   A reference sentence can be a "short answer". When selecting the reference sentence on which a question is based, the lecturer can add some further sentences which are not co-referenced to REF (in-text reference answer), but which make the statement in REF more

precise, and which could then be used by the student to build an ambitious answer. Actually, at this point, such a trick could enable the creation of inferential questions (textual implicit) of greater difficulty – but little more.

- To check the parses and enrich them.

In a real-world perspective, as difficult as it is to obtain perfect parses, it is reasonable to think that a lecturer could participate in the syntactic analysis and provide, for instance, information to specify the function of complements by simply attaching together constituents in a given syntactic function.

B. What can be done easily by a lecturer and only with great difficulties or costly computation by a computer

- Some of the above…

Such as enriching the parses or properly disambiguate words. Some words can be disambiguated only through enlarged context, calling for world knowledge valuation (e.g., a classic case-study: *La petite ferme la porte*, where every word is ambiguous), making automatic disambiguation very difficult. Similarly, properly catching ambiguous syntactic information can be inaccessible to what is presently possible in automation (e.g., *Tarte à la crème* = NP + PP/NounComplement, while *Tarte à la table* = NP + PP/LocationVerbComplement)

- Provide semantic information on words in REF.

To tell the machine that a word stands for an animate or inanimate, countable or non-countable object provides useful information, as lots of verbs do not behave in a similar fashion depending on this kind of information. Also, to state that a polysemous verb topically expresses itself under its intransitive or transitive form represents valuable information, as lots of verbs shift between the two modes depending on their exact meaning (e.g., *Je rentre* vs. *Je rentre les chaises*), in order to specify the auxiliary admissible for the verb under a given sense (e.g., *Je suis rentré* vs. *J'ai rentré les chaises*). The more this process of enriching vocabulary is achieved, the less it is needed next time. But this issue has more to do with language analysis in general than with CALL in particular.

Finally, after having studied what the lecturers could do for the system, it is worth examining what the system can do for the lecturer. The most obvious benefit relieves a lecturer of part of the assessment burden. A lecturer can provide students with formative drills and only verify the quality of the automatic assessment. She can also obtain a report on a student's achievements over a timeline and automation can provide information on points of progress or stagnation, or even on more subtle points such as the ability to reformulate. Any piece of information that can be evaluated can be integrated into abstraction and consulted by the lecturer.

II.3.1.2. Students

Any tutoring system has to be positioned with respect to the type of assessment it addresses. There are two modes: Formative and Summative assessement. The first aims at practicing and the second at marking or grading. Naturally, Summative assessment requires error-proof systems and cannot include experimental solutions. Formative assessment is the right field in which to test solutions in assessment.

Formally speaking, formative assessment is a diagnostic use of assessment to provide feedback to teachers and students over the course of instruction, while summative assessment takes place after instruction and requires making a judgment about the knowledge of a student, such as grading a test. According to Black and William (1998), formative assessment follows *assessment* which includes activities that teachers and students undertake to obtain evaluation of teaching and learning. It becomes formative when the evaluation serves to adapt teaching and learning to meet student needs. There is strong common ground between Formative Assessment and ICALL, as both insist on the importance on feedback, especially when providing specific comments about errors and specific suggestions. Another objective of Formative Assessment is to drive students to focus their attention thoughtfully on the task rather than on simply getting the right answer. As it favours frequent short tests over infrequent long ones, it seems to provide a good pedagogical framework for ITS.

In formative assessment, what matters is more the quality of the feedback than the marking. In summative assessment, any student material has to be considered under the aspect of correction: capitalize on the good. But in the paradigm of formative assessment, there is no such thing as a

good answer: capitalize on the bad. Especially in a topic such as ours, a good answer is always to be an alternative between several good ones. Using reformulation, an answer might lack economy in the display of the contents, and choosing not to use reformulation exhibits poor ambition and language skills. Formative assessment is directed at improving students' level, not judging them (Schmidt & al. 1990). However, formative and summative assessment are not mutually exclusive, and *scored-formative* assessment encompasses a hybrid approach (MacKenzie, 1999).

We are far from the successful implementation of the ultimate CAA tool for FSL, or SLL – a system capable of emitting any pertaining remark on the quality of what an answer is and is not. But that is nevertheless the ultimate purpose of such formative tools – to gloss over the material in the absence of the trainer/lecturer who remains sole decider of where goodness starts and ends.

Coming with specific assessment products conditions the feedback, the pedagogical task conditions specificity, and the pedagogy is at last governed by the target audience of the project. So there is (again) a trade-off between the specificity of the feedback and the expected audience, and economically speaking, the feedback appropriate enough to the audience only should be provided. This helps to deal with the shortcomings of computerization: only implement solutions up to where the system can cope with the didactics and pedagogy. In other words, there should be a level at which automation can meet learning activities.

In the context of this project, the audience consists of "intermediate-advanced" students of French. This conditions what should be expected in terms of good and bad answers. But for now, it simply means that the audience factor holds a stake in the definition of this project as an engineering problem[2].

[2] Note that the « intermediate-advanced » nature of the audience has been an imperative on this project, though indirectly, following DidaLect.

Chapter III. Free-Text Answers and Text Comprehension

This chapter presents the problem from the perspective of didactics and linguistics, especially in their relation and with respect to the problem of Text Comprehension – a branch of Language Learning and Education. First, its purpose is to examine the problem of question answering in didactics, and the kind of solutions that didactics could offer to solve the problem of reformulation. Then, it presents the problem in terms of linguistics: what constitutes a correct or an incorrect answer.

III.1 Texts, Words and Second-Language Learning

III.1.1. Writing and Written Material in Second Language Learning

In the approach proposed by DidaLect, and directed towards a given category of students (intermediate or advanced), language learning is conceived as primarily a task of developing cognitive abilities to build and record contextual sense with limited linguistic knowledge – and the application is in Text Comprehension. What is meant by *contextual sense* is both the sense of subset of a text with respect to the text as a complete set, as well as the sense of this textual complete set within, let's say, *world knowledge*. In other words, in such an approach the activity of learning and understanding is conceived as a task of disambiguation.

The procedure of cognition drives a student to examine the vocabulary in order to understand structure and therefore meaning (Meyer, 1987, Duquette & St-Jacques, 2005). The enhancement

of these cognitive skills should enable the student both to draw conclusions based on partial understanding and literally to learn the specifics of a language, or part of it, through deduction, and through enhancing deduction abilities, rather than through formal instruction.

Although this approach is mainly concerned with understanding, adding a free-text input processing capacity can further test this paradigm for language production objectives, as well as control the linguistic inferences drawn by the student. The latter task can also be controlled through multiple-choice questions (MCQs), and perhaps with better results as MCQs enable presentation of categories of questions of greater difficulty, in a linguistic or even more cognitive sense. But the production task can only be tested through free-text. This approach is however not understood as a substitute for other methods of language learning, but as a helper.

III.1.1.1. Modalities of Text Comprehension Testing

There are a good number of strategies to teach or enhance Text Comprehension (TC) skills, following the cognitive school of thought as most applications now do (see Duke & Pearson 2002 for an overview). The modalities of testing are however quite straightforward and are sound enough by themselves to allow the economy of the cognitive justification. This is not directly relevant to this project, but the interest to Computer Science and to CALL is to find applications possible to automate, and to locate the modality of our testing paradigm within the framework of TC.

Note that in TC testing there are strategies and routines. Strategies are the different types of exercises. Routines are sets of strategies reputed to work well in the field.

- Prediction (Anderson & Pearson, 1984):

  The student is asked to make *predictions* about what is to come in the course of reading a text. This applies to narrative texts.

- Think-aloud:

Teacher think-aloud or *student* think-aloud (see for example Kucan & Beck, 1997). The teacher or learner expresses verbally what she has retained (who, what, where, etc.) in the course of reading a text.

- Text-structure (see Duke & Pearson 2002 for an overview):

Useful for both narrative and explanatory texts. This goes from explaining, possibly formally, the structure of a text, to creating a hierarchical summary (ideas of importance, and order of importance) and possibly visual representations, such as semantic maps or story maps.

- Visual representations (again Duke & Pearson 2002 for an overview):

Following the above, but more formally defined. The debate evolves around the question of whether it is the mapping itself which favours comprehension or the process of constructing the map, in which case Text-structure accounts for the benefit.

- Summaries (From Kintsch and Van Dijk 1978):

Again, taking on Text-structure. Here, the methods of summarization make the most of the interest.

- Questioning (See Durkin 1978):

As MCQs or open questions. The activity can follow three modalities: answering questions on text, asking questions on the text (as in QAR, *Question-answer relationship*, Ref), and QtA or *Questioning the author* (Beck & al, 1996).

In the most advanced testing routines, free-text answering to open questions is the ultimate test (see for instance, Vocabulary Knowledge Scale, Wesche & Paribakht 1996). In Section III.1.4 we present the approaches to asking questions on vocabulary.

III.1.1.2. Text Comprehension and Questions

Theories of Text Comprehension, or discourse comprehension, or even reading comprehension, can be divided between processes and objectives. Processes can in turn be divided between abstract and concrete processes – cognitive and linguistic, though the two tend to overlap.

Enright (2000) has been responsible for the design of the tests for the TOEFL (Test of English as a Foreign Language). At the most practical level, he identifies four tasks to be tested in Reading comprehension: 1/ to find information, 2/ for basic comprehension, 3/ to learn and 4/ to integrate information across various texts.

This is not the place for a full review of TC in details. But aspects of TC relevant to our problem of open-questions are presented in this section.

In an interesting article, Clifton and Duffy (2001) sum up the situation of TC with respect to linguistics, drawing attention to particular aspects of language that help or make more difficult the understanding of a text – and consequently, aspects which are more or less worthy of being tested. Following this review, four aspects of language do impact on text comprehension: syntactic, lexical and semantic, as well as prosodic – this latter aspect being of interest to the spoken language or to narrative texts.

In syntax, there is a "garden path" theory which stipulates the construction of sentence meaning in a word-by-word fashion, following the canonical syntactic structure: this favours meaning attachment of a new item to the most recent correspondent; hence, sentences producing a revision of the meaning encountered necessitate more attention. There is a *locality* factor in sentence phrasal structure which makes sentences more difficult to understand when disrupted. The question of anaphora resolution is emblematic of this problem of structure building: distant attachments create ambiguity, and this is where a student's comprehension of a text might be questioned.

Similarly, there is a *frequency* factor in the lexicon that may influence the readability of a sentence: in the interpretation of a sentence "a given lexical item passes more activation to its more frequently used structures, and [...] frequent structures should therefore be preferred and more easily processed"[3]. So, ambiguous words may cause a disruption in the understanding of a sentence – an evidence which becomes even stronger in SLA. In relation to the lexical aspect of language, the semantics of the lexicon used is an influence on understanding. To illustrate this, and to fully elucidate a concern which might sound as another obviousness, the authors mention an explanation of the difficulty to properly analyze a sentence as "*the horse raced past the barn*

[3] The authors report an interesting debate – the lexical-head debate – around the question of whether a sentence is first understood by syntax or by the lexicon.

fell" – here, verbs rarely used transitively tend to show a complex sub-categorization structure when transitive. They use examples of similar idiosyncrasy to show that unusual subject quantification can also disrupt sentence understanding.

We found these considerations worthy of mentioning because they drive the interest of question asking to linguistic concerns rather than to contents. And this should push the student to reconstruct the sentence following linguistic reformulation (possible to process via NLP) rather than semantic reformulation (not possible to process), although no studies exist to our knowledge that support this claim.

III.1.1.3. Texts

As "Usage cannot be invented, it can only be recorded" (J. Sinclair, in Binon & Verlinde, 1999) authentic texts should be used in the assembling of learning material for reading activities – "authentic texts" are texts written by native speakers for native readers. Criteria for text selection include concerns for topic familiarity and interest (to the students), clear sequential development, as well as length and discourse concerns (coherent or/and well-balanced discursive quality) (Swaffar, 1986, 1991, in P. Raymond, 1995). The length problem is particularly crucial to our subject where vocabulary acquisition is linked to text comprehension, as shorter texts tend to promote vocabulary acquisition purposes as they favour a closer word for word reading, and hence facilitate the inference process from known to unknown words associated to meaning construction in the reading activity (Raymond 1995). A small-scale problem such as word understanding is better dealt with in small language contexts such as a single sentence.

This reading activity can be accounted for by two differing approaches (Kintsch, 1982). The first one is a text-centered approach, in which the student is presented a text as a fixed set of words and sentences presenting an immutable meaning. The second one is a reader-centered approach, where the construction of meaning is understood as a transaction between reader and text, an "interaction between reader, text, classroom, teacher and context" which occurs through open dialogue between students and between students and teachers, according to P. Raymond (Raymond, 1995). Though this point of view seems naïve in the presence of scientific or pseudo-

scientific reading material, it is certainly of great value in the perspective of reading activity as support to language learning, and this is generally the approach adopted in present language teaching strategies.

Our subject of interest here is not directly in the field of didactic motivations for text selection, so we will not go further than to say that "true contexts" and a minimal familiarity on the part of the student with the reading material are imperative with respect to our present didactic concern. We will come back briefly to the textual material retained for our objective when presenting the system.

III.1.1.4. Words

Texts exhibit a dual structure, divided between a micro-structure and a macro-structure, and that the link between the two levels is set at word level (Kintsch, 1983). Broadly speaking, the micro-layer encompass a morphological level and the macro-layer a general sense level, so that sense comes into form gradually from propositions to sentences and from sentences to text.

Lexicology is a field in itself, and vocabulary can be formally taught if students memorize word senses in the form of more or less formal definitions, created with a more or less controlled vocabulary – which is the field of lexicography. Under this approach, a word goes with its definition, a context example, and possibly in the case of SLL, its translation in the native language (J. Binon, S. Verlinde, 1999).

This approach fits well automatic teaching/learning as vocabulary is to be learned as made of single-sense words, which permits setting of fill-in-the-blank style exercises easy to implement (their completion is easy to check for correctness). This kind of teaching has its advantages and can be very efficient for vocabulary acquisition. It presents however several drawbacks, as this method works well for the learning of simple words like "birthday" or "donkey", but less well for more complex words or signal words (or grammatical words, or function words, as opposed to content words as nouns, most adjectives – with the exception of non-qualifying adjectives – or verbs). Knowledge of the vocabulary as discrete items (for instance a word and some of it

synonyms) is known as *passive knowledge* (Laufer & Goldstein, 2004). Knowledge of how to use a word in context is *active knowledge*.

Richards (1976), a pioneer in the teaching of vocabulary, identifies the following items in the knowledge of a word: frequency, register, syntax, derivation, association, semantic features and polysemy. Read (1988) divides the knowledge of vocabulary between depth and breadth. The knowledge in breadth has to do with the size of the lexicon known with at least superficial knowledge. Knowledge in depth has to do with how well one knows a word. Enright (2000) has defined ten aspects of word knowledge, which more or less cover those of Richards: frequency, meaning, collocability, register and functional constraints, syntactic behaviour, basic forms and derivational possibilities, semantic association, idiosyncratic features, homonymy, degree of abstractiveness. Of these ten, he identifies three factors as the most important: synonymy, polysemy and collocation. Two of these three factors are instances of active knowledge.

Read and Chapelle (2001) see language proficiency as the development of communicative skills. Lexical skill should therefore incorporate a communicative competence in addition to the knowledge of words as discrete items. Plus, they see vocabulary in texts as in relation to a context matching the user's need. Following these authors, activation of passive knowledge can be obtained by asking a student either to produce or to identify a list of synonyms to a given word; activation of active knowledge can be motivated through asking a student to produce a sentence using a target word.

As studied in the context of a text – especially authentic – vocabulary acquisition (learning words from context) has broader implications than new word meaning acquisition (learning words from their definition), as the method is inverse to that of lexicography: the latter advocates a method going from word to context, while the former advocates a method going from context to words. It forces the examination of morphology in order to search for semantic clues, it casts light on the lexical phenomena of false friends and cognates, as the student is driven to commit sense misinterpretation and errors through this phenomenon (and learn from his/her errors). It forces the student to look for informative clues in the textual material in order to make sense of unknown words. This replaces the apprenticeship of vocabulary in a cognitive perspective, which itself participates in a general sense acquisition strategy that is valuable both to words and text. Making the effort to understand a content word through context eases word and text

understanding while making the effort to understand signal words through context eases the understanding of text and discursive expression.

In this perspective, asking questions over authentic textual material is a good teaching/learning strategy in the field of SLL. It partly incarnates a reader-centred approach through which the student's personal knowledge and understanding of the world is valuated (Giasson, 1990).

III.1.2. Texts and Questions

As we said, the feasibility of the project relies on the fact that students answer questions of limited type, over texts of restricted categories. Its justification rests on the fact that the pedagogy of asking questions based on real texts is thought to be beneficial for the learning activity (Wixson, 1983). This principle of using real-world texts is being put in practice in CALL in at least one project already (Metcalf, 2006).

The typology of both questions and texts comes from the field of Education, as seen through the prism of psychology – that is, through the cognitive means involved in the activity of learning and grasping sense. Texts are categorized according to the kind of information they carry. Questions are categorized according to the type of information they refer to in text (identity, causality, etc.), or to their relation to the text's structure (number of sentences needed to answer).

III.1.2.1. Texts Categories

There are two main kinds of texts: narrative and informative texts. Examples of narrative texts include legends, novels, fables, and so on. They tell a story made of ordered elements, according to a structure more or less defined (called the tale grammar). In this regard novels offer more freedom than, say, the Russian tale (Propp, 1928), but all do obey given rules in their structure. Canonical elements of a tale grammar are: Exposition, Triggering event, Complication, Resolution, End, Moral. Such texts are thought in theory to be easier to handle than informative texts (Giasson, 1990).

Informative texts are thought to be more difficult to understand. This is because they exhibit content unfamiliar to the reader, or new concepts and specific vocabulary, or syntax of higher complexity (Muth, 1987). Whatever the reason, informative texts follow varying structures depending on what is to be communicated – and knowing these structures helps the understanding (Richgels 1987).

Meyer (1985) proposed a classification of informative texts as follows:

1. Description

   Gives information on a subject, with part or all of its features and specifics.

2. Collection

   Gives a list of elements which all share a common feature. Texts for which the common feature is a time span are called "sequential".

3. Comparison

   Compares elements against each other, based on their commonalities and differences.

4. Cause-Effect

   Presents ideas linked to each other through a relation of causality, where one idea is the cause (causal antecedent) and the other the effect.

5. Problem-Solution

   Also known as "question-answer". These are very similar to Cause-effect texts, and the problem is antecedent to the solution. Practically, these texts start where abstract Cause-Effect texts stop, and show information of a more practical nature.

This has been acknowledged as the best categorization (Giasson, 1990), though the categories are not exclusive of each other and in practice informative texts show a mixture of these categories.

The set of our 96 texts is entirely informative, showing an approximately even distribution between the above categories – with the exception of "collection" texts, which are included together with "description" texts. They have been gathered primarily to be included in DidaLect.

They originate from publications of general information (*L'Actualité*, *Le Point*) or of popular scientific writing (*Sciences Et Vie*, etc.).

They all have been evaluated against a scale of difficulty (Desrochers, 2006), computing a complexity score based on surface clues, such as the average complexity of sentences (through PoS), length, number of rare words, and the like. Words are defined as rare according to a word frequency list for the general language, below a given threshold (for French, see for instance New & Pallier, 2001). This complexity aspect has been studied in order to enhance the adaptability of the system: in a context of a fully implemented computer assessment system, a student may be oriented towards texts of greater or lesser difficulty according to their current performance.

III.1.2.2. Questions

Science of Education also offers a comprehensive theory for the categorization of questions related to reference texts (Giasson, 1990). There are roughly two scales of categories. The older one is based on Bloom's taxonomy of cognitive process levels and has been reduced to 1/ literal comprehension, 2/ interpretative comprehension and 3/ critical comprehension (Bloom, 1956). The more recent is more objective and easy to apply; it follows the various possible relations between the subject of a question and the answer as available in the reference text: explicit or inferential. Pearson and Johnson (1978) have proposed the RQA classification (Relation-Question-Answer). Formally, the categories are as follows:

- Textually explicit

  Question and answer both come from the text, and the relation between the two is clearly indicated through surface language, such as in a cause-effect sentence where a question states the cause and asks for the effect.

- Textually implicit

  Question and answer also come from the text, but the relation between the two is not indicated through surface language, which implies that the student draws an inference between several parts of the text (two to three, e.g. sentences). Accordingly, answering the

question could require the student to put together parts of the texts which are remote from each other.

- Script implicit:

Only the question comes from the text, and the student has to infer the answer from the text and their own knowledge.

These categories find echoes in other areas of the broad field of Text Comprehension and Language Learning. In the TOEFL reading comprehension section, Sheehan et al. (1999) differentiate between MC questions that ask readers to identify the main idea of a passage, to directly search a specific piece of information, to disambiguate vocabulary in context, and to resolve anaphoric or pronominal references. Newest version of the TOEFL contains a wider range of item types and includes three basic comprehension, three inference, and one reading-to-learn item types (see also Enright & al., 2000).

In the QAR (Question-Answer-Relationship, Raphael & McKinney. 1983), a method in reading Comprehension advocating the creation of questions on a text by the student themselves, the relation to the RQA is even more direct, and the author has also identified three types of questions to be asked : "right-there", "think-and-search", "on my own"[4].

Automation of answer validation constrains strongly the admissible types of questions. At this point, we can only speculate that as long as the elements necessary to produce an answer are present in the text, the job can be done – therefore the first two categories of the above are feasible to process. Note that in our case, textually implicit questions can be produced in two ways: either by calling for the student to link several sentences to produce an answer, or through the question being reformulated compared to the segment of text on which it is based.

III.1.2.3. Examples

- Textually Explicit

[4] Note that QAR was designed for children in a context of FLA, which might explain the more relaxed terminology.

Here, the question is interesting in that it reformulates REF (the reference segment from the text) at the lexical level (both *acidité* and *lacs* become adjectives from Q to REF) as well as at the discourse level (the cause-effect answer relation is expressed lexically in Q's form, through *pourquoi*, while it is expressed pragmatically in REF, through juxtaposition of the cause and effect clauses).

(2) Q: *Pourquoi l'acidité des lacs de l'Outaouais s'est-elle multipliée par 10 au cours du XXème siècle?*

(Why has Ottawa region's lakes acidity multiplied by 10 along the XXth century ?)

REF: *les précipitations acides agissent sur les écosystèmes lacustres depuis 75 ans, soit depuis l'essor de l'industrialisation et du transport automobile.*

(Acid rains have been influencing lake ecosystems for 75 years, that is since the surge of industrialization and automotive transportation)

- Textually Implicit:

Here the question is textually implicit in that the student has to link several sentences to answer the question.

(3) Q: *Pourquoi les voix en faveur de l'abolition du bilinguisme se multiplient-elles?*

(Why do voices in favour of abolishing bilinguism multiply?)

REF: *Le mot d'ordre de ses partisans:* English only! [...] *Autrement dit, que les immigrants apprennent l'anglais et s'assimilent. Du coup, les voix en faveur de l'abolition de l'enseignement bilingue se multiplient.*

(the byword for its supporters : *English only !* [...] In other words, let the immigrants learn English and assimilate. In consequence, the voices in favour of abolishing bilinguism multiply)

Below, the question is textually implicit in that the question is reformulated lexically and pragmatically on top of bearing on two sentences. Q is a question of manner form while the sentences don't express manner at the discourse level, but an author's personal view in direct style.

(4) Q: *Comment l'auteur juge-t-il l'infanticide sacrificiel?*

→ Que faut-il être vraiment pour sacrifier son fils à Dieu?

(How does the author judge sacrificial infanticide ?)

(What must one really be to sacrifice his/her son to God ?)

REF: *Vouloir sacrifier son fils à son dieu. Il faut vraiment être primitif.*

(To want to sacrifice his/her son to god. One must really be primitive.)

As the system knows which segment(s) of text to attach to which question, the level of reformulation is as desired by the composer of the question. In cases where the question should be automatically traced to the text, in an effort towards full automation, questions should follow the text's form more directly. This means that the lecturer, or the person who produces the questions, has to do the task of connecting the two when creating a question (see section II.3.1.A).

There is another feature over which to categorize questions – but as this does not really have implications for the possibility of automation, this aspect has not received as much attention. The categorization holds on the communication aim of the sentence from which question and answer are drawn; it covers categories close to the above for the texts themselves: cause-effect (and causal antecedent), problem-solution and identity. Typically, cause-effect questions follow a *Pourquoi X?* pattern, problem-solution follow a *Comment X?* pattern, while identity questions follow a *Qu'est-ce que X?* pattern (after Johnson & Johnson, 1986).

III.2. Good and Bad Answers

III.2.1. What's in an answer?

Answering the questions from the part of the student/trainee amounts to selecting one sentence (textual explicit) or a set of two to three sentences (textually implicit) and transform that material into a one-sentence answer. Turning the text's sentences into answers can be usually done with

just "pasting" the REF sentence into an answer slot, though the students are encouraged to reformulate REF's material. To reformulate REF can bring extraneous textual material or leave aside some of REF.

III.2.1.1. Reformulation

In March 2005, a small experiment was run with a group of Alliance Française FSL students – another one was run in February 2006 with students from U of Ottawa's Second Language institute. In order to evaluate the results of the implemented program, one last run of experiment has been conducted in November 2007 with a new group of students from U of Ottawa's Second Language institute. The procedure is described in more details in section VI.4.1.

In all cases, students were asked to read a text randomly picked and answer a question based on that text, following the Textual explicit / implicit pattern. It appeared that the groups were almost equally divided between those who chose not to reformulate and those who decidedly did (no indication to this respect was given to the students).

Due to the simplicity of the question-answering technique (one sentence in text as an answer, at least for Textually Explicit questions), reformulation on the part of the students occurs as expected on the linguistics rather than on the content. Below are two examples showing the highest degree and lowest degree of reformulation – this, in terms of distance from the student's answer to the original REF answer sentence. Differences can occur due to syntactic change, or semantic change where the reformulation tends to reduce a sentence's clause to its main idea only – REF stands for Reference sentence, Q for question and S for student answer. Difference can also be due to the presence of error, even when directly pasting the answer.

III.2.1.1.1 Reformulation in a 3-sentence TI question

(5) REF: *Le nez est un organe très spécifique chez l'homme, et il nous inspire pas mal de méfiance. Y toucher dans le cadre d'une rhinoplastie, c'est s'attaquer au centre du visage*

et à l'un des fondements de l'identité. Bien souvent, ce type de demande masque des problèmes psychologiques graves, de l'ordre de la schizophrénie.

(The nose is a very specific organ in men, and it arouses suspicion. To touch it through rhinoplasty is to tackle the middle of the face and one of the foundations of identity. Very often, this kind of demand conceals serious psychological problems, in the order of schizophrenia.)

Q: *Pourquoi une demande de rhinoplastie masque-t-elle des problèmes psychologiques graves?*

(Why does a demand for rhinoplasty conceal serious psychological problems ?)

S: *Le nez est au centre du visage et il est très lié avec l'identité.*

(The nose is in the middle of the face and it is strongly linked to identity)

This is a good answer, though *lié à l'identité* is more correct than *lié avec l'identité*. Here the student made the effort to synthesize in order to produce an answer. As close as it appears to REF, this sentence represents the highest degree of admissible reformulation shown in the experiments (although we did admit as good answers sentences showing still higher degrees of reformulations, such as full abstractions – but these slightly violates the rules of the exercise).

III.2.1.1.2. No reformulation in a one-sentence Textually Explicit question

(6) REF: *les précipitations acides agissent sur les écosystèmes lacustres depuis 75 ans, soit depuis l'essor de l'industrialisation et du transport automobile.*

(acid rains have been acting on the ecosystems since 75 years, that is since the surge of industrialization and automobile transportation)

Q: *Pourquoi l'acidité des lacs de l'Outaouais s'est-elle multipliée par 10 au cours du XXème siècle?*

(Why has the acidity of the Ottawa region multiplied by 10 in the course of the XXth century)

S: *à cause de l'essor de l'industrialisation et du transport automobile.*

(Since the surge of industrialization and automobile transportation)

The student has simply pasted the answer directly from the text, and used a prepositional anchor to link the form of the question to the answer. Here, the distance between S and REF is zero and content correspondence is fine.

These two examples are just meant to show what an answer can be, as well as to exemplify the idea of reformulation – which can be *acceptable*, as in Example 5, or *necessary*, as in Example 6. Reformulating sense is usually optional, but reformulating some grammar can be compulsory as Ss are evaluated against REFs in terms of sense, but possibly against Q in terms of form.

III.2.1.2. Extraneous or missing material

Whether sense is reformulated or not, answering a question can introduce extraneous or missing material. For instance, answering Example 6 question by *à cause de l'industrialisation* involves no reformulation in sense, but still shows missing material. Conversely, reformulating in S the sense of REF can bring on extraneous material, as in the following (errors in S are left untouched).

(7) REF: *[...] et d'un chaperon qu'il lui met pour éviter les distractions de son œil de faucon pendant le transport jusqu'au lieu de «travail».*

([…] and with a hood he puts on the animal to spare him the distractions of his hawk eye during transportation to the work "place")

Q: *A quoi sert un chaperon?*

(what's the use of a hood ?)

S: *Éviter les distractions de son oeil de faucon pendant le transport,* ***donc sa aide avec les distractions****.*

(To spare the animal the distractions of his hawk eye during transportation, so **its help** with distractions)

Here, the clause *donc sa aide avec les distractions* is extraneous to REF. Most of the words in this clause are grammatical words and can be processed through lists from the lexical resources, with the exception of *distractions* and of *aide*. *Aide* is unknown to the system as it appears nowhere in REF: the system has no means to decide whether *aide* is an appropriate word for an answer in this context.

Note, first, that extraneous material's importance to the meaning of the sentence is harder to analyze than missing material, and that whether missing or extraneous, this material can have varying syntactic relationships to the core of the sentence (what appears in REF). As we have little possibility to grasp its sense through automated processing, this kind of material is to be evaluated with respect to syntax only, verifying how is the material linked to the core of the sentence.

This means that sense is left without evaluation, and that extraneous material can only be processed for evaluation of grammar.

III.2.1.2.1. Extraneous material

a. Syntax

The syntax of extraneous material is to be evaluated in itself and with respect to the rest of the sentence.

i. Relationship

The extraneous material can be included syntactically in the core of the sentence, which means that removing it would make the sentence ungrammatical, or it can appear as sentential complement, or even just verbal complement, and removing it would leave the sentence grammatically correct, as it is the case in the above example.

ii. Constituency

Supplements can be themselves grammatically correct or not. This aspect is more important in cases where the supplement is not mandatory to the sentence's grammatical correctness, and where it appears as a sentence/verbal complement. When supplements are adjuncts to

constituents present in REF, evaluating their grammatical correctness amounts to evaluating that of the constituent to which they are bound, which is an instance of REF, even when reformulated – as extraneous material and reformulation are two different things.

For instance, in Example 7, the extraneous clause is grammatically incorrect, and this is graspable through automatic analysis: if *aide* is a noun, then the possessive pronoun is incorrect in gender, and a noun cannot receive a noun complement beginning with preposition *avec* introducing a determined NP; and if it is a verb, the student selected the wrong subject through an error of homonymy, and *aider* is not to receive a verb complement beginning with preposition *avec*. The information necessary to draw these conclusions automatically is part of our resources[5], but these resources are very unexploitable due to the possible optional nature of prepositional phrases (a verb, noun or adjective's preposition subcategorization restrictions are only valid for compulsory complements).

In any circumstance, extraneous material calls for heavy processing as the system has no reference with which to draw comparison. This has the advantage of actually assessing the supplement's linguistic structure, and teaching the student proper grammar usage. In this perspective, apart from "bad faith" answers, the student has enough good sense to feel whether the supplement is in accordance with the rest in terms of sense. Recall that this is supposed to help formative, not summative assessment.

b. Lexicon

The vocabulary used in supplements cannot be evaluated with respect to sense, but it can still be evaluated against given broad semantic categories: attribution and state. These categories are often used discursively in utterances. Attribution can be formulated through verbs or prepositions. Verbs are content words and prepositions are function words – although the distinction is not so clear, as rich prepositions are often seen as semi-content words. Therefore, verbs or nouns can be used to reformulate prepositional phrases, without implicating adjunction of semantics in the sentence. The same is quite true for the state semantic relation, which can be used to enrich the form of a sentence without really modifying its sense.

---

[5] This kind of information can be accessed through XIP lexicon file or through Memodata synonymy bank. Precisely, here, *aider* could only receive an *avec* verb complement if it were a manner circumstancial. Manner complements reject determined nouns: *avec grâce*, *avec bonne humeur*, etc.

So, we have planned to model the forms of attribution and state so as to enable the recognition of lexical supplements that do not really change the meaning, or semantic scope, of a sentence, but rather, reformulate or enrich phrases carrying meanings otherwise equivalent to the reference. But the processing of such lexical reformulations is superficial, so we have not planned to go beyond treatment of state and attribution.

For now, this is as far as we plan to go on the question of extraneous material. In conclusion, and to replace the above in a more general perspective, the procedure of assessing extraneous material works on assessment of syntax, or on the recognition of superficial lexical state and attribute reformulations.

III.2.1.2.2. Missing material

Processing missing material is simpler. The student has the freedom to reduce REF to a minimum. Therefore, in terms of sense, as long as an answer exhibits the nucleus of REF, the answer is considered correct. The nucleus corresponds to the syntactic nucleus of REF[6], with respect to what has already been expressed of the nucleus in the question. In this respect, there is no need to distinguish lexicon from syntax when dealing with missing material. In the context of a triangular relation between REF, Q and S, a S nucleus is composed of a noun at the least, and of a VP + NP structure at the most, depending on what has already been expressed in the question. A nucleus, like a verb in a simple sentence, can be modified, through noun, verb or sentence complements, or a nucleus can be coordinated, with another part of the sentence which then constitutes another nucleus within a complex sentence. Anything that is a modifier can be discarded, while anything that is a coordinate to the nucleus has to appear in S.

Back to Example 6, the answer *à cause de l'essor de l'industrialisation et du transport automobile* is maximal, *à cause de l'industrialisation et du transport* is minimal, and *à cause de l'industrialisation* is insufficient, even though acceptable.

[6] The nucleus of a sentence is formally its syntactical head, and is detected in XIP's analysis as the top-most lexical head or clause in the dependency analysis.

III.2.2. What's in an error?

This question has already been loosely defined when studying the examples. But this should be carefully stated, and categorized, as the error types determine the structure and content of the feedback. The two following parts deal with the form and nature of errors. This part is concerned with form of errors in the context of this project: how they show on surface; the next part discusses the nature of language errors to be expected based on a typology of English native FSL students.

There are two general categories of errors: content and form (or *content and style*, in the literature). There is a distinction between form with respect to language and form with respect to the question's formulation. All these are examined in turn.

III.2.2.1. Content

As we said, an answer can be insufficient, minimal, maximal or partial. What is coming on top of that falls beyond the deterministic analysis of content, as the system simply cannot do it. Therefore an answer can show erroneous content in two ways: as incomplete or as wrong. To be judged incomplete, an answer must lack at least one of REF's nuclear elements. An answer can be judged wrong in two ways, according to observations:

A. the answer is drawn outside REF's boundaries, using a segment of the text that simply does not answer the question. Whether the student has been fooled by surface indices or simply answered randomly is interesting to know, but there is no plan for now to develop this.

B. the answer has been selected in REF, but as either wrongly put in correspondence with the question or as building on a misunderstanding of the question. The two causes are hardly distinguishable, but they are nevertheless two distinct causes of error. Here is an example to make things clear:

(8) REF: *la présence de sang et de protéines dans les fèces humaines est un indice de nombreuses maladies, par exemple des saignements associés à un ulcère.*

(the presence of blood and proteins in human faeces is an evidence of numerous diseases, for instance bleeding associated with an ulcer)

Q: *Qu'indique la présence de sang et de protéine dans les fèces humaines?*

(What does indicate the presence of blood and proteins in human faeces ?)

S: *Des traces de myoglobine humaine dans un coprolithe indiquent la présence de sang et de protéines dans les fèces humaines.*

(Traces of human myoglobin in a coprolyth indicate the presence of blood and proteins in human faeces)

Here, REF refers to the portion of the text that constitutes the reference answer, Q refers to the question and S to the student candidate answer. These abbreviations will be used from now on.

III.2.2.2. Form

An answer that is wrong in form can be so in three ways: lexically, syntactically and with respect to question formulation. Of course, the three can overlap, specially as a lexical error can result in bad syntactic structure. Errors that put an answer in correspondence with question formulation represent the lowest degree of error, and do not produce syntactic errors. It is a special case of lexical error, contextual.

The seriousness of the errors follows the informal rule:

Question_lexical < Lexical < Syntactic.

III.2.2.2.1. Errors with respect to question formulation

Producing an answer while following the question's formulation can imply a slight modification of REF's answer segment, depending on the lexical form of the question, as seen in Example 6. More or less, each question can play on this slight manipulation, and can be formulated in order to call or not to call for modifying REF's answer segment. The interest of the variations (of question formulations) is to use as many interrogative adverbs and pronouns as possible, whose

mastering in French can be difficult. Particularly, average-level students find it a challenge to deal with the interrogative pronouns *lequel*, *duquel*, *quel*, etc. Questions formulated accordingly can always be replaced acceptably by similar questions beginning in *que/qui* or *qu'est-ce-que/qu'est-ce qui.*

III.2.2.2.2. Lexical errors

Lexical errors can occur in many ways – the above being a special case. Sometimes they interfere with proper lexical analysis, or not. Lexical errors can all be placed in the three categories shown below, which are not to be confused with the categorization of English native FSL student's language errors presented in section IV.1. Here, errors are categorized as they appear, while in section IV.1, they are categorized as they are.

- Right word, wrong PoS:

E.g. *Le **cardio-vasculaire** d'un rat* [...] (a rat's cardio-vascular [...])

Student made the error of directly translating from English to French: in French *cardio-vasculaire* can only be an adjective. This has no negative effect on parsing, as XIP considers the word under its actual PoS during tagging, but treats it as a NP item during parsing: an example of XIP's robust parsing.

Example: *un système cardio-vasculaire très **semblant que** celui* [...]

(a cardio-vascular system very **similing** that [...])

The same kind of error, but doubled, and having negative impact on parsing.

Typically, this kind of error has no effect on parsing provided the error does not occur on a grammatical word. These errors are easy to detect.

- Wrong word, good PoS:

Example: *c'est le **temps** dans la pratique médicale* [...]

(it is the **time** in the medical practice)

The student made the effort to reformulate, but replaced *période* by *temps*, which is incorrect. No impact on parsing, but difficult to detect. Some function word errors, such as preposition errors fall within this category.

- Wrong word, wrong PoS:

Example: ***Un retour*** *leur hermaphrodisme*. (A return their hermaphroditism.)

In such cases, it is as impossible for the machine as for a lecturer to understand what the student has been trying to say. The parse is done, but meaningless, as it barely goes past PoS tagging – in such an example the only parsing will consist in binding the nouns to their determiners. Spelling mistakes are a special case of this category, as a wrongly spelled word appears as non-existent.

III.2.2.2.3. Syntactic errors

There are four superficial forms of syntactic errors: 1/ error due to a lexical error, 2/ error due to reproducing an English structure, 3/ error due to wrong word order, and 4/ error due to one or more missing or additional word.

The syntactic errors can result in halted parsing. Detecting the errors is easy when parsing is halted, but more of a problem when parsing goes on. Thankfully, the more serious the error, the greater the chances of it to halt parsing. So, eventually, when parsing has not been halted, the errors should be detected by heuristics based on grammar rules evaluated through the words' grammatical features.

Another source of error comes from lack of agreement, whether subject/verb, determiner/noun, or noun/adjective error. Although these have no negative impact on parsing, it is not very clear whether these errors originate from syntax or lexicon – but this is of little importance. Besides, these are the easiest errors to detect automatically.

Chapter IV. Free-Text Answering and NLP

In chapter II, we have seen the relation of this project to the domains of Information Technology as CALL and CAA, and in chapter III, its relation to didactics and linguistics. The relation of this project to Computational Linguistics or Natural Language Processing is presented in this chapter. It is preceded by a section on language errors and Error-Analysis. EA is an instance of corpus linguistics, and although it is of primary interest to teachers, it is necessary to any attempt at designing automatic procedures to correct language errors, as in a CALL context, and it requires means of NLP, mainly parsing, to be exploited. This is the reason of this section in this chapter.

This chapter does not present solutions, but rather, the problems encountered in this work as seen from the side of NLP – even when the link is indirect, as it is the case for EA and language error correction. A description of the solutions used for this work is presented in the next chapter.

Eventually, a case is made for a syntactical approach to the problem of free-text assessment in the context of CALL in the last section.

IV.1. Student's Language Errors

In such a task as assessing automatically student's free-text input, it is mandatory to know in advance the kind of language errors to expect. Teachers and lecturers, or so we found by experience, tend not to classify the errors they find in given error categories, or very loosely. Rather, the errors are corrected one-by-one, and the weight of categorized errors in the overall marking of a drill or essay is subject to 1/ the total amount of errors, 2/ their variety, and 3/ the

general quality of the drill or essay. A good student will always pay more for a error which is one error among good answers/material (in a given category) than a poor student who constantly repeats the same error pattern. This is true to such an extent that a good student will be judged on all the errors she committed, while a poor student would be judged more on a loose general ratio of errors with respect to good material. Typically, in a poor student's essay or drill, lots of errors remain uncorrected, as the teacher gets tired of correcting them. In this respect, error categories are of variable importance, which may explain why the teachers make little use of such formal categories of error other than informally.

Hence our system is formative rather than summative: to train rather than to mark – marks are too sensitive a problem to treat it as deterministic only. The activity of marking is too much an instance of compromise politics to be left to a computer, at least for the present time.

But on the other hand, due to focus, speed and variable-geometry physical space[7], a computer is a proper medium to perform formative assessment – perhaps better than a teacher, who can hardly split the time unequally between students (a stereotype is that of the teacher cursing a student who causes him/her to spend more time on this student's sheet than on others due to overload of errors – resulting in even poorer evaluation). Plus, a computer has no problem with being repetitive. A computer would be naturally driven to spend more energy on bad students than on nerds – that is, to the ones who need it the most.

Summative and formative assessments tend to overlap, in practice. A good example of this trend is the possibility for a lecturer or teacher to obtain individual students' reports on formative computerized activities. Having well-defined categories therefore also enables both trainee and trainer to observe a topical approach to progress, based on given categories of linguistic problems.

Using automation requires examining errors under strict categorization, were it only in order to provide the machine with proper depiction of the errors, to be able to actually catch them, and store them. In a human perspective, the interest is to lay ground for complete, formal and detailed assessment, such as a teacher could not provide, without sending the student to reference manuals. Here, the computer can act as both the teacher, or evaluator, and the reference manual.

---

[7] Here we mean that a computer can *naturally* allocate more time to process some student's answer, when correcting it. The speed of a computer enables vast discrepancies in per-student time allocation without them noticing.

But there is a major drawback: that not all categories of error are suited to formalization or lend themselves to automatic detection.

IV.1.1. Error Analysis

Error Analysis (or EA – see Heift & Schulze, 2007, for an overview) consists in categorizing errors, and it has little to do with computation. Studies in EA concentrate on the study of errors with respect to language learning; therefore it is often understood as categorizing errors, as made by learners 1/ of a target language, and 2/ coming from a given linguistic background. Errors as thought of by EA are conceived in an inter-linguistic perspective. We can see from the categorization in next section that some types of errors are indeed motivated by the fact that learners are Anglophones. However, a problem with EA is that it only records acknowledged types of errors, falling under known categories (for instance agreement errors). So, EA can fail to categorize some kinds of errors – or errors can lend themselves to categorization only ambiguously.

But error classifications are used in CALL, and in Language Acquisition in general. S. Granger (2003) has used a set of error categories to annotate the FRIDA corpus (French Interlanguage Database). This fine-grained categorization contains nine error domains subdivided into categories. The domains comprehend Form, Morphology, Grammar (for instance agreement or voice errors), Lexical, Syntactic (for instance word order errors), Register, Style, Punctuation and Typo errors. The purpose of the annotated corpus is to produce a typology of frequent errors to tune an error-diagnosis system based on a parser within the FreeText project, as the corpus is too small to train a true classifier. D. Kies (2007) has produced another typology of frequent errors in order to test grammar checkers, such as programs embedded in the most popular text editors (Word, Wordperfect, etc.). This typology contains only the 20 most frequent errors, as made by learners of English-as-a-Second-Language (ESL). D. Anctil (2005) has produced a similar typology, but exclusively to address lexical errors.

In spite of the resources cited above, we have chosen to create our own error classification. The absence of any consensus on the question of EA is the main reason for that. Heift and Schulze (2007) also identify a lack of common ground in this domain. They blame equally the lack of

coverage and homogeneity in the corpora used to record errors, and the problem of classification characteristics. We use this judgment to advocate a personal approach to the problem. Particularly, the fact that the corpus used for the error classification below comes from the same environment as our test set of answers seemed relevant (intermediate-advanced students from University of Ottawa's Institute of Second Languages) in the absence of consensus. This also had the advantage of confronting us repeatedly with erroneous material, and therefore helping the design of solutions.

However, a much larger study is required to pass definitive judgment on this type of error correction – probably a study lager than any existing – and this study does not claim to establish a standard. Ultimately, in the rise of interest for error detection and correction for CALL, a comparison of the various classifications existing to-date should be a necessity.

IV.1.2. Error Categories

To establish categories, an ensemble of student's session's test material has been gathered by this researcher. It is composed of about thirty individual students' booklets containing each all of what a student has been producing in terms of drills and essays over a full session (3 to 4 months) – so-called "journals". The level of the students is intermediate-advanced, corresponding to the target population of the system. Errors have been extracted from the booklets, and categorized. It is expected that the range of error categories gathered this way should be larger than in one-sentence answers to questions-over-text, as, in essays, students express to full capacity (or incapacity) their linguistic skills. Therefore, the categories established through this means are more indicative than anything else, and we feel free to manipulate them in the actual implementation of the system. They represent however the full scope of what is to be tracked dynamically, as far as possible.

Two kinds of errors are not accounted for in this study: word order errors and homophony errors. There are two reasons for that. The first is that these have shown insignificantly in the learners' corpus, at least in what concerns main constituents order: subject, verb and complements. The ambiguity of word order inside phrases is another reason. For instance, in French, adjectives and adverbs can appear before or after nouns and verbs respectively, depending on the adjective or

adverb itself – most can appear equally well before or after. For instance, *un énorme problème* or *un problème énorme* are equal, while *un petit homme* or *un homme petit* are very different. The same is true for adverbs to a lesser extent: most should appear after the verb, but there are lots of semantically motivated exceptions, depending on the sense of the sentences or on the semantics of the adverb itself. We have therefore chosen to leave this problem aside for now, in the absence of a lexical knowledge base, or model, describing the behaviour of such words. Errors in word order happening at the level of constituency produce either ambiguity (such as in the assertion/question subject-verb inversion), or a break in the syntactic analysis, which will then appear as a syntactic error. Errors of homophony are quite common in FSL – FreeText error detection module is good at detecting them. To detect or correct errors of homophony, one needs a dictionary with a repertory of homophonic form. Homophonic errors are detected as a syntactic error in the parsing, as homophones have different PoS. The correct form is sent as feedback to the user when she commits an error by way of PoS replacement. We do not have such a dictionary, and this has been left aside for the moment.

Finally, some types of errors are not accounted for due to the difficulty to place them in a grammatical category. Such is the case with adjuncts – to incorrectly add a word in the sentence. In such a case, the category of the error depends on the word's P-o-S. Consider the example:

Elle ça l'intéresse

(She it interests her)

The student has put in two subjects, and there is an ambiguity in the feminine personal pronoun *elle*, which could be topicalized should it be separated from the rest of the sentence by a comma. This can yield severe errors and should produce breaks in the analysis. It is therefore treated as a high-level error and participates in the complexity of the kind. Again, high-level errors are difficult to name with a grammatical category and should therefore constitute a research subject.

There are all-in-all 12 error categories, which are summed up below. The format is *error* → correction. Then, the *NLP* heading introduces a note on how the problem could be addressed in the context of automation. The fragments are not translated as they are erroneous.

1. Preposition errors in front of noun or verbs

Due to the polysemy of prepositions in general, and kinship of English to French, students are driven to errors when using prepositions to introduce nouns such as names of locations, or verb complements.

(a) e*n Ottawa* → à Ottawa

(b) *ceci est facile de comprendre* → ceci est facile à comprendre

NLP: This is a serious problem in error diagnosis – only to be addressed by corpus-based methods, at least (see Hermet and al. 2008).

Calls for a systematic procedure.

2. Confusion over tense and forms of verbs.

Given genres of discourse call for given verb tenses and forms – this commands, for instance, the usage of simple past in French as dated, or very specific. This also governs tense concordance.

(a) *mon école s'est appelée [...]* → mon école s'appelle [...]

(b) *il y a fait des achats et elle l'ai reconnu* → ... et elle l'a reconnu

NLP: The pragmatic aspect of the problem is not really solvable (case a), but the tense concordance is (case b). However, in case b, the first verb must be retained as the reference, which induces a bias.

Calls for a systematic procedure, or else the parser should be modified to signal an error.

3. Errors in verb agreement

There are roughly four kinds of such errors:

- a/ picking up the wrong agreement with subject
- b/ taking one verb form[8] for another (like between 2nd and 3rd group)
- c/ tendency to generalize conjugation over one verb person, typically, that of singular 3rd, or using the infinitive as past participle to skip the trouble of putting it in agreement with auxiliaries.

[8] In French, verbs are separated in 3 groups according to their morpho-syntactic behaviour: 1st group for verbs ending in *–er*, 2nd group for verbs in *–ir* and 3rd group for other verbs.

- d/ wrong agreement between auxiliaries and past participle, especially with *avoir*

NLP: Correcting case errors is straightforward – note that case c is an instance of case a, except for the infinitive / past participle generalization problem. Case b is actually a spell-checking problem, as conjugating a verb over forms that do not exist results in an unknown word. Case d is also straightforward, although slightly more complicated than case a, as possibilities of anteposed object complements have to be examined.

Calls for a systematic procedure, or else the parser should be modified to signal an error.

4. Low-level syntactic errors

These are syntactic errors which do not affect the overall syntactic well-formedness of a sentence.

(a) *n'oublier pas* → ne pas oublier.

(b) *qui veut s'en profiter des autres* → qui veut profiter des autres.

NLP: These can be addressed by heuristics, but correcting all such possible errors would require a complete error grammar, which we do not have – so the solution can only be partial.

See 5.

5. High-level syntactic errors

The errors are affecting the sentence well-formedness

(a) *quand elle c'était nécessaire* → quand c'était nécessaire

NLP: The same as 4, but in the absence of an error grammar, it should be treated as a parsing problem based on the detection of breaks in the analysis. To some extent, this is also true for some cases of 4, as in practice cases 4 and 5 can amount to the same. Actually, cases 4 and 5 show errors of similar kind, but ranging on a scale of increasing seriousness.

The errors should trigger a break in the analysis, therefore the procedure is restricted to an error alarm.

6. Anglicism – clause

The student pastes an English form over French, resulting in a French sentence with English syntax.

(a) *Ma mère et ma grand-mère les deux [...]* → My mother and my grand-mother both [...]

NLP: These can be addressed by corpus-based methods, or error grammars (which eventually amounts to the same as error grammars should be based on corpora) – only, the corpus or grammar should be large. So, this is perhaps a research problem in the domain of FSL.

The errors could trigger a break in the analysis, and the procedure is restricted to an error alarm. If the error does not disturb the syntax, then a systematic procedure is needed.

7. Anglicism – lexicon

The same, but only at word level.

(a) *une salle foncée* → A dark room

NLP: The same as 6, but the problem here is simpler – with the proviso of an included annotated corpus.

Calls for a systematic procedure.

8. Generalization of a lexeme over one of its PoS

The student always uses the same lemma PoS realization to make use of some lexeme, for lack of knowing the appropriate form, regardless of the syntactic context.

(a) *elles étaient la Belgique* → elles étaient belges

(b) *nous sommes déménagement* → nous avons déménagé

NLP: This might cause ambiguity, if the mistaken word form is also subcategorized by its governor (as in the two examples). Then, the sentence would look correct. Otherwise, the error should cause a syntactic error. In all circumstances, some correcting procedure could be provided, based on derivates. But here, the error could only be detected with respect to a reference – as in itself the segment could look correct.

The errors should trigger a break in the analysis, therefore the procedure is restricted to an error alarm, or else it is undetectable.

9. Lexico-grammatical errors

The student makes wrong use of a word due to lack of knowledge of its behaviour with respect to grammar.

(a) *mes erreurs étaient toujours les mêmes problèmes* → mes erreurs étaient toujours les mêmes.

(b) All-time classic: *de le* → du.

NLP: This can be addressed by heuristics, or error grammars, but this is all error dependent, as some errors are easy to detect (case b), or impossible (case a).

Calls for a systematic procedure.

10. Lexical errors

Student does not know a word and uses another one, or confuses homonyms.

(a) *si je recevais l'emploi* → si j'obtenais l'emploi.

NLP: Corpus-based only. And yet the error could also cause ambiguity.

Calls for a systematic procedure.

11. Pragmatics

Pragmatics errors, of word usage, or of clause form – difficult to automate but nevertheless very real.

(a) *c'est très important aussi que [...]* → il est également important que [...]

NLP: This could be addressed by heuristics or corpus-based methods – and here, ambiguity should be less of a concern, as the particularity of these errors is that the form is correct but the probability of occurrence is very low, and only possible if the corpus contains improper language.

Calls for a systematic procedure.

12. Agreement errors

Very common errors of confounding a gender for another, or of missing number agreement. Note that these errors are presented independently of verbal agreement errors. We believe verbal agreement errors belong to the category of conjugation errors, which is conceptually different. Conjugation errors are agreement errors, but the reverse is not true. Henceforth, both classes of errors will be presented as agreement errors – and in fact, the solution to detect them remains the same.

(a) *quelque chose d'effrayante* → quelque chose d'effrayant

NLP: Straightforward.

Calls for a systematic procedure, or else the parser should be modified to signal an error.

Eventually, language-specific errors are common in Second-Language-Learning. By this term, we refer to errors committed by applying forms of a native language to a second language. Anglicisms above are an example of this kind of errors in FSL. These errors can fit in a strictly linguistic category, as they usually translate into the violation of a syntactic form in the second language. But the means to correct these errors are first-language dependent, and should motivate specific research.

IV.2. Semantic Similarity

This project of assessing free-text answers to questions using unique textual references falls into the domain of Semantic Similarity. Semantic Similarity focuses on finding computational techniques to assess the similarity of texts fragments. We can roughly distinguish two approaches to solve this problem: purely statistical on the one hand and corpus-based or Knowledge-Based methods on the other.

Texts, or fragments of texts that can be compared by lexical set difference, are an illustration of the first method. On the other hand, words can only be compared using corpus-based or knowledge-based methods to judge of their degree of similarity with other words.

Semantic Similarity is currently used in Automatic Summarization, Question Answering, Information Retrieval and Entailment. Besides, it is used as well in the field of CAA, although not all CAA systems refer to semantic similarity as a parent field of research (they actually rarely do: Burstein and al., 2003 and Landauer and al., 2003, refer to – the others do not). Some projects in CAA come from semantic similarity (Landauer and al., 2003), while at least one project from Intelligent Tutoring makes its way into semantic similarity, in the field of Entailment (Rus and al., 2005).

Here, the review deals with English as very little has been done outside English, or otherwise the techniques are extended to other languages following what has been done for English.

IV.2.1. Words

IV.2.1.1. Semantic Similarity

The notion of semantic similarity of words is very restrictive and amounts to detecting or acquiring links of synonymy, or to links of hypernymy/hyponymy between candidate terms. Or at least it should, as we will see. In NLP, this field of research has concentrated around the means necessary to process such formal semantic knowledge resources as WordNet (Miller, 1995), or domain-specific thesauri such as the Medical Subject Heading thesaurus (Schulman, 2000). Ontology analysis and modelling is also a domain related to semantic similarity, although the two are not explicitly associated. The aim of this field is nevertheless to build resources of formal description of a domain's lexicon. A detailed review of ontologies won't be given here as this is a vast subject (see for instance Sowa, 2009, for an overview of the concept). However, Ontology modelling only pushes further the techniques employed in Semantic Similarity (see for instance the relation between Milne and al., 2006 and Gabrilovitch & Markovitch, 2007, both using Wikipedia as a knowledge reference: the first to acquire ontological relations between concepts, the second to put semantic tags on concepts).

On the side of theoretic semantics, the NLP approach to semantic similarity has attached the notion of lexical-semantic similarity to unequal relations, as synonymy and hypernymy/hyponymy – synonymy shows strict similarity, while hypernymy/hyponymy shows

only partial similarity. Therefore, the research has also shifted to investigating the field of Semantic Relatedness, which takes interest in the closeness of terms rather than in their strict similarity. Now, this notion of semantic relatedness has subsumed to a large extent that of semantic similarity, so that they are largely confused in the literature. Morris (2004), brings a lexicological point of view on the question, and distinguishes between classical and non-classical lexical relations. Classical relations refer to synonymy, hyper/hyponymy, meronymy and holonymy, as well as antonymy by extension. This distinction is confused in NLP applications through the notion of semantic relatedness.

IV.2.1.2. Semantic Relatedness

Semantic Relatedness is more complex than semantic similarity. In SR, the terms are evaluated along a quantitative rather than qualitative line, and the relations between words are not named following a semantic terminology, as it is the case in semantic similarity with synonymy or hypernymy/hyponymy – although some formal semantics relations as holonymy and meronymy encompass aspects of semantic relatedness – hence, the notion of semantic distance (Budanitsky & Hirst, 2006, for a mention on the difference between similarity and relatedness). In a review of an article presenting a LSA-driven study of the impact of measures of word co-occurrence on the quality of *similarity* (Lemaire & Denhiere, 2006), Turney implicitly highlights the difference between relatedness and similarity in the expression of quantity, and by consequence the greater suitability of statistics to assess the first (Turney, 2007).

SR makes heavy use of corpus-based methods, although methods relying on paths lengths and patterns have been used to detect relatedness using such knowledge-bases as WordNet; interestingly, in the same line, WordNet has also been used as a corpus through the glosses associated with the words (see Pedersen and al., 2004, or again Budanitsky & Hirst, 2006, for an overview on computing relatedness with WordNet). In corpus-based methods, candidates are semantically associated in a term/document matrix based on their tf.idf values (Sparck Jones, 1972) within a set of documents – a threshold on this tf.idf value yields a set of terms related to the candidate. This value highlights the tendency of candidates to appear together within textual frames of selected topics – common words are well spread across documents while specific terms are local to given topics: comparing their value within one document over the set enables

to establish association. Turney has tested the use of pointwise-mutual-information (PMI) to compute similarity between two words queries using the web as a corpus, but this has explictly been cast as to assess *similarity* (Turney, 2001).

The scientific value of this association as a valid semantic relation depends in turn on the referential value of the corpus. Here, the quality of a corpus depends on its size or/and on its content.

Semantically generic corpora are used, where the quality of the associations comes from the size of the corpus as well as from the sampling of contained documents. The British National Corpus is an example of such a corpus (Ref British National Corpus). It contains 100 million words to match the quantity criterion, and the words come from samples of a maximum of 45000 words to ensure variety of topics and genres, matching the quality criterion. Corpora can however be more restricted in their content scope, as well as form. For instance, Bodenreider and his colleagues have used the Medline resource to build a 3 million noun-phrases corpus in the domain of medicine, to serve in a thesaurus modelling project (Bodenreider and al., 2002). An example of the corpus-based approach in semantic similarity applications is presented in the next section (Mihalcea & al., 2006). A new approach relies on Wikipedia as a knowledge backbone – a promising approach, as Wikipedia offers both the size and the reference of contents (Gabrilovitch & Markovitch, 2007, or Strube &Ponzetto, 2006). The similar or related nature of the association between two words is only a function of the threshold techniques employed to decide of their association. Explicit Semantic Analysis (Gabrilovitch & Markovitch, 2007) draws relatedness between words based on the occurrence of high tf/idf valued words in given articles. Then, the words are stored into a database as vectors of weighted concepts – the concepts correspond to the titles of the articles.

IV.2.2. Texts

To compare texts, or documents, according to Semantic Similarity amounts to comparing their lexical content, i.e. their words as a set. However, the lexical complexity of texts permits a purely statistical comparison of their contents, making the use of a reference corpus or a

Knowledge-base superfluous. Two or more texts can be compared through the overlap of their content's words (Graesser and al., 2004), as well as through the cohesion of common words – the fact that words common to both texts appear in both texts close to each other. Latent Semantic Analysis (Landauer and Dumais, 1998) is a technique commonly employed for that matter, as mentioned in section II.2.1.1.

Using statistical techniques to achieve text similarity computation has been investigated to the point of practicality – this is for instance the way in which news articles are associated in such programmes as Google News. Google News works on Storyrank, a graph representation of news articles' contents based on PageRank (Brin & Page, 1998). Pushing the technique to finer granularity calls for the inclusion of lexico-semantic relatedness methods, which is the current way of research now. In practice, adding the power of a knowledge base to compute a relatedness value on words strengthen the results on the task of text similarity. Another reason for the addition to measures of lexico-semantic relatedness in Text Similarity is the length problem associated with purely statistical measures (Dennis, 2006, Penumatsa & al. 2004, the latter in the context of ITSs). With the increasing of the lexical space of a given content, the chances grow that another content will yield a degree of similarity when compared to the first (MacCarthy & Jarvis, 2007). The chances that two sentences randomly extracted from the Web have a similarity value higher than 0 are scarce – but spliting the Web in two, and comparing the two halves against each other would produce a value of similarity close to 1.

As for lexico-semantic similarity studies, the development of corpus-based methods has induced a conceptual shift from the notion of text similarity to that of relatedness, and the techniques employed to assess text relatedness follow that of lexico-semantic relatedness (Gabrilovitch & Markovitch, 2007, or Budanitsky & Hirst, 2006, for a counter-example: how Knowledge-based methods are insufficient to capture relatedness). The relatedness of texts is computed over the relatedness of words (Gabrilovitch and Markovitch, 2005). A major difference between computing text and lexical similarity/relatedness is that of categorization. The process of assigning a category to a text is usually supervised and delegated to a classifier, as it is the case in news article classification (see Sebastiani, 2002, for an overview – Wermter & Hung, 2002, for an application to News articles classification). It can however be unsupervised, and the process of assigning a category to a text amounts to cluster a set of texts with respect to their lexical content (Korhonen and al., 2000).

Mihalcea (Mihalcea & al., 2006) proposes a method to compute text similarity. She and her team use the British National Corpus to compute values of semantic similarity between words with LSA, concurrently with computing semantic similarity over WordNet. These values are combined with measuring word specificity using *idf* to produce a value of similarity between segments of words of varying length – whether sentences or texts. Explicit Semantic Analysis is used to convert texts or sentences into vectors of semantic concepts (Gabrilovitch and Markovitch, 2007) – these vectors are in turn compared by cosine values to establish their relatedness or similarity. Another prototypical system, Entailer (Rus & al., 2006), adds Rules of syntactic equivalence to a graph difference similarity measure to assess text similarity. It also uses heuristics to take into account negation and antonymy in the process. Interestingly, Entailer comes from the field of Intelligent Tutoring Systems and aims at *assessing* student's inputs, but it is not concerned by CAA, and even less so in a free-text assessment context – rather, its authors intend to use the system to analyse students' questions in a tutoring environment in order to retrieve documents of interest by measure of entailment. Syntax is coming into the field of Text similarity/relatedness, in applications to entailment especially for the probable reason that the format of texts to be processed in the task are short (see for instance, Wang and Neumann, 2007, or Snow and al., 2006).

IV.2.3. Sentences

Sentence Similarity is strongly related to Text Similarity as a field of research, and is often concealed by TS. Also, just as the newer notion of lexical-semantic similarity overlaps on the classic notion of synonymy, sentence similarity covers, at least partly, the linguistic notion of paraphrase, where it takes particular acceptance as a sub-field in Semantic Similarity. Sentence similarity is a term of common usage in the present state of research (for instance Mihalcea and al., 2006 does not explicitly refer to it otherwise than through Text Similarity, while Wang and Neumann, 2007, refer to it in the domain of Entailment – Li and al., 2006, present it as a full concept), but so are the concepts of Paraphrase Recognition/Generation, while the boundary between the two does not seem to be explicitly drawn. The techniques employed in sentence similarity are similar to text similarity, and often the two notions are confused (again, Mihalcea

and al. 2006, or Gabrilovitch and Markovitch, 2007 – See Selvi and Gopalan, 2007, for an explicit assertion of the case) – at least when sentence similarity is not presented as paraphrase R/G. The three systems presented in the previous section claim to deal with sentence similarity as well. However, as the size of the textual input is very influential on the quality of statistical techniques' results, sentence similarity is never treated solely by statistical methods. LSA does not give good results on sentence similarity, nor does Content Word Overlap (MacCarthy and al., 2007). Here comes the specificity of Sentence Similarity as a field of study. However, as we mentioned, Sentence Similarity amounts to Paraphrase – or, to put it differently, the most ambitious approaches to Sentence Similarity are to be found in paraphrase R/G. Therefore it is more interesting to study it under the paradigm of paraphrase, also as it permits to draw a distinction between paraphrase as seen from Computer Science and from formal Linguistics (which ignores Sentence Similarity). So, we will come back to this problem next chapter on Paraphrase.

IV.2.4. Semantic Similarity and question answering in CALL

In the light of applications to Semantic Similarity (or Relatedness), Question-Answering and Entailment are fields of studies where questions and answers are analyzed for correspondence. In Entailment, this relationship is even assessible through a reference. Although our problem is in some aspects related to QA and to Entailement, there are however two fundamental differences in their relation to this project.

The first is that contrary to QA, in our context the candidate answer is present as much as the question, in the form of the student's answer. There is a trend of research going on in the automatic assessment of answers in QA, using means of Entailment (see for instance the reports of the Answer Validation Exercise, as Peñas & al., 2007), but that is not necessary here as the student answer is only to be compared to a reference segment as an exact match. Therefore, the point of having to match pieces of semantics arises at a level of semantic completeness: the two utterances, student answer and reference in the text, are semantically equal – or at least, the content of the second should subsume the content of the first, as a correct answer can be a partial answer.

The second difference has to do with the nature of the assessment. On top of the semantic accuracy of the answer content with respect to the question comes the problem of the language correctness, or grammaticality. We are not aware of QA systems that tackles the issue of a potentially flawed language of production in the utterances of the answers, and in these are actually two very different way of research (see section V.3 for a presentation of language error detection and correction). Besides, not only should the framework provide means to analyze the language's form correctness, but the analysis should also lend itself to comprehension so as to produce feedback of interest to the student, or user.

IV.3. Paraphrase

The relation of this subject to Computational Linguistics has been presented in the preceding chapter, which leads to the conclusion that our problem is an instance of sentence similarity, or paraphrase in formal linguistic terms.

Briefly, paraphrases are sentences which mean the same while being different in form (see definition in section IV.3.1 below). There has been no theory of paraphrase in language, until recently, and historically, it has been seen as a means of language production more than as a field of language studies. Formal linguistics and computational linguistics have a different perspective on paraphrase, as will be shown in the definition hereafter. There is little communication between the computational and linguistic views of paraphrase. Both act separately, and both have their limitations.

The first approach has been showing promising results in recent years – while the second has barely shown any results in terms of practical applications. But the second truly addresses the question of what is paraphrase – while the first has loosened the premises of the definition in order to accommodate the quantitative bias of its statistical modus, as we will show in the following sections.

In this study, we have chosen to borrow material from the linguistic side to process the recognition of paraphrase. This is far from perfection, but it offers the security of an exact model.

This model is expressed in terms of syntax, which also suits our syntactic approach to assessment. In this chapter, to justify our choice, we will first try to give a formal definition of paraphrase – a task rarely addressed in practical studies on paraphrase. Then we give a review of related work in the domain of paraphrase recognition and generation from the computational and linguistic studies. The exact description of the model on which we have based the implementation is postponed to section V.4.

IV.3.1. A Definition of Paraphrase

Paraphrase is a well-known mechanism in language production: how to express the same meaning with different surface forms. This is a loose definition, perhaps as vague as the field itself, as paraphrase actually offers little formal study, as we will see. Linguistics and Natural Language Processing have different judgements on the nature of paraphrase, and we review them in this section.

IV.3.1.1 Qualitative definition of paraphrase

Even more than syntax, paraphrase is mastered by speakers implicitly – paraphrase is practiced heavily and naturally, but not taught. We subscribe however to the point of view of Igor Mel'čuk:

One of the most important tasks in contemporary theorical linguistics is to design a theory of language paraphrase (1992)

In example 9, the various ways to express the same thing, namely that "Ulysses will come back", are subject to a shift of focus, as well as to different syntax and a change in rhetorics to a lesser extent.

(9) a . *Pénélope est sûre qu'Ulysse reviendra.*

(Penelope is sure that Ulysses will come back)

b. *Pénélope, elle ne doute pas du retour d'Ulysse.*

(Penelope, she does not doubt that Ulysses will come back)

c. *Pénélope croit qu'Ulysse reviendra à coup sûr.*

(Penelope thinks that Ulysses will come back against all odds)

d. *Il est sûr pour Pénélope qu'Ulysse reviendra.*

(it is a sure thing for Penelope that Ulysses will come back)

e. *Selon Pénélope, le retour d'Ulysse est un fait certain.*

(According to Penelope, the return of Ulysses is a sure fact)

f. *Le retour d'Ulysse ne soulève chez Pénélope aucun doute.*

(The return of Ulysses raises no doubt for Penelope)

g. *Le retour d'Ulysse, Pénélope en est sûre.*

(The return of Ulysses, Penelope is sure about it)

h. *Ulysse reviendra: c'est ce dont Pénélope est sûre.*

(Ulysses will come back: That's what Penelope is sure about)

i. *Pénélope a la certitude qu'Ulysse reviendra.*

(Penelope is certain that Ulysses will come back)

Through these examples, we can see that paraphrase authorizes any kind of reformulation admitted by syntax, in order to express a shift 1/ in focus, 2/ in rethorics or 3/ a change in lexicon, as long as no information is lost or added.

In terms of Natural Language Processing, paraphrase is thought (Mel'čuk 1992) to be of primary interest to Language Generation and Translation. Translation can even be seen as "interlinguistic paraphrase" (Milićević 2003, after Mel'čuk and Wanner 2001). Although, some linguists do see paraphrase as one of the major difficulties to be mastered by learners of a second language, we feel that mastering the syntax and the lexicon should be enough: paraphrase follows (Milićević 2003).

Briefly, according to J. Milićević, the study of paraphrase presents the following challenges:

1 – Theoretically, several complex questions arise, such as:

a – the nature of the paraphrasal link between sentences of quasi-equivalence (identity, equivalence, semi-equivalence).

b – contextual neutralisation of semantic differences in the expressions. (in other words: sentences of different forms and contents can be paraphrases within some contexts)

c – the factors at stake in the choice between possible paraphrases (the role of rhetorical and communicative information, of context, etc.).

d – Cognitive and linguistic paraphrase.

2 – On the side of description:

a – to give an account of the variety of the paraphrasal means and to produce a typology.

b – to study those means in an interlinguistic perspective.

3 – formally, to produce tools for the modelling of paraphrase precise enough to guarantee automatic processing of the models.

In order to define and model paraphrase properly, a number of distinctions should be made, as follows:

1. Cognitive vs. Linguistic paraphrase:

Linguistic paraphrases are paraphrases which can be produced or recognized through linguistic knowledge only. Example 9 above shows linguistic paraphrases.

Cognitive paraphrases, or extra-linguistic, can express the same content but differ in their language sense. They necessitate the activiation of extralinguistic knowledge (world knowledge) on top of linguistic knowledge. The example below shows paraphrases of the cognitive sort (adapted from Milićević):

(10) a. One more half-dinar for a piece of bread!

b. The price of the wheat rose by 20% in Serbia.

According to Milićević (2003), a complete model of paraphrase should take into account cognitive paraphrase, as it is frequent, and more importantly, the boundary between linguistic and cognitive paraphrase is narrow. Now, she admits that cognition is too vast for cognitive

paraphrase not to be studied in an interdisciplinary context. In other words, we are sent back to the modelling of the world.

2. Paraphrase as a relation vs. Paraphrase as an operation:

Paraphrase can be thought of in a static way, as a relation of quasi-synonymy between sentences, or in a dynamic way, as an operation of production of quasi-synonymic sentences.

3. Virtual vs. Reformulative paraphrase:

Reformulative paraphrase is an operation belonging to writing and translation (as seen as "interlinguistic paraphrase").

Virtual paraphrase is in charge of reproducing a semantic content through different expressions.

Virtual paraphrase generates, and reformulative paraphrase selects. What is meant here is that there are two mechanisms at stake : one to master the construction of paraphrases, any kind, and one to appropriately select the right paraphrase at the right time.

These are general and theoretic considerations – and this is the point which the study of paraphrase has reached presently. The next section provides an overview of how this point has been reached.

IV.3.1.2. Quantitative Definition of Paraphrase

There is no formal definition of paraphrase in the domain of NLP. For example, in their presentation of the Microsoft Paraphrase Corpus – widely used in the NLP community – Dolan & al. (2004) assert:

The decision to tag sentences as being "more or less semantically equivalent", rather than "semantically equivalent" was ultimately a practical one: insisting on complete sets of bidirectional entailments would have ruled out all but the most trivial sorts of paraphrase relationships, such as sentence pairs differing only a single word or in the presence of titles like "Mr." and "Ms.". Our interest was in identifying more complex paraphrase relationships, which required a somewhat looser definition of what "semantic equivalence" means. In an effort to focus on these more interesting pairs, the dataset was restricted to pairs with a minimum word-based Levenshtein distance of ≥ 8.

As the work of Dolan is acknowledged in NLP as seminal in the domain of paraphrase, this comes as a definition. They add :

In the end, we're tagging something that's not quite paraphrase, but something like "semantic near-equivalence" – sentences pairs that ideally involve complete sets of bidirectional entailments, but which in fact often have some entailment asymmetries or other mismatches. The issue here is when those asymmetries/differences become significant enough to make the pair different enough that you don't think they mean more or less the same thing anymore, where "more or less" becomes a personal judgment call.

This definition of paraphrase is quantitative. The approach is certainly interesting to build corpora with reference material (here sentences extracted from news articles) on which to train paraphrase models, but then the corpus is too small (about 5800 sentences). Ultimately, the notion of statistical near-equivalence is dangerous to use in the assessment of sentence similarity among intermediate-to-advanced SL Learners.

IV.3.2. Approaches in the theory of paraphrase

IV.3.2.1. Natural Language Processing

All studies on paraphrase emerging from NLP are corpus-based, and aim at training models for the recognition and/or generation of paraphrases. We present a few of them in this section – the most important to our knowledge. The first study presented in this section justifies its approach through the theory of linguistics, but the others don't. So, ultimately, the notion of paraphrase as understood here is very broad and amounts to a *least common denominator* – roughly, sentences whose meaning is more or less equal, or of which one's sense is subsumed by the other's.

IV.3.2.1.1. DIRT – Discovery of Inference Rules from Text (Lin and Pantel, 2001)

This work is based on Harris's Distributional Hypothesis which states that the words occurring in the same context should be similar. The algorithm of Lin & Pantel is unsupervised and

automatically extracts lexical patterns of paraphrases or inferences from a corpus of parsed newspapers stories. The patterns are recognized via syntactic dependency paths viewed as binary relationships between two variable pairs of nouns. The assumption is that two different syntactic paths linking the same pair of nouns should represent a pattern of semantic reformulation. Above a certain threshold, a pattern is recorded as an inference rule. For example, their system enables the recognition of paraphrase patterns as:

(11a) X *is the author of* Y → X *wrote* Y

Or so-called *inference* patterns as:

(11b) X *caused* Y → Y *is blamed on* X

This is a simple and promising approach – however, the number of patterns extracted from the corpus is modest and the authors don't provide global results although they evaluated the output of their system against a series of TREC8 questions manually generated paraphrases, and acknowledge that their method would be only useful to assist humans in the manual creation of paraphrase patterns.

IV.3.2.1.2. Barzilay and Lee (2003)

The work of Barzilay and Lee adds finer granularity and more sophisticated computation to the method of Lin and Pantel. It also increases the training dimension through building a language model over two corpora. Here, the corpora are decomposed into separate word lattices. These are in turn clustered using a similarity metric based on word *n*-gram overlap. To build the patterns, a technique known as Multiple Sequence Alignment is applied over the sentences of the clusters, resulting in new lattices of lexical possibilities over a common VP path. The lattices from different corpora comprising the same *role* slots are understood as paraphrases. The authors tested their method on a set of later news stories on the same topic (Suicide bombings), and found that the system's acquired patterns was able to paraphrase more than 60% of sentences in short articles (less than 10 sentences), but the rate drops quickly with longer articles, which tends to indicate that the approach is too domain-dependent and suitable only to learn patterns of pattern sentences.

IV.3.2.1.3. Quirk and al. (2004)

Quirk and al. cast the problem of paraphrase generation as one of monolingual statistical machine translation. That is, the paraphrase patterns could be acquired by translation probabilities selection over sets of aligned sentences rather than through paired multiple sequence alignments over clusters of sentences as in Barzilay & Lee (2003). The authors make use of the availability of news sources on the Web to gather news stories data from multiple sources, instead of just a couple – but the training material is equivalent to Barzilai & Lee. Here, also, the gathering is very large in terms of topics so as to prevent the resulting paraphrase patterns from being too domain-restricted, and the collected articles are organized in clusters. Levenshtein distance is applied to detect close sentences within the data clusters. These sentences are paired and word-aligned using Giza++ (Och and Ney, 2000). Then an IBM model1 (Brown and al. 1993) is applied to the alignments to compute translation probabilities. The authors themselves acknowledge that the simplicity of the chosen translation model has some drawbacks, especially as due to its monolingual tone, no inter-phrase reordering is taken into account, which limits paraphrase detection to linearity: synonymy, phrasal replacements and phrase words' reordering. This is tested for paraphrase generation: given an input sentence, a lattice is built with nodes as phrase boundaries and phrase candidates from the DB as weighted edges – then the optimal path in the lattice is selected as a paraphrase candidate. Unsurprisingly due to their closeness, this system was compared to Barzilai & Lee's MSA in the evaluation phase where paraphrase generated by both methods have been submitted to human judges. The performances of this system are very low, especially when paraphrases express more than just lexical change – lower than MSA – but its coverage is greater due to the greater generality of its training corpus.

IV.3.2.1.4. Qiu, Kan & Chua (2006)

In this study, a reverse approach to the previously mentioned ones is followed: to detect dissimilarities between sentences and to judge whether these dissimilarities are expressive of paraphrase relations. Besides, the authors use syntactic analysis to model the data. Again, they do not define paraphrase qualitatively but rather quantitatively, refering to their approach as *sentence-level* paraphrase recognition. In their presentation, synonymy is understood as a kind of paraphrase. Another consequence of this quantitative view of the question is that the same sentences are also understood as paraphrase in their framework. The processing of data is two-phased. In the first phase, sentences are syntactically parsed and represented as *nuggets* of lexical information, mainly consisting of predicate-arguments tuples – i.e., the head verb and its arguments. Then the tuples are put into equivalence based on the similarity of the arguments. In an example, the predicate-arguments relations a *young man hurt Richard Miller* and *a young man injured Richard Miller* are put into equivalence in this way.

Candidate paraphrase sentences are therefore represented as sets of paired predicate-arguments tuples. This amounts to the alignments of Barzilai & Lee (2003) or Quirk & al. (2004), except that the method adds the qualitative filter of syntax, which can take into account re-orderings of the sentence words. The real novelty of the approach is that a second phase is applied to judge whether extraneous material, showing dissimilarity, is a barrier for paraphrase. This is treated as a ML problem, and the authors apply SVM fed with the syntactic paths from a given constituent adjunct to the head (predicate-argument in the paired tuples) as the main feature. The classifier is trained on a corpus of paraphrase pairs to avoid the cost of human annotation. The framework is evaluated on the MSR paraphrase corpus and results vary from a 65 to 42 % accuracy in recognizing paraphrases, depending on the presence or absence of unpaired tuples (extraneous material) in the pairs. The approach of Qiu & al. Should be very interesting to capture the specifics of conversive verbs equivalence (for ex.: *to buy* vs *to sell*) to build a model of paraphrase, but it barely goes beyond that, so that paraphrase here is restricted to synonym and conversive verbs, as observed on a corpus of limited size. In conclusion, the framework of Qiu does not really address paraphrase recognition in the linguistic sense, while it brings an interesting contribution to NLP, and to our knowledge unique, only misidentified.

Paraphrase Recognition is also used in the task of Entailment – usually involving simpler approaches. Bosma & Callison-Burch (2006) use the length of the Longest Common Subsequence (LCS) as their decision criterion in text entailment – to refine the alignment of common sequences, they employ paraphrase patterns rather than strict word matching. The paraphrase patterns are acquired from bilingual parallel corpora (Bannard & Callison-Burch, 2005), as probabilities for two candidate paraphrase in a source language to translate into a single sentence form in a target language – the procedure yields a 61.7% accuracy in the validity of paraphrases produced. Finally, Snow (2006) uses a set of heuristics to judge sentence similarity in order to detect false entailments. Although the approach does not explicitly refer to Paraphrase Recognition, it is very close to it – with the reservation that its aim is to assess or refute the a-priori hypothesis of paraphrase in candidate pairs. Furthermore, Snow's approach is probably the closest to our own, being strictly rule-based, and using only heuristics applied over the syntactic and lexical forms of the textual material. However, it would necessitate a preliminary phase of paraphrase candidate detection to come as a proper Paraphrase Recognition framework. The approach yields results in the line of concurrent systems in the RTE task (62.5% accuracy), which advocates the need to take syntax and language analysis into account in Entailment or Paraphrase Recognition.

IV.3.2.2. Linguistics

Paraphrase has rarely been studied for itself, at least until a recent time, and historical linguistics – say from Saussure to Chomsky – have not produced any decisive theory on paraphrase. But in their studies of language in general, some linguists have taken interest in paraphrase. In this section, we review theories on paraphrase in formal linguistics. These theories are hardly NLP-ready, so the purpose of this review is to support a sound methodology for automatic recognition of paraphrases. Besides, these studies have all more or less complemented and enriched each other through time.

The review is established after J. Milićević.

IV.3.2.2.1. Harris and the Pennsylvania School

Through his studies on language and discourse, Harris (1968) and the Pennsylvania School has introduced the notion of *transformation. Transformation* was at first conceived as a mechanism enabling the reduction of a sentence to a kernel, containing the objective information.

In the sixties, Harris enlarged his theory by including orientation: transformation now occurs from one single source sentence and can follow two different ways, *incremental transformation* and *paraphrasal transformation*. For instance, a sentence such as:

I say this.

Can be re-written as, or transformed into, *This I say*. The principle of orientation means that the opposite is not true, and *This I say* cannot be transformed into *I say this*, following the theory. The principle of orientation starting from a source makes possible to define operators in order to control the transformations.

Incremental Transformation adds information to the source sentence, and Harris defined types of incremental transformations – for, instance example 12 shows the transformations for a source sentence using the verbal and nominal operators ; note that the added information is implicit and not factual:

(12) a. *He **studies*** → *He **is studying*** ; *He **has studied***

→ *He **is** a **student**.*

Paraphrasal transformation does not add information to the basic content of the source sentence. Actually, and perhaps by excess of formalism, there are only three transformation operators, which would not be sufficient to cover all kinds of paraphrases.

- Permutation:

(13) a. *He will speak **only here**.*

b. ***Only here** will he speak.*

- Zeroing:

(14) a. *I oppose **John's** drinking.*

b. *I oppose drinking.*

- Morphophonemic change:

(15) a. *His behavior **awoke** suspicion.*

b. *His behavior **awaked** suspicion.*

Note that if *paraphrasal transformation* does not authorize the adjunction of information, it authorizes suppression of information, as in example 14. Also, Harris does not include synonymy in his description: paraphrase works at constant lexicon. As a dedicated syntactician, Harris does not conceive lexical variation other than induced by morphological change – example 15 shows the theory's greatest effort to include synonymy in the study of paraphrase.

IV.3.2.2.2. Generative Semantics

Lakoff (1963) and Mc Cawley (1976a, b) are responsible for the introduction of semantics within the study of paraphrase – not as much in order to explore new paths of research than because of a refusal to distinguish formally syntax from semantics. The theory points out that paraphrase, or rather, following the terminology of this theory, semantic equivalence, cannot be reduced to logical equivalence and explores concepts such as *nominalization*, which can cause a change in the lexicon, as well as used notions such as *predicate raising* to illustrate the falsity of the idea of the reduction of sentences of similar contents to logical equivalence. An illustrative example of this is the definition of the word *kill* according to GS:

CAUSE TO BECOME NOT ALIVE → CAUSE TO DIE → KILL.

The work amounted to adding rules to existing transformational schemes. But their work on paraphrase has not been followed. Perhaps, it could be said that GS has served a purpose of contesting Chomsky's standard theory rather than one of self-sustainability.

IV.3.2.2.3. Functional Sentence Perspective

This theory (see for instance, Daneš 1974, Sgall *et al.* 1986) stresses the importance of functions (theme and rheme) and of the communicative dimension in the sentence. It introduced the idea of *communication dynamism.* This amounts to the extent a given element in the sentence contributes to the development of communication. Formally, three levels of description of a sentence are defined: *Semantic Sentence Pattern*, *Grammatical Sentence Pattern* and *Communicative Sentence Pattern.* Therefore, three dimensions of a sentence are expressed at a deep level to be realized on surface: semantics, syntax and communication. This frames the paraphrase possibilities for a given sentence. For instance, sentences with synonymous words are given similar deep representations, as well as active and passive sentence. The precision of the approach limits the expression of paraphrases to *exact-synonymy* (between sentences), and therefore has more to do with the description of language (and of the *act of speech*) than with monitoring of paraphrase production. In this respect, illustrations of the works of the FSP use parallel multi-lingual texts rather than sets of paraphrases (Firbas 1992).

Milićević reports on Adamec giving an account of the communication role in the study of paraphrase, and advocates the necessity to include the communicative angle in the description of the deep structure. Following this, the following examples are not expressible through a single deep stucture (adapted from Adamec):

(16) *The officer proceeds to the interrogation of the prisoner*

(17) *It's the prisoner that the officer interrogates*

(18) *The officer submits the prisoner to interrogation*

As paraphrase is seen in the FSP as a mechanism to express exact synonymy – an example of such a mechanism relies on the verb's subcategorization frame, as for the verb *suggest*:

(19) a. *Jan* ***suggested*** *to his son* ***to arrange*** **[Vinf]** *the exhibition.*

b. *Jan* ***suggested*** *to his son* ***that he arranges*** **[Prop]** *the exhibition.*

c. *Jan* ***suggested*** *to his son* ***the arrangement*** **[N]** *of the exhibition.*

IV.3.2.2.4. Halliday: Functional-Systemic Theory

This theory (See Halliday 1976 and 1985) defines a general model language. Its main characteristics are 1/ a semantic orientation of the model: the language is thought as a system of choices to express meaning, 2/ its interest for *Language in use* and therefore to the situation context, and 3/ the use of stratification models as well as elements of dependency grammars.

It is not directly a study on paraphrase. But the use of functions in the description of sentences as semantic structure to be realized lexico-grammatically enables a representation of paraphrases. Functional-Systemic means that language is thought of in the theory as 1/ multi-Systemic: a system of mutually exclusive options to be selected in order to express sense, and 2/ Multi-functional: language has three major meta-functions:

a) Ideational: takes in charge the communication goal of an utterance.

b) Interpersonal: in charge of the speaker-interlocutor relation in the utterance.

c) Textual: in charge of setting the utterance into a context.

The model has two major dimensions: semantic and lexico-grammatical, of which the latter only is precisely described. For instance, the lexicon is thought of as the "most delicate grammar" and categories are described in order to frame the lexicon into syntax. Propositions are organized in types where semantico-syntactic roles around the predicate are specified – as in the example below:

(20) a. *The lion* [**Actor**] *caught* [**material process**] *the tourist* [**Goal**].

b. *Mary* [**Senser**] *liked* [**mental process**] *the gift* [**Phenomenon**].

This kind of representation should permit the production or recognition of paraphrases. But that is for the theory. We cannot describe in full details the FST theory, due to its richness and complexity. But this should be its major drawback: as formal as it is, it cannot lend itself to automation. However, an ideal model of paraphrase in NLP could resemble that made possible by the FST.

IV.3.2.2.5. Fuchs

C. Fuchs (1980) has exclusively concentrated her research on paraphrase. She stresses the fact that linguistic equivalence is neither a necessary nor sufficient condition to establish a relation of paraphrase, as well as the role of the context of utterance and that of the speaker. She insists on making a distinction between what belongs to language and to discourse in paraphrase.

For instance:

(21) a. *Paul est à Paris*

b. *il est ici*

(Paul is in Paris) → (he is here)

in this example, according to the author, the sentences are linguistically equivalent, but they are not *linguistic* paraphrases. She uses the notion of *plans* to explain this level of variations which can cancel the paraphrase link between two sentences still carrying the same content. Four plans can be identified: what we would call a *master* plan, the *locution* plan, and three contextual plans: *referential* (which example 21 illustrates, from *locution* to *referential*), *symbolic*, and *pragmatic*. Paraphrases can only happen between equal plans.

The linguistic equivalence on which resides a relation of paraphrase is to be found at the phrase level – her theory being more an act of analysis than one of description. A paraphrase is always a *predicative equivalence* (equivalence between expressions from the point of view of the operations needed for predicative relation: diathesis, thematization, focalization, etc.) and an *utterance equivalence* (equivalence between expressions from the point of view of the operations giving referential values: person, aspect, tense, determination, etc.). In this, her theory is close to that of the FSP. Below are examples to illustrate this:

Predicate-equivalence and Utterance-identity

(22) a. *Le chat a mangé la souris.*

(The cat ate the mouse)

b. *La souris a été mangée par le chat.*

(The mouse has been eaten by the cat)

Predicate-identity and Utterance-equivalence

(23) a. *Il aurait pu venir.*

(he could have come)

b. *Il avait la possibilité de venir, mais il ne l'a pas fait.*

(he had the possibility to come but he didn't do it)

Predicate-equivalence and Utterance-equivalence

(24) a. *C'est une chance que Paul ait refusé de recevoir ce cadeau de Marie.*

(It's lucky that Paul has refused to receive this gift from Mary)

b. *Heureusement que Paul n'a pas laissé Marie lui offrir ce cadeau.*

(it's fortunate that Paul did not let Mary give him this gift)

C. Fuchs has worked a lot on paraphrase and this is not the place to present the details of her studies. See C. Fuchs 1980, 1994. Note that C. Fuchs's work does not really lend itself to formalization and automation, but her purpose is descriptive, and still constitutes a valuable contribution.

Another idea of C. Fuchs is that of the orientation of the paraphrase link, taking on Harris, which she refines. When paraphrasing implies a shift from one semantic plan to another (phrasal, symbolic, referential or pragmatic), the paraphrase link is oriented and therefore non-reversible. In the following example:

(25) a. *"he's an Harpagon" means "he's stingy".*

SYMBOLIC PHRASAL

b. *"he's stingy" means "he's an Harpagon".*

PHRASAL SYMBOLIC

According to the author, a paraphrase link such as in 25b is *over-determining*, while it is *sub-determining* from 21b to 21a. To be reversible, the link must happen on equal plans:

(26) a. *"he's an Uncle Scrooge" means "he's an Harpagon"*

IV.3.2.2.6. Moscow Semantic School

This work on paraphrase has been pushed to the end of this brief presentation, as the model we have decided to use has been produced by people coming from the Russian Laboratory for Automatic Translation, namely the Meaning-Text Theory (MTT). The research took interest in the study of semantic representation and paraphrase in order to develop models for automatic translation. This work gave birth to the Moscow Semantic School from which comes Igor Mel'čuk. A. Žolkovskij and I. Mel'čuk are responsible for the formulation of the MTT (Mel'čuk 1981 for a sum-up of his work in English).

The Meaning-Text theory is a theoretical frame for the description of a functional model for natural language. Its specifics are as follows:

1. Globality: interest in the development of universal concepts.
2. Semantic bases, interest in linguistic synthesis, essential role of paraphrase and communication structure.
3. Focus on the lexicon
4. Relational approach to the description of language: more than categories, relations are considered as the most important factor in the organization of language – this favours dependency grammar over phrase grammar, due to portability and non-linearity.
5. Modular organization of the models.
6. Formalism.
7. Implementability.

Let us point out that in its present state, the paraphrase models of the meaning-text theory are far from being implementable, and no successful implementation of the theory has ever been produced, though some attempts have been made (of which the oldest was intended for paraphrase generation, see Boyer and Lapalme, 1985 – for a more recent implementation, see Apesjan and al., 2003). However, they are intended to be implementable and are therefore highly formal. The toy implementations based on the MTT are no less convincing than what has been done in paraphrase R/G in the field of NLP, and yet they have received much less publicity.

None of the NLP studies presented in section IV.3.2.1. mention any as related work. What has retained our attention is that the theory already offers models and rules for the production or recognition of syntactic paraphrase. Besides, it has brought weight to our choice of using a syntactic approach to solve our problem. This will be explained and justified in detail in the next chapter, and we will present the Meaning-Text theory in more details in section V.4.

In conclusion to this review, computational approaches to PR can present promising results in some aspects of paraphrase, but they show two major drawbacks. First they are too domain-dependent: the patterns of paraphrases are expressed lexically and are therefore constrained by the corpora on which they are acquired. Only Snow (2006) does not concentrate on a pattern acquisition problem, but his approach is incomplete as PR system. Second, the coverage of paraphrase as a linguistic phenomenon in these approaches is too scarce, and limited to lexicality – all the frameworks presented above have in common that they barely go beyond phenomena of extended synonymy. But this might be their strength, and their interest in PR. They are suitable to capture expressions of semantic paraphrase difficult to model theoretically, as shown in examples 11a or 11b, or show potential to detect and record conversive verb forms, as Qiu & al. (2006). Otherwise, they show a way to PR that casts paraphrase as a translation problem rather than a linguistic one. This is not the place to put a final judgement on this question, but in the state of the art in PR, this argues in favour of our choice to use a syntactic model of paraphrase, which offers the best compromise between comprehensiveness, simplicity and stability. This is in accordance with our conclusions on semantic similarity applied to short texts in general.

IV.4. A Syntactic Approach to Assessment

To assess the similarity of an answer to a reference sentence amounts to sentence similarity. To frame the problem of sentence similarity for a resolution, it should be cast as a paraphrase problem as the question offers analyses and solutions from both formal linguistics and NLP. Our approach is new in that it encompasses a conservative approach. This is justified by the fact that

in their mode of comparison, approaches to sentence similarity are either too permissive and only examine the overlap of lexical material between sentences, or too specific, as we will see in methods of sentence similarity as applied to paraphrase, which is the subject of the next chapter. The approach we defend is conservative in three ways: 1/ in the simplicity of the methods employed to assess similarity (comparison of patterns of syntax similarity and synonymy), 2/ in its restrictive view of similarity (due to impossibility to rely on a fair evaluation of semantic relatedness), and 3/ in the categories of language description it uses to achieve assessment (syntax and strict similarity).

There are four technical arguments in favour of a syntactic approach to solving our problem: the first one comes from the field of ITS itself (hence, CALL and CAA as well), the second has to do with the natural bias of sentence similarity methods on false-positive assessments, the third with the over-dependence of paraphrase R/G on lexical models difficult to acquire and insufficient in coverage of reformulation possibilities, and the last is related to the requirement of assessing not just content but form also. These come on top of the main practical argument in favour of a syntactic approach to similarity: the absence of a corpus of pre-recorded answers.

A review of these arguments is given by turn in this chapter.

IV.4.1. Tutoring Systems

As seen already, the present state-of-the-art in tutoring systems offers no definitive solution to the assessment of free-text answers. And no syntactic approach has been tried up-to-date. There are two major arguments for a syntactic approach: a direct argument and an indirect argument that happen to coincide.

IV.4.1.1. ITS and Technology

The fact that most tools are dedicated to the marking of essays does not reflect a market preference, but rather the natural course of technology. Marking essays calls for feature evaluation simpler than marking short answer, as K. Kukich stresses it (Kukich, 2000):

To the uninitiated, it might seem counter-intuitive that scoring short answers poses a greater challenge than scoring essays. But it should be clear [...] that automated essay-scoring techniques can rely on statistical averages of general features such as overall vocabulary content and syntactic variety to derive evidence of writing quality. In contrast, short answers seem to provide little textual evidence of the writer's underlying meaning, hence the need for finer-grained measures and deeper analysis.

Kukich surveyed various essay grading systems, which all have in common marking essays based on superficial features evaluation – IEA, e-rater and PEG to name them (see section II.2, subsection 2, 3 and 6 respectively). The features of decision have been, in all cases, selected through means of statistical analysis, from sets of pre-marked essays. These features are in turn evaluated statistically in the automatic marking process – hence her comment on the way essays can be graded automatically. Her remark nevertheless suggests that scoring shorter pieces of texts should rely more on symbolic evaluation.

Having studied the feasibility of scoring students' short-answer responses to end-of-chapter textbook questions, Kukich identifies several aspects of language evaluation specific to the marking of short-texts answers, concluding for the need to add NLP power to already existing free-text marking solutions:

1. pronoun and co-reference resolution, as she has found that short text answers did contain a higher level of anaphoric language than longer texts.

2. use of subject-specific thesauri on top of general lexicon databases.

3. use of special tokenizing and tagging techniques to derive parts of speech and partial parses from incomplete sentences and clauses.

4. some approximation of underlying predicate-argument or propositional structure to make judgment of how much the short-text answer represents the target-concepts that constitute a correct answer.

These are conclusions based on first-language textual productions, but they all imply that the material be analyzed syntactically. Point 3 is very dependent of the type of parsing technique

used, which in the case of Kukich's study presupposes that a piece of text must be formally complete to lend itself to syntactic analysis. This is not a problem in our case (see section VII.1.3). But in our case, lifting this difficulty leaves us with the supplementary task of having to address second-language learners' error detection.

IV.4.1.2. A bias on the assessment of correctness

In tutoring systems, our approach encompasses the strategy opposite to that exemplified by Bailey's CAM (Bailey & Meurers, 2008), see section II.2.2.2: a bias on incorrectness to make sure that no incorrect material is assessed as correct. However debatable this approach is, it follows observations of the science that incorrect feedback affects students' motivation (Graesser & al., 1995). In an instructional context such as that of ITS, a consequence of this is to privilege the deterministic assertion of correctness over incorrectness at the time of feedback, as incorrectness should be expressed in indirect terms and correctness only in partial terms.

The focus on assessing similarity through syntax and a strict semantic similarity (synonymy) makes processing incorrectness-prone. Other approaches only compare semantics in the assessment process. The tendency of this bag-of-word similarity approach to over-indulgence is the main reason for Kukich to support a syntax-controlled way of assessment. Here semantics is compared in an orderly way: that is, through syntax. The chances are that correct material will be assessed as incorrect, but there are almost no chances that incorrect material will be assessed as correct. One can argue against this, but the fact is that all other approaches are correctness-biased (greater chances that incorrect material is assessed as correct than the contrary). With this argument of ITS science for a bias on the precision of the assessment of correctness first, this project finds legitimacy in its approach. This bias is confirmed by studies in semantic similarity as we show in the next section.

IV.4.2. Semantic similarity and sentence similarity

IV.4.2.1. Similarity and Relatedness

A major problem in semantic similarity, as applied to text or sentence similarity, is the natural bias towards false positives. Actually, there is no or very little machine understanding of the textual material processed. First, the sentences, or texts, to be compared are processed shallowly based on token similarity or relatedness. Second, the decision on the similarity is calculated over a quantitative threshold applied to the distribution of the relatedness values. This makes the systems tolerant to some changes in the semantics between answer and reference, which increases recall of good answer, but at the cost of precision.

For instance, current sentence similarity systems judge:

Ex (27a) *Microsoft will purchase Yahoo, in spite of its resistance*

Ex (27b) *Microsoft will not purchase Yahoo, because of its resistance*

as similar sentences.

As we have seen, there is a gap in the measurement of similarity between strict similarity and relatedness. As useful and promising measures of semantic relatedness can be, they don't really lend themselves to scalability against the reference of similarity: we can know that two sentences are related, but it is still difficult to know how close this *relatedness* puts these sentences to *similarity*. 27a and 27b above should show close values of relatedness, in spite of their difference, while 27c below would show a lesser value of relatedness to 27a even though it is semantically closer than 27b.

Ex. (27c) *The Redmond firm has bought Yahoo.*

Sentence similarity can only be measured as the sum of sentences' constituent words' similarities. Therefore, a further problem is that of the order of these constituents: *Yahoo has bought the Redmond firm* is quantitatively equivalent to 27c in terms of similarity, but the two sentences mean different things. As simple as this seems, the problem can not so easily be addressed by heuristics (as some studies already do – for instance Snow and al., 2006, or McCarthy and al., 2007) – here for instance, a rule saying that order should be taken into account in the measure of similarity would not be able to handle an example as:

Ex. (27d) [...] *Yahoo which the Redmond firm has bought.*

These examples tend to show that a statistical approach to sentence similarity is insufficient to capture actual similarity for the reason that Semantic Similarity imprecision in form adds to Semantic Relatedness imprecision in contents.

As Semantic Similarity-based systems base their evaluation of similarity on the ratio of common contents between answer and reference, adding parts of the question in the answer affects the results as well. MacCarthy (MacCarthy and al., 2007), a member of U of Memphis' Graesser team shows how assessment of free-text answers by means of statistical methods in an ITS context (and therefore CALL) "appears to be problematic".

There are two major impediments to this association. The first is the *length problem*: statistical methods favors length (Wiemer-Hastings, 1999 – implicitly confirmed by Kukich 2000). According to MacCarthy, a longer but wrong answer showing greater overlap with the reference segment can receive more favorable judgement by a statistical system than a good but short one. We formulate the second as the *quantitative problem*: assessing quantity equivalence does not assess quality equivalence, as shows example 27 a, b, c and d above. In this respect, just as they judge 27 a and b equal, these systems are unsuitable to handle some aspects of syntactic reformulation (See section V.4.1.4).

It is true that the statistical systems produced by research tend to be backed by rule-based mechanisms to correct the false positive bias (See section IV.2.2), but these are very recent developments. This is for instance the case in the Textual Entailment system developed by Snow, which uses rules of syntactic equivalence as well as lexical heuristics to address phenomena of negation and antonymy to detect false entailments (Snow and al., 2006) – the tendency is yet stronger in paraphrase R/E. We believe that the addition of rule-based components to existing statistical systems in the domains of Entailment and paraphrase R/E only strengthen our case that a strictly rule-based approach should be tried in Sentence Similarity, and that our context of CAA is an especially suitable field, as it has been advocated in the preceding section.

IV.4.2.2. Paraphrase

Judging by the criterion of Semantic Similarity, the problem here is one of sentence similarity. Therefore it should be cast as a paraphrase problem.

Rather with than heuristics or ad-hoc rules, the best way to deal with such phenomena of language form is to use a language model. Whether this model should be probabilistic or deterministic is another question. Such models should be evaluated with respect to two criteria. The first is the reformulation possibilities it depicts for a given utterance, and the second is its coverage of the language – in our case, French. It could be a perfect model, putting in ordered correspondence fully lexicalized utterances in a given language with respect to measured similarity. This is the approach followed by statistical NLP in Paraphrase Recognition/Generation, but these models are still costly while largely insufficient to cover the whole language, neither in the possibilities of reformulation for a given utterances, nor in the coverage of language itself, i.e., world knowledge. This has been presented in section IV.3.2 on NLP approaches to paraphrase. Another language model is syntax itself. A flaw of the syntactic model is that it fails to capture the semantic aspect of paraphrase, but its great quality is that the reformulation possibilities depictable by its formalism cover the whole of the language. For this reason, we believe that syntax still offers a better account of paraphrase possibilities than statistical models. Another advantage of syntax is that information of basic lexical-semantic similarity can easily be added so that synonymy and antonymy are accounted for in the model. The one main difficulty in using syntax to model paraphrase is however to find a subset of syntax that actually expresses the reformulation possibilities with respect to similarity. A syntactic model for paraphrase following the Meaning-Text Theory is presented in section V.4.

In conclusion to this section, our approach represents the most severe way to assess similarity – as at every step of comparison, we make sure that the contents do match in sense. This is indeed a tough restriction on the flexibility of some statistical methods found in Semantic Similarity and Relatedness, or on the lexical richness of PR methods. That prevents our system from admitting answers slightly reformulated in their semantics (See section VI.5.1). But in this way, we make sure that assessed material is correct.

IV.4.3. Evaluation of Language Form

The last point concerning the portability of semantic similarity techniques in the context of CALL is that QA, Entailment or PR systems are all tested on common test sets (see for instance Dagan and al., 2005, For an overview of the Pascal textual entailment challenge), which all presents material of undisputable syntactic quality – clumsy utterances (see section VI.5.2.2) are absent from these sets, not to mention bad faith answers. As clumsy utterances do not usually exhibit syntactic errors, they pass the barrier of grammatical error detection unnoticed and increase the detection of false positive answers.

In her study, S. Bailey (2008) does not mention grammatical errors as an impediment to assessment of correctness, although she acknowledges that about 66% of the answers of her data set do show errors of various kinds, and although she uses syntax in his method. Perhaps the form of the English language is less sensitive to syntactical errors in the readability of content's meaning. In this study, answers have been gathered from real students, and the importance of grammatical errors made it compulsory to take error detection into account in the assessment process.

Using a syntactic approach to manipulate the language material in the assessment of answer rightness also provides material for the assessment of grammatical correctness. Detection and Correction of grammatical errors is a proto-field of research. It slowly arises as a trend of its own within NLP[9] and is presented here in section V.3. If statistical methods are used to address some issues of lexico-grammatical errors, to deal with higher-level errors (such as affecting the syntax of a whole sentence) requires analysis at the level of a sentence syntax, if only to express a statement on the error in a comprehensible, syntactic, way at the time of feedback creation. Provided the availability of a good syntactic parser – which are, if not freely, at least available – the detection of syntactic errors is easier than in any other way using the form and structure of syntax itself, especially as there exist no comprehensive models of errors to date.

Ultimately, statistical NLP could be used for this kind of exercise as purveyor of knowledge bases, or models. We saw in section IV.3.2.1 that some statistical systems in the domain of paraphrase recognition were giving interesting results, for instance to record lexical conversive associations that no knowledge bases up-to-date take into account, as well as to record phrasal-

9 The first shared task of the workshop on "using NLP to build Educational Application" will be on *grammatical error detection of articles and prepositions* and will take place in 2010 as part of the NAACL conference.

synonymic relations as in example 11 above. Therefore, statistics should be of great help in the building of lexical-semantics relations models. This would be beneficial to language technologies in general, and is actually on the research agenda as we have seen. But this is still work-in-progress even in statistical NLP and all present models suffer from too great a domain specificity (see Quirk and al., 2004, or Qiu and al., 2006). A remark on the research in statistical NLP is in this respect that it focuses on high-level problems, as paraphrase recognition where we saw that the systems lack a perspective on sentence-level syntax. Perhaps, general linguistics should be of help and direct statistical NLP towards lower-level tasks as conversives or phrasal synonymy, or other categories simply ignored by statistical NLP in the present state of things.

Part 2.

Resources, Processing and Evaluation

Chapter V. Resources

This chapter presents the various NLP solutions used in this project. It starts with parsing and with an evaluation of several parsers considered for the task. Then, the lexical resources used are presented. These resources and processes are employed for spell-checking, as well as to map lexical items following relations of synonymy and antonymy, including across Part-of-Speech. In the third section, the processing of language errors is described, and finally, we present our solution for paraphrase recognition. These resources, in order, correspond to the processing chain as implemented, and answer comparison, between Ref and S, is actually a function of the paraphrase module. Three processing examples are provided in Appendix C to illustrate the behaviour of the system.

V.1. Syntactic Analysis

V.1.1. Parsers

For the purpose of syntactic analysis, three parsers for French have been tested on our set of questions, corresponding text segments and student answers. In this paper, a large part is devoted to presenting the parsers as the quality of parsing is fundamental to our method of assessment. As well, in some systems, students are invited to examine directly parsers' production in order to enhance their syntactic skills and knowledge (Vinther 2004, Knutsson & al., 2003). This seems at the least to take part in a trend to include the teaching of formal syntax as part of language acquisition (Amaral & al., 2006a, 2006b).

First, in order to meet our needs, a good parser should be precise enough to capture proper grammatical functions, i.e. fine-grained enough to make a distinction between functions establishing relations between the words present in the sentences (such as noun or verb complements, appositions), as well as between the clauses present in the sentences (such as sentence complements). It should be precise enough to capture what is moveable and what is not in the words/clauses order in the sentence: a direct/indirect object or a noun complement is not moveable, while a sentence or verb complement typically is; a noun complement, or a sentence complement, may be removed without changing the overall meaning of a sentence – objects and some verb complements cannot be removed without changing the meaning of the sentence. Most parsers actually do not make a difference between verb complements and sentence complements. Second, the parser should be "robust" enough to be able to process a sentence's analysis even in the presence of linguistic errors, at least to a point sufficiently advanced to lay ground for error analysis.

The parsers are Syntex, a new parser called Syntex, built as a joint effort between University of Toulouse, University of Stuttgart and a company named Synomia, the C101 parser (actually a part of a software suite designed for redaction of English by second-language English speakers), and XIP (Xerox Incremental Parser). We will now examine these tools by turn.

V.1.1.1. Syntex

Syntex (Bourigault 2006) is a symbolic parser, dedicated to French, whose chain of analysis works in four phases:

1. Pre-tagging: performs sentence segmentation, word tokenization as well as word pre-tagging.
2. Morpho-syntactic tagging: performs the actual PoS tagging on words.
3. Conversion: between outputs produced by second layer and inputs needed for actual syntactical analysis.
4. Parsing: to define the relations between the PoS.

And that's about all the information we have on the internal machinery deployed by Syntex. At this point, note that a further requisite for a good parser concerns available documentation on the way the system actually works. But as these tools are often proprietary (the case for our three parsers), proper technical documentation is scarce, or divulged with parsimony.

That said, Syntex still provides analysis of decent quality, making few errors; on the other hand Syntex does not go very deep into the analysis, as shown in the example provided below. Another drawback of Syntex is the lack of readability of the outputs – only a partial problem in the face of processing, but a real one in the face of exemplifying. Below is an example of an analysis by Syntex.

<TXT>*plus les réserves en fer de la femme sont faibles dans la première moitié*

de la grossesse , plus le temps de gestation est écourté.

(The lower a woman's level of iron in the first half of pregnancy, the shorter the gestation time.)

```
<ETIQ>Adv|plus|plus|1||     DetFP|le|les|2|DET;3|
NomFP|réserve|réserves|3|SUJ;9|DET;2,PREP;4 Prep|en|en|4|PREP;3|NOMPREP;5
NomMS|fer|fer|5|NOMPREP;4|PREP;6 Prep|de|de|6|PREP;5|NOMPREP;8
DetFS|le|la|7|DET;8|  NomFS|femme|femme|8|NOMPREP;6|DET;7
VCONJP|être|sont|9||SUJ;3,ATTS;10,PREP;11   NomMP|faible|faibles|10|ATTS;9|
Prep|dans|dans|11|PREP;9|NOMPREP;14    DetFS|le|la|12|DET;14|
Adj?S|premier|première|13|ADJ;14|
NomFS|moitié|moitié|14|NOMPREP;11|DET;12,PREP;15,ADJ;13
Prep|de|de|15|PREP;14|NOMPREP;17 DetFS|le|la|16|DET;17|
NomFS|grossesse|grossesse|17|NOMPREP;15|DET;16    Typo|,|,|18||
     Adv|plus|plus|19||
DetMS|le|le|20|DET;21|      NomMS|temps|temps|21|SUJ;24|DET;20,PREP;22
Prep|de|de|22|PREP;21|NOMPREP;23 NomFS|gestation|gestation|23|NOMPREP;22|
VCONJSp|être|est|24||AUX;25,SUJ;21     PpaMS|écourter|écourté|25|AUX;24|
Typo|.|.|26||
```

Table 1. An example of Syntex Output.

The output comes as a chain of lexical items depicted accordingly:

```
PoS|lemma|form|position_number|headed_by|heads
```

Here, the analysis fails to provide a single line of entry in order to understand what is going on. A good syntactic analysis should provide such a header to properly enroot the sentence into syntactic coherence. Formally speaking, the above tree has two roots:

```
VCONJP|être|sont|9||SUJ;3,ATTS;10,PREP;11
```

and:

```
VCONJSp|être|est|24||AUX;25,SUJ;21
```

These say that the first clause receives a subject, a subject's attribute, and calls for a preposition carrying what is a verb complement (circonstancielle de temps), and that the second commands a subject and is auxiliary to a past participle predicate. This is true but incomplete. These two sub-roots (which are branches) should be themselves attached by a single root – here, the comparative adverb *plus*, acting syntactically as a coordinator. The analysis catches the assertions but misses the overall comparison. In addition, Syntex is not able to draw the distinction between an indirect object and a circumstantial complement (verb complement – can possibly be either discarded or displaced without affecting the meaning of a sentence, which is not the case of indirect objects).

To finish with Syntex, and to be fair, the parser is good at nouns-to-adjuncts relationships, such as in:

[…] *Ce qui* […] *rend difficile leur interprétation.*
([…] which […] renders their interpretation difficult.)

```
[…]
VCONJS|rendre|rend|40|CC;39|OBJ;43,ATTO;41
      AdjMS|difficile|difficile|41|ATTO;40|
DetMS|leur|leur|42|DET;43|
      NomFS|interprétation|interprétation|43|OBJ;40|DET;42
[…]
```

Here, *difficile* is an attribute of the object *interpretation*, and has actually no direct relationship to the verb, which is what the analysis says. The other parsers fail to capture this kind of inverted Adjective/noun relationship.

V.1.1.2. C101

C101 – standing for *Correcteur 101*[10] – is a Canadian, computer-assisted redaction tool, and includes a parser that works for French and/or English. We have been able to have our set of textual material analyzed by this parser, without having ever the opportunity of a closer look at it or its technical specs. Therefore, there is very little we know about C101 – other than that it is a partially lexicalized, error-correcting, statistical parser: outputs come in the form of a set of analyses, each ranked according to an index of confidence computed over a mix of word PoS confidence value and syntactic relationship confidence value. On top of it comes an annotation of corrected, or not, errors at word level: orthographic and agreement errors are corrected when all analyses (for one sentence) agree on the presence of an error and when the parser "thinks" it has reached a conclusion on the exact nature of the error and on how to correct it. This holds ground for further errors, numerous, and introduced by the parser itself. True errors can be left unseen, and correct material can be treated as errors, ruining the whole analysis.

This said, C101 produces splendid parses – above anything we tried. A superb quality of C101's parser is that sentences are analyzed with respect to the discourse type of a sentence: a straight assertive sentence is analyzed under its verbal head, but a comparative sentence, for instance, is analyzed under a head constituted of the sentence's comparative adverb(s); similarly, a sentence/clause containing a coordination is headed by the coordination conjunction: if the coordinates are sentential clauses, the conjunction heads the whole sentence – and so on.

Below is an example analysis based on the sentence shown above. For concision, only the most confident analysis for this one sentence is shown here, out of eight; note that the output is hardly more reader-friendly than Syntex's:

```
PhraseOriginale: plus les réserves en fer de la femme sont faibles
dans la première moitié de la grossesse, plus le temps de gestation
est écourté.
( arbre_dep_valide ( plus/plus/et/prop/22/18/19 ( R_PlusProp (
plus/plus/advComparSuper/advComparSuper/100/0/1 ) ) ( R_Coord (
sont/être/verbe/verbe/63/8/9 ( R_Sujet (
réserves/réserve/nom/nomDeter/86/2/3 (
R_Déter ( les/le/deter/deter/100/1/2 ) ) ( R_CmplDuNom (
en/en/prep/sPrep/93/3/4
```

[10] See References. Or an evaluation of C101 can be found on CALICO's site, at : https://www.calico.org/p-73Le%20Correcteur%20101%20Ver%203.5%20Pro%20(91998).html

```
( R_CmplDePrep ( fer/fer/nom/nom/95/4/5 ( R_CmplDuNom (
de/de/de/deSN/96/5/6 (
R_CmplDePrep ( femme/femme/nom/nomDeter/98/7/8 ( R_Déter (
la/le/deter/deter/100/6/7 ) ) ) ) ) ) ) ) ) ) ) ( R_Attrib (
faibles/faible/adjectif/adjectif/100/9/10 ) ) ( R_CCirc (
dans/dans/prep/sPrep/90/10/11 ( R_CmplDePrep (
moitié/moitié/nom/nomDeter/92/13/14 ( R_Déter (
la/le/deter/deter/100/11/12 ) )
( R_Ordinal ( première/premier/ordPremier/ordPremier/100/12/13 ) ) (
R_CmplDuNomQuant ( de/de/de/deSN/96/14/15 ( R_CmplDePrep (
grossesse/grossesse/nom/nomDeter/98/16/17 ( R_Déter (
la/le/deter/deter/100/15/16 ) ) ) ) ) ) ) ) ) ) ) ( R_VirguleSepar (
,/,/virgule/virgule/100/17/18 ) ) ( R_Coord (
écourté/écourter/verbe/verbe/74/24/25 ( R_Sujet (
temps/temps/nom/nomDeter/95/20/21 ( R_Déter (
le/le/deter/deter/100/19/20 ) ) (
R_CmplDuNom ( de/de/de/deSN/98/21/22 ( R_CmplDePrep (
gestation/gestation/nom/nom/100/22/23 ) ) ) ) ) ) ( R_Aux (
est/être/aux/aux/85/23/24 ) ) ) ) ( R_FinProp (
././finProp/finProp/100/25/26 )
) ) )

NbAnalysesCompletes: 8
PhraseCorrigee-0: <em>Plus</em><ERR/> les réserves en fer de la femme
sont faibles dans la première moitié de la grossesse, plus le temps de
gestation est
écourté.<br/>
NbAnalysesVisibles: 1
NbErreursMin: 1
<em>Plus</em><ERR/> les réserves en fer de la femme sont faibles dans
la première moitié de la grossesse, plus le temps de gestation est
écourté.<br/>
```

Table 2. C101 Parser Output Example

The two layers shown here should not be read in full correspondence. The first part is the most confident analysis, and the second is a statement of completeness and errors. `NbAnalysesVisibles` tells you that there is only one trustworthy analysis, out of 8 – meaning that level of confidence is controlled by a threshold rather than by top-of-the-list only. The parser exhibits an error thought to have been averted: `<em>Plus</em><ERR/>`, although it is left untouched. This is no error, but the parser seems to be fooled by the capital letter. The actual analysis, as a tree, correctly interprets *plus*, and has the whole sentence headed by it, providing one single line of entry to understand the analysis, and the sentence's structure. Note that this

analysis has only a 22% level of confidence, while interpretation of *plus* as a comparative adverb is trusted at 100%:

```
( arbre_dep_valide ( plus/plus/et/prop/22/18/19 ( R_PlusProp (
plus/plus/advComparSuper/advComparSuper/100/0/1 ) )
```

This heads the two verbs of the two sentential clauses, analyzed as coordinates:

```
( R_Coord ( sont/être/verbe/verbe/63/8/9
```

and

```
( R_Coord ( écourté/écourter/verbe/verbe/74/24/25
```

This in turn heads the other constituents of their respective sentential clauses.

This is no place to show in details the skill of C101 at correctly analyzing complex and deep structures. Let's just add that it did make an error here: *de la femme* is analyzed as a noun complement to *fer*, instead of *réserves* (an error also made by Syntex).

Although C101 seems to provide the best analysis, we chose not to use it for three main reasons. The first and most trivial is that the error-correcting module is not very useful and predicts a catastrophe when it comes to parsing our student set of answers. The second is that the statistical mode of functioning would render difficult the selection of a single analysis when the parser cannot decide between several – again, a major drawback when it comes to analyzing flawed linguistic material, which drives the tool to greater operative tolerance. The last reason is that we have no access to the piece of software itself. There are other reasons; they will become evident while presenting XIP. But it should be stated clearly that C101's analyses are actually superior to those of XIP.

V.1.1.3. XIP

XIP – Xerox Incremental Parser (Ait-Mokhtar & al., 2001) – is an incremental parser, providing "robust" analysis of a text's syntactic structure. The robustness is actually guaranteed by the tool's incremental processing, which can produce analysis whatever the linguistic quality of textual material. The rules are layered, and applied successively; therefore, the parser will work up to where it can when analyzing text, and provide a more-or-less complete analysis. Although the parser is not presented as such, in fact it performs a kind of upper-grade shallow parsing, building two-level syntactic trees: one level for main categories, phrases, and another one for the PoS involved in the syntactic phrases. The possible dependencies between these syntactic phrases are provided separately from the tree itself, as flat relations between two dependents.

This tool will be presented here in more detail, as it has been selected as the parsing module in our system. The first reason for this choice is that it does perform robust incremental parsing, which makes it suited for analyzing flawed material; a second reason is that the whole parser can be modified, in order to produce custom parsing – it can be modified under all processing aspects, or at selected levels only. This last quality has been decisive, and the parser has already been modified.

V.1.1.3.1. XIP's processing

XIP is composed of three core modules that are included with XIP by default and two optional pre-processing modules. The core modules are:

♦ Contextual disambiguation module: this module uses XIP rules to assign words features or categories according to their context.

♦ Chunking module: this module segments linguistic units as a series of chunks or other groupings.

♦ Dependency module: this module uses rules to identify dependencies between linguistic units.

The optional pre-processing modules are:

♦ Normalization, Tokenization, and Morphology (NTM): this module provides the normalized form and all potential lexical information for each word identified. (included in our version but not modifiable)

♦ Hidden Markov Model (HMM) disambiguator: this module uses the Hidden Markov Model algorithm to find the most probable grammatical category of a word according to its immediate context. (not included)

Figure 1 shows module interaction.

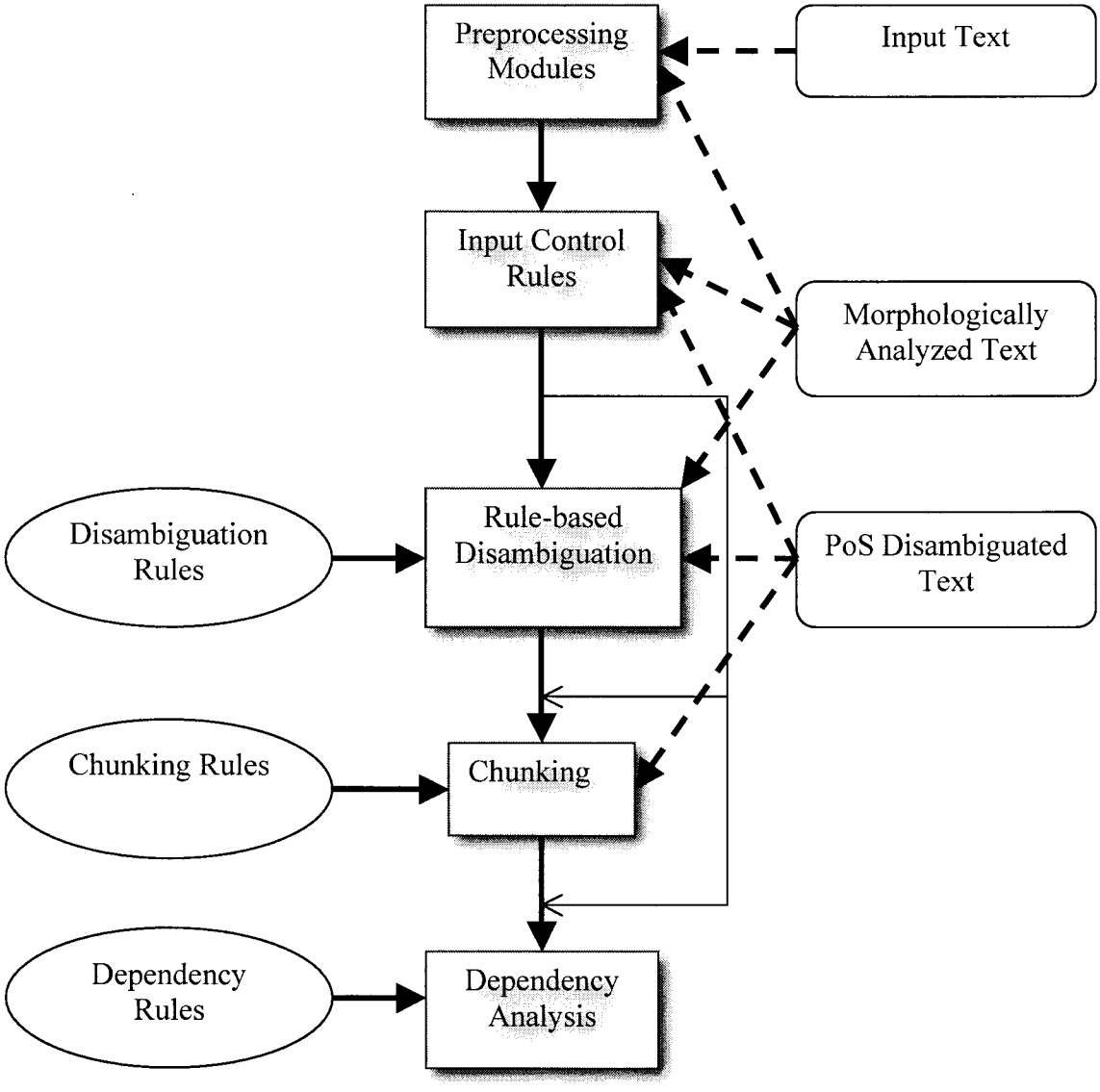

Figure 1: Architecture of XIP.

More than just a parser, XIP is a true specialist's tool for grammar analysis design and requires a good degree of expertise. In the above diagram, the ovals show modifiable modules. The non-arrowed lines show alternative points in the process where to introduce input textual material which can come as mere text, or as pre-processed files (by another program) containing either

PoS or constituents, or XML-structured material. A `translation file` can be filled in order to put in correspondence XIP's nomenclature with that of any tool providing pre-processed input files.

♦ Pre-processing tokenizes a piece of text into lexical nodes, each depicted by features recording the word's PoS, number and gender as well as information on lexical compatibility, as in the following example:

```
est   être
      +en+pourSN+aSVINF+il+deSN+queP+avoir+aSN+SADJdeSVINF+SADJ+c
ontreSN+locSN+deSVINF+SN+IndP+SG+P3+Verb+VAUX_P3SG
```

Note that the above mentions that this verb (*être*) is recorded as compatible with such prepositions as *à*, *de*, *pour*, *contre*, calling for either a NP (`pourSN+`) or an infinitive afterwards (`deSVINF`).

♦ Input control rules contain such files as the translation file, to make sure that the input corresponds to what is expected at the level input is introduced. These rules can therefore come into play at three different levels.

♦ Disambiguation rules come on top of the disambiguation module performing at the pre-processing step. This further disambiguation works on immediate context and is part of XIP custom modules, to be created by the user/administrator.

♦ The chunking module uses chunking rules to group sequences of categories into structures that can be processed by the dependency module. The rules are organized in layers and are applied to sequences of categories one after the other on the disambiguated input text. This one module contains basic rules, to be possibly augmented by custom rules in a layered manner. It produces outputs such as:

```
NP{XIP} FV{est} NP{un bon analyseur}
```

Or according to custom specifications:

```
NP{XIP} FV{est} NP{un AP{bon} analyseur}
```

♦ The dependency module produces the dependency relations between words or chunks. These dependency relationships can be specified by the grammar writer and may include standard

syntactic dependencies (subject, object, etc.), lexical modifications (noun modifiers, verb modifiers, etc.) or even broader relationships, such as inter-sentential co-references. Below is an example of dependency:

`SUBJ(est,XIP)` and `OBJ_SPRED(est,analyseur)`

As well as,

`NMOD_POSIT1(analyseur,bon)` and `DETERM(analyseur,un)`

Note that `OBJ_SPRED` means that the object actually stands for a subject predicate.

V.1.1.3.2. XIP Modification

In order to enhance the quality of parsing, we have extensively modified XIP. The task of comparing two segments of text which should carry similar meanings, of which one is possibly/probably flawed, requires that the Reference segment REF is properly parsed. This is no place to show the details of the modifications, only relative to the amount, and type, of errors that come with standard XIP parsing. We only present the types of the errors, and the options XIP offers for customization.

a. Errors

i. Compound or complex nouns

XIP is currently unable to recognize compounds or complex nouns. An example compound is *singe lion* or *voiture pilote* – the second word being either a noun or an adjective. An example complex noun is *tête de lecture* or *réaction en chaîne* (« read head » or « chain reaction »). The major syntactic characteristic of such lexemes is that only the first noun can participate in sentence dependencies, as head of the second – the second having no other value than to modify the first.

ii. Noun complements

XIP records noun complements as NMOD (noun modifiers). This has the consequence that adjectives or nouns as well as noun complements and appositions are all analyzed as noun

modifiers – such as in the following example for *les faibles réserves en fer de la femme* (woman's scant iron level):

```
NMOD_POSIT1(réserves,faibles)
NMOD_POSIT1(réserves,fer)
NMOD_POSIT1(fer,femme)
```

This is insufficient, because it puts on the same level elements which should not be. And it also drives the analysis to straight error – here, putting in a relation of dependency *fer* and *femme*.

iii. Past Participle

XIP systematically analyzes past participles as verbs, even when in adjectival relations. This produces major cascading errors. Similarily, when placed after an auxiliary, XIP is not able to detect a predicative function when the attribute is an adjective under past participle form.

iv. Appositive Clauses

An apposition is, typically, an adverbial phrase, or a clause headed by a present participle and introducing a noun modifying phrase, such as *des femmes ayant de faibles réserves en fer* (women having scant iron reserves). This also produces wrong attachments in cascade as the parser will take any forthcoming Prepositional Phrase as modifying the present participle.

v. Verb Modifiers

XIP takes anything placed after a verb that does not conform to the object pattern for a verb modifier (VMOD). These includes indirect objects, verb or phrase complements (old school adverbial complements), or straight noun modifiers. This is by far the most difficult issue to tackle, as these categories do not always lend themselves to description in terms of syntactic or PoS features, but are to be understood under lexico-semantic constraints. A typical example put in opposition *manger une tarte à la crème* (to eat a cream pie) and *manger une tarte à la table* (to eat a pie at the table).

However, between dealing with an indiscernible lot of VMODs and coming with perfect parsing, a lot can be achieved refining XIP rules.

b. Modifications

XIP is configured and customized by the following files:

File Type	Function
Feature Declaration file	to declare the features used to describe words and to create rules working on them
Category Declaration file	to declare the categories under which to record words and sequences of words
Dependency Declaration file	to declare the tags used to describe dependencies
Custom Lexicon file	to declare features and/or categories to be added or replaced for existing words
Controls file	Contains information about the automatic features of XIP, including features that are appended to nodes and substrings used to split input text into a sequence of categories
Translation file	to declare translation rules for translating tags coming from an external source to XIP tags
Grammar Configuration file	to declare all the files constitutive of the grammar – not all need to be present
Rules file(s)	Contains the layered rules

Table 3. XIP's files for grammar customization.

Therefore, XIP can be modified to a various extent and in various scope. Using variables and so-called TRE (Tree Regular Expression), one can create a whole variety of rules, designed to manipulate data at any step of the tasks necessary to syntactic parsing (tokenization and PoS tagging, disambiguation and dependency analysis). They can intervene before, during or after

standard parsing. Below the rule types are briefly presented, without going into the details of admissible operators and computational scope.

♦ Valence rules: to modify a word's feature set.

1. Default Feature Specification (DFS): composed of two lists of features. The list on the left provides a condition and the list on the right provides the set of features that should be instantiated.
2. Feature Co-occurrence Restriction (FCR): same syntax as DFS; but the two lists are conditions: if the left side applies, then the right side must too.

♦ Split rules: break the input stream into processing units. These rules are part of the input control rules.

♦ Disambiguation rules: disambiguate the most likely category(s) given a word's context. These rules are layered, and can thus be mixed with other types of rules:

```
layer> readings_filter = |left_context| selected_readings |right_context|.
```

The reading filter is expressed in terms of categories and features. Conditions can be optionally implemented using the `where(condition)` keyword, for instance to test the values of features coming from different words.

♦ Chunking rules: to group PoS under syntactic categories.

1. Immediate Dependency (ID): layered rules to assign PoS to a syntactic category, regardless of the order.
2. Linear Precedence (LP): associated with the preceding. Used to put order in the words/PoS coming in the syntactic categories. They can be layered or not, depending on whether the rule should apply to a given ID rule, or be a general constraint on the whole grammar.

♦ Sequence rules: a refinement over the above, as these rules can express context conditions. A good way to reshape chunks based on very specific patterns, but these rules do basically the same job as chunking rules. They are also layered, and can for instance isolate specific sets of words from general chunking. Plus, they can work on either a shortest match or longest match mode. Also handles the `where(condition)` keyword.

♦ Dependency rules: they take the sequences of constituent nodes identified by the chunking rules and identify relationships between them. These rules can also reshape already established dependencies based on conditions expressed at node level (i.e., on the features values of words).

♦ Marking rules: used to instantiate specific features on nodes (words already placed on the tree) or sub-trees – therefore different from Valence-rules.

♦ Reshuffling rules: used to reshape sections of the tree. They are layered, and can be mixed with chunking rules. In a way this is similar to sequence rules, but acts on syntactic categories already detected by chunking rules.

We have modified XIP in order to meet as much as possible our needs: having reference's material well parsed. The tools offered by the software to manipulate the analysis are good to work at local level: to disambiguate words, to change one-to-one dependencies between phrases or word's nature and to refine over nodes amount and order inside a phrase. Defining long-distance attachments between sentence's kernel and sentence's modifiers – that is, between two or more *groups of phrases* – has been left to post-processing independent of parsing.

We have succeeded in correcting some of XIP's analysis flaws. One type of error that cannot be corrected with total accuracy concerns verb modifiers, in that they can be mistaken for some noun modifiers. Below is an example.

Le tabagisme, par exemple, gêne le passage du sang à travers le placenta.

(Tobacco use, for instance, interferes with the flooding of blood through the placenta)

Note that *par exemple* isn't properly caught by XIP – this is an adverbial expression (even if there is no adverb in it – the function is the same), and the parser does not know how to treat such expressions when they are distanced from the head of the sentence (here, by commas). This is typically the kind of problem left to post-processing, as modifying XIP for that purpose would require too much machinery and expose to too big a risk of error (due to distance) for the simplicity of the problem. In addition, XIP ignores the notion of *sentence* modifier.

However, in the above example, *à travers le placenta* is analyzed by XIP as VMOD to *gêne*, while it is actually NMOD to *passage*. XIP would properly analyze a sentence as *passer le bout*

du nez à travers la porte, but would equally wrongly analyze *proposer une idée de promenade à travers les bois* – and so would C101. This can only be controlled by lexico-semantics means, and is therefore left unsolved in our project.

V.1.2. Post-processing

As the parser does not always produce analyses that are both correct and complete at the same time, a layer of post processing for syntactic analysis is necessary. A property of the parser is that it will stop parsing when confronted with attachments it cannot resolve, instead of trying to find the most likely attachment; in other words, the parser stops before committing an error in the analysis. Post-processing the output of XIP can help to clarify distant attachments – specially as XIP is merely fooled by commas; or rather: XIP does not know how to use comma positions in order to disambiguate between possible attachments. The parser cannot make a difference between the two complements.

(28) *Le chat a laissé filer la souris ayant mangé sa pâtée.*

(The cat let go away the mouse which has eaten its slops)

Le chat a laissé filer la souris, ayant mangé sa pâtée.

(The cat let go away the mouse, having eaten its slops)

Solving such a problem evidently looks straightforward. If the complement is not separated from the main proposition by a comma, the attachment belongs to the first noun to its left – while in the presence of a comma, the attachment belongs to the head of the main proposition. In its present state of modification, the parser only properly captures the first structure – the second is left unanalyzed (no attachments between *ayant* and any left element). Therefore, we believe it is easier to solve the issue through post-processing than through modifying XIP, which would involve a high risk of error due to the generality of XIP Tree Regular Expressions.

Another purpose of post-processing is to properly map the sentence into richer dependency depiction of the relations between the sentence words (or analysis nodes). This requires that co-reference relations be solved. It is only partially done by XIP: the parser only solves subordinate conjunction co-reference phenomena. Eventually, VMODs relations should be expressed in

terms of the preposition that introduces the PP rather than in terms of the noun that heads the PP-included NP, to differentiate between various prepositions which do not bear the same semantic charge (think of the meaning difference between preposition *de* and prepositions *à travers de* or *dans le cadre de*).

Below are two examples of XIP's incomplete analysis (only trees are shown here, to save space). As we succeeded in modifying the parser up to dealing with cross-punctuation attachments, such as shown below (and the unsolvable confusion between some VMODs and NMODs introduced by some prepositions), hereafter are shown the kinds of problem we encounter due to the presence of commas – therefore, the shown errors are the only ones.

- Example 29: Comparison Prepositional Phrase.

```
0>GROUPE{SC{Dans ces conditions , NP{la circulation} AP{sanguine} PP{du
NP{foetus}} FV{est détournée}} PP{vers NP{les organes}} AP{nobles} , PP{comme
NP{le coeur et le cerveau}} , PP{au détriment du NP{rein}} .}
```

(Under those circumstances, the fetus's blood circulation is turned away in the direction of noble organs, such as the heart or the brain, at the expense of the kidney.)

Prior to modification, parsing would produce an attachment between *Comme le coeur et le cerveau* and *est détournée*. Modified, it leaves the PP as a free clause, to be properly put into dependency during post-processing. Note that this is left to post-processing, as simple as it is, because this actually recovers other PP attachment problems, such as *au détriment du rein.* Treating this with a separate program seemed simpler.

- Example 30: Apposition:

```
0>GROUPE{SC{NP{Ils} FV{revendiquent}} NP{le droit au labeur} PP{dans NP{un
univers}} SC{BG{oú} NP{l' âge} FV{constitue}} NP{un handicap} , GV{affirmant}
SC{BG{qu'} NP{ils} FV{en ont}} assez IV{de devoir} IV{recourir} PP{aux
NP{artifices}} AP{cosmétiques et chirurgicaux} IV{pour travestir} NP{le leur}
.}
```

(they claim the right to work in an environment where age represents a handicap, assessing they are fed up with having to use cosmetic and surgical artifices in order to modify theirs)

This roughly corresponds to Example 12. Here, *affirmant* attaches to subject *ils*.

- Example 31: Sentence Apposition

```
0>GROUPE{SC{NP{la présence de sang} et PP{de NP{protéines}} PP{dans NP{les
fèces}} AP{humaines} FV{est}} NP{un indice} PP{de NP{AP{nombreuses}
```

```
maladies}} , par exemple NP{des saignements} AP{associés} PP{à NP{un ulcère}}
.}
```

(the presence of blood and proteins in human faeces is an evidence of numerous diseases, for instance bleeding associated with an ulcer)

Note that this is a different case from Example 29, as the clause is not headed by a preposition, and the structure is not recorded at parsing as a single P phrase. XIP does not know yet again how to deal with clauses headed by *par exemple*. This adverbial expression is recorded as an adverb and therefore acts as a VMOD on its own (to *est*); while *des saignements associés à un ulcère* (bleedings associated with an ulcer) is separately attached to the same verb. Note also that *des saignements* can be analyzed as a PP (as *de nombreuses maladies* (numerous diseases)) or as an NP (the case here), which is very much debatable. Actually the whole *par exemple* clause needs to be attached to the whole sentence – it is trivial but has to be done.

The two phases of parsing and postParsing processing amount to a mapping of REF's material. This needs to be done as perfectly as possible as the quality of the mapping puts a condition on the overall quality of the assessment. The postparsing module is currently being built. It uses rules showing a mix of syntactic and lexical clues. The syntactic categories work at the phrase level, therefore, the attachment to be recognized involve large chunks of text. The lexical clues are all grammatical words – for now prepositions, and it does not seem we need more.

V.2. Dynamic Lexical Resources

The term denotes third-party software-output lexical information rather than just text form encodings. Semantic Similarity and spellchecking both call for the inclusion of such resources, as the lexical items for which to obtain similarity or orthographic information are numerous (max = all of a language words). The fixed lists previously mentioned as lexical resources (verbs and prepositions) include a maximum of several hundred words. The reason for this is both to set a limit on manual additions to the system as well as to minimize the transformations necessary for portability from one language to another.

V.2.1. Orthography

The spell checking module contained in the system is custom-made, based on pre-existing components. It follows a design common to most spell-checker, particularly Aspell. In this design, a word is first converted to a code; the code is then compared to a dictionary, or lexical basis, of correctly spelled word. If no match is found for the word, the words with equal code value compose the correction suggestion. Equality is computed by means of the Levenshtein distance. Therefore, the implementation requires the following resources:

1. a correspondence basis between letters and symbols
2. an exhaustive lexical basis of correctly spelled words

V.2.1.1. The Soundex algorithm

The soundex algorithm, or method, dates back to 1918 (R. Russell, M. Odell, US pat 1261167). Soundex converts consonants to symbols, to create a phonetic transcription of a word, following the idea that errors are made on vowels rather than consonants – with the exception of the first letter, converted directly. The symbols can vary – in the original version, letters were used only for the first *soundex* of a word, that is the first letter ; following consonants were transcribed as numbers. Standard versions of soundex converters produce four-symbols codes only.

We use a version of soundex adapted to French pronounciation. We borrowed the code as a perl package from J-M. Pennel (2006) – although the implementation is straightforward: conversion of a word's consonants to symbols. In this version, the symbols are all capital letters ; this boosts the combination possibilities from 26 000 to 160 000 for four-letters codes.

V.2.1.2. The Levenshtein Distance

The Levenshtein distance has been named after its author, V. Levenshtein (1966). It measures the edit distance between two strings. *Editing* here corresponds to operations of deletion, substitution or insertion – and *distance*, to the number of such operations needed to convert one word into another. The editing distance between two words is proportional to letters difference,

in number or form, or to the rank of these letters. The maximum edit distance between two words corresponds to the length of the longer one of these words.

The implementation we use (D. Mistrut, 2002) is based on the Wagner-Fischer dynamic programming bottom-up algorithm, in which words' letters are compared orderly within a matrix, and the number of conversion operations adds up.

V.2.1.3. Lexical base

Our lexical base was so complete that we encountered a problem of over-fitting. Based on Lefff (2007) French lexicon, first intended to help PoS annotation and syntactic disambiguation for syntactic parsing. The words comprise information on grammatical values of number and gender, as well as sub-categorization frames. It includes lots of technical words directly borrowed from English, which can create problems in a project like this one where most students are Anglophones (for instance, *system* is included). However, we've extracted the words from the base to create a lexicon for the spell-checker. It currently includes 352 964 tokens.

V.2.2. Synonymy

As we have seen in section IV.2.1.1, the notion of Word Similarity coincides with that of synonymy. Synonymy can be extended to relations of hypernymy/hyponymy. In METEOR (Lavie & Argawal, 2007), a new metric to assess the quality of Machine Translations, words contained in source and target sentences are aligned according to three factors, one of which is an extended notion of synonymy including hypernymy/hyponymy (the authors never mention Semantic Similarity or Relatedness). These synonymic relations are acquired through WordNet. However, extending their approach to other languages including French, the authors failed to find freely available thesauri save WordNet for English. Thesauri for some language of the European Community exist, as for French, but they are costly and perhaps obscure (EuroWordNet 99). Notably, these thesauri seem to be rarely mentioned in the scientific literature on semantic similarity as we have found no mention of their usage. This metric is

worth being mentioned as the older metric it challenges in the field of MT, BLEU, has been used to assess the validity of free-text answers in a CAA context (Pérez & al., 2004).
As well, the relation of semantic similarity is logically three-way: similarity, non-similarity and opposite (not taking into account the possibility of relatedness). In semantic-taxonomic terms, the relation between two words with respect to similarity can be antonymic.

Memodata is a French provider of various NLP resources. The Alexandria suite includes a synonymy bank built over multiple resources coming on top of a reference synonymy dictionary (CRISCO)[11]: these include several other synonymy and standard dictionaries, bilingual dictionaries, as well as the WordNet (Miller, 1995) and Balkanet ontologies (Tufiş and al. 2004).

The bank is actually installed on a server and sends an XML output file when interrogated about a given word. The files are intended for machine processing and are therefore hardly eye-readable. These outputs provide information on 1/ definition(s), 2/ synonyms, 3/ translations, 4/ usage, 5/ derivates, and 6/ antonyms.

An example of the original Memodata output file for word *approche* (to come/get close to) is shown in Appendix B.

This file is recorded in the entry *approche* and exhibits the word in noun and verb forms, together with synonyms for its various senses, idiomatic usage and translations.

Three main pieces of information are of interest in our project: synonymy, antonymy and grammar-lexical preference (prepositions). Thus, in its verbal form, we can learn from the list of synonyms <SYN> that *rapprocher* is synonymous with *approcher* and that *s'approcher* comes with preposition *de*:

```
<SW ENC='n'>006A006F0069006E006400720065</SW></WD><WD L='fr' W='4'><W
L='fr'>rapprocher</W>
```

And

```
<SW ENC='n'>0073002700610070007000720006F0063006800650072</SW></WD><WD
L='fr'><W>s'approcher de</W>
```

As the basis includes information on derivates, it is quite easy to detect synonymic links across PoS – although the list of derivates is usually limited to a maximum of four lexemes. The method simply consists in adding the derivates of word1 in the search for a match with the synonyms of

[11] As this dictionary of synonyms is nameless, we will thereby refer to it as *Memodata*.

word 2. By the time of the system's completion, we found that this approach has been followed by Snow (Snow and al., 2006) together with lexical-syntactic alignment in his work on Entailment.

Now, a possibility that Memodata does not offer, at least under the terms of our contract, is the exploitation of Wordnet's indices, mentioned in the information. This would be useful, as we would enable to enlarge the list of derivates and add hypernyms by performing searches on the maximum common index sequences on synonyms and derivates.

So, eventually, the lexical relations graspable through the use of the Memodata bank are limited to derivates, synonyms and antonyms, with the addition of compound forms.

V.3. Error Processing

Error Processing amounts to the following activities: error detection and error identification, also named error diagnosis, and finally, if technically feasible, error correction. In practice, error identification and error correction come together. In the domain of CALL, these notions are associated with that of Error Analysis (EA).

Although this project is not directly concerned with EA, we did produce an error analysis (section IV.1). The analysis helped us to define broad error categories, necessary for our project. Now, although error analyses can result in typologies (Anctil 2005), they are rarely formal – particularly, a disturbing point is that there are no standards on the question (other than that of grammar), and these analyses generally result in informal rankings of quantities of language errors. Therefore, we did not try any further to formalize our own analysis. Rather, we tried to set the error types into abstract categories corresponding to possible automatic treatments for detection, and diagnosis when possible. Here, let us say that categorizing errors with respect to the type of process best-suited to detect or correct them is an interesting issue – but it depends very much on the means of language analysis, as well as on the studied language itself. A formal comparative study of the performance of various processing solutions on a common training set of errors would make a promising challenge, but interestingly, no one seems to have taken up this research challenge.

V.3.1. Error Diagnosis

The research in Error Diagnosis is mostly associated with CALL and ITS. This is surprising because other domains in NLP, or even perhaps information technology in general, could use Error Detection/Diagnosis techniques in some applications. In a way, this is already done to some extent: think for instance of spelling suggestions provided by search engines. In the domain of text editors, some tools offer grammatical checking help, whether in the form of integrated processing modules, such as Word grammar checker, or in the form of associated programs, as editing aids. But there is hardly a field of research dedicated to error detection/diagnosis, and it only seems to emerge within the waters of CALL (see for instance, CALICO's special issue on Error Diagnosis, 2003). Perhaps, in NLP's branch of syntactic analysis, the notion of Robust Parsing copes with the problem, but rather than treating it as a language error problem, it treats it as a parsing problem: this is the reason why, in Robust Parsing, breaks may not coincide with the actual error, as wrong attachments are forced when possible, and the benefit of break position is consequently cancelled.

Here is what Bruce Wampler, the author of the Grammatik program (the grammar checker used in WordPerfect), said about grammar-checking to a NY Times journalist in 2002:

[...] *the grammar checking available today, 2002, is essentially no different than it was in 1992* [...] *Essentially, what has happened is that Microsoft has decided that its version of a grammar checker is 'good enough' and has stopped significant work on improvement. No one else in the world has the resources to build a better grammar checker. I'm fairly confident that if you ran your test through versions of Grammatik or other grammar checkers of around 1992-1994 that the results would be almost the same as you got with the most recent versions. And I'd bet you'll get the same results 5 years from now.* (cited in Kies 2007)

This to illustrate that Error Diagnosis – a major component of grammar checkers – has not received much attention as a field of research.

V.3.2. Solutions

There are two main solutions that address the problem of language error in textual inputs: corpus annotation and syntactic constraint relaxation. The first consists in annotating corpora with types of errors, and, possibly, training a classifier on the annotated corpus. Golding (1995) has been the first to think of applying Machine Learning Word-Sense-Disambiguation (WSD) methods to the case of error diagnosis. He showed how the correction of homophonic errors could be done using a corpus on contexts around the tokens *they're*, *their* and *there*. This technique of using corpora to help error diagnosis has been explored but only superficially. As we saw in section IV.1, the FRIDA corpus annotated for the FreeText project has only been used to create a typology of errors (Granger, 2003), and so has Anctil's lexical typology (2005), which has not even been set within a defined CALL or ITS project. Now, we know that such tools as the editing aid/grammar checker Antidote use corpus mining techniques to assess cases of lexical collocation – a task for which corpus-based techniques are appropriate. The art barely ventures beyond data mining. A true application of ML to error diagnosis is to be found in the ALEK (Assessing LExical Knowledge) error detection system (developed at ETS, Leacock & Chodorow, 2000), and builds on Golding's initial idea to produce an unsupervised classifier. The ALEK team uses enriched PoS tag bigrams and trigrams to compute the mutual information of the corresponding words in two corpora: one general and one composed of word-specific sentences (to help model some words' usage idiosyncrasies). Then these values are examined to detect improbable collocations, understood as errors, in a test set of TOEFL essays. The system obtained a 80% precision over a 20% recall. In spite of the results, a clear advantage of this method is its ability to work without a-priori knowledge of the errors – even if this poses the problem of adequate diagnosis.

The second solution consists in augmenting a parser rule set with rules authorizing the syntactic inclusion of errors patterns, to be signaled in the output (see for instance L'Haire & Vandeventer 2003b). This is known as *syntactic constraint relaxation*. After PoS tagging (or after chunking, depending on the systems), the binding of the elements (or phrases) is performed via context-free rules (or dependency rules), and these rules are selected by the application of grammatical

constraints. These grammatical constraints constitute the part to be relaxed in order to handle possible language errors. The relaxation of constraints is therefore done at chunking time (or dependency analysis time), and theoretically, easing chunking possibilities should lead to greater ambiguity. So, the relaxation of constraints asks for great care, based on the *a priori* knowledge of the error types. Here, detection and diagnosis go together.

A major issue with this technique is its dependence on the parser – the parser and the error detection module are merged. There are two advantages to the technique. The first is to be able to produce full parses in spite of the presence of errors in the input (provided that the errors are of a handled type), with the addition of an error signal, and possibly diagnosis. The second is processing time, as parsing and error detection/diagnosis are the same process.

Another parser-based technique for error detection/diagnosis uses *error grammars* (see for instance Schneider & McCoy, 1998). These are simply sets of rules to be applied as post-parsing input processing. The rules are called *mal-rules* or *buggy-rules*. Again, they necessitate the *a priori* knowledge of error types.

Both approaches have flaws, as possibilities of errors are endless and corpora limited in size on the one hand, and on the other, augmenting a parser's rule set requires knowing the types of all possible errors, which seems quite impossible. According to our experience, there can be two major categories of errors, conceptually: errors following a given pattern of error, and errors disruptive of a given pattern of correctness. The two can coincide, but not always. For instance, in French, the use of the inquantifiable determiner form *de le* instead of the contracted determiner *du* is a pattern of error, even if it can be as well understood by the pattern of correctness it violates. Agreement errors are another example, and can conveniently be named as such. Now, a sentence like 32:

(32) *Elles étaient la Belgique* (They were Belgium)

cannot so easily be categorized for the reasons that there can be two correct forms given the error, and that the error is simply unusual.

Having to fit these two categories within detection/diagnosis techniques is quite a challenge. Actually Error Diagnosis is a field of research which shows very few formal studies, and none complete, to our knowledge.

V.3.3. The present approach

Our approach to error diagnosis is hybrid. In section V.1.2, we showed how the parser was modified. But this modification cannot really be understood as relaxation of syntactic constraints. First, the parser was modified to enhance the quality of correct material: to provide good parses for each reference text segment, based on the syntactic nature of incorrectly parsed segments so as not to overfit parsing (to enhance parsing of syntactic types, not just of given sentences). Second, we do not have access to the set of syntactic constraints used by XIP, as already mentioned. Third, XIP is already a robust parser, and it is quite tolerant with respect to language errors – the problem here is not to provide XIP with rules of constraint relaxation.

When in the presence of an error, XIP produces a break or a hole in the syntactic analysis of the input. Therefore, the method of error detection/diagnosis consists more in exploiting these holes and breaks to try to diagnose errors. Add that XIP manages to provide good parses in spite of the presence of lower-level errors. For instance, agreement does not come into account in producing good parses (in Vandeventer 2001, the case study uses agreement errors as a demonstration of the method). This goes up to XIP tolerating confusions of content word types: for instance, XIP usually provides good parses even when the input has an adjective for a noun. So, our work on the parser has consisted in adding more constraints on the parsing, so as to force XIP to create holes or breaks when a rule is breached. This is described in section V.1.1.3.2.

In order to detect errors using XIP as a parser amounts therefore to providing means of error diagnosis for errors that do produce breaks or holes (syntactic errors) as well as for errors that do not (such as agreement errors). Error grammars are probably the best method to address both cases.

V.4. Paraphrase and the Meaning-Text-Theory

In section IV.3 was presented a review of paraphrase in linguistics, both formal and computational. In this chapter the Meaning-Text Theory is presented in more details in a first

section. In the second section, we show the rules derived form this theory as well as some adjuncts.

V.4.1. The Theory

The Meaning-Text Theory (henceforth, MTT) does not explicitly provide a model for the production or recognition of paraphrases, though its models naturally give directions and procedures for it. MTT considers language as a correspondence between sense and text. The correspondence is formally described by Meaning-Text Models (henceforth, MTM). In the models, there are seven levels of representation of utterances: semantic, deep and surface syntax, deep and surface morphology, deep and surface phonology. The levels are linked by rules, called *modules*. The semantic module of a MTM links a semantic representation (formally called RSem) to all deep syntactic representations (RSyntP) carrying the corresponding sense; the deep syntactic module produces for a given RSyntP all the synonymous surface syntactic representations (RSyntS) and so on. The representations are expressed by structures: the semantic structure is a non-linear network showing the propositional sense of an utterance in terms of *semantemes* and semantic relations between them; the syntactic structure is a dependency tree, showing lexemes and the syntactic relations uniting them; the morphological structure is a linear set of lexemes, and the phonological structure is a linear set of phonemes (See for instance Mel'čuk 1981 and 1997, Steele 1990, Polguere 1998, Kahane 2003 and Milićević 2003b).

V.4.1.1. Semantic Structure

At sentence level, the semantic structure is decomposed into both a rhetorical semantic structure and a communicative semantic structure. The rhetoric structure is very theoretical and slightly undefined, the communicative is more practical and rather well defined. It controls the paraphrase possibilities and shows the following information, called *oppositions*, of which only the first two elements are compulsory:

- Thematicity: {theme, rheme, specifier}
- Assertivity: {presupposed, asserted}
- Given/New
- Focalisation: {focalized, non-focalized}
- Perspective: {front, back, neutral}
- Emphasis: {emphatic, neutral}
- Locutionality: {communicated, signalled, performative}

This division of the semantic information frames the paraphrase possibilities within limited boundaries, and the more a sentence is specified, the more restricted are the possibilities. The richness of the model is expressive enough to show the difficulties the automation of this theory would create.

V.4.1.2. Syntactic structure

There is not much to be said of the syntactic structure in the MTT other than that it uses the theory of dependency grammar, and not phrasal grammar. The difference between the deep and surface levels of representation is that function elements of a sentence do not appear in the deep structure. Their definition is the task of the linkage from deep to surface level, and depends on the sub-categorization possibilities of the content lexicon involved in the deep syntactic representation. One must add however that the lexicon is also described in terms of lexical functions which can subsume the lexical representation of the trees, at deep level (think of WordNet categories subsuming words further below in the ontology). This is explained in greater details in the next section.

This reliance on dependency grammar is where the MTT becomes of interest to our project. The research on MTT is quite fragmented, some researchers working on some aspects of the theory. This explains why in the space of the theory, some aspects are more defined than others. The MTT already offers rules of correspondence to express paraphrase possibilities for syntactic paraphrase – while there are very few such rules in what concerns semantic paraphrase, and these

rules call for the lexicon to be modelled following patterns beyond the reach of NLP, or in a form that we would call *pseudo-formal*. The semantic paraphrase rules use a lexicon defined in terms of lexical functions; therefore a dedicated lexical base has to be created, manually – the reason why we judge the model as pseudo-formal. Before listing the rules we use, the next section proposes an overview on the lexicon and the notion of lexical functions.

V.4.1.3. Lexical Functions

Lexical functions are formal tools used for the modelling of lexical relations, such as the phenomenon of restricted lexical co-occurrence and of semantic lexical derivation (Mel'čuk 1996). There are three levels of lexico-functional specification:

- Phrasal vs. Paradigmatic
    - Phrasal: specifiers, modifiers
    - Paradigmatic: synonyms, antonyms
- Standard vs. non-standard
    - Standard LFs can modify a large number of words and take a large number of values in any language. An example is LF *Magn* specifying adjective or adverbs to be applied to nouns or verbs for the expression of "intensity" – *very* is a *Magn* specifier to any adjective.
    - Non-standard LFs apply to a limited number of words. Also called *private*. such is the case for the LF applicable to French: {has 366 days} → "bissextile" ("année bissextile" = *leap year*)
- Simple vs. Complex vs. Configuration
    - Simple: an example of simple LF is *A* which puts in correspondence a noun word and its *owner* or *actor*: under this LF, *A(war)* corresponds to *belligerent*.
    - Complex: a combination of LFs. The complex LF *IncepOper* means "to begin to have" – an example of such a complex LF puts in relation the verb "to contract" to the source noun "disease".

- Configuration: for instance, *Magn(disease)* + *A(disease)* equals *struck down*. This seems to be highly theoretical, and not only difficult to formalize but also quite ambiguous.

Here are some examples of lexical functions in context:

Magn -> sense (intense/very)

Magn(success) = great, unequivocal | antepos (in front of)

Magn(to sleep) = profoundly , like a baby, like a stone, on his/her two ears, etc.

S1 -> sense (person/object doing [what is denoted by] L (current term))

S1(crime) = author [of ART ~] // criminal

S1(murder) = murderer[N], assassin

The set of LFs applies to a lexicon. In the MTT, the lexicon is carefully defined following the categories of the LFs. It's actually the cornerstone of the MT buiding, and we will now show some examples of words defined in the DEC (Dictionnaire Explicatif et Combinatoire – Mel'čuk & al. 2000) – a dictionary in-progress gathering the definitions needed for the theory to be put in practice[12].

Below is an example of a DEC entry. This is incomplete, as it does not include examples of lexical functions other than synonyms (it could as the list of LF is not fixed and one can think of modifiers to *teacher* requiring the definition of a LF: think of *professor emeritus*), and no collocations.

TEACHER[N] **profession**
def
([X is a] teacher of Y to Z (at W)) ≈ ([X is an] individual who professionally teaches subject matter Z (at the institution W))
f l
QSyn: INSTRUCTOR, EDUCATOR, LECTURER

[12] A paper version of the DEC exists up to volume 4 – each volume containing about 2500 words. There is also a lexical base for the domain of Computer Science at the OLST (Observatoire de Linguistique Sens-Texte, Université de Montréal), DicoInfo, which contains 523 words.

sr Y = II = N Z = III = *to* N (W = IV = Locin N) **ex** *Baltazar wanted to become a teacher of physics. He teaches English to adults <business people>. He teaches at the University.*

Table 4. An example of a DEC definition for word *teacher*.

V.4.1.4. MTT and paraphrase

MTT is presented by its advocates as the best model for paraphrase (Milićević 2003a). The main reason for this is that in the MTT, the sense carried by language is considered as the invariant of paraphrases and language is understood as a mechanism to produce paraphrases.

The MTT has created, or used, formal means usable for the modelling of paraphrase – notably, the languages for the representation of linguistic objects, semantic networks and syntactic trees, to be related by the necessary rules and controlled by lexical functions. The theory is attached to the definition of a lexicon oriented on semantics and rich in lexicographic information, contained in the DEC. True enough, till now there has been no implementation of an automated system of paraphrase production/recognition built on the principles of the MTT – at least no other than toy projects (see for instance Nasr, 1996). However, it is true that the theory took very early on interest in the study of paraphrase – in the perspective of language generation as the whole theory originates from work on automation, as said earlier (summed up in Mel'čuk 1981).

J. Milićević (2003a) has dedicated her work on the MTT to the study of paraphrase. She acknowledges herself that in its present state, the model is not complete enough to depict the phenomenon entirely – according to her, the study of paraphrase within the MTT is to be done in breadth more than in depth for reason of the theory lack of completeness. She insists however that the MTT is a suitable theoretical frame for the *production* of paraphrases in the following terms:

The production of speech is an activity more linguistic than the comprehension of speech. Ideally, the speaker knows in advance what she wants to say and to whom she speaks, and only needs purely linguistic knowledge to produce an utterance staring from a given sense (pre-built). Contrary to this, to understand an utterance implies, on top of linguistic knowledge, the resource

of extra-linguistic knowledge – logic, pragmatic, etc. For this reason, the linguistic correspondence is easier to study on the direction of synthesis. Some linguistic phenomenon can only be discovered from the point of view of synthesis [...]. Therefore, for the MTT, the main question is how to express a sense S in a language L? *rather than* What can an utterance U in a language L possibly mean? *In that sense, the language activity amounts to no more than to the production of synonymous expressions, that is paraphrases. From that, to describe a language is to describe its synonymic means.*

The description of the means of production, based on patterns of synonymy, should be more than enough, theoretically, to make possible the activity of paraphrase recognition. Note that, here, synonymy is to be understood as more that just lexical synonymy, but this theory represents the best we have found in order to address our present problem of paraphrase recognition.

V.4.1.4.1. Syntactic Paraphrase in the MTT

The rules we used are adapted from Milićević (2003a). The rules are organized according to two main parameters, as follows:

- Lexical:
    - Synonymic substitutions:
        - Synonyms: *X opposes Y → X fights against Y*
        - Synonymic expression: *photosynthesis → photosynthesis phenomenon*
    - Antonymic substitutions:

      X is likely to stay

      X is unlikely to leave
    - Converts substitutions:
        - Simple: *X fears Y → Y scares X*
        - Complex: *X fears Y → Y is scary for X*

- Derivative substitutions:

  The socialist voters → *the socialist vote*

- Structural:
  - Fission / Fusion

    A lexeme, or phrase if illustrated from the point of syntax, is broken into several related pieces.

    Ex: *aristocrat* → *person from the aristocracy*

  - Re-tagging

    A lexeme, or phrase if illustrated from the point of syntax, is assigned a new category.

    Ex: *cat* → *a feline mammal* (here, *fission* implying *re-tagging*)

  - Transfer of a dependant to a new governor

    Ex: *My neighbour's cat* → *my neighbour has a cat*

  - Inversion

    Ex: *The flag of Canada* → *Canada's flag*

The structural categories usually combine themselves; examples of the categories and their combination will be shown with the enumeration of the rules covered in this project. Note that one lexical category cannot be automatically recognized: that of converts. This is a problem that has not found a solution to date in NLP.

The rules are exploited in their dependency form: two sets of dependencies are put in relation, with common words represented under their indexed PoS category, the index is that of the position of the word in the ordered dependency chain. In the system, a sentence is first parsed – the resulting dependencies of Ref and S are compared to each other; should a synonymic / antonymic / derivate relationship be discovered between the two sentences, the synonyms/antonyms/derivates are replaced by tags such as: SYN*n* / ANTO*n*. Derivates are treated as synonyms. For instance, the sentences:

They are friends → *They are not foes*

are first parsed:

1. SUBJ(be,they)
   OBJ(be,friends)

2. SUBJ(be,they)
   OBJ(be,foes)
   NEGAT(be,not)

and then normalized over indexed PoS categories and lexical relation (here, antonymy between *friends* and *foes*):

1. SUBJ(VERB1,pron)
   OBJ(VERB1,ANTO1)

2. SUBJ(VERB1,pron)
   OBJ(VERB1,ANTO1)
   NEGAT(VERB1,adv)

The dependencies, or a subset of them, are then sent to a paraphrase module which compares the pair with rules such as this:

OBJ((VERB[0-9]),(ANTO[0-9]))→ OBJ(z0,z1),NEGAT(z0,pron)

Note that the z*n* symbols are indices referring to items identified in the left portion of the rule. VERB*n*, ANTO*n*, etc. are anchors for the vocabulary common to the sentences candidate to paraphrase. The above rule recognizes that *are friends* and *are not foes* are equal in term of sense. The subject dependency is discarded for perfect equality in the two sentences: *they are*. If the subject relations were not equal, they would not be discarded and the paraphrase rule would not be able to empty the dependency sets of S and Ref alone.

Note that in some cases, the PoS categories can be lexicalized to enforce a greater degree of control – for instance: NEGAT(z0,pron_*not*).

V.4.2. Rules

Below are the rules supported in the system. All the rules stored in the paraphrase base are presented. However, due to some lexical relations being difficult, or impossible, to detect automatically (converts, support verbs), some rules can only control the PoS of an item, and not its lexical validity. We signal those cases by the † symbol. Another limitation is also due to the quality of the synonymic base, which provides support words with inconstancy - the *Republic*

entry provides *Republican State*, or *fog* provides *curtain of fog*, but *Photosynthesis* does not provide *Photosynthesis Phenomenon*. These are severe limitations to the module, but at least it enables the syntactic validation of the compared structures.

Instead of the actual rules, expressed in terms of dependencies, we only show examples of sentences covered by the rules, for clarity. Each example corresponds to one rule. Note that some categories of paraphrase are naturally equivalent in the dependency analysis; this is the case for subject omission:

Balthazar took his pill and he calmed down → *Balthazar took his pill and calmed down*

as well as for some subordination correspondences:

The woman I love lives next door → *I love a woman who lives next door*

V.4.2.1. Rules from the MTT

The examples are English examples – as the rules are interlingual, they hold for French comparable utterances.

1. Synonymic Substitution:

    a. Simple:

    *Where's my **car**?* → *Where's my **drive**?* → *Where's my **coach**?* etc.

    b. Change of function:

    *Balthazar calmed down **as** he **breathed** deeply*

    *Balthazar calmed down **after** a deep **breathe***

2. Synonymic Substitution with Fission:

    a. Noun to Noun:

Photosynthesis → *Photosynthesis* ***Phenomenon***

b. Noun to Adjective:

Republic → *Republican* ***State***

c. Metaphoric

Fog → *a* ***curtain*** *of fog*

3. Synonymic Substitution with Support Verb Fission †:

a. Simple Support Verb:

I ***respect*** *him* → *I* ***have*** *respect for him*

He ***limps*** → *he* ***walks*** *with a limp*

The samples must be ***analyzed*** → *the samples must be* ***submitted*** *to* ***analysis***

To ***translate accurately*** → *To* ***make*** *an* ***accurate translation***

b. Support Verb of an Agent Noun:

Oswald has ***murdered*** *Kennedy…* → *Oswald* ***is*** *the Kennedy's* ***murderer****…*

Balthazar ***cooks well*** → *Balthazar* ***is*** *a* ***good cook***

c. Support Verb calling for an adjectival form of the initial verb

He ***knows*** *about the case* → *He* ***is aware*** *of the case*

He ***forgets*** *things* → *He* ***is forgetful***

4. Antonymic Substitutions:

a. Simple Antonymic Substitution:

*He is **likely** to **stay** → He is **unlikely** to **leave***

Employment** goes **down** → **Unemployment** goes **up

5. Antonymic Substitution with Fission:

a. Simple:

*To **like** → **not** to **dislike***

*Balthazar **keeps** his temper → Balthazar **does not lose** his temper*

b. Propagated to Dependants

*To feel **good** → **Not** to feel **bad***

*They are **friends** → They are **not foes***

*I find her **attractive** → I do **not** find her **unattractive***

*I am doing this **with** hesitation → I am **not** doing this **without** hesitation*

***Everybody knows** the rules → **Nobody ignores** the rules*

6. Convert Substitution †:

a. 2 roles:

*I **fear** the consequences → The consequences **scare** me*

b. 3 roles:

*The duke **has** given that teapot to my aunt*

→ *My aunt **was** given that teapot by the duke*

→ *That teapot **has been** given to my aunt by the duke*

c. 4 roles:

*They **charged** me 100 dollars for the repair*

→ *I **payed** them 100 dollars for the repair*

→ *I **spent** 100 dollars on the repair*

d. Convert Support Verb

*The general **gave** this order to Wolfgang*

→ *Wolfgang **received** this order from the general*

*The proposal to **modify** the status **comes** from the assembly*

→ *The proposal of the assembly **bears** on a **modification** of the status*

e. Simple Non-Verbal Convert †

*Penelope is **sure** that Ulysses will come back*

→ *That Ulysses will come back is **sure** to Penelope*

→ *Ulysses' return is **sure** to Penelope*

*Gus is Balthazar's **employer** → Balthazar is Gus' **Employee***

f. Support Non-Verbal Convert †

*Canada is a big **exporter** of wheat to the USA*

→ *The USA is a big* ***importer*** *of wheat from Canada*

7. Convert Substitution with Fission †:

a. Patient Convert.

Gus ***hugged*** *Balthazar* → *Balthazar* ***got*** *a* ***hug*** *from Gus*

The police ***are investigating*** *the case*

→ *The case* ***is*** *under* ***investigation*** *by the police*

b. Adjective Support Verb Convert

I ***doubt*** *the truth of his statement* → *The truth of his statement is* ***doubtful*** *to me*

8. Simple Derivate Substitution:

The socialist ***voters*** → *the socialist* ***vote***

9. Derivate Substitution with Fission:

The latest measures will affect 50% of the ***workers***

→ *The latest measures will affect 50% of the* ***work force***

10. Derivate Substitution with Head-Switching:

a. to Adverb[13]:

*They **retaliated** by **burning** the village → They **burned** the village **in retaliation***

*Balthazar **ballooned across** the Pacific → Balthazar **crossed** the Pacific **ballooning***

*It is **clear** that he will come back → He will **clearly** come back*

*A **huge number** of people showed up → People showed up **in huge numbers***

b. to Adjective:

*The **insufficiency** of the resources may cause problems*

*→ **Insufficient** resources may cause problems*

*The **excessive use** of this drug may cause...*

*→ **Used excessively**, this drug may cause...*

Japanese countryside → Rural Japan

11. Head-Switching with Fission:

*This is **known intuitively** → This **knowledge** is **intuitive***

*He **sleeps soundly** → He is a **sound sleeper** → ! His **sleep** is **sound***

12. Implication Rules:

a. Causation of Changing State → Change of State †:

*He put the baby to **sleep** → The baby fell **asleep***

*He **started** the engine → The engine **started***

[13] In the MTT, adverbs are understood as verb modifiers which do not fill an agent position. They include adverbial phrase.

b. Causation of State → State †:

*He **was sent** to jail → He **is** in jail*

c. Start of State → State †:

*The baby fell **asleep** → the baby **sleeps***

13. Translation Rules:

a. Autonomous:

*She wishes to **travel** → She wishes for a **trip***

*The dogs **bark** → The dog's **bark***

*This is a **truth** → This is **true***

b. Induced †:

*She **analyses** the samples in a standard way*

*→ She **submits** the samples to a standard **analysis***

*This is **evidently true** → This is an **evident truth***

14. Verbalization/Nominalization of the Syntactic Head:

*The prices **rise** → The price **rise** → The **rise** of prices*

15. Syntactic Restructuring:

a. Coordination → Subordination:

The temperatures cooled down after the 40's ***but*** *started to rise again after the 70's*

→ *The temperature* ***which*** *cooled down after the 40's started to rise again after the 70's*

b. Conditional → Relative:

People can join ***if*** *they have a ticket* → *People* ***who*** *have a ticket can join*

c. to Subordination †:

The temperature of the warm period

→ *The temperature* ***that prevailed*** *in the warm period*

d. Subordination to Participle:

Orders ***which comes*** *from the top* → *Orders* ***coming*** *from the top*

The rise in temperatures ***which was observed***

→ *The* ***observed*** *rise in temperatures*

As the above shows, two kinds of lexical relations cannot be automatically captured: converts and support verbs (the paraphrase relations which cover those cases are noted with a †). Therefore, the detection of paraphrases obeying these two patterns can only happen syntactically – this yields possibilities of not detecting errors, of the lexical kind: the presence of a bad convert or support verb. But at least, the admissibility of a paraphrase structure is accounted for.

V.4.2.2. Adjunction of rules

On top of the paraphrases patterns adapted from the MTT, several other patterns have been included in the base. Sometimes the MTT provides patterns of paraphrase equivalence which show a quite high level of complexity, as the MTT's rules are built on examples – therefore simplified versions of the paraphrase rules have been added.

Also, we added some paraphrase rules resting on lexical information, when it was felt that the lexical adjuncts involved in the transformation did not as much add lexical or semantic content than induce a shift in the discursive paradigm. Actually, these cases are treated in the MTT as patterns of *semantic paraphrase*. But we have no means to process these cases as 1/ the semantic paraphrase rule set is incomplete in the MTT, and as 2/ for lack of a lexical function definition procedure, we would have very little means to control the validity of the lexical adjuncts in the transformations (converts and support verbs are already an instance of these adjuncts, but they occupy a syntactic position necessary to the well-formedness of the sentence with respect to the original lexical information, and this principle of composition is constant, independently of semantics). So these cases of paraphrase patterns have been created from the experience we have of the observed paraphrases in the answer test set, and not adapted from the MTT – though, in practice, it makes little difference.

Below are listed these two kinds of rules, divided in two categories: purely syntactic and lexical.

V.4.2.2.1. Syntactic

1. Transformations from Adjective *Attribut* → Adj *Epithète* → Infinitive V → conjugated V → Noun

E.g. *gestation est écourtée* → *gestation écourtée* → *écourter la gestation* → *écourte la gestation* → *raccourcissement de la gestation*

(pregnancy is shortened → shortened pregnancy → to shorten pregnancy → pregnancy shortening)

2. Transformation from Noun Complement → Simple Sentence – the noun complement is preceded by an attribute or action verb

E.g. *[...] la baisse des eaux* → *les eaux baissent* – note that this is slightly different from 1, here no transformation to Adj is possible; therefore, it's an addition to 1.

(lowering/dropping of the water level → The water level lowers/drops)

3. Transformation from Noun Complement, to be reduced to only one of the head/modifier noun + noun structure.

E.g. *Des cris de détresse* → *Des cris*

(crys of distress → crys)

E.g. *Des enregistrements de cris de détresse* → *Des enregistrements* or *Des cris* (*de détresse*)

(recordings of crys of distress → crys (of distress) *or* recordings)

4. Transformation from a Definite Article → Indefinite

This type of reformulation is actually taken into account inside higher-level paraphrase possibilities (see Lexical ex 4).

V.4.2.2.2. Lexical

1. Transformations from *Attribut du Sujet* + Subordinate → Simple sentence

E.g. *Le rat est un animal qui possède [...]* → *Le rat possède [...]*

(The rat is an animal which has [...] → The rat has [...])

2. The same, but where the *Attribut du Sujet* is modified by a PP

3. Transformation from Noun + PP → NP + VP + NP (V replaces Prep)

E.g. *Des couples avec de la difficulté [...] → Des couples qui ont de la difficulté [...]*

(Couples with troubles [...] → Couples who have troubles [...])

4. Abstraction on the paradigm of *possibility*

E.g. *Le temps de gestation est écourté → Le temps de gestation peut être écourté → [...] La possibilité d'avoir un temps de gestation écourté*

(Pregnancy is shortened → pregnancy can be shortened → [...] possibility to have a shortened pregnancy)

5. Transformation from *C'est... que/qui* (Focus → Simple Sentence)

E.g. *C'est la couleur que je préfère → je préfère cette couleur*

(It's the color I prefer → I prefer this color)

With elision of the thematic noun:

E.g. *C'est la couleur rouge que je préfère → je préfère la rouge*

(It's the red color I like best → I prefer the red)

6. Polymorphism of attribution

E.g. *le rat possède un système cardio-vasculaire → Le système C-V du rat*

(the rat has a cardiovascular → The rat's cardiovascular)

E.g. *une fleur avec épines → une fleur qui a des épines.*

(a rose without thorns → a rose which has no thorns)

Note that this is only to control reformulation, not to monitor it[14], as a vast number of counter-examples can be found. The system relies on the student's good sense. Plus, the sets do not pretend to exhaustion of paraphrase possibilities – a research subject in itself, and one which passes the scope of a PhD thesis, or so we feel.

Explanations of an S/Ref pair paraphrase processing will be given in the *corresponding* section, as all patterns of paraphrase are not equally processed.

In conclusion to this section on paraphrase, only a partial answer has been found to our problem of paraphrase recognition. We showed that theories and even formalisms to explain or express the various kinds of paraphrases do exist, and that in the course of linguistics history, these theories and formalisms have evolved into a more and more precise definition of paraphrase – but also that they are far from being machine-ready. We believe that the formalism and theory proposed by the MTT encompass the only comprehensive approach to paraphrase processing.

Practically, the set of paraphrases assembled in this way has been sufficient to enable recognition of syntactic paraphrases coming from our set of student answers, as will be shown in the Results section, although cases of failure did happen and will be explained.

[14] Think of the difference between controlling and monitoring as the same between language production and understanding: monitor rules could participate in producing language, as they are deterministic, while control rules can only help understanding as their validity is context dependent. Therefore these rules do not fit within the criteria of MTT – but this is one limitation of *adaptation* anyway.

Chapter VI. Evaluation and Results

In this section, we present the results of the automatic assessment of our student's answers set. To evaluate properly the automated validation of free-text answers amounts to an evaluation of all aspects of the process, as well as their interaction. Some aspects have been defined and implemented for this system, while others come from third parties. So, along with a single measure of accuracy, various measures of the system performance will be given following the definitions mentioned in the introduction. As well, methods for the definition of gold standards and baselines at the various steps of the evaluation are discussed subsequently.

VI.1. Introduction to the aspects of the evaluation

In our work, the notion of success or failure does not apply to an answer, but to the verdict of the system. Besides, as the system has certain capacity to detect errors, and correct some of them, error detection is also to be evaluated in terms of success or failure. We should also evaluate the language modules that help the analysis: the parser, the synonymy bank and the paraphrase pattern bank.

Note that in an appendix E all the answers are given together with an explanation of the given verdict.

So, theoretically, success can occur in the following quantitative terms:

- Assessing the student's answer to the question:
    - Good answer

    - Wrong answer
- Detection of syntactic errors (high-level syntactic errors):
    - Provoking a break in the syntactic analysis
    - Or, excluding words from the analysis
    - Provoking a correct but inappropriate analysis
- Detection of a language error (Low-level syntactic errors):
    - Orthographic
    - Agreement
    - PoS selection
- Synonymy detection:
    - Retrieval of a good synonymic link (from synonymic bank)
    - Retrieval of a wrong synonymic link
    - Failure to detect a good synonymic link
- Paraphrase:
    - Recognition of a paraphrase pattern, or failure

Some of these terms could also be evaluated qualitatively. This would amount for the system to giving a judgment *correcting* the error(s), or driving the user to correct the answer provided with adequate material, accordingly:

- Assessment of the answer
    - What's missing if the answer is partially good
    - What's correct in a wrong answer, if anything
- Syntactic or language error:
    - Solving the error, either by:
        - Identifying precisely the nature of the error
        - Giving a solution, or all

The remaining terms are deterministic and so cannot be evaluated in another way than binary. We mentioned the aspect of *qualitative* success as, in this project, some work has been done in order to provide judgment on the error type when high-level syntactic errors happens and are detected. But this is quite unreliable and very incomplete as the task is difficult. Commercial editing aids simply do not cope with errors at this level, so we are unashamed of having barely succeeded. An interest of this partial work is to provide tactics to cast light on the sentence's elements which are implicated in that kind of errors: the words at error position, the subcategorization frames of these words, etc.

Practically, the results should also be evaluated relative 1/ to the form of the answers, and 2/ to the question, taking into account the following:

1. Whether the answer, if good, is an exact match or a reformulation of the reference – in other words, the share of reformulations correctly evaluated.
2. The share of syntactic reformulation within the set of correct answers – semantic reformulations simply cannot be automatically recognized.
3. The share of correct answers which have not been detected due to problems with third party components:
    - direct errors:
        - Parser errors
        - Synonymy errors
    - indirect errors: due to the limitation in the system's modules responsible for the exploitation of these third party components.
4. At question level:
    - The ratio of correct judgments
    - The ratio of semantic reformulations
    - The ratio of syntactic over semantic reformulations
    - The ratio of non-grammatical answers

- The ratio of clumsy utterances[15]

The figures for items 1, 2 and 3 are of interest to the evaluation of the system itself, while some aspects of item 4 are of interest to the activity (of free-text open-question answering assessment) in general. The idea is that as the science can only correctly evaluate some types of answers (and this is an observation not limited to this system), it should be interesting to distinguish between types of answers which more or less motivate semantic reformulation, or reformulations in general.

Finally, figures of correctness should be given for the third party components per se: this amounts to the accuracy of the parses for the parser, and the presence or absence of a sound synonymic link within the synonymy basis.

VI.2. Baseline

Baseline measures can be defined for some aspect of the evaluation only – those aspects for which baseline evaluation methods can be devised. As it seems, such is not the case for synonymy detection, paraphrase recognition or error detection, although this depends on the definition of the concept of baseline itself. Usually, a baseline for a given computing problem is established on the basis of competing or anterior processing approaches to this problem, but here, there are none. So the baselines are conceived as the simplest possible process to solve a problem. We present below the baseline methods for each aspect of the evaluation when possible or else explain why no baseline could be defined.

---

[15] *Clumsy utterances* are sentences showing a somewhat correct but inappropriate language form, either for reasons of lexical selection – selecting words contextually misplaced or misused – or for reasons of bizarre syntactic usage. The main challenge with such answers is that they may show structures impossible to assess in terms of paraphrase. Therefore, the candidate answer cannot be *corrected* nor be judged wrong, as it is not incorrect syntactically, and it cannot be, or can barely be, recognized as a correct answer – or when it is, the judgment of correctness turns to be abusive. To sum up, such sentences are neither good or bad, and impossible to assess in principle.
A not too extreme example of such a sentence is (independent of the gender agreement errors): *la système cardio est la même système pour l'homme et pour le rat.* (The cardio system is the same system for the man and for the rat).

VI.2.1. General assessment of answer correctness

We use the simplest way to measure that an answer is the right answer to a question: exact match with the reference answer. By exact match we mean that the student answer should be equal to the reference both in lexical content and in syntactic structure. However, we point to two remarks which should further constrain the baseline.

VI.2.1.1. Partial answers

The first and most constraining remark concerns the partial or complete nature of the answer compared with the reference. The overlap between S and Ref can be complete or partial, as most questions only necessitate a fragment of the reference answer to be correctly answered. But this aspect of completeness is dual, at least in principle: completeness can be read at phrase level, or at intra-phrasal syntactic level. To admit intra-phrasal partial completeness amounts to including the means of syntactic pruning in the baseline. As free-text answer validation is new, even if this project is not the only one of its kind, and as the research starts from scratch, including syntactic pruning amounts already to a departure from a baseline, and to the inclusion of a degree of reformulation in the baseline, so it is excluded from this baseline measure. This means that overlap can only happen at phrase level in the definition of this baseline.

To admit that overlap can be partial at phrase level implies that the use of a parser is necessary to the definition of a baseline – otherwise, the delimitation of phrases in the sentence cannot be performed and there are no means to distinguish between the phrasal units composing the sentence. Choosing to include means of parsing in the baseline amounts to a debate between evaluating the answer at sentence or dependency level, content or structure. We believe that both are admissible, but both approaches induce a bias in the baseline, and the preferred bias should follow the requirement of the application (actually, this question of bias is also valid for syntactic pruning).

If the answer material is to be evaluated at sentence level, good answers only can be detected, as exact matches are always good answers; this would induce a precision bias. If the answer

material is evaluated at dependency level, some dependency relations could be suppressed in a rule-based manner. But then, the baseline could find correct answers which aren't, as we have no means to distinguish necessary parts from optional adjuncts (think for instance of verbs which can be both transitive and intransitive – then, cutting the complement is admissible, but what if the complement is contextually necessary?). This approach would induce a recall bias. It is another debate whether the application requires giving preference to precision over recall, or the contrary. But we chose to favour the precision bias, as otherwise favouring the recall bias could open a door to other means of computing a baseline which would drive the measure away from the needs of the application: to judge an answer to a question as correct or incorrect. In practice, the results as they are show a 100% precision in what concerns success: answers assessed as correct. Therefore, correctness of the answers drives the baseline method, which should maintain perfect precision.

VI.2.1.2. Extraneous material

The second remark has to do with the admissibility of extraneous material. The question is here whether a candidate answer with material absent from the reference answer can be judged correct in the baseline if this candidate answer also shows an exact match with the reference. This remark is secondary to the remark on partial answers, as in any case, whether augmented or not, the candidate answer has to be a correct answer. Here, we believe that the solution to the problem should follow the way in which the same problem is solved in the implemented system. The system halts the evaluation of the answer material's correctness to what it can assess as correct; therefore extraneous material cancels any judgment of correctness for what has been otherwise recognized in the candidate answer. The system in its state is quite conservative: any answer that shows content material absent from the reference is judged wrong, even if it shows otherwise what should be of the reference (See section III.2.1). This is the procedure.

Now, note that some extraneous material is admissible in the system, when it appears as intra-sentential adjuncts such as adjectives or adverbs – or noun complements in general. In other words, adjuncts are admissible only when they do not affect the thematic structure of a sentence with respect to the reference answer. To control that extraneous material does not affect the thematic structure, extraneous material can only modify words which are themselves modifiers

in the reference at the level of subjects, objects and verb modifiers. So, in order to reflect this state of things, the baseline should also rule out answers containing extraneous material affecting the referential thematic structure.

The example (33) shows a reference sentence – examples (34) show a sentence whose thematic structure is not affected by an adjunct (34a), and two whose themes are affected (34b, c) :

(33) *Un chaperon est mis pour éviter les distractions de son oeil de faucon pendant le transport sur son lieu de travail.*

(A hood is put to stop the distractions of its hawk's eye during transportation to its workplace)

(34) a. *Un chaperon est mis pour éviter les distractions de son œil de faucon pendant le transport* ***de l'oiseau*** *sur son lieu de travail.*

(A hood is put to stop the distractions of its hawk's eye during **the bird's** transportation to its workplace)

b. *Un chaperon est mis pour éviter les distractions de son oeil de faucon pendant le transport sur son lieu de travail,* ***donc sa aide avec les distractions.***[16]

(A hood is put to stop the distractions of its hawk's eye during transportation to its workplace, **so it helps with the distractions**)

c. *Un chaperon est mis* ***par le technicien*** *pour éviter les distractions de son oeil de faucon pendant le transport sur son lieu de travail.*

(A hood is put **by the technician** to stop the distractions of its hawk's eye during transportation to its workplace)

In conclusion to this remark, the baseline is an exact match to the reference – in other words, the simplest way to automatically assess the correctness of a segment with respect to a reference. This excludes parts of the question in the candidate answer. Syntactic pruning is ruled out at both sentential and phrasal level, but its opposite, the presence of extraneous material (which amounts

[16] Examples are presented as they are in the student's set, inclusive of possible language errors, which is the case here.

to syntactic augmentation), is admissible when it does not affect the thematic structure of the sentence. This last point implies that eventually the baseline should be measured from the answer as a set of dependencies rather than as an answer, and so it admits at least the sophistication of syntactic parsing.

VI.2.2. Specific baselines

The question of defining baselines for the subtasks involved in the general assessment of correct answers is examined in this section, and we explain why no baselines have been devised for these subtasks. The subtasks are error detection, synonymy recognition and paraphrase recognition. Note that the questions of baseline measures for the performance of third-party components are not considered here, but only the processes exploiting these components.

VI.2.2.1. Error detection

As it is, there are two kinds of errors to be detected, as we make a distinction between syntactic errors which produce a break in the analysis and errors which do not produce such a break.

In the system, the first kind of errors are errors which challenge the sentence's correct syntactic analysis and are detected by means of one single process. This process can detect either breaks in the analysis or words excluded from the analysis. The second kind of errors deals with language inconsistencies within dependencies, such as agreement, and are detected by heuristic means of a set of rules. The rules work on both the set of dependencies and the set of grammatical features attached to each word, i.e, a set of dependencies whose words are composed of sets of attributes. As a matter of fact, applying these rules to the sentences is a primary task which could not work in any other way, except by means of machine learning. A rule attribute is atomic, and the sets of attributes compose a closed item to be matched binarily with a canonical reference – only the elements on which rules work could be different. However simplistic baseline measures can be, they still have to emit a judgment based on some measurable material. So, the aspects of errors are presented in turn, to see if any measure simpler than what we use can be defined.

- Syntactic errors producing either a break or exclusion of a word:

These errors are to be detected in terms of position, as we said. This can only work on the syntactic analysis provided by the parser, or else the baseline would be more complicated than the process as it is: it could be 1/ pattern matching, but the set should contain a possibly infinite set of structures, not to speak of ambiguity, or 2/ a machine learning approach, but we have no corpus. So, the process already represents a baseline, so to speak. Besides, as an error appears as a break or a missing word, the question is more to know whether the parser, as modified, makes proper way to error detection by systematically producing breaks or excluding words when in presence of an error. So, there is no baseline for syntactic errors detection.

- Orthographic errors:

Here, baseline measures could be defined, but only for orthographic error detection, and not correction. For instance, the words in the candidate answer could be compared with those in the reference and judged erroneous when showing only a single-character difference with the correspondent in the reference. But then again, the baseline would be more complicated and costly than the process as it is, and we know in advance that the results of this baseline would be worse than those of the process measured, which gives perfectly good results except in one or two cases. So, here again, no baseline is provided.

- Agreement errors:

Agreement errors are evaluated based on the sentence's words' feature sets. The definition of a baseline can only rest on the comparison of features, however simplified, so then again, the way in which agreement errors are evaluated in the system already constitutes a baseline. Now, the feature set is acquired through parsing, therefore a baseline could be defined using features simpler to access, such as morphological suffixes, or the use of a lexical base – all to be exploited in rules. But that would be another complex procedure for a quasi non-problem, as the agreement error detection module works very well as it is.

- Part-of-Speech selection:

Some categories are incompatible for syntactic dependency, such as determiners and adjectives. In the system, the rules to check this matter of consistency constitute a set by itself. But the rules work on the indices of PoS categories present in the syntactic feature set obtained after parsing for each word, so the problem is the same as that of agreement errors, one of binary control, and no baseline is defined here.

VI.2.2.2. Synonymy Recognition

There is no baseline for synonymy detection. The problem of detecting synonyms is of the same kind as detecting agreement errors: discrete – either the information is present or absent, and there is no refinement in between. Besides, baselines are interesting to problems only solved up to a certain degree of success. Here, the synonymy detection procedure is often defaulted, but not for reason inherent to the detection procedure in itself. A baseline could be provided using a simpler bank of synonyms, as that of Memodata is already quite complex, in spite of its incompleteness, but that would be a big effort for results of little interest, or so we believe. So, no baseline has been defined for synonymy detection.

VI.2.2.3. Paraphrase Detection

The problem is different than that above, although here too the decision is binary: a sentence B is a paraphrase of sentence A or it is not – recall that the paraphrase recognition process works on maximally pruned sentences, so that the question of partiality does not hold. The difference is that paraphrase recognition is no primary task and involves processing steps that make it complex: paraphrase recognition is no heuristic task.

As implemented in the system, paraphrase detection works at dependency level, and it involves the possibility of lexical reformulation. Sets of dependencies are paired in relations of paraphrase. But this could be depicted in other ways – for instance lexically, putting pairs of sets of ordered words in relation. This at least says that a simpler paraphrase recognition process

could be defined – as it is, the process is already slightly above the level of a baseline; or else it would make a good baseline measure for a machine learning paraphrase recognition system, or for a system motivated by lexical functions. It is located at an intermediate point of complexity. Another difference is that of the nature of error: when a student produces a sentence that is thought to be a paraphrase of a reference, in case of error, is the error an error of paraphrase or an error of language? This is debatable: the error could produce a syntactic malfunction, or only an ambiguous structure. Working at the level of dependencies minimizes the impact of such errors, as low-level syntactic errors are bypassed at parsing time, which enables both to recognize the paraphrase relation, as well as to detect or solve the language error in some other step of sentence processing. Therefore, simplifying the process of paraphrase recognition would reduce the parallelization possibilities, and the comparison with the process as it is would be skewed. To devise a baseline for paraphrase recognition is an interesting challenge, but a complex one. As the only cases of failure to recognize paraphrase in the system come from reasons other than procedural, we leave this question aside.

Eventually, we only come with baseline figures for the system as a whole. The fact that no baselines are provided for the subtasks is due to two main reasons: 1/ that establishing baselines would require processing efforts exceeding those in the system itself, and that 2/ the subtasks show very few direct errors – the errors are due to parsing or clumsy language. This means that at best the baseline could match the performances of the subtasks as they are. It seems that the question of gold standards is more important.

VI.3. Gold Standard

All the answers have been evaluated manually. In this section, we explain how this has been done, and the kind of information selected for establishing the gold standard. This also gives us the occasion to take a look at what is not evaluated by the system, as some aspects of language correctness are not. The gold standard measures are richer than baselines, and we believe they are more important.

VI.3.1. Marking Scheme

The question of the marking scheme follows the orientation on quantitative or qualitative evaluation. Here, the marks are binary: good or bad – but the items of evaluation, as presented in the preceding section, are multiple, so that there is a balance between quality and quantity in the scheme. The exact items evaluated in the gold standard are presented at the end of this section. A mark established on a numeric scale could even be attached to each example, but that is not our priority. This would be interesting if the system were more than a prototype study. Besides, if all aspects of the quality of an answer are equally interesting in order to judge an answer in the context of Second-Language Learning, such is not the case in this context of automation, as we know that some processes work better than others. Orthographic and agreement errors are easier to detect and to solve than other types of errors, so they are of a lesser scientific interest.

As it is, we believe that a binary marking scheme of the diverse aspects of correctness is enough to provide a precise evaluation of the system.

The evaluation of the answers was done by this researcher as well as by a former FSL teacher of the University of Ottawa's Second Language Institute. The evaluation was conducted on the basis of agreement between the two parties, so that the question of consistency of the manual evaluation does not arise. The procedure of agreement between annotator seemed realistic due to the small size of the test set, showing no real conflicting cases, and any case of ambiguity has been resolved by assigning the answer a label of correctness. We are, however, aware that any larger experiment would have to take into account the rate of annotator disagreement.

The next sections deal with how errors should be identified and the modalities of the identification, with respect to comparing a gold standard to the system's results.

VI.3.1.1. Identification of errors

- Content:

Some answers showing contents which might be interpreted as good answers have been judged wrong when the contents reflect parts of the reference text which have not been retained as part of the reference answer. This is questionable, but the gold standard is used to assess the system's performance rather than answers. Besides, these cases are recorded as such and constitute a specific subset of "wrong answers".

- Language:

The question of style and of language usage, or cohesion, has been left aside, as this is not dealt with in the automatic process, even if the style and usage are important in the context of SLL. Our aim here is to only evaluate what is evaluated by the system itself. Therefore language correctness amounts here to syntactic and grammatical correctness.

VI.3.1.2. Mode of identification

- Detection vs. correction of language errors:

Another important aspect of the evaluation has to do with the expected gap between detection and correction of language errors, as already mentioned. So a distinction between both should be made at the time of evaluation. Some language errors are easy to detect and correct while others can only be detected. Editing aids are good indicators of what automated results should be squarely put against a human evaluation, and what should not, or at least, cannot be hold as being a product of design inconsistencies.

Therefore, in the experiment, detection only should be pertinent to the evaluation for high-level errors, such as those producing breaks in the analysis, while detection and correction are pertinent to the evaluation for lower-level errors. This is a tricky aspect of comparing a human gold standard with the system's evaluation, as human correctors do not detect errors but identify them, and most often through correction. This represents a limit of the computationally-driven approach. See the next section.

- Format of high-level errors detection:

The question of the format of feedback to high-level syntactic errors is important when comparing the results of the evaluation results by the system vs. the humans. In the system, feedback on these kind of errors is provided in the form of a structure break position in the sentence. This is not how humans work. Rather, the error is immediately identified in terms of its nature: i.e., the correct reference is what enables the human judge to identify an error, based on the context surrounding the error. Here is an example:

(35) *On peut les tirer parce que les rats sont des animaux qui ont un système cardio-vasculaire * très * semblant * que celui des êtres humaines.*

(We can draw them because rats are animals which have a cardio-vascular system very *seeming* that of the human beings.)

The error is that 1/ the student took a present participle, *semblant*, for an adjective, and 2/ she used as a connector to the attached adjectival modifier, *celui des êtres humaines*, a subordinating conjunction instead of a preposition. That is the *likeliest* case intuitively. But there could be other analyses of the error: for instance, that the student made a lexico-semantic error in selecting verb *sembler* instead of *ressembler* in this context, or mistook the lexico-syntactic behavior of a comparative quantity adverb (*plus/moins...que* – more/less...than) with that of simple quantity adverb (*très* - very). These analyses are mutually exclusive.

The system will detect a break at the * points, and will provide feedback based on this information – namely, that sentence analysis is split into two unrelated parts of which the words preceded by * are excluded. Therefore, the pedagogical value of the feedback is a function of the procedure's ability to break at the point of error in the sentence. A human judge would evaluate things quite differently, at best recording all the aspects of the error, and possibly providing a solution. The system judges the error as it is structurally, while the human judges as the sentence should be. The system is still able to tell the student that all words excluded are inadequate, but can barely state what they should be[17].

[17] Actually, here the system does provide qualitative feedback: that *celui des êtres humaines* can be attached to the verb without use of a conjunction, which is structurally true. The problem is that the student mistook adjective

Now, an interesting aspect of error identification by human correctors is that they tend to put errors within categories. In example 35, the errors can be characterized either as syntactic or lexical. Such is not the case in the system.

In the case of correction evaluation, the rule is the gold standard, beyond human gold standard. In this way, if the system provides a good correction, it matches the gold standard even if the human evaluation has missed one possible explanation for having preferred another explanation. This is a clear advantage of our loose approach to human evaluation. Practically speaking, the sentence is evaluated by a human judge *after* the system has produced its own evaluation.

- Multiple Errors:

Errors are recorded as multiple when the sentence contains only one high-level error. Agreement and some lexico-grammatical errors do not trigger an exit from the program, so they can be detected together with high-level errors. Spelling mistakes have been corrected manually as the correction appears in a list form, following Levenshtein distance, which forces the program to exit – this not to reduce further an already small set. Eventually, low-level errors are counted as they appear, while high-level errors are only counted once per sentence. The motivation for this follows the procedure itself, which stops at the first error detected.

VI.3.2. Topography of the gold standard

To sum up and end this part on gold standard, the items of evaluation participating in the gold standard are:

- Contents
    - Good or bad
- Language

---

semblable for verb *semblant* which is semantically wrong here. An admissible sentence using a present participle would be: *ressemblant à*.... As a matter of fact, the system can correct *ressemblant que* by *ressemblant à*.

- Low-level syntactic errors:
  - Orthography, identification and correction
  - Agreement, identification and correction
  - PoS Selection, identification and correction
- High-level syntactic errors:
  - Detection
  - Correction

So, some aspects of the system's evaluation have been left out of matches within the gold standard. It is the case for synonymy detection and paraphrase recognition, as the gold standard evaluates language at the scale of *sentences* and contents only at the scale of *answers*. Therefore, a synonymy error is a content error, and a paraphrase error can only be a language error, indistinct from a language error made on a near-paste of the reference answer (that is, without paraphrase).

VI.4. Results

VI.4.1. Evaluation Set

For the evaluation, a set of 276 answers has been gathered among students of French as a Second Language of intermediary-advanced level, at University of Ottawa Second Language Institute. Six classes were visited, for a total of 180 students. The majority of students were females (only 36 males participated). This is representative of a general tendency in FSL studies (or SLL), as language courses usually attract more females than males. As well, most of these students were English native speakers.

This researcher went in class and presented each student with one text to read, accompanied by one or two questions to answer on paper. The origin of the texts is described in section III.1.2.1,

but the number of texts used here has been reduced from 96 to 9 to limit the number of questions, and maximize the number of answers per question. 14 questions have been kept to ensure diversity of material.

Students were asked to read the text and answer the questions in sessions of no more than 20 minutes. 10 students answered only one of the two questions presented to them. Three answers have been discarded from the set resulting in a final test set of 273 answers. One was too long and resembled more a short essay than a one-sentence answer, one stated the inability of the student to understand the text and the last one, by the same student, was ill-formulated and incomplete to the point of unintelligibility.

The texts numbered from 1 to 9 in table 5 below corresponds to:

1. *Comme un poisson dans l'eau acide*
2. *Coussin gonflable : Pourquoi il peut tuer*
3. *Le cannibalisme humain à l'épreuve des faits*
4. *L'homme et son corps*
5. *Les fauconniers de l'aéroport*
6. *Carence en fer et temps de gestation*
7. *Plus fertiles au sud*
8. *L'hypertension déterminée dès la vie fœtale*
9. *Epidémie d'hermaphrodisme chez les ours polaires*

VI.4.2. General Results

The general results are shown in table 5. The rows showing the figures are to be read as follows:

1. Answers: number of answers, correct answers and (/) wrong answers. The number in parenthesis expresses the total.

2. Reform. (reformulations): number of reformulated sentences among the answers, syntactic or semantic – simply the answers which are not exact matches. Expressed in %, as share of *good* answers.
3. Lang. E. (language errors): number of answers containing language errors, of the non-detectible kind. Expressed in %.
4. Syn. E. (syntactic errors): number of answers containing syntactic errors, detectable. Expressed in %.
5. Success: figure of recall for answers recognized as good answers. Expressed in %.
6. (and 10) Errors 1: share of good answers intercepted by error detection[18]. Expressed in %.
7. Errors 2: share of good answers evaluated as wrong answers for reason of the presence of language errors – the figures are not shown for wrong answers as these answers are judged wrong in all cases. Expressed in %.
8. Failure: figure of recall for answers recognized as wrong answers. Expressed in %.
9. Precision: figure of precision for answers recognized as wrong answers only – precision for good answers is 100% in any case. Expressed in %.
10. Errors 3: share of wrong answers intercepted by error detection. See Errors 1.

	Text 1 – Q 1	**Text 1 – Q 2**	**Text 2 – Q 1**	**Text 3 – Q 1**	**Text 4 – Q 1**	**Text 4 – Q 2**	**Text 5 – Q 1**	**Text 6 – Q 1**
Answers	14 / 4 (18)	10 / 5 (15)	10 / 7 (17)	11 / 2 (13)	16 / 3 (19)	20 / 2 (22)	12 / 4 (16)	11 / 9 (20)
Baseline	21.42	30	23.52	15.38	5.26	10	16.67	9.09
Reform.	78.58	70	76.47	84.61	94.73	90	83.32	90.91
Lang E.	5	0	0	7.69	0	0	6.25	5
Syn Err.	5	20	17.64	0	11.52	22.72	12.5	20
Success	42.85	50	60	63.63	12.5	45	41.67	36.36
Errors 1	7.14	30	20	0	12.5	25	16.67	9.09
Errors 2	7.14	0	0	9.09	0	0	8.33	9.09
Failure	100	80	85.71	100	100	100	100	66.67
Precision	36.36	66.66	75	33.33	20	25	44.44	50

[18] Syntactically erroneous answers are evaluated as good or bad following the gold standard, that is human intuition. Very often, the sentences are purely ungrammatical and difficult to correct (see Marking Scheme section), so these answers are left out of content evaluation.

Errors 3	0	20	14.29	0	0	0	0	33.32

Table 5a. Results for Assessment of answers as correct or incorrect.

	Text 6 – Q 2	**Text 7 – Q 1**	**Text 7 – Q 2**	**Text 8 – Q 1**	**Text 8 – Q 2**	**Text 9 – Q 1**	TOTAL
Answers	16 / 5 (21)	17 / 4 (21)	18 / 3 (21)	22 / 1 (23)	18 / 2 (20)	23 / 4 (27)	218 / 55 (273)
Baseline	37.5	29.41	27.78	18.18	11.11	8.69	19.72
Reform.	62.5	70.59	72.22	81.82	88.89	91.31	80.28
Lang. E.	4.76	9.52	0	8.69	25	33.33	8.42
Syn Err.	14.28	14.28	23.8	17.39	10	18.51	15.75
Success	68.75	47.05	50	63.63	27.77	21.73	**44.03**
Errors 1	18.75	17.64	22.22	13.63	11.11	17.39	16.05
Errors 2	6.25	11.76	0	9.09	27.77	39.13	10.55
Failure	100	100	66.67	0	100	75	85.45
precision	71.42	40	28.57	0	15.38	17.64	35.63
Errors 3	0	0	33.32	100	0	25	14.54

Table 5b. Results for Assessment of answers as correct or incorrect.

We did not give a value of precision when measuring the assessment of correct answers because all answers assessed as correct are correct (row *Success*). This is due to the system's strictness, and this value of precision is therefore 100%.

Now, a measure of the system's overall performance is difficult to establish precisely because some answers, good or bad, are blocked at error detection stage due to the presence of high-level errors (35 correct answers and 8 incorrect answers). Accordingly, we give two measures of accuracy: the first records blocked answers as missed and constitutes a conservative value, while the second, more liberal, does not take them into account, recording only answers that have passed the error detection barrier. Table 6 gives the performance values in absolute counts with respect to the assessment of both correct and incorrect answers.

	Total (273)	Blocked	True Positive	True Negative	False Positive	False Negative
Correct	218	35	*96*	*47*	*0*	*87*
Incorrect	55	8	*47*	*96*	*87*	*0*

Table 6. System's assessment performance in terms of answers absolute counts. Assessment performance counts are given in italics.

We can add the number of blocked answers to the performance counts to reflect the fact that these answers have actually not been assessed and that an ideal system should be able to assess content correctness even when language errors are present. In this case, the accuracy formula is as follows (TP, TN, FP, FN denote true/false negative/positive):

$$(TP + TN) / (TP + TN + FP + FN + \text{Blocked Answers})$$

Blocked answers constitute a sort of third label, always to be added to the divisor. This will give a conservative 0.524 accuracy. Conversely, we can discard blocked answers on the basis that these serve to calculate the error detection performance value (see section VI.4.3), and that the presence of language errors made ambiguous the correctness of an answer. This will give a more liberal 0.62 accuracy. This last value is retained as our final accuracy, but we believe that both are actually valid results, and that the ultimate performance measure lies somewhere in between.

This ambiguity on the value of blocked answers with respect to the assessment of content is also true for the rows Success and Failure in Table 5, which records blocked answers as missed. For convenience, Table 7 below sums up the performance values.

	Correctness	Incorrectness	All
Conservative	44%	85.45%	52.38%
Liberal	52.45%	100%	62.2%

Table 7. Content Assessment Performance values summed-up. Values are given in accuracy.

Comments on the number of incorrect answers

All answers do not yield the same degree of incorrectness with respect to the reference. Particularly, question 1 from text 2 and question 1 from text 6 show a high ratio of incorrect student answers. Now, this seems to be independent of the Textual Explicit and Textual implicit distinction in question categories. Even though Q1/Text2 is a Textual Implicit question, Q1/Text6 is not, and the other Textual Implicit questions show a level of failure rather on the average of all questions. This is not the place to go into too great details concerning the causes of the discrepancies in failure rate between questions – a task for didactics. But we think it is

nevertheless a point of interest in the study of tutoring question answering strategies in the field of education, and we give some hints here.

It seems that the syntactic complexity of a sentence affects the quality of the understanding students have of this sentence. The reference of Q1/Text2 is composed of two sentences linked by a relation of anaphoric co-reference. Besides, the succession of elements of causality in sentence 1 is tedious, but strict. In this case, there are several common causes of error in the student answers, originating from the syntactic complexity. The first is incompleteness where completeness is needed. Some questions can be answered with only a fragment of the reference, while Q1/Text2 needs the formulation of the full chain of causality contained in the reference to be intelligible as an answer to the question. Students tend to cut short and omit clauses of the needed answer (4 cases). The second is the student's inability to handle reformulation of the chained causality's form while preserving the sense. Here the error is more syntactic, but the mishandling of the syntax does not only yield error but skews the sense as well. More generally, the language error rate increases with the size of the complexity of the question, which seems normal.

Now, for both Q1/Text2 and Q1/Text6, the questions are quite reformulated with respect to what is in the text. For instance, in Q1/Text6's question:

(36) *sur quel aspect de la grossesse les réserves en fer ont-elles une incidence* ?
(On which aspect of pregancy does iron level impact ?)

aspect de la grossesse (aspect of pregnancy) is to be linked to *temps de gestation* (gestation time) – and the word *incidence* refers to a concept of logic and never appears in the text. This proves to be challenging for some students and it demonstrates the interest of this type of exercises.

Finally, there is an ambiguity in Q1/Text6 on what constitutes the answer – again due to a causality chaining effect going from *iron level* to *pregnancy* and to *women*. As the question only refers to the first two, and as students tend to answer by explanation of the link of causality between *iron level* and *women*, they fall into the trap of ambiguity.

Eventually, syntactic complexity, logical complexity in the relations between the clauses, and the degree of reformulation of the question compared with the text's reference, all seem to impact on

the level of failure to provide a correct answer. This is actually in relation to the degree of semantic reformulation in the answers (see section VI.5.1.2).

VI.4.3. High- Level Error Results

High-level errors are errors causing a disruption of the sentence syntactic structure. We gave a characterization of these errors in section IV.1.2. Note that here, the category covers most syntactic errors – 1/ errors where a wrong or extraneous PoS prevent correct dependency association of clauses and constituents, or 2/ errors where the wrong function word is used for a given function PoS (here prepositions and conjunctions).

VI.4.3.1. Results

These errors are errors which *should* produce a break in the analysis around the point of error, for means of detection – the parser having been modified accordingly. But XIP is a complex piece of software, and sometimes bypasses errors.

- Total number of such erroneous sentences: 47
- Number detected: 33
- Undetected: 14
- False errors: 3
- Precision: 90.9 %
- Recall: 70.2 %

Three errors were wrongly reported due to XIP's behavior: one shows a case of bizarre usage, another shows a square incapacity of XIP to analyze a conjunction of adverbs, and finally one is due to a polysemic word. The undetected errors show errors quite similar to detected errors, but in different contexts – XIP being very context sensitive.

Multiple high-level errors are another problem. Although it proved insignificant in our case (in only one case did the presence of multiple errors prevented the correct detection of the first error), it should be in a larger context. The only remark in this experiment is that a minimum distance should happen between two errors in order not to blur the whole sentence's structure. But our study is too small to come with to a final word on the problem – which is otherwise very parser dependent, as different parsers can follow different chunking strategies.

VI.4.3.2. Comparison with Antidote

To be able to judge the performance of the language error correction module, the results have been compared with the output of a commercial language checker – Antidote[19]. Antidote is a parser and rule-based checker which aims at correcting errors of grammar, spelling and style. The software has been available for more than ten years and should constitute a very tough baseline against the error correction module included here.

Surprisingly, Antidote does not really do better. True, the checker does correct some errors that our module does not, such as very specific error of infinitive as well as homophonic errors. But Antidote has a tendency to over-detect errors, and sometimes signals errors where there are none. Now, this does not reveal the quality of our approach as much as the extreme difficulty of language error correction, which accounts for the tendency of Antidote to over-correct: with greater sophistication rises the probability to fall into the traps of language ambiguity. This is particularly true in the context of SLL, and we believe that such tools as Antidote are inadequate to address the specificities of SL error correction.

Here, we left aside spelling and agreement errors as we consider this a non-problem, at least in this context. The kind of errors that are of particular interest are high-level errors, affecting the well-structuredness of a sentence, as they are very common in SLL, and the hardest to detect and correct. Results on the performance of our framework are given in Section VI.4.2., and show a 90.9% precision on a 70.2% recall in correctly identifying the location of the error in the sentence. Antidote works in the similar way and reverts to only giving a syntactic analysis break point as feedback when caught short of error correction capacity. Instead, an indication of

[19] Antidote, by druide, can be found at: http://www.druide.com/antidote.html

syntactic analysis break is given. Note that the feedback is not expressed in terms of error possibility, but in terms of syntactic analysis incompletness. Antidote has values of:

- Recall: 40.4
- Precision: 82.6

But Antidote corrects 84.2% of the identified errors, so that errors are not presented in terms of a point in the analysis, but as a corrected sentence. In other cases, the checker either declares itself unable to analyze a sentence, or provides a wrong analysis. In the eight cases where Antidote provides feedback as a break point, it is correct only three times.

Table 6 shows a detailed view of the performance. *Wrong Analysis* means that the syntactic analysis is wrong, whether splitting the tree at the wrong place, or giving a complete but flawed analysis. *No Analysis* means that the parser does not output any analysis.

	Precision	Recall	Detected /47	Corrected	Wrong Analysis	No Analysis	False Positive
Antidote	82.6	40.4	19	16	9	19	5
Custom	90.9	70.2	33	1	14	0	3

Table 8. Comparison with Antidote for high level errors. *Custom* refers to the error correction module implemented for this study.

Ultimately, the judgement on the comparison should be mixed. Our implementation does better than Antidote at detecting locations of errors, and Antidote does better at correcting the errors. The latter was expected as Antidote is a professional product. We believe however such tools as Antidote not to be adequate to address the problem of second-language error correction directly, as they have not been designed for that purpose (see Kraif and Ponton, 2007), although students could use them in other writing contexts, as essay writing for example. The major impediment to a serious use in SLL has however more to do with the state of error detection and correction processing than with design questions.

VI.4.4. Agreement and orthographic errors

PoS selection errors have been discarded of this part of the evaluation, as indeed they sometime produce breaks in the syntactic analyses, naturally, and therefore fall in the category evaluated in the previous section. Orthographic error evaluation results are expressed in terms of *detected* and recall, as well as *undetected* when a wrongly spelled word is still present in the word bank (usually for reason of being a word coming from another language and used in French in very specific contexts). Agreement errors are expressed in terms of recall only as detection and solution comes in pair. PoS selection errors constitute a category intermediate between low and high-level syntactic errors. Typically, some of these errors do produce breaks in the analysis, and have been taken into account in the preceding section. Here are only errors which do not produce breaks.

VI.4.4.1. Orthographic

- Total: 18
- Undetected: 5.55 % (1)
- Recall (the output shows the correct form): 80 % (15)

Some orthographic errors are not solved either because the form appears as a neologism (out of context in the answers), or as word so misspelled that its correct form cannot be traced by Levenshtein distance.

VI.4.4.2. Agreement

- Total: 27
- Recall: 85.18 %
- Undetected due to parsing error: 7.4 % (2)
- Undetected: 7.4 % (2)

Undetected agreement errors are due to wrong attachments in the XIP output.

VI.4.4.3. PoS Selection

- Recall: 62.5 % (5)
- Undetected: 37.5 % (3)

Usually, PoS selection errors have to do with basic forms such as DET + NOUN structures, here mismanaged. When they are not detected, it is because of ambiguity in the selector. For instance, "de" can be both a determiner and a preposition.

VI.4.5. Synonymy Results

The evaluation is done based on the number of actual synonymic links in the texts. The measures of evaluation are precision and recall. Precision measures the number of correct links in the hits, and recall measures the share of links which have been recognized. Another interesting figure is that of erroneous links: words judged as synonyms in spite of any contextual link in the sentences. Those links cannot be put against the figure of actual links, as they are unintended. For this reason we give two figures of precision in the detected synonyms, one for the precision of the detections with respect to the existing links (i.e., detections based on a word which is involved in a relation of synonymy in the sentence), and one general figure of precision.

Note that direct synonymy relations and cross-PoS relations (cross-synonymy) are merged here. Please refer to the section on Synonymy for the details of the detection procedure (Section V.2.2).

- Total number of links: 32
- Detected: 16
    - Recall: 40.6%
    - Precision:

  - Wrt actual links: 100%
  - General: 81.25%
- Undetected because of a parsing/Language error: 15.6% (5)
- Undetected because of a match in Q: 12.5% (4)
- Undetected because of absence in Memodata[20]: 21.9% (7)

We also give figures of recall and precision of the synonymy programming function per se as most undetected relations are not detected because of 1/ either parsing breaks, orthographic or language errors in the answers, and 2/ the presence of the S' synonym candidate in the question Q.

- Recall: 75%
- Precision: 91.7%

Eventually, there are only four links which has not been detected for reasons other than those mentioned. One is due to the detection method which triggers cross-synonymy detection based on a word's governing constituency – governors only are searched for cross-synonymy. Therefore a word which is not governing in a relation of dependency is discarded as a cross-synonymy candidate. This is a limitation, but the contrary would increase the number of improper relations, as the cross-synonymy detection procedure increases the searches for matches exponentially. The three other failures are due to matters of ambiguity. In one case, a synonym in S has been put in relation to the bad one of two possible synonyms in Ref. In the two last cases, the synonymic relations are ambiguous in themselves, and the words happened not to be in Memodata (link between "*to have*" and the two words "*to present*" and "*to develop*", in their medical sense).

VI.4.6. Paraphrase

[20] But Memodata remains a quite good synonymy bank – it is only that one of the relations appeared four times, and it happened not to be in Memodata.

There are two distinct procedures to match paraphrases in the system. The first one, and most comprehensive, puts in relation sets of answer dependencies with rules of paraphrase equivalence, using specific bases of such rules. The second one puts in correspondence dependencies paths in a procedure specific to the mapping of the thematic structure of the sentence. This is a means to deal with paraphrases existing only at the level of phrases in the sentences: "sentence paraphrase" vs. "phrase paraphrase" so to speak.

Here, as for synonyms, the results are merged as all phrases showing some kind of paraphrase pattern are defined as paraphrases. They are expressed in terms of recall only, as precision is 100%.

- Total number: 34
- Detected (Recall): 41.2% (14)

As poor as the recall figure appears, the failures are not due to the paraphrase detection procedure itself. Logically, all sentences showing paraphrase patterns and reaching the processing point of paraphrase recognition in good syntactic or lexical shape are recognized as paraphrases. All failures are due to 1/ errors in synonymy detection, 2/ bad or broken syntactic analyses, 3/ orthographic errors, or 4/ clumsy formulations. Concerning the last point, a lot of sentences have not been recorded as paraphrases for reason of clumsiness in the form – so that they are difficult to match against any pattern.

As good or bad as the results may appear, the interest is more to know where and why failures to assess correct answers do happen. This is the subject of the next section.

VI.5. Comments on the Results

As shown in the preceding section, there can be several reasons to false judgments of failure in the process of assessing correct answers. These are mainly due to 1/ the use of semantic reformulation, 2/ the presence of language errors or clumsiness of usage in the answers, and 3/ the use of textual parts picked outside the textual part making the reference – a phenomenon we

refer to as *inter-textuality*. In this chapter, we review these reasons and indicate possible enhancements where possible, although these will be summed up in the Future Work section.

VI.5.1. Semantic Reformulation

The impact of semantic reformulation in answering is a severe qualitative limitation to the whole process of automatic assessment. In a way, it could be said that the incapacity to handle the assessment of such answers puts a question on the interest of any project of this kind – but we believe that this is a short answer. Little in Computer Science would exist today if the research community had expected immediate capacity to handle all aspects of computing problems – and even less in Natural Language Processing (think for instance of automatic translation).

Semantic reformulation is not a simple notion – It too can be complete or partial, and it can depart more or less abruptly from the reference's content. Here, we retain as a definition of semantic reformulation a definition by default: any sentence that cannot be put in correspondence to the reference by means of lexical synonymy/antonymy as well as by means of syntactic reformulation equivalence – independently of this implementation.

Here, we will measure the exact impact of semantic reformulation on the performance, and try to figure how this obstacle can be minimized.

XV.5.1.1. Semantic reformulation in the results

Table 7 shows the extent to which semantic reformulation affects the answers, both in general and at the level of each question. The figures indicate the share of partial and complete semantically reformulated answers within the set of all reformulated good answers for each question – *reformulated* includes exact match with extraneous material. Textually Implicit questions are in italics. The *PC ratio* row show figures for the ratio of partial reformulations with respect to all reformulations.

	Text 1	Text 1	*Text 2*	Text 3	*Text 4*	Text 4	Text 5	Text 6

	– Q1	– Q2	– *Q1*	– Q1	– *Q1*	– Q2	– Q1	– Q1
Nbr	13	10	*10*	11	*16*	19	11	10
All	38.5	30	0	9.1	*81.25*	47.35	27.3	50
PC Ratio	60	100	0	100	*53.8*	55.5	100	60
Complete	15.4	0	0	0	*37.5*	21.05	0	20
Partial	23.1	30	0	9.1	*43.75*	26.3	27.3	30

Table 9a. presence of semantic reformulation in the answers set.

	Text 6 – Q2	Text 7 – Q1	*Text 7 – Q2*	Text 8 – Q1	*Text 8 – Q2*	Text 9 – Q1	Total
Nbr	13	17	*18*	20	*18*	23	209
All	15.4	29.5	*27.8*	5	*16.7*	26.1	29.1
PC Ratio	100	60	*39.9*	100	*66.5*	100	67.35
Complete	0	11.8	*16.7*	0	*5.55*	0	9.5
Partial	15.4	17.7	*11.1*	5	*11.1*	26.1	19.6

Table 9b. presence of semantic reformulation in the answers set (cont'd).

Several remarks can be made on these results, and these remarks can drive to conclusions even if careful as the quantity of data is scarce.

- Complete vs. Partial reformulation

The figure of partial semantic reformulation is double that of complete semantic reformulation. This means that in more than 9 cases on 10, an automatic assessment system can still find content material to work on, as partial semantic reformulations respects, at least partly, Ref's thematic structure. Complete semantic reformulations simply cannot be processed. Note that there is a correlation between the degree of semantic reformulation, or reformulation in general (as expressed in table 7), and the value of success recall – see preceding section table 6.

There can be two kinds of partial semantic reformulations – following the definition of extraneous material (see section III.2.1.2.1): one where a unique thematic structure (that of Ref) is modified, and one where Ref's thematic structure is doubled by another thematic structure (See examples 34 c and b). Dealing with the second kind is easier than with the first, as in this case the proposition making Ref's theme is usually respected – only expanded with an adjunct proposition. Dealing with the first kind of reformulations is more risky as the adjunct modifies the proposition making Ref's theme.

- Discrepancies between questions

An interesting remark is that not all questions motivate the same degree of semantic reformulation, or so it seems. For instance, text 8's question 1 has a 5% semantic reformulation ratio, all partial, while text 4's question 1 has a 81.25% ratio, well spread between partial and complete. This difference naturally produces a wide gap between performances: respectively 60% vs. 12.5%. This also seems to be independent of the Text-Explicit/Implicit paradigm: text 4's question 1 is TI, but so is text 8's question 2 with only a 16.75% ratio.

A simple variance calculation, adjusted by the quantitative importance of each question, gives a 21.72 standard deviation value – a figure which does not require further tests of statistical significance.

This should be tested more thoroughly, first to see if the fact is hard, and then to correlate it to question types – but that's not in the domain of computer science. What's in the domain of computer science is that a selection of questions could minimize the number of semantic reformulations, therefore the feasibility of computerization, should a discrepancy be confirmed between questions types in the motivation of reformulation.

XV.5.1.2. Ratio of reformulations

An interesting observation is that, just as not all questions yield the same failure rate, not all questions motivate the same degree of reformulation in the answer. Q1/Text8 has a level of semantic reformulation higher than any other question, and by a wide margin. An explanation for that is in the complexity of the reference, composed of three sentences showing content of high specificity. Students take short cuts from one point in the chain of causality to another, and this departs from the system's dependency analysis of the sentence. Besides, the complexity of the sentence contents drives students to use simpler reformulation. In other words, students tend to simplify the sentence at the two levels of language and of contents, and ultimately the reformulation is too deep for the system to handle.

Now, there is a difference in the nature of complexity, compared to the complexity that motivated higher failure rate (see section VI.4.2). Here, the chained causality is more conjunctive than transitive, and the relations of cause-effect stand as different ways to explain a phenomenon at different levels. A transitive causality is a logic relation where the elements are linked together so that each element is the cause of the next and the effect of the preceding (Hermet and Barrière, 2002). Transitive causal relations show a different aspect of complexity and seem to drive the student towards error.

Although Q1/Text8 is a Textual Implicit question, this does not seem to motivate reformulation particularly. As well, there is no real correlation between a question's rate of reformulation and the failure rate. For instance, Q1/Text2 has as null reformulation ratio in spite of it being textually implicit and of showing the second highest failure rate. On the other hand, Q6/Text1 is the second most reformulated and a Textual Explicit as well as the question yielding the highest failure rate. Q6/Text1 shows another motivation for reformulation. The question being itself reformulated compared with what is in the text, both in content and in form, students tend to follow this as an example and provide reformulated answers. Moreover, the question shows a degree of abstraction (*incidence*) to characterize the relation from the cause to the effect (from *iron level* to *gestation time*) – this could also explain the higher reformulation.

The question of the motivation for varying reformulation rates is outside the scope of this thesis, and is the work of didactics, especially as our evaluation set is quite small. But it seems that the two factors of 1/ complexity of the reference in form and content, and 2/ abstraction of the question do come into the distinction of questions as *open* questions, yielding more personalized answers through higher reformulation.

VI.5.2. Language incorrectness

The second reason for failure is that of language incorrectness, which can be of two kinds. Some sentences have broken syntactic structures, as produced by XIP. Others show clumsy usage of the language. The first kind is easy to detect and the processing of such cases is not error prone:

the system does its job when halting the process before any judgment of content assessment. This is not an instance of failure. The second poses another challenge to CALL activities in general. We present these issues in turn.

VI.5.2.1. Bad Syntax

Within the set of student's answers, the share of syntactically broken sentences is 15.75%. The fact that these sentences cannot be judged as correct is not a limitation of the system – here, the job is done, at least up to the system's capacities. Also, as important as it is in such systems, the question of feedback does not come into account to evaluate success or failure – it is another aspect of the problem.

Therefore, answers which have been intercepted due to broken syntax should be evaluated only with respect to the real presence of syntactic errors – the figure of recall is 70.2% (section VI.4.2).

Now, all these errors are not equal. Some are real syntactic inconsistencies, while others are errors of usage. For instance, the parser has troubles to handle usage of oral forms of language in sentences which are otherwise syntactically correct. We depict such sentences as *clumsy utterances* or *language*. Here is an example:

(37) *Dans ces deux périodes l'interprétation est trop difficile à faire parce que le volume plasmatique augmente beaucoup et ça * dilue les paramètres biochimiques.*

(During these two periods the interpretation is to difficult to make because the plasmatic volume increases a lot and it * dilutes biochemical parameters)

Example 37 does not show any syntactic errors, but the language is clumsy, and the parser breaks the analyses at the * point. At least, the break occurs at a pertinent location, and inviting the student to examine his/her sentence advocates a *lesser-is-better* approach to error correction (Ponton and Kraif, 2007).

The varying quality of XIP's outputs only adds to this kind of problem. For instance, the fact that XIP does not make the same attachments between optional complements, depending on the location of the complement to the verb, before or after. This does not rely on a sentence quality

as correct or incorrect, but is a mere limitation of the parsing technology, independently of XIP's specifics.

VI.5.2.2. Clumsy Utterances

Clumsy utterances are not really incorrect syntactically but show improper lexico-syntactic usage. For instance, preposition errors are a case of clumsy language[21]. Another cause of such errors comes from the length of sentences: sentences showing aggregated clauses without proper punctuation usually fail at parsing. The adverb *usually* says it all: the behavior of the parser is quite unpredictable concerning these errors, and as we don't have XIP's full version, we could not activate an option which permits the output of the exact rule on which the parser has failed: i.e., the step in the analysis at which XIP has failed. Another possible problem of clumsy language is that it forces the parser to flaw analyses – the analyses come without breaks but the dependencies are erroneous. In all cases, assessment of the answers is impossible.

Such sentences represent 18.68% of all sentences (51 cases). Within this subset, come:

- 25.5% of correctly assessed answers
- 52.95% of incorrectly assessed answers
- 21.55% of broken analyses

In table 5, the incorrectly assessed answers are either counted by the *Language Errors* row or amounts partly to the remainder of the failures *precision* row. This in turn annihilates paraphrase recognition or simple dependency equivalence recognition capacities. Below is a good example of such an error (simplified example):

(38) Question : *A quoi sert un chaperon ?*

(What's the use of a hood ?)

S : *quelqu'un qui est mis pour éviter les distractions pendant le transport d'un faucon.*

[21] Preposition errors are actually a complex case. They appear as clumsy language when the semantics of the erroneous preposition is close to the correct preposition (think for instance of *to* and *at*), especially when the governor of the preposition subcategorized both prepositions (think of verb *to go*). In which case, the syntax of the sentence is not broken. In more severe cases, preposition errors can turn into a high-level error. See Hermet and al, 2008.

(someone who is put on to prevent distractions during transportation of a hawk)

Such an answer would be recognized as a good answer, due to the pronoun nature of the compound *quelqu'un*.

Providing solutions for such cases is doable – but it amounts to an area of research in itself within Second Language Acquisition: a typology of language errors as committed by learners 1/ in general, or 2/ according to their cultural background. This is out of the field of CS, but conditions the feasibility of CALL projects.

VI.5.3. Inter-Textuality

By this expression, *inter-textuality*, adopted form Literature studies, we refer to the act of using parts of the text outside the referential answer, to make a reformulation of this reference. In appearance, the reformulation is semantic – but actually, the parts used can be found back in the text. The parts used outside of Ref can be parts of content, as single words, or structural, where whole phrases are used to add to or reformulate Ref. There are 37 such cases, which amounts to 13.55% of all sentences, and more importantly to 24.34% of all rejected sentences (including sentences halted by syntactic breaks).

Therefore, the question is to know whether these can be linked to the reference by lexical co-occurrence, or by any means. Here is an example to illustrate this:

(39) Ref : *le tabagisme, par exemple, gêne le passage du sang à travers le placenta* [...]

(Tobacco use, for instance, blocks the flooding of blood through the placenta)

S: *la nicotine qui se trouve dans le tabac est absorbée par la mère et atteigne le foetus par la circulation sanguine, perturbant le métabolisme.*

(the nicotine contained in tobacco is absorbed by the mother and reaches the fetus through blood circulation, perturbing the metabolism)

The student uses parts of the text which reformulates the contents of Ref (*tabagisme* is textually equal to *la nicotine qui se trouve dans le tabac est absorbée par la mère*).

Therefore, the information is present to solve the problem, which is one of text mapping (see for instance Dennis and Kintsch, 2003) – and the question to know whether the means exist to map text segments to each other is again a research topic in itself.

A problem linked to the one of inter-textuality is that of multiple possible answers. In its state, the system uses only one reference, constituting of one or several co-referenced sentences. Adding the possibility to map answers to different possible references would increase performance considerably. But this is in turn question dependent: some questions call for the use of multiple references while others don't.

VI.5.4. Feedback

The question of feedback is not treated in this study. Our goal is to provide the content of possible feedback, and not really the feedback structures – this is a matter of user interface and of pedagogy. Now, the quality of feedback conditions the success of an ICALL application to a large extent, so this section proposes a review of the system's capacities with respect to the creation of feedback.

The first most important task in a tutoring context like this one is the report of language errors. In its present state, the system can provide material to feedback to the students in order to inform them of their language errors. This information can be specific as in cases of agreement or spell-checking, or it can be only general, as in the case of high-level errors – but in the latter case, this is an insufficiency of the science more than of this one system. Particularly, the errors are not corrected, they are detected. In this context, any feedback would come in the form of an error message. The providing of means for the students to correct their errors depends on the support installed in the tutoring environment and is outside the scope of this study.

The information on the adequacy of the answer in terms of content is binary, and the answer is judged as correct or incorrect with respect to the question. The only refinement comes in the form of extraneous material reported for the student's answer. This has not resulted in any false positive, but it could theoretically – for instance due to addition of content tempering or

contradicting the correct part. However, the definition of a category of errors with respect to content assessment remains to be done.

Due to the architecture of the system, it is possible to highlight which part of the sentence the system cannot analyze, although that information comes in the form of relation tuples. In that sense, the system can output a partial assessment report, but it is not possible to qualify a partial assessment, and this is beyond our reach.

Ultimately, the system keeps track of the analysis so as to be able to build feedback on lower-level errors nature, higher-level errors position in the sentence and success (adequate answer). It can also send feedback on the unrecognized content. Eventually, in such a context, and due to the system's limitations (sensibility to semantic reformulations and absence of content error typology), reports of incorrectness should not be expressed as such, but rather as an inability of the system to process the answer.

In conclusion to this section, the performance of the system is affected by language issues which are more or less controllable. The problems range from semantic reformulation which can't find realistic solutions, to clumsy language problems which could be addressed by some solutions (starting by more robust parsing) as well as inter-textuality, which is often solvable by simple co-occurrence solutions.

However, the single most important issue regarding the feasibility of such projects as this one is to provide a characterization of language errors, both in general and with respect to cultural idiosyncrasies.

Chapter VII. Conclusion

The project presented in this thesis represents a computerized application of Language Learning. As seen from the perspective of Language Acquisition, or Education in general, it encompasses an approach to Learning Technology.

More specifically this project implements a Computer-Assisted/Aided-Assessment program to be applied to the restricted Computer-Assisted-Language-Learning problem of a student's free text input processing in the didactic paradigm of text comprehension for French as a Second Language. The learning activity on which the project takes place follows that of DidaLect, a tutoring software in the family of Intelligent Tutoring Systems.

Following theories of didactics, this learning activity consists in having students of FSL, of an intermediary and advanced level, to answer questions based on texts. In the DidaLect project, questions were multiple-choice, and the goal of our project has been to adapt the methodology to open answers. This effort casts the project in the area of ICALL research by investigating a way to increase the possibilities of answer variation in a tutoring context.

This goal has called for the exploration of two paths. The first stretches along the computational means to address the problem of language content comprehension and form assessment. And the second is parallel to the first: how to restrict as much as possible the scope of reformulation of the answer material necessary to answer the question, in order to cope with the limitations of the computational means of language analysis.

NLP and Didactics

Natural Language Processing provides the computational means of dealing with shallow language comprehension. Statistical NLP offers solutions to deal with approximate similarity, but although we have carefully studied these solutions, we have selected a symbolic approach, as we felt a symbolic approach only could provide means for an in-depth analysis of the written material. Therefore, the means of syntactic parsing, synonymy detection and paraphrase recognition have been employed to solve the case of language content and form assessment.

In addition to these techniques, we took interest in a less studied domain of NLP: error diagnosis. These means of NLP provide "intelligence" to the CAA/CALL component, and Computational Linguistics (or *Human Language Technologies*) is often seen in the community of CALL researchers as a branch of Artificial Intelligence.

Our approach represents a conservative strategy to assess similarity, as we do not make use of statistical methods. The strategy behind the project is syntactic and symbolic. Although this may sound limited in theory, it is justified in our view by three major reasons. The first is to achieve independence from the cost of gathering training material. The second has to do with the bias of statistical method to generalize correctness based on surface lexicon, independently of language errors, of misinterpretation and of clumsiness. Eventually, the goal of the exercise would be to help students in second language acquisition, and so the system should use the same description of language as instructors – here syntax is part of the model of language.

On the side of didactics, and more precisely text comprehension, we have found topics to address the problem of limiting reformulation possibilities. We employed a typology of question types to restrict the correspondence between questions and the text. This typology posits three types of relations to textual contents, more or less direct. By choosing only question types which restricts this relation to factual, or direct, correspondence, we have been able to provide firm theoretical grounds likely to be addressed by computational solutions. This is one aspect of the novelty of this project.

Of the NLP techniques used for this project, some are old and tested, and others are new, at least as applied in a CAA/CALL context. Syntactic parsing and synonymy detection, not to mention spell-checking, are applications which have been widely used for a variety of tasks – including for CAA/CALL. Conversely, paraphrase recognition is a field of computational linguistics which

has received attention only recently – it is not clear that it *is* a field of CL. Whatever the answer, we believe it should be.

Results

Although the system has not been conceived for direct deployment, it has been tested with material gathered in-class, among intermediate-advanced students. In this way the evaluation has partly reproduced class conditions. The overall results of the system justify moderate enthusiasm only (44% of good answers correctly assessed – 62.2% of answers correctly assessed with respect to content, good or bad). The level of the results is due, at least partly, to the importance of semantic reformulations, but we showed that the results are also very dependent on the question asked, with a 21.7 standard deviation value for the per-question recall values. As modest as it looks, this result of a 44% recall for the assessment of correct answers is to be compared to a 19.6% baseline, and so represents a serious improvement.

The results concerning language error detection are acceptable, ranging between 62.5% and 85.2% for the various types of errors – especially as undetected errors are undetected due to ambiguity or priorities between error alarms. But we stressed how error detection should be evaluated with respect to error diagnosis, and that this poses problems in what concerns syntactic errors.

Example 40 shows two correctly assessed answers – although S2 contains an undetected preposition error. This illustrates the capacities and limits of the system.

(40) Q: "*pourquoi peut-on tirer les mêmes conclusions pour l'homme et pour le rat* ?"

(Why can the same conclusions can be drawn for men and for rat ?)

Ref : "*le rat est un animal qui possède un système cardio-vasculaire très semblable à celui de l'homme.*"

(The rat is an animal which have a cardio-vascular system very similar to that of human beings)

S1: " *le rat possède un système cardio-vasculaire très semblable à celui de l'homme, donc on peut tirer les mêmes conclusions.*"

S2: " *le système cardio-vasculaire d'un rat s'approche à une personne humain.*"

Both S1 and S2 are syntactic reformulations well assessed. In S1, the paraphrase does not involve cross P-o-S synonymy of a Ref's word. S2 does (from *semblable* to *ressembler* to *s'approcher*) – unfortunately, the correct preposition is lost in the process as *s'approcher* subcategorizes both prepositions *de* and *à*.

VII.1. Problems and Limitations

The main obstacle to a satisfactory processing of all possible answers to a question is the limited capacity in NLP to deal with semantic reformulation. There are two ways of coping with this problem.

The first is direct and should aim at enhancing recognition of content correspondence by the use of semantic paraphrase recognition. The difference between syntactic and semantic paraphrase is mainly that the first can be read at the level of syntactic categories, with the vocabulary changing only following patterns of synonymy, antonymy or function word correspondence. Semantic paraphrase further implies the recognition of content lexicon addition and/or suppression between two or more corresponding sentences – the changing lexicon being controlled by means of lexical functions. Now, semantic reformulation does not always imply a dramatic change in the lexicon of two comparable sentences. Two sentences such as *X causes Y* and *X can cause Y* show an instance of semantic paraphrase, and we see that the lexical reformulation is minimal, and easy to model. Therefore, processing of paraphrase recognition can be initiated by a divide-and-conquer solution, using already existing successful means of detecting semantic similarity.

The second way is indirect and relative to the activity paradigm. A semi-legendary story is told among candidates of the baccalauréat in the secondary school system in France. Some forgotten year, the subject of the philosophy dissertation was "What is courage?" A student handed back a blank sheet, otherwise saying: "This". Is this an admissible answer? It could be, but it nevertheless violated the principle of the philosophy dissertation composition: the paradigm of

the activity (thesis, antithesis, synthesis, etc.). Quite similarly, a FSL student participating in one of our data-gathering sessions answered the question:

pourquoi avoir retenu une période d'un an pour étalon épidémiologique de l'étude?

(Why has a one-year period been retained as a epidemiological standard for the study?)

(Ref: *cette période de un an correspond à ce qui est observé dans la pratique médicale de consultations des couples ayant des difficultés à avoir un enfant.*)

(This one-year period corresponds to what has been observed in the medical practice of consultations for couples having trouble to have a child.)

By:

ils ont retenus une période d'un an pour étalon épidémiologique de l'étude pour standardiser le variable temps.

(They have retained a one-year period … in order to standardize the time variable.)

This is an admissible answer (retained as a miss in the success recall value), but it violates slightly the implicit rule of the game by adopting too great a level of abstraction; and we lose information compared to what is present in the reference answer. Particularly, the reformulation here does not conform to the definition of paraphrase, as expressed in section IV.3. This is to illustrate that providing a process-friendly answer is part of the activity in such a context – which still authorizes the use of semantic reformulation.

In-depth analysis of content is of limited reach in what concerns semantics. It should remain so for a while before technology brings better solutions. Particularly, statistical similarity methods are too tolerant in a context such as CALL, which requires a precise analysis of similarity. Therefore, the interest of exercises like those presented in this study are never stronger than as a linguistic reformulation practice. That is, the activity should be constrained by some degree of guidance, possibly restricting it. Exercises that motivate semantic reformulation should be less ambitious for now. A middle ground would be to constrain free-text production exercises to the application of linguistic patterns, and to extend already existing exercises as fill-in-the-blank to more complex patterns, so as to involve some degree of free-text.

Another weakness of the program, and perhaps in CAA/CALL in general, is the limitations of error-diagnosis performance. Now, this is an area of research which should be explored extensively. We showed how error detection is quite easy to do, although it is very parser-dependent. Identification, which amounts to diagnosis, is a more difficult process. Error diagnosis should involve a good deal of collaboration between computer scientists and teachers as the process calls for a definition of possible errors, and this does not exist to a full extent yet. Once such a resource is to be produced, defining and providing the means to successfully exploit this resource is only a matter of time – which can eventually find only one unknown, familiar to NLP researchers: ambiguity.

Ambiguity comes as a disruptor of correct processing in almost all of its aspects. In some aspects, ambiguity is induced by a design bias, for instance in the parser's favoring of some types of attachments. The parser has a tendency to attach complex prepositional phrases to a verb rather than to a noun (by complex, we mean introduced by a complex preposition, usually a compound). In others, the ambiguity is a consequence of time and location. For instance, this happens in the synonymy detection process where some wrong synonymy relations could be preferred simply because the mistaken token appears before the correct one in the list of possibilities. Therefore, ambiguity can be addressed with more or less success depending on the modality of processing which prompts it. Finally, ambiguity comes from the language specifics – for instance, some students systematically rewrite a passive verb pattern with an active form, otherwise leaving unchanged the order of constituents : this results in two syntactically correct sentences which are opposite in sense, and the assessment fails. The same is true for verbal reflexivity.

We have tried to prevent possible ambiguities to arise where possible, that is, where a given process has been custom-defined. This is not the case of the parser; therefore, parsing ambiguities have been corrected only modestly, through XIP's customization files, following observations made on the small test set. We have tried to bias the dependency relations to systematically favor attachment to verbs rather than nouns (to increase an internal tendency of the parser), so as to be able to put the compared sentences in correspondence – here, regularity comes before exactitude.

At the level of spell-checking, ambiguities happen due to phonemic confusions. For instance, the misspelled word *resemble* (*ressemble*) cannot be corrected due to the soundex algorithm, which converts the word into RZMB instead of RSMB. At the level of parsing post-processing (to attach segments separated by commas when no attachment is provided), we have devised a method by which a syntactic attachment is created for all possible candidates, so as to be sure not to miss the right one. But this is limited to cases of present participle governing relative subordinate-like phrases (for instance, in the example above, *des personnes* ***ayant des difficultés***) – this has been sufficient for our needs[22]. At the level of synonymy detection, we have chosen to use a limiting factor for examined words based on the governing quality of the words: only governing words are sent to the cross-synonymy processing step, in order to limit the confusion possibilities. But this should create as much confusion as it solves.

So, as in many NLP tasks or applications, the potential for ambiguity is everywhere, but some procedures could help to solve the issue, whether by implementing more complex algorithms (spell-checking), or by providing lists of candidates when the importance is to enable comparison more than to provide a fact of exactitude (what has been done for syntactic post-processing could be applied to synonymy). In what concerns language form errors, like passives and reflexivity, comparing the student input with the reference could help highlighting cases of errors to some extent.

Now, a severe case of ambiguity comes with error diagnosis. As we did not go any further in the operational correction of syntactic errors, we have remained below the ambiguity zone – again, this is a research topic in itself. But were one to deal with the issue of error diagnosis, based on parsing breaks (which is a common strategy, as we saw), ambiguity should come as a major issue – just as it can be for humans.

VII.2. Contribution

This work has made a number of contributions, especially to CALL, but not only. The first and perhaps main contribution is as we said the system itself, which is entirely new. It is of greater

[22] The cases for post-processing have been established after the set of questions and reference sentences.

interest to CALL. Only CAM (Bailey, 2008) and C-Rater (Leacock and Chodorow, 2003) propose a similar tool, although dedicated to English. CAM, in spite of spectacular results, does not deal with grammatical errors, and performs evaluation of reformulation only through overlap (lexical and grammatical) which suits a test set showing a balance favorable to correctness. C-Rater is a solid software but requires reference answers to be encoded with a special tool, which permits to create a set of reference answers for each questions. A direct competition of this system, FreeText (L'Haire and Vandeventer-Faltin, 2003a), has not reached a state of completion although it involved a team of computer scientists, teachers and linguists. In this respect, the present system constitutes a baseline, where the necessary processing steps are recognized and implemented using symbolic methods. Let statistical NLP lend a hand where possible in future research.

Having gone to the point of completeness, the project allowed us to come with some results that could help to understand the process of question answering. As we mentioned, all questions do not yield the same rate of reformulation, and it should be of interest to examine why. This accounts for didactics.

In what concerns NLP, we have studied the possibilities of automating paraphrase recognition after the theories of formal linguistics on paraphrase to avoid the specificity of statistical models of paraphrase. We have proposed a model for paraphrase based on syntax, and built after the Meaning-Text-Theory. Coupled with the cross-PoS synonymy detection procedure, this paraphrase recognition module represents a quite complete model of paraphrase, and is unique to our knowledge. Although this aspect of processing is limited in its coverage (with respect to the scope of language production possibilities), it nevertheless suffices to fulfill its role in the chain of processes. An interesting fact is that although the recognition procedure relies on surface-syntactic categories of limited number and generic type, the rate of false positive paraphrase recognitions is null, while recall is near-perfect, in spite of the poor operational recall value (41.2% - see section VI.4.6 for an explanation). While this should certainly be tested in a specific and exhaustive framework, it nevertheless represents a novelty. As it does not depend on training, and does not therefore reflect the specificities of the training material, it covers the whole of a language – here French. Although it is slightly lexicalized (see section V.4.2, or Appendix D), the lexemes are function words only and the rules could easily be adapted to other languages, in theory.

VII.3. Findings and Future Work

This project has permitted some findings and lines of inquiry for future work. In the domain of NLP, error diagnosis and paraphrase recognition are certainly fields worth of being explored. As we said, there is a gap in what exists concerning paraphrase in NLP between formal linguistics and computational linguistics. We believe that promising results could be achieved refining on a model such as the MTT, and especially on the question of semantic paraphrase by way of incorporating resources of statistical lexical-semantic similarity. Furthermore, it seems that this notion of semantic paraphrase should be further decomposed, both in its relation to semantics itself (take for instance the expression of condition), as well as in its relation to syntax: for instance, what are the syntactic variations at different levels of semantic reformulation?

The question of semantic reformulation/paraphrase is a vast one: it should motivate a field of research in itself (which is the case in the community of MTT). To study semantics at the level of the sentence, through semantic paraphrase, could help to cast the problem within a more limited communication perspective – more limited than the *world knowledge*. This program would require collaboration between research in statististical and non-statistical NLP – the first to annotate vast amounts of textual data and build a language translation model, and the second to sort out the relative importance of function and content words in models of equivalence, and thus to distinguish what is generalizable in the model (syntax) and what is less or not at all (semantics).

Concerning Error Diagnosis, we can only repeat what has already been said: this is a case for collaboration with members of the Education community, as the set-up of any successful application requires a fair amount of training data. We should add that any application should provide ways to deal with errors falling out of any typology, so that processing of language errors should consist in providing means for correcting what is known, as well as means to cope with errors of unknown types. In other words, to transform into correctness, or else explain incorrectness, which can only be done with respect to what is correct in the evaluated piece of text.

Now, in the domain of CALL, this study has made possible an observation that all questions did not motivate the same degree of reformulation. This is quite interesting, as processing of reformulation is a major barrier to raising the quality of assessment. Studying the reasons for this discrepancy in the level of reformulation is not really in the domain of NLP, but it certainly conditions the extent to which applying NLP solutions to CAA/CALL is feasible, all the more in the context of open-answer assessment, which has been the subject of this thesis.

References

Ahmed A., Pollitt A. *The Support Model for Interactive Assessment.* UCLES – University of Cambridge Local Examination Syndicate. IAEA Conference, Hong Kong, 2002.

Ait-Mokhtar S., Chanod J.-P., Roux C. *A Multi-Input Dependency Parser.* In Proceedings of the Seventh IWPT (International Workshop on Parsing Technologies), Beijing. 2001.

Aldridge N., Broomhead P., Mitchell T. *Computerised Marking of Short-Answer Free-Text Responses.* IAEA Conference. 2003.

Aleven V., Popescu O., Koedinger K. P. *Pedagogical Content Knowledge in a Tutorial Dialogue System to Support Self-Explanation.* In Working Notes of the AIED 2001 Workshop "Tutorial Dialogue Systems". 2001.

Aleven V., Rosé C. P. *Towards Easier Creation of Tutorial Dialogue Systems: Integration of Authoring Environments for Tutoring and Dialogue Systems.* Workshop on Dialog-Based Intelligent Tutoring Systems: State-of-the-Art and New Research Directions. 7th International Conference on Intelligent Tutoring Systems. Maceió, August 31 2004.

Alexandria. A demo can be found at: http://elsap1.unicaen.fr/alexandria2.html

Amaral L., Metcalf V., Meurers D. *Language awareness through re-use of NLP technology.* Ohio State University. NLP in CALL Workshop, CALICO, May 2006a.

Amaral L., Meurers D. *Where Does ICALL fit into Foreign Language Teaching?* Ohio State University. NLP in CALL Workshop, CALICO, May 2006b.

Amaral L., Meurers D. *Using Foreign Language Tutoring Systems for Grammatical Feedback.* Ohio State University. EUROCALL Conference, September 2006c.

Anctil D. *Characterization of the Notion of Lexical Error in the Framework of Explanatory Combinatorial Lexicology.* OLST-RALI, Université de Montréal. 2005b.

Anderson, R.C., Pearson, P.D. *A schema-theoretic view of basic processes in reading comprehension.* In P.D. Pearson, R. Barr, M.L. Kamil, & P. Mosenthal (Eds.), Handbook of reading research. pp. 255–291. New York: Longman. 1984.

Andrenucci, A, Sneiders, E. *Automated Question Answering: Review of the Main Approaches.* Proceedings of the 3rd International Conference on Information Technology and Applications (ICITA'05), July 4-7, Sydney, Australia, IEEE, Vol. 1, pp. 514-519. 2005.

Antidote, by Druide Informatique, at: http://www.druide.com/antidote.html

Apresjan, J. and al. *ETAP-3 Linguistics Processor: a Full-Fledged Implementation of the MTT.* In: Kahane, S. & Nasr, A., eds., pp 279-288. 2003.

Attali Y., Burstein J. *Automated Essay Scoring with E-Rater* V. 2. In JTLA, Journal of Technology, Learning and Assessment. Vol 4, No 3, pp , February 2006.

Bailey, S.M. *Content Assessment in Intelligent Computer-Aided-Language-Learning: Meaning Error Diagnosis in English as a Second Language*. PhD thesis. Ohio State University. 2008.

Balcom P., Copeck T., Szpakowicz S. *DidaLect: Conception, implantation et évaluation initiale.* In Technologies langagières et apprentissage des langues: actes du colloque tenu dans le cadre du Congrés de l'ACFAS, le 11-12 mai 2004, sous la direction de Lise Duquette et Claude St-Jacques, Montréal: ACFAS Cahier No. 105. 2006.

Bannard, C. and Callison-Burch, C. *Paraphrasing with Bilingual Parallel Corpora.* Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL 2005): 597-604. 2005.

Barrière C., Duquette L. *Cognitive-Based Model for the Development of a Reading Tool in FSL.* Computer Assisted Language Learning, Vol. 15, pp 469-481. 2002.

Barzilay, R. and Lee, L. *Learning to Paraphrase; an unsupervised approach using multiple-sequence alignment.* In Proceedings of HLT/NAACL, pp 16-23. 2003.

Beck, K., Andres, C. *Extreme Programming Explained: Embrace Change*. Addison Wesley Professional, 2nd edition. 2004.

Beck, I.L., McKeown, M.G., Sandora, C., Worthy, J. (1996). *Questioning the author: A yearlong classroom implementation to engage students with text.* In The Elementary School Journal, 96, pp. 385–414. 1996.

Bescherelles. *La Grammaire pour Tous*. Hatier, Paris, 1997.

Binon J., Verlinde. S. *La contribution de la lexicographie pédagogique à l'apprentissage et à l'enseignement d'une langue étrangère ou seconde (FLES)*. In Études de linguistique appliquée. Pp. 447-463. 1999.

Black, P. and Wiliam, D. *Inside the black box: Raising standards through classroom assessment.* Phi Delta Kappan, 80 (2): 139-148. 1998.

Bloom B. *Taxonomy of Educational Objectives. Handbook 1: Cognitive Domain.* David Mackay, New York. 1956.

Bodenreider, O, Mitchell, JA and McCray, AT. *Evaluation of the UMLS as a Terminology and Knowledge Resource for Biomedical Informatics.* In Proc AMIA Symp, pp 61–65. 2002.

Bogdan Florin M., Hunger A. *Intelligent Agents: a New Paradigm to Support Collaborative Learning in Distance Education Systems.* In Learning Technology Newsletter, Vol. 7, Issue 2, Pp 18-20, April 2005.

Borin L. *What have you done for me lately? The fickle alignment of NLP and CALL.* Eurocall pre-Conference Workshop on NLP in CALL. August 14, Finland, 2002a.

Borin L. *Where will the Standards for Intelligent Computer-Assisted Language Learning Come from?* LREC 2002, Workshop Proceedings. International Standards of Terminology and Language Resources Management. ELRA, pp 61-68. Las Palmas, Spain. 2002b.

Bosma, W. and Callison-Burch, C. *Paraphrase substitution for recognizing textual Entailment.* In Proceedings of CLEF. 2006.

Bourigault D. *Documentation Syntex* V 1.0. Université de Toulouse 2. February 2006.

Boyer, M. & Lapalme, G. *Generating Paraphrases from Meaning-Text Semantic Networks.* Montreal: Université de Montréal. 1985.

Boytcheva S., Kalaydjiev O., Strupchanska A., Angelova G. *Between Language Correctness and Domain Knowledge in CALL.* In Proceedings of the EuroConference RANLP-2001, Tzigov Chark, Bulgaria, pp. 40-46. September 2001.

Brin, S. and Page, L. *The Anatomy of a Large-Scale Hypertextualc Web Search Engine.* In Proc. WWW7 Conf. Computer Networks 30, pp 107-117. 1998.

British National Corpus, at: http://www.natcorp.ox.ac.uk/

Brown, P., Della Pietra. S.A., Della Pietra, V.J. and R.L. Mercer. *The Mathematics of Statistical Machine Translation.* Computational Linguistics 19(2). pp 263-311. 1993.

Budanitsky, A. and Hirst, G. *Evaluating wordnet-based measures of lexical semantic relatedness.* Computational Linguistics, 32(1). Pp 13–47, 2006.

Bull J., McKenna C. *A Blueprint for Computer-Assisted Assessment.* RoutledgeFalmer, London. 2004.

Bull J., Mc Kenna C. *CAA Centre Update.* Proceedings of the 4th International Computer-Assisted Assessment Conference, Loughborough, 2000.

Bunke H., *Recent Developments in Graph Matching.* In the Proceedings of ICPR'00, 2, pp 117-124. 2000.

Burstein, J., Chodorow, M., and Leacock, C. *CriterionSM: Online essay evaluation: An application for automated evaluation of student essays.* In J. Riedl & R. Hill (Eds.), Proceedings of the Fifteenth Annual Conference on Innovative Applications of Artificial Intelligence, Acapulco, Mexico. pp. 3-10. 2003.

Burstein J., Leacock C., Swartz R. *Automated evaluation of essays and short answers.* In the 5th International Computer Assisted Assessment Conference. Loughborough University, 2001.

Burstein, J., Kaplan, R., Wolff, S. and Liu, C. *Using lexical semantic techniques to classify free-responses.* In Proceedings of the ACL SIGLEX Workshop on Breadth and Depth of Semantic Lexicons, University of California, Santa Cruz. June 1996.

Burston, J. *Exploiting the potential of a computer-based grammar checker in conjunction with self – monitoring strategies with advanced-level students of French.* CALICO Journal, Vol. 18 No. 3, pp 499 – 516. 2001.

Chapelle, C.A. *Computer Applications in Second Language Acquisition. Foundations for Teaching, Testing and Research.* Cambridge, CPU. 2001.

Chapelle, C.A. Multimedia CALL : *Lessons to be learned from research on instructed SLA.* in Language Learning & Technology, vol. 2, nu. 1, pp.22-34. 1998

Chen L., Tokuda N. *A New Diagnostic System for the Japanese-English Translation ILTS by Global Matching Algorithm and POST Parser.* In Proceedings of the Machine Translation Summit, pp 608-616. 1999.

Chodorow, M., & Leacock, C. *An Unsupervised Method for Detecting Grammatical Errors.* ANLP 2000. pp. 140-147. 2000.

Chomsky, N. Aspects of the Theory of Syntax. Special technical report. MIT Press. 1965.

Christie J. R. *Automated Essay Marking – for Both Style and Content.* In Proceedings of the 3rd Computer Assisted Assessment Conference, pp 39-48, 16-17th june 1999.

CISCO, http://cisco.netacad.net/

Clifton, C., Jr., Duffy, S. *Sentence comprehension: Roles of linguistic structure.* In Annual Review of Psychology, 52. pp. 167-196. 2001.

Colpaert, J. *Design of online interactive courseware.* Antwerp: Universiteit Antwerpen. 2004.

Coniam D. *Using Language Engineering Programs to Raise Awareness of Future CALL Potential.* In Computer Assisted Language Learning, Vol 17, No 2. pp 149-175. 2004.

Correcteur 101. www.mysoft.fr/**correcteur**.htm

CRISCO Université de Caen: Lettre d'Information: http://elsap1.unicaen.fr/dicosyno/

Cushion, S. *What does CALL have to offer computer science and what does computer science have to offer CALL?* Computer Assisted Language Learning, 19:2, pp 193 - 242. 2006.

Dagan, I., Glickman, O. and Magnini, B. *The PASCAL Recognizing Textual Entailment Challenge.* In Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment. 2005

Daneš, F. *Papers on Functional Sentence Perspective.* Prague, Academia. 1974.

Davies G. *Computer Assisted Language Learning: Where are we now and where are we going?.* Keynote paper originally presented at the UCALL Conference, University of Ulster, Coleraine in June 2005, revised in March 2007.

Davies W.M., Davis H.C. *Designing Assessment Tools in a Service-Oriented Architecture.* In 1st International ELeGI Conference on Advanced Technology for Enhanced Learning, Vico Equense, (Napoli), Italy, 2005.

Dennis, S. & Kintsch, W. (submitted). *The text mapping and inference generation problems in text comprehension.* 2003.

Dennis, S. *Introducing word order in an LSA framework.* In Handbook of Latent Semantic Analysis. T. Landauer, D. McNamara, S. Dennis and W. Kintsch eds.: Erlbaum. 2006.

DeSmedt, W.H. *Herr Kommissar: An ICALL Conversation Simulator for Intermediate German.* In V.M. Holland, J.D. Kaplan & M.R. Sams (Eds). Intelligent Language Tutors: Theory Shaping Technology. Pp. 153-174. Mahwah, NJ: Erlbaum Associates. 1995.)

Desrochers, A., Duquette, L. *Comment appuyer la compréhension en lecture.* In Technologies langagières et apprentissage des langues: actes du colloque tenu dans le cadre du Congrés de l'ACFAS, le 11-12 mai 2004, sous la direction de Lise Duquette et Claude St-Jacques, Montréal: ACFAS Cahier No. 105. 2006.

Dolan W. B., Quirk C., and Brockett C. *Unsupervised Construction of Large Paraphrase Corpora: Exploiting Massively Parallel News Sources*. COLING 2004, Geneva, Switzerland. 2004.

Duke, N.K., Pearson, D. *Effective Practices for Developing Reading Comprehension.* In Farstrup & Samuels (Eds.). What Research Has To Say About Reading Instruction. 3rd ed., pp. 205–242. Newark, DE: International Reading Association. 2002.

Duquette L., St Jacques C. *Identification de la Macrostructure des Textes: Computation et Cognition*. In Technologies langagières et apprentissage des langues: actes du colloque tenu dans le cadre du Congrés de l'ACFAS, le 11-12 mai 2004, sous la direction de Lise Duquette et Claude St-Jacques, Montréal: ACFAS Cahier No. 105. 2006.

Durkin, D. *What classroom observations reveal about reading comprehension instruction.* In Reading Research Quarterly, 14, pp. 481–533. 1978.

Duscherer K., Chevaux F., Flad D., Mounoud P. *Normes d'Associations Verbales pour 165 Verbes Additionnels*. Document de Travail. Université de Genève. 2006.

Duscherer, K., Mounoud, P. *Normes d'associations verbales pour 151 verbes d'action.* In L'Année Psychologique, Vol.106, pp 397-413. 2006.

Enright, M., Grabe, B., Koda, K., Mosenthal, P., Mulcahy-Emt, P., Schedl, M. *TOEFL 2000 reading framework: A working paper* (TOEFL Monograph Series, Report No. 17). Princeton, NJ: Educational Testing Service. 2000.

EuroWordNet, at the University of Amsterdam:
http://www.illc.uva.nl/EuroWordNet

Farmer R., Gruba P. *Towards Model-Driven End-User Development in CALL.* In Computer Assisted Language Learning, Vol 19, No 2 & 3. pp 149-191. 2006.

Farmer, R.A., Gruba, P., & Hughes, B. *Towards principles for call software quality improvement*. In CALL and Research Methodologies: Proceedings of the Eleventh International conference on CALL, J. Colpaert, W. Decoo, M. Simons, & S. Van Beuren (Eds.), pp. 103-113. University of Antwerp. 2004.

Fenton-Kerr T., Koppi T., Chaloupka M., Cattle S., Anderson A. *ALLIANCE: A Student's Best Friend.* Technical Report of the Centre for Teaching and Learning, University of Sydney. 1998.

Firbas, J. *Functional Sentence Perspective in Written and Spoken Communication*. Cambridge: Cambridge University Press.1992.

FreeText. FreeText : *French in Context: An advanced hypermedia CALL system featuring NLP tools for a smart treatment of authentic documents and free production exercises*. Genève : Université de Genève. http://www.latl.unige.ch/FreeText. 2002

Fuchs, C. *Paraphrase et énonciation*. Ophrys, Paris. 1994.

Fuchs, C. *Paraphrase et théorie du langage. Contribution à une histoire des théories linguistiques contemporaines et à la construction d'une théorie énonciative de la paraphrase* [these de doctorat]. Université Paris VII. 1980.

Fuller, M. *Assessment for real in Virtual Learning Environments – how far can we go?*, Interface - Virtual Learning and Higher Education, Mansfield College, Oxford 2002.

Gabrilovich, E. and Markovitch, S. *Computing semantic relatedness using Wikipedia-based explicit semantic analysis*. Proceedings of IJCAI. pp 1606-1611. 2007

Gabroliovitch, E. and Markovitch, S. *Feature generation for text categorization using world knowledge*. In IJCAI'05. pp 1048–1053. 2005.

Gagnon, M., Da Sylva, L. *Text Compression by Syntactic Pruning*. In: Proceedings of the 19th Conference of the Canadian Society for Computational Studies of Intelligence (AI'06). 7-9 juin 2006, Université Laval, Québec, pp. 312-323. 2006.

Gamper, J. and J. Knapp. *Review of intelligent CALL systems*. Computer Assisted Language Learning. Vol. 15 No. 4, pp. 329-342. 2002.

Gentner D., Markman A.B. *Analogy - Watershed or Waterloo? Structural Alignment and the Development of Connectionist Models of Cognition.* In ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS 5, Morgan Kaufmann Publishers Inc. San Fransisco. 1992.

Giasson J. *La Compréhension en Lecture*. Gaëtan Morin Editeur, Montréal. 1990.

Giouroglou, H. & Economides, A.A.: *State-of-the-Art and adaptive open-closed items in adaptive foreign language assessment.* Proceedings 4th Hellenic Conference with International Participation: Informational and Communication Technologies in Education, Athens, pp. 747-756, 2004.

Golding, A. *A Bayesian hybrid for contextsensitive spelling correction.* Proceedings of the 3rd Workshop on Very Large Corpora. Cambridge, MA. pp. 39-53. 1995.

Graesser A.C., Person N., Lu Z., Jeon M. G., McDaniel B. *Learning While Holding a Conversation with your Computer*. In Technology-based Education: Bringing Researchers and Practitioners Together, pp 143-167. Information Age Publishing. 2005.

Graesser, A., McNamara, D., Louwerse, M., and Cai, Z. *Coh-Metrix: Analysis of text on cohesion and language*. Behavioral Research Methods, Instruments, and Computers 36, pp 193-202. 2004.

Graesser A.C., Person, N.K., and Magliano, J.P. *Collaborative dialogue patterns in naturalistic one-on-one tutoring*. Applied Cognitive Psychology 9: 495-522. 1995.

Granfeldt, J., Nugues, P. et al. *CEFLE and Direkt Profil: A New Computer Learner Corpus in French L2 and a System for Grammatical Profiling*. In Proceedings of the 5th International Conference on Language Ressources and Evaluation, pp. 565-570. Genoa, 22-28 May 2006.

Granger, S. *Error-Tagged Learner Corpora and CALL : A Promising Synergy*. In CALICO Journal. 20:3. pp. 465-480. 2003.

Hall, R., P. *Computational Approaches in analogical Reasoning: A Comparative Analysis*. In Artificial Intelligence, pp 39-120. Elsevier Science Publishers B.V (North-Holland). 1989.

Halliday, M.A.K. *System and Function in Language*. London: Oxford University Press. 1976.

Halliday, M.A.K. *An Introduction to Functional Linguistics*. Arnold, London. 1985

Harris, Z. *Mathematical Structures of Language*. Interscience Publishers, New York. 1968.

Hartley J.R., Sleeman D. H. *Towards intelligent teaching systems*. International Journal of Man-Machine Studies Vol. 5, pp 215–236. 1973.

Heift, T. *Intelligent language tutoring systems for grammar practice*. Zeitschrift fur Interkulturellen Fremdsprachenunterricht [Online]. Vol 6 No 2. 2001.

Heift, T. *Multiple learner errors and meaningful feedback: A challenge for ICALL systems*. CALICO Journal. Vol 20 No 3, pp 533–548. 2003a.

Heift, T. and Nicholson D. *Web delivery of adaptive and interactive language tutoring*. International Journal of Artificial Intelligence in Education. Vol 12 No 4, pp 310–325. 2003b.

Heift, T., & Schulze, *M. Errors and Intelligence in Computer-Assisted-Language-Learning, Parsers and Pedagogues*. Routledge, New York. 2007.

Hermet, M. And Barriere. C. *Causality Taking Root in Terminology*. In Proceedings of TKE'2002 (Terminology and Knowledge Engineering), Nancy, 2002.

Higgins, D., Burstein, J., Marcu, D., & Gentile, C. *Evaluating multiple aspects of coherence in student essays*. In Proceedings of the Annual Meeting of HLT/NAACL, Boston, pp 185-192. 2004.

Hincks R. *Tutors, Tools and Assistants for the L2 User.* In Phonum Vol 9. pp 173-176. 2003.

Hirschman L. & al., "Deep Read: A Reading Comprehension System," *Proc. 37th Ann. Meeting Assoc. Computational Linguistics*, Assoc. for Computational Linguistics, College Park, Md., pp. 325–332. 1999.

Hirschman L, Breck, E., Light, M., Burger, J.D., Ferro, L. *Automated Grading of Short Answers Tests*. In The Debate on automated essay Scoring, IEEE Intelligent Systems. pp. 31-35. 2000.

Holland, M., Kaplan, J. and Sams R. (eds.). *Intelligent Language Tutors: Theory Shaping Technology*. Lawrence Erlbaum Associates, Inc. 1995.

HotPotatoes, http://web.uvic.ca/hrd/hotpot/

Hubbard P. *A Review of subject Characteristics in CALL Research.* In Computer Assisted Language Learning, Vol 18, No 5. pp 351-368. December 2005.

Hubbard P., Levy M. *Why call CALL "CALL"?* In Computer Assisted Language Learning, Vol 18, No 3, pp 143-149, Editorial. July 2005.

Johnson D., Johnson B. *Highlighting Vocabulary in Inferential Comprehension Instruction.* In Journal of Reading, Vol. 29, No. 7, pp 622-626. 1986.

Kahane, S. *The Meaning-Text Theory*. To appear in Dependency and Valency. Handbooks of Linguistics and Communication Science. De Gruyter, Berlin, New York. Ágel, Vilmos et al. (eds.). 2003.

Kaplan, J.D., Sabol M.A., Wisher R.A. and Seidel R.J. *The Military Language Tutor (MILT) Program: An Advanced Authoring System*. Computer-Assisted-Language-Learning, 11(3), 265-287. 1998.

Kies, D. Evaluating Grammar Checkers: A Ten-Year Study. http://papyr.com/hypertextbooks/grammar/gramchek.htm

Kintsch, W. *Memory for Text*. In Discourse Processing, A. Flammer & W. Kintsch (Eds.). North Holland Publishing, pp 186-204. 1982.

Kintsch W., Dijk T. *Toward a Model of Text Comprehension and Production.* In Psychological Review, Vol. 85, pp 363-394. 1978.

Knight P. *Being a Teacher in Higher Education.* Society for Research in Higher Education and Open University Press, Buckingham. 2002a.

Knight P. *Summative Assessment in Higher Education: Practice in Disarray.* In Studies in Higher Education, Vol. 27 No. 2, pp 275-286. 2002b.

Knutsson O. Ceratto Pargman T., Eklundh K. S. *Transforming Grammar Checking Technology into a Learning Environment for Second Language Writing.* In Proceedings of the HLT-NAACL 03 Workshop on Building Educational Applications Using Natural Language Processing, pp. 38–45, 2003.

Korhonen A., Briscoe T. *Extended Lexical-Semantic Classification of English Verbs.* In Proceedings of the HLT/NAACL Workshop on Computational Lexical Semantics, Boston. 2004.

Korhonen, T. Kaski, S. Lagus, K. Salojarvi, J. Honkela, J. Paatero, V. Saarela, A. *Self organization of a massive document collection.* In IEEE Transactions on Neural Networks. 11(3). Pp 574-585. 2000.

Kraif O., Ponton C. *Du bruit, du silence et des ambiguïtés : que faire du TAL pour l'apprentissage des langues ?* in Actes de TALN 2007. 5-8 juin 2007. Toulouse. 2007.

Kucan, L., Beck, I.L. *Thinking aloud and reading comprehension research: Inquiry, instruction and social interaction.* Review of Educational Research, 67, pp. 271–299. 1997.

Kukich, K. *Beyond Automated Essay Scoring.* In The Debate on automated essay Scoring, IEEE Intelligent Systems. pp. 22 – 27. 2000.

Lakoff, G. *Toward Generative Semantics.* In McCawley 1976. pp. 232-296. 1963.

Landauer T.K., Laham D., Foltz P.W. *Automated Essay Scoring: Applications to Educational Technology.* 2003a.
http://www.psych.nmsu.edu/~pfoltz/reprints/Edmedia99.html

Landauer, T. K., Laham, D., & Foltz, P. W. *Automated scoring and annotation of essays with the Intelligent Essay Assessor.* In M. D. Shermis & J. Burstein (Eds.), *Automated essay scoring: A cross-disciplinary perspective.* Lawrence Erlbaum Associates, Inc. Mahwah, NJ. pp. 87–112. 2003b.

Landauer, T. K., & Dumais, S. T. *A solution to Plato's problem: The Latent Semantic Analysis theory of the acquisition, induction, and representation of knowledge.* Psychological Review, 1 0 4 , pp 211-240. 1997.

Larkey L. S. *Automatic Essay Grading Using Text Classification Techniques.* In Proceedings of the 21st ACM-SIGIR International Conference on Research and Development in Information Retrieval. Melbourne, Australia, August 24-28 1998.

Laufer, B., Goldstein Z. *Testing vocabulary Knowledge.* In Language Learning 54:3. pp 399-436. 2004.

Lavie, A. and Agarwal, A. *METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments*. In Proc. Of the ACL/WMT, pp 228–231. 2007.

Leacock, C. *Scoring Free-Responses Automatically: A Case Study of a Large-Scale Assessment*. Examens, 1:3. 2004.

Lefff. Can be found at: http://www.labri.fr/perso/clement/lefff/

Lemaire, B. and Denhière, G. *Effects of High-Order Co-occurrences on Word Semantic Similarity*. in Current Psychology Letters, 18, Vol. 1. 2006.

Levenshtein, V.I. *Binary codes capable of correcting deletions, insertions, and reversals*. In Soviet Physics Doklady 10. pp. 707–710. 1966.

L'Haire S. & Vandeventer-Faltin A. *Using NLP tools in a CALL software: the FreeText project*. In EuroCALL 2003, Limerick, September 3-6, 2003a.

L'Haire S., Vandeventer-Faltin A. *Diagnostic d'Erreurs dans le projet* FreeText. In Revue ALSIC, Vol 6, N. 2, pp 21-37. December 2003b.

Li, Y. McLean, D. Bandar, Z.A. O'Shea, J.D. Crockett, K. *Sentence similarity based on semantic nets and corpus statistics*. In IEEE Transactions on Knowledge and Data Engineering. 18(8). Pp 1138-1150. 2006.

Lin, D. and Pantel, P. *DIRT - Discovery of Inference Rules from Text*. In Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Mining. Pp 323-328. 2001.

McCarthy, P. M., and Jarvis, S. *A theoretical and empirical evaluation of vocd*. Language Testing. Vol. 24. pp 459-488. 2007.

McCarthy, P.M., Rus, V., Crossley, S.A., Bigham, S.C., Graesser, A.C. and Mc-Namara, D.S. *Assessing entailer with a corpus of natural language*. In Proceedings of the 20th International Conference of the Florida Research Society of Artificial Intelligence. May 7-9, Key West, FL, pp. 247-252. 2007.

McCawley, J. The Role of Semantics in a Grammar. In: McCawley, J., *Grammar and Meaning*. New York: Academic Press. pp. 59-99. 1976a.

McCawley, J. Semantic Representation. In: McCawley, J., *Grammar and Meaning*. New York: Academic Press. pp. 240-257. 1976b.

Mackenzie D. *Recent Development in the Tripartite Interactive Assessment Delivery System* (TRIADS). In Proceedings of the 3rd International Computer-Assisted Assessment Conference, Loughborough, 16-17 june 1999.

MacWhinney, B. *Evaluating foreign language tutoring systems*. In V.M. Holland, J.D. Kaplan, & M.R. Sams (Eds.), Intelligent language tutors, pp. 317–326. Mahwah, NJ: Lawrence Erlbaum Associates. 1995.

Mel'čuk, I., Wanner, L. *Towards a Lexicographic Approach to Lexical Transfer in Machine Translation* (Illustrated by the German—Russian pair). In Machine Translation, 16. pp. 21-87. 2001

Mel'čuk, I., Arbatchewsky-Jumarie, N., Iordanskaja, L., Mantha, S., et Polguere, A. *Dictionnaire explicatif et combinatoire du français contemporain*. In Recherches lexico sémantiques IV. Les Presses de l'Université de Montréal. 2000.

Mel'čuk, I. *Vers une linguistique Sens-Texte. Leçon inaugurale*, College de France, Paris. 1997.

Mel'čuk, I. A. *Lexical Functions : A Tool for the Description of Lexical Relations in a Lexicon.* In Wanner, L., ed. pp.37-102. 1996.

Mel'čuk, I. *Paraphrase et lexique: la theorie Sens-Texte et le* Dictionnaire explicatif et combinatoire. In: Mel'čuk *et al.* pp. 9-59. 1992.

Mel'čuk, I. (1981), Meaning-Text Models: A Recent Trend in Soviet Linguistics, Annual Review of Anthropology, 10. pp. 27-62. 1981.

Memodata: http://www.memodata.com/

Metcalf V., Meurers D. *Generating Web-based English Prepositions Exercises from Real-World Texts.* Ohio State University. EUROCALL Conference, September 2006.

Meyer B. *Prose Analysis: Purposes, Procedures and Problems.* In "Understanding Expository Text", B. Bitton and J. Black (Eds). Hillsdale, NJ, Lawrence Erlbaum. 1985.

Meyer B. *Following the Author's Top-Level Organization: An Important Skill for Reading Comprehension.* In "Understanding Reader's Understanding", R. Tierney, P. Andres and Mitchell J. Hillsdale, NJ, Lawrence Erlbaum. Pp 59-77. 1987.

Mihalcea, R., Corley, C., and Strapparava, C. *Corpus-based and knowledge-based measures of text semantic similarity.* In Proceedings of AAAI'06, pp 775-780. 2006.

Michaud L. N., McCoy K. F., Stark L. A. *Modelling the Acquisition of English: An Intelligent CALL approach.* In Proceedings of the 8th Interneational Conference on User Modelling. Bauer, Gmytrasiewicz & Vassileva, Eds. Springerverlag, pp 14-23, 2001.

Michaud, L.N., and McCoy, K.F. Supporting Intelligent Tutoring in CALL by Modeling the User's Grammar. In *Proceedings of the Thirteenth International FLAIRS Conference*, Orlando, pp 50-54, 2000.

Milićević, J. *Modélisation sémantique, syntaxique et lexicale de la paraphrase.* Thèse de Doctorat, Université de Montréal. 2003a.

Milićević, J. *A Short Guide to the Meaning-Text Linguistic Theory.* Topics in Computational Linguistics and Intelligent Text Processing (CIC-ing 2000 post-conference book), Gelbukh, A., ed.. Springer-Verlag. Berlin, Heidelberg, New York. 2003b.

Miller, G. A. *WordNet: A Lexical Databases for English.* Communications of the ACM, pages 39–41, 1995.

Milne, D., Medelyan, O. and Witten, I.H. *Mining Domain-Specific Thesauri from Wikipedia: A Case Study.* In IEEE/WIC/ACM International Conference on Web Intelligence. Pp 442-448. 2006.

Mistrut, D. Implementation of the Levenshtein edit distance can be found at: http://search.cpan.org/~jgoldberg/Text-Levenshtein-0.05/Levenshtein.pm

Mitchell T., Russell T., Broomhead P., Alridge N. *Towards Robust Computerized Marking of Free-text responses.* In the 6th International Computer Assisted Assessment Conference. Loughborough University, 2002.

Mitkov, R. *Anaphora resolution.* Studies in Language and Linguistics. Longman. 2002.

Morris, J. *Readers perceptions of lexical cohesion in text.* In: Proceedings of the 32nd annual conference of the Canadian Association for Information Science, Winnipeg. 2004.

Muth, D. *Structure Strategies for Comprehending Expository Text.* In Reading Research and Instruction. Vol. 27, No. 1, pp 66-72. 1987.

Nakov, P., Angelova G., Strupchanska A., Kalaydjiev O., Yankova M., Boytcheva S., Vitanova I., Nakov P. *Towards Deeper Understanding and Personalisation in CALL.* In Proceedings of Workshop 6 "eLearning for Computational Linguistics and Computational Linguistics for eLearning", COLING-04, Geneva, August, pp 45-52. 2004.

Nerbonne, J. *Computer-Assisted Language Learning and Natural Language Processing.* In R. Mitkov (Ed.). The Oxford Handbook of Computational Linguistics, pp 670-698. OUP. 2003.

Nerbonne, J. *Computer-Assisted Language Learning and Natural Language Processing.* In Handbook of Computational Linguistics, R. Mitkov (Ed.), Oxford Univ. Press, pp. 670-698. 2002.

New, B., and Pallier, C. *Liste des Mots du Français et leur Fréquence par Million d'Occurence.* CNRS. Can be retrieved at http://www.lexique.org/listes/liste_mots.php. 2001.

Och, F. and Ney, H. *Improved Statistical Alignment Models.* In Proceedings of the ACL. pp 440-447. Hong Kong, China. 2000.

Page, E.B. *New computer grading of student prose, using modern concepts and software.* In Journal of Experimental Education, Vol. 62 No. 2, pp 127-142. 1994.

Page, E. The imminence of grading essays by computer. Phi Delta Kappan 47, 238-243. 1966.

Papineni, K., Roukos, S., Ward, T., and Zhu, W. J. *BLEU: a method for automatic evaluation of machine translation.* in ACL: 40th Annual meeting of the Association for Computational Linguistics pp. 311–318. 2002.

Pearson D., Johnson D. *Teaching Reading Comprehension.* Holt, Rinehart and Winston, New York. 1978.

Pedersen, T., Patwardhan. S. and Michelizzi, J. *WordNet::Similarity – Measuring the relatedness of concepts.* In Companion Volume of the Proceedings of the Human Technology Conference of the North American Chapter of the Association for Computational Linguistics, pp. 267–270. 2004.

Pennel, J.M. fr_soundex Perl module can be found at: http://sqlpro.developpez.com/cours/soundex/

Peñas, A., Rodrigo, A. and Verdejo, F. *Overview of the Answer Validation Exercise 2007.* In the CLEF 2007 Working Notes. Peters, C. And al. (Eds). Vol. 5152, Springer, Heidelberg. 2008.

Penumatsa, P., Ventura, M., Graesser, A.C., Franceschetti, D.R., Louwerse, M., Hu, X., Cai, Z., and the Tutoring Research Group. *The right threshold value: What is the right threshold of cosine measure when using latent semantic analysis for evaluating student answers*? International Journal of Artificial Intelligence Tools, vol. 12. pp 257-279. 2004.

Pérez D., Alfonseca E. *Adapting the automatic assessment of free-text answers to the students.* In Proceedings of the 9th international conference on CAA, Loughborough, UK, July 2005.

Pérez D., Postolache O., Alfonseca E., Cristea D., Rodriguez P. *About the Effects of Using Anaphora Resolution in Assessing Free-Text Student Answers.* Proc RANLP-2005, pp 380-386. 2005.

Pérez D., Alfonseca E., Rodríguez P. *Application of the sc Bleu method for evaluating free-text answers in an e-learning environment.* Proceedings of the Language Resources and Evaluation Conference (LREC). 2004.

Polguère, A. La théorie Sens-Texte. *Dialangue* 8-9. Université du Québec à Chicoutimi. pp. 9-30. 1998.

Powers, D. E., et al. *Stumping e-rater:® Challenging the validity of automated essay scoring* (GRE No. 98-08bP, ETS RR-01-03). Princeton, NJ: Educational Testing Service. 2001.

Propp V. *Morphologie du conte Russe.* Ed. Le Seuil. Paris. 1970.

Qiu, L., Kan, M., and Chua, T. *Paraphrase recognition via dissimilarity significance classification.* In Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, pp 18–26. Association of Computational Linguistics. 2006.

Qualrus: www.ideaworks.com/**qualrus**/index.html

QTI: IMS Question & Test Interoperability: An Overview. Final Specification Version 1.2. 2002. http://www.imsglobal.org/question/qtiv1p2/imsqti_oviewv1p2.html

Quirk, C., Brockett, C., Dolan, W. *Monolingual machine translation for paraphrase generation.* In Proceedings of EMNLP, pp 142.149. 2004.

Raphael, T.E., & McKinney, J. *An examination of fifth- and eighth-grade children's question answering behavior: An instructional study in metacognition.* In Journal of Reading Behavior, 15. pp. 67–86. 1983.

Raymond P.M. *Reader Response Theory in an Advanced ESL Course : An Application.* Technical Report, University of Ottawa Second Language Institute. 1995.

Read, J. Measuring the Vocabulary Knowledge of Second Language Learners. In RELC Journal. 19:2. pp. 12-25. 1988.

Read J., & Chapelle, C.A. *A framework for second language vocabulary assessment.* Language Testing. 18. pp. 1-32. 2001.

Richards , J.C. *The Role of Vocabulary Teaching.* In TESOL Quarterly, 10:1. pp. 77-89. 1976.

Richgels D., McGee L., Lomax R., Sheard C. *Awareness of Four Text Structures: Effects on Recall of Expository Text.* In Reading Research Quarterly, Vol. XXII, No. 2, pp 177-197. 1987.

Roos B. *ICT, Assessment and the Learning Society.* ICT network at the European Conference on Educational Research (ECER), Lisbon, 11-14 September, 2002.

Rosé, C.P., Gaydos, A., Hall, B.S., Roque, A., VanLehn, K. *Overcoming the Knowledge Engineering Bottelneck for Understanding Student Language Input.* In Artificial Intelligence in Education. Sydney, Australia: Amsterdam, Kay, J., editor. IOS Press. 2003a.

Rosé P.C., Roque A., Dumisizwe B., Van Lehn K. *A Hybrid Approach to Content Analysis for Automatic Essay Grading.* In Building Educational Applications Using Natural Language Processing, pp 68–75. 2003b.

Rudner, L.M. & Liang, T. *Automated essay scoring using Bayes' Theorem.* The Journal of Technology, Learning and Assessment, Vol. I No. 2, pp 3-21. 2002.

Rus V., MacCarthy P.M. and Graesser A.C. *Analysis of a Textual entailer.* In Computational Linguistics and Intelligent Text Processing. Springer Berlin/Heidelberg. pp 287-298. 2006.

Rus, V., Graesser, A. C. and Desai, K. *Lexico-Syntactic subsumption for textual entailment.* In Proceedings of the International Conference on Recent Advances in Natural Language Processing. Borovets, Bulgaria. Pp 444-452. 2005.

Schmidt N. G., Norman G. R., Boshuzen H. P. A. *A Cognitive Perspective on Medical Expertise: Theory and Applications.* In Academic Medecine Vol. 65, pp 611-621. 1990.

Schneider, D. and McCoy, K. F. *Recognizing syntactic errors in the writing of second language learners.* In Proceedings of COLING/ACL 98. 1998.

Schulman, J.L. *Using Medical Subject Headings (MeSH) to examine patterns in American medicine.* [course paper, STS 5206]. Falls Church: Virginia Polytechnic Institute and State University, Northern Virginia Center; May, 2000.

SCORM/ADL (Advanced Distributed Learning) : http://www.adlnet.org.

Sebastiani, F. *Machine Learning in automated Text Classification.* In ACM Computing Surveys, 34(1). Pp 1-47. 2002.

Selvi, P. Gopalan, N.P. *Sentence Similarity Computation Based on Wordnet and Corpus Statistics.* In Proc of the International Conference on Computational Intelligence and Multimedia Applications. Vol. 1, pp 9-14. Dec 2007.

Sgall, P, Hajičova, E. & Panevova, J. *The Meaning of the Sentence in its Semantic and Pragmatic Aspects.* Dordrecht, Reidel. 1986.

Sheehan, K.M., Ginther, A., Schedl, M. *The development of a proficiency scale for the TOEFL reading comprehension section.* Paper presented at the Annual Conference of the Association for Applied Linguistics, Stamford, CT. 1999.

Snow, R., Vanderwende, L. and Menezes, A. *Effectively using syntax for recognizing false entailment.* In Proc. Of HLT/NAACL, pp 33-40, New York, 2006.

Sowa, J.F. Building, Sharing, and Merging Ontologies. Last update 2009. Can be found at: http://www.jfsowa.com/ontology/ontoshar.htm

Sowa J.F., Majumdar, A.K. *Analogical Reasoning, VivoMind LLC.* In the Proceedings of the International Conference on Conceptual Structures: *Conceptual Structures for Knowledge Creation and Communication.* A. Aldo, W. Lex, & B. Ganter, eds. LNAI 2746, Springer-Verlag, pp. 16-36. Dresden, 2003.

Sparck-Jones, K. *A Statistical Interpretation of Term Specificity and Its Application in Retrieval.* Journal of Documentation 28 (1). Pp 11-20. 1972.

Steele, J. *Meaning-Text Theory: Linguistics, Lexicography and Implications.* University of Ottawa Press. 1990.

Strube, M. and Ponzetto, S. *WikiRelate! Computing semantic relatedness using Wikipedia.* In AAAI'06, pp 1419-1424, Boston, MA, 2006.

Sturgeon C. M. *Computer Assisted Language Learning: the integration of instructional design concepts into language learning.* 2003. http://faculty.leeu.edu/~msturgeon/CALL%20Term%20PaperITCE679-Final-Version.doc

Sukkarieh J.Z., Pulman S.G., Raikes N. *Auto-marking: Using Computational Linguistics to score short, free text responses.* Cambridge University. 2003.

Swaffar J., Arens K., Byrnes H. *Reading for Meaning. An Integrated Approach to Language Learning.* Prentice Hall, NJ. 1991.

Swaffar, J.K. *Reading and Cultural Literacy.* In Journal of General Education, Vol. 38, pp 70-84. 1986.

Tufiş D., Cristea D., Stamou S. *BalkaNet: Aims, Method, Results and Perspectives. A General Overview.* In Romanian Journal of Information Science and Technology. Volume 7, Numbers 1-2, pp 9-43. 2004.

Turney, P. Review of LSA. At: http://apperceptual.wordpress.com/2007/01/19/effects-of-high-order-co-occurrences-on-word-semantic-similarity/

Turney, P. *Mining the web for synonyms: PMI-IR versus LSA on TOEFL.* In Proc. ECML-01, pp. 491–502. 2001.

Valenti S. *Information Technology for Assessing Student Learning.* Special Series on Approaches to Applying IT to the Assessment Process of Student Learning. In Journal of Information Technology Education – Introduction to Vol. 2, 2003a.

Valenti S., Cucchiarelli A., Neri F. *An Overview of Current Research on Automated Essay Grading.* In Journal of Information Technology Education. Vol. 2. pp 319-330. 2003b.

Van Dijk, T. A., & Kintsch, W. *Strategies of discourse comprehension.* New York: Academic Press. 1983.

VanLehn K., Jordan P., Rosé C.P., and The Natural Language Tutoring Group. *The architecture of why2-atlas: a coach for qualitative physics essay writing.* In the Proceedings of the Intelligent Tutoring Systems Conference, pp 158-167, 2002.

Vinther J. *Can Parsers Be a Legitimate Pedagogical Tool?* In Computer Assisted Language Learning, Vol 17, No 3 & 4, pp 267-288. 2004.

Vitanova I. *Evaluating Integrated NLP in Foreign Language Learning: Technology Meets Pedagogy.* In Proceedings of InSTIL/ICALL 2004 – NLP and Speech Technologies in Advanced Language Learning Systems – Venice. 17-19 June 2004.

Wang, R. and Neumann, G. Recognizing *Textual Entailment Using Sentence Similarity based on Dependency Tree Skeletons.* In Proceedings of the Workshop on Textual Entailment and Paraphrasing, pp 36–41, Prague, June 2007.

WebCT, http://www.webct.com/

Wermter, S. and Hung, C. *Selforganizing classification on the Reuters news corpus.* In Proc. Of Coling-02. 2002.

Wesche, M., Paribakht, T.S.. *Assessing L2 Vocabulary Knowledge: Depth versus Breadth.* In The Canadian Modern Language Review, 53(1), pp. 13-40. 1996.

Wiemer-Hastings, P., Wiemer-Hastings, K., and Graesser, A. *Improving an intelligent tutor's comprehension of students with Latent Semantic Analysis.* In Lajoie and Vivet, Artificial Intelligence in Education. Pp 535-542. Amsterdam: IOS Press. 1999.

Wordnet. Documentation can be found at: http://wordnet.princeton.edu/doc

Wixson K. K. *Postreading Question-Answer Interactions and Children's Learning from Texts.* In Journal of Education Psychology, Vol. 5, pp 413-423. 1983.

Yang Y. W., Buckendahl C. W., Juszkiewicz P. J., Bhola D. S. *A Review of Strategies for Validating Computer Automated Scoring.* In Applied Measurement in Education, Vol. 15, pp 91-412. 2002.

Appendix A. XIP Output

```
C:\xip-9.59-5\sample\grammar\sample8>echo le but du projet mené par
l'ESIS est d'aider les couples qui ayant des difficultés à avoir un
enfant.   | parse -t

C:\xip-9.59-5\sample\grammar\sample8>echo off
<XIPRESULT math="0">


GROUPE
                                        +--------------------------------
+---------------+----------+------+------------+----------------+------
--------+---------+
                                        |                                |
|          |      |            |                |              |
|
                                      SC                                IV
NP          BG     GV            NP                IV             NP
SENT
            +-----------+------------+---------+--------------+     +---
---+       +-------+       +      +      +---------+        +-------+
+-------+     +-
            |           |            |         |              |     |
|       |       |       |      |      |         |        |       |     |
|     |
          NP           PP           VERB       PP             FV   PREP
VERB    DET    NOUN    PRON   VERB    DET      NOUN     PREP    VERB
DET    NOUN    .
         +-----+     +-------+       +     +---------+        +     +
+       +       +       +      +      +         +        +      +-     +
+
         |     |     |       |       |     |         |        |     |
|       |       |      |      |      |         |        |      |      |
|
       DET  NOUN  PREP     NP     mené  PREP       NP      VERB   d'
aider   les   couples   qui   ayant   des   difficultés   à    avoir
un    enfant
         +     +     +       +            +     +------+     +
         |     |     |       |            |     |      |     |
       le    but   du     NOUN          par   DET   NOUN   est
                           +                  +      +
                           |                  |      |
                        projet               l'    ESIS


<LUNIT language="French"><NODE num="44" tag="GROUPE" start="0"
end="101"> <FEATURE attribute="CAT" value="0"/>
<NODE num="58" tag="SC" start="0" end="36">
```

```
<FEATURE attribute="QUEP" value="+"/>
<FEATURE attribute="POURSN" value="+"/>
<FEATURE attribute="LOCSN" value="+"/>
<FEATURE attribute="IL" value="+"/>
<FEATURE attribute="EN" value="+"/>
<FEATURE attribute="DESVINF" value="+"/>
<FEATURE attribute="DESN" value="+"/>
<FEATURE attribute="CONTRESN" value="+"/>
<FEATURE attribute="AVOIR" value="+"/>
<FEATURE attribute="ACEQUEP" value="+"/>
<FEATURE attribute="ASVINF" value="+"/>
<FEATURE attribute="ASN" value="+"/>
<FEATURE attribute="SVINFDIR" value="+"/>
<FEATURE attribute="SVINF" value="+"/>
<FEATURE attribute="SN" value="+"/>
<FEATURE attribute="SADJDESVINF" value="+"/>
<FEATURE attribute="SADJ1" value="+"/>
<FEATURE attribute="SADJ" value="+"/>
<FEATURE attribute="FIN" value="+"/>
<FEATURE attribute="VERB" value="+"/>
<NODE num="50" tag="NP" start="0" end="6">
<FEATURE attribute="DESN" value="+"/>
<FEATURE attribute="PASSIVE" value="+"/>
<FEATURE attribute="FONC" value="FSUBJ"/>
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="NOUN" value="+"/>
<FEATURE attribute="DET" value="+"/>
<NODE num="0" tag="DET" start="0" end="2">
<FEATURE attribute="STARTBIS" value="+"/>
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="MASC" value="+"/>
<FEATURE attribute="SG" value="+"/>
<FEATURE attribute="DEF" value="+"/>
<FEATURE attribute="DET" value="+"/>
<TOKEN pos="DET" start="0" end="2">
le
<READING lemma="le" pos="DET">
</READING>
</TOKEN>
</NODE>
<NODE num="2" tag="NOUN" start="3" end="6">
<FEATURE attribute="DESN" value="+"/>
<FEATURE attribute="SFDE" value="+"/>
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="MASC" value="+"/>
<FEATURE attribute="SG" value="+"/>
<FEATURE attribute="P3" value="+"/>
<FEATURE attribute="NOUN" value="+"/>
<TOKEN pos="NOUN" start="3" end="6">
but
<READING lemma="but" pos="NOUN">
</READING>
</TOKEN>
</NODE>
</NODE>
<NODE num="55" tag="PP" start="7" end="16">
<FEATURE attribute="DESVINF" value="+"/>
```

```
<FEATURE attribute="SVINFDIR" value="+"/>
<FEATURE attribute="FNPP" value="+"/>
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="PREP" value="+"/>
<FEATURE attribute="NOUN" value="+"/>
<FEATURE attribute="DET" value="+"/>
<NODE num="4" tag="PREP" start="7" end="9">
<FEATURE attribute="SFDE" value="+"/>
<FEATURE attribute="FORM" value="FDE"/>
<FEATURE attribute="MASC" value="+"/>
<FEATURE attribute="SG" value="+"/>
<FEATURE attribute="DEF" value="+"/>
<FEATURE attribute="PREP" value="+"/>
<FEATURE attribute="DET" value="+"/>
<TOKEN pos="PREP" start="7" end="9">
du
<READING lemma="de" pos="PREP">
</READING>
</TOKEN>
</NODE>
<NODE num="49" tag="NP" start="10" end="16">
<FEATURE attribute="DESVINF" value="+"/>
<FEATURE attribute="SVINFDIR" value="+"/>
<FEATURE attribute="NOUN" value="+"/>
<NODE num="6" tag="NOUN" start="10" end="16">
<FEATURE attribute="DESVINF" value="+"/>
<FEATURE attribute="SVINFDIR" value="+"/>
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="MASC" value="+"/>
<FEATURE attribute="SG" value="+"/>
<FEATURE attribute="P3" value="+"/>
<FEATURE attribute="NOUN" value="+"/>
<TOKEN pos="NOUN" start="10" end="16">
projet
<READING lemma="projet" pos="NOUN">
</READING>
</TOKEN>
</NODE>
</NODE>
</NODE>
<NODE num="8" tag="VERB" start="17" end="21">
<FEATURE attribute="LOCSN" value="+"/>
<FEATURE attribute="AVOIR" value="+"/>
<FEATURE attribute="ACEQUEP" value="+"/>
<FEATURE attribute="ASN" value="+"/>
<FEATURE attribute="SVINFDIR" value="+"/>
<FEATURE attribute="SVINF" value="+"/>
<FEATURE attribute="SN" value="+"/>
<FEATURE attribute="SFA" value="+"/>
<FEATURE attribute="FONC" value="FPOST"/>
<FEATURE attribute="MASC" value="+"/>
<FEATURE attribute="SG" value="+"/>
<FEATURE attribute="PARTPAS" value="+"/>
<FEATURE attribute="VERB" value="+"/>
<TOKEN pos="VERB" start="17" end="21">
mené
<READING lemma="mener" pos="VERB">
```

```
</READING> <READING lemma="mener" pos="VERB">
<FEATURE attribute="LOCSN" value="+"/>
<FEATURE attribute="AVOIR" value="+"/>
<FEATURE attribute="ACEQUEP" value="+"/>
<FEATURE attribute="ASN" value="+"/>
<FEATURE attribute="SVINFDIR" value="+"/>
<FEATURE attribute="SVINF" value="+"/>
<FEATURE attribute="SN" value="+"/>
<FEATURE attribute="SFA" value="+"/>
<FEATURE attribute="MASC" value="+"/>
<FEATURE attribute="SG" value="+"/>
<FEATURE attribute="PARTPAS" value="+"/>
<FEATURE attribute="VERB" value="+"/>
</READING>
</TOKEN>
</NODE>
<NODE num="54" tag="PP" start="22" end="32">
<FEATURE attribute="FVPP" value="+"/>
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="PREP" value="+"/>
<FEATURE attribute="NOUN" value="+"/>
<FEATURE attribute="DET" value="+"/>
<NODE num="10" tag="PREP" start="22" end="25">
<FEATURE attribute="PREPINF" value="+"/>
<FEATURE attribute="SFPAR" value="+"/>
<FEATURE attribute="FORM" value="FPAR"/>
<FEATURE attribute="PREP" value="+"/>
<TOKEN pos="PREP" start="22" end="25">
par
<READING lemma="par" pos="PREP">
</READING>
</TOKEN>
</NODE>
<NODE num="48" tag="NP" start="26" end="32">
<FEATURE attribute="MAJ" value="+"/>
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="PROPER" value="+"/>
<FEATURE attribute="NOUN" value="+"/>
<FEATURE attribute="DET" value="+"/>
<NODE num="12" tag="DET" start="26" end="28">
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="MASC" value="+"/>
<FEATURE attribute="FEM" value="+"/>
<FEATURE attribute="SG" value="+"/>
<FEATURE attribute="DEF" value="+"/>
<FEATURE attribute="DET" value="+"/>
<TOKEN pos="DET" start="26" end="28">
l'
<READING lemma="le" pos="DET">
</READING>
</TOKEN>
</NODE>
<NODE num="14" tag="NOUN" start="28" end="32">
<FEATURE attribute="TOUTMAJ" value="+"/>
<FEATURE attribute="MAJ" value="+"/>
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="GUESSED" value="+"/>
```

```
<FEATURE attribute="MASC" value="+"/>
<FEATURE attribute="FEM" value="+"/>
<FEATURE attribute="PL" value="+"/>
<FEATURE attribute="SG" value="+"/>
<FEATURE attribute="P3" value="+"/>
<FEATURE attribute="PROPER" value="+"/>
<FEATURE attribute="NOUN" value="+"/>
<TOKEN pos="NOUN" start="28" end="32">
ESIS
<READING lemma="ESIS" pos="NOUN">
</READING>
</TOKEN>
</NODE>
</NODE>
</NODE>
<NODE num="56" tag="FV" start="33" end="36">
<FEATURE attribute="QUEP" value="+"/>
<FEATURE attribute="POURSN" value="+"/>
<FEATURE attribute="LOCSN" value="+"/>
<FEATURE attribute="IL" value="+"/>
<FEATURE attribute="EN" value="+"/>
<FEATURE attribute="DESVINF" value="+"/>
<FEATURE attribute="DESN" value="+"/>
<FEATURE attribute="CONTRESN" value="+"/>
<FEATURE attribute="AVOIR" value="+"/>
<FEATURE attribute="ASVINF" value="+"/>
<FEATURE attribute="ASN" value="+"/>
<FEATURE attribute="SVINFDIR" value="+"/>
<FEATURE attribute="SN" value="+"/>
<FEATURE attribute="SADJDESVINF" value="+"/>
<FEATURE attribute="SADJ1" value="+"/>
<FEATURE attribute="SADJ" value="+"/>
<FEATURE attribute="ARGSUBJ" value="+"/>
<FEATURE attribute="FIN" value="+"/>
<FEATURE attribute="VERB" value="+"/>
<NODE num="16" tag="VERB" start="33" end="36">
<FEATURE attribute="QUEP" value="+"/>
<FEATURE attribute="POURSN" value="+"/>
<FEATURE attribute="LOCSN" value="+"/>
<FEATURE attribute="IL" value="+"/>
<FEATURE attribute="EN" value="+"/>
<FEATURE attribute="DESVINF" value="+"/>
<FEATURE attribute="DESN" value="+"/>
<FEATURE attribute="CONTRESN" value="+"/>
<FEATURE attribute="AVOIR" value="+"/>
<FEATURE attribute="ASVINF" value="+"/>
<FEATURE attribute="ASN" value="+"/>
<FEATURE attribute="SVINFDIR" value="+"/>
<FEATURE attribute="SN" value="+"/>
<FEATURE attribute="SADJDESVINF" value="+"/>
<FEATURE attribute="SADJ1" value="+"/>
<FEATURE attribute="SADJ" value="+"/>
<FEATURE attribute="SFPOUR" value="+"/>
<FEATURE attribute="SFDE" value="+"/>
<FEATURE attribute="SFCONTRE" value="+"/>
<FEATURE attribute="SFA" value="+"/>
<FEATURE attribute="FORM" value="FETRE"/>
```

```
<FEATURE attribute="SG" value="+"/>
<FEATURE attribute="P3" value="+"/>
<FEATURE attribute="PRE" value="+"/>
<FEATURE attribute="IND" value="+"/>
<FEATURE attribute="AUX" value="+"/>
<FEATURE attribute="COPULE" value="+"/>
<FEATURE attribute="VERB" value="+"/>
<TOKEN pos="VERB" start="33" end="36">
est
<READING lemma="être" pos="VERB">
</READING>
</TOKEN>
</NODE>
</NODE>
</NODE>
<NODE num="53" tag="IV" start="37" end="44">
<FEATURE attribute="SNAINF_SO" value="+"/>
<FEATURE attribute="SE" value="+"/>
<FEATURE attribute="DESN" value="+"/>
<FEATURE attribute="AVOIR" value="+"/>
<FEATURE attribute="ACEQUEP" value="+"/>
<FEATURE attribute="ASVINF" value="+"/>
<FEATURE attribute="ASN" value="+"/>
<FEATURE attribute="SVINFDIR" value="+"/>
<FEATURE attribute="SN" value="+"/>
<FEATURE attribute="ARGCOMP" value="+"/>
<FEATURE attribute="FONC" value="FINFCOMP"/>
<FEATURE attribute="INF" value="+"/>
<FEATURE attribute="VERB" value="+"/>
<FEATURE attribute="DIR" value="+"/>
<NODE num="18" tag="PREP" start="37" end="39">
<FEATURE attribute="PREPINF" value="+"/>
<FEATURE attribute="SFDE" value="+"/>
<FEATURE attribute="FORM" value="FDE"/>
<FEATURE attribute="PREP" value="+"/>
<FEATURE attribute="DIR" value="+"/>
<TOKEN pos="PREP" start="37" end="39">
d'
<READING lemma="de" pos="PREP">
</READING>
</TOKEN>
</NODE>
<NODE num="20" tag="VERB" start="39" end="44">
<FEATURE attribute="SNAINF_SO" value="+"/>
<FEATURE attribute="SE" value="+"/>
<FEATURE attribute="DESN" value="+"/>
<FEATURE attribute="AVOIR" value="+"/>
<FEATURE attribute="ACEQUEP" value="+"/>
<FEATURE attribute="ASVINF" value="+"/>
<FEATURE attribute="ASN" value="+"/>
<FEATURE attribute="SVINFDIR" value="+"/>
<FEATURE attribute="SN" value="+"/>
<FEATURE attribute="REFLEXTYPE" value="+"/>
<FEATURE attribute="REFLEXIVE" value="+"/>
<FEATURE attribute="SFDE" value="+"/>
<FEATURE attribute="SFA" value="+"/>
<FEATURE attribute="INF" value="+"/>
```

```
<FEATURE attribute="VERB" value="+"/>
<TOKEN pos="VERB" start="39" end="44">
aider
<READING lemma="aider" pos="VERB">
</READING>
</TOKEN>
</NODE>
</NODE>
<NODE num="47" tag="NP" start="45" end="56">
<FEATURE attribute="FONC" value="FCOMP"/>
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="NOUN" value="+"/>
<FEATURE attribute="DET" value="+"/>
<NODE num="22" tag="DET" start="45" end="48"> <FEATURE
attribute="CLOSED" value="+"/>
<FEATURE attribute="MASC" value="+"/>
<FEATURE attribute="FEM" value="+"/>
<FEATURE attribute="PL" value="+"/>
<FEATURE attribute="DEF" value="+"/>
<FEATURE attribute="DET" value="+"/>
<TOKEN pos="DET" start="45" end="48">
les
<READING lemma="le" pos="DET">
</READING>
</TOKEN>
</NODE>
<NODE num="24" tag="NOUN" start="49" end="56">
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="MASC" value="+"/>
<FEATURE attribute="FEM" value="+"/>
<FEATURE attribute="PL" value="+"/>
<FEATURE attribute="P3" value="+"/>
<FEATURE attribute="NOUN" value="+"/>
<TOKEN pos="NOUN" start="49" end="56">
couples
<READING lemma="couple" pos="NOUN">
</READING>
<READING lemma="couple" pos="NOUN">
<FEATURE attribute="MASC" value="+"/>
<FEATURE attribute="PL" value="+"/>
<FEATURE attribute="P3" value="+"/>
<FEATURE attribute="NOUN" value="+"/>
</READING>
</TOKEN>
</NODE>
</NODE>
<NODE num="57" tag="BG" start="57" end="60">
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="REL" value="+"/>
<FEATURE attribute="PRON" value="+"/>
<FEATURE attribute="SCBEGIN" value="+"/>
<NODE num="26" tag="PRON" start="57" end="60">
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="MASC" value="+"/>
<FEATURE attribute="FEM" value="+"/>
<FEATURE attribute="PL" value="+"/>
<FEATURE attribute="SG" value="+"/>
```

```
<FEATURE attribute="NOM" value="+"/>
<FEATURE attribute="INT" value="+"/>
<FEATURE attribute="REL" value="+"/>
<FEATURE attribute="PRON" value="+"/>
<TOKEN pos="PRON" start="57" end="60">
Qui
<READING lemma="qui" pos="PRON">
</READING>
</TOKEN>
</NODE>
</NODE>
<NODE num="52" tag="GV" start="61" end="66">
<FEATURE attribute="QUEP" value="+"/>
<FEATURE attribute="ILY" value="+"/>
<FEATURE attribute="EN" value="+"/>
<FEATURE attribute="CONTRESN" value="+"/>
<FEATURE attribute="AVOIR" value="+"/>
<FEATURE attribute="ASVINF" value="+"/>
<FEATURE attribute="SVINFDIR" value="+"/>
<FEATURE attribute="SN" value="+"/>
<FEATURE attribute="SADJ" value="+"/>
<FEATURE attribute="ARGCOMP" value="+"/>
<FEATURE attribute="PARTPRE" value="+"/>
<FEATURE attribute="VERB" value="+"/>
<NODE num="28" tag="VERB" start="61" end="66">
<FEATURE attribute="QUEP" value="+"/>
<FEATURE attribute="ILY" value="+"/>
<FEATURE attribute="EN" value="+"/>
<FEATURE attribute="CONTRESN" value="+"/>
<FEATURE attribute="AVOIR" value="+"/>
<FEATURE attribute="ASVINF" value="+"/>
<FEATURE attribute="SVINFDIR" value="+"/>
<FEATURE attribute="SN" value="+"/>
<FEATURE attribute="SADJ" value="+"/>
<FEATURE attribute="SFCONTRE" value="+"/>
<FEATURE attribute="FORM" value="FAVOIR"/>
<FEATURE attribute="PARTPRE" value="+"/>
<FEATURE attribute="AUX" value="+"/>
<FEATURE attribute="VERB" value="+"/>
<TOKEN pos="VERB" start="61" end="66">
ayant
<READING lemma="avoir" pos="VERB">
</READING>
</TOKEN>
</NODE>
</NODE>
<NODE num="46" tag="NP" start="67" end="82">
<FEATURE attribute="DESVINF" value="+"/>
<FEATURE attribute="DESN" value="+"/>
<FEATURE attribute="ASVINF" value="+"/>
<FEATURE attribute="SVINFDIR" value="+"/>
<FEATURE attribute="FONC" value="FCOMP"/>
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="NOUN" value="+"/>
<FEATURE attribute="DET" value="+"/>
<NODE num="30" tag="DET" start="67" end="70">
<FEATURE attribute="FORM" value="FDES"/>
```

```
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="MASC" value="+"/>
<FEATURE attribute="FEM" value="+"/>
<FEATURE attribute="PL" value="+"/>
<FEATURE attribute="INDEF" value="+"/>
<FEATURE attribute="DET" value="+"/>
<TOKEN pos="DET" start="67" end="70">
des
<READING lemma="un" pos="DET">
</READING>
</TOKEN>
</NODE>
<NODE num="32" tag="NOUN" start="71" end="82">
<FEATURE attribute="DESVINF" value="+"/>
<FEATURE attribute="DESN" value="+"/>
<FEATURE attribute="ASVINF" value="+"/>
<FEATURE attribute="SVINFDIR" value="+"/>
<FEATURE attribute="SFDE" value="+"/>
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="FEM" value="+"/>
<FEATURE attribute="PL" value="+"/>
<FEATURE attribute="P3" value="+"/>
<FEATURE attribute="NOUN" value="+"/>
<TOKEN pos="NOUN" start="71" end="82">
difficultés
<READING lemma="difficulté" pos="NOUN">
</READING>
</TOKEN>
</NODE>
</NODE>
<NODE num="51" tag="IV" start="83" end="90">
<FEATURE attribute="QUEP" value="+"/>
<FEATURE attribute="ILY" value="+"/>
<FEATURE attribute="EN" value="+"/>
<FEATURE attribute="CONTRESN" value="+"/>
<FEATURE attribute="AVOIR" value="+"/>
<FEATURE attribute="ASVINF" value="+"/>
<FEATURE attribute="SVINFDIR" value="+"/>
<FEATURE attribute="SN" value="+"/>
<FEATURE attribute="SADJ" value="+"/>
<FEATURE attribute="ARGCOMP" value="+"/>
<FEATURE attribute="FONC" value="FINFCOMP"/>
<FEATURE attribute="INF" value="+"/>
<FEATURE attribute="VERB" value="+"/>
<FEATURE attribute="DIR" value="+"/>
<NODE num="34" tag="PREP" start="83" end="84">
<FEATURE attribute="PREPINF" value="+"/>
<FEATURE attribute="SFA" value="+"/>
<FEATURE attribute="FORM" value="FA"/>
<FEATURE attribute="PREP" value="+"/>
<FEATURE attribute="DIR" value="+"/>
<TOKEN pos="PREP" start="83" end="84">
à
<READING lemma="à" pos="PREP">
</READING>
</TOKEN>
</NODE>
```

```
<NODE num="36" tag="VERB" start="85" end="90">
<FEATURE attribute="QUEP" value="+"/>
<FEATURE attribute="ILY" value="+"/>
<FEATURE attribute="EN" value="+"/>
<FEATURE attribute="CONTRESN" value="+"/>
<FEATURE attribute="AVOIR" value="+"/>
<FEATURE attribute="ASVINF" value="+"/>
<FEATURE attribute="SVINFDIR" value="+"/>
<FEATURE attribute="SN" value="+"/>
<FEATURE attribute="SADJ" value="+"/>
<FEATURE attribute="SFCONTRE" value="+"/>
<FEATURE attribute="FORM" value="FAVOIR"/>
<FEATURE attribute="INF" value="+"/>
<FEATURE attribute="AUX" value="+"/>
<FEATURE attribute="VERB" value="+"/>
<TOKEN pos="VERB" start="85" end="90">
avoir
<READING lemma="avoir" pos="VERB">
</READING> </TOKEN> </NODE> </NODE>
<NODE num="45" tag="NP" start="91" end="100">
<FEATURE attribute="FONC" value="FCOMP"/>
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="NOUN" value="+"/>
<FEATURE attribute="DET" value="+"/>
<NODE num="38" tag="DET" start="91" end="93">
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="MASC" value="+"/>
<FEATURE attribute="SG" value="+"/>
<FEATURE attribute="INDEF" value="+"/>
<FEATURE attribute="DET" value="+"/>
<TOKEN pos="DET" start="91" end="93">
un
<READING lemma="un" pos="DET">
</READING>
</TOKEN>
</NODE>
<NODE num="40" tag="NOUN" start="94" end="100">
<FEATURE attribute="CLOSED" value="+"/>
<FEATURE attribute="MASC" value="+"/>
<FEATURE attribute="FEM" value="+"/>
<FEATURE attribute="SG" value="+"/>
<FEATURE attribute="P3" value="+"/>
<FEATURE attribute="NOUN" value="+"/>
<TOKEN pos="NOUN" start="94" end="100">
enfant
<READING lemma="enfant" pos="NOUN">
</READING>
</TOKEN>
</NODE>
</NODE>
<NODE num="42" tag="SENT" start="100" end="101">
<FEATURE attribute="SENT" value="+"/>
<TOKEN pos="SENT" start="100" end="101">
.
<READING lemma="." pos="SENT">
</READING>
</TOKEN>
```

```
</NODE>
</NODE>
<DEPENDENCY name="SUBJ">
<PARAMETER ind="0" num="16" word="être"/>
<PARAMETER ind="1" num="2" word="but"/>
</DEPENDENCY>
<DEPENDENCY name="OBJ">
<PARAMETER ind="0" num="28" word="avoir"/>
<PARAMETER ind="1" num="32" word="difficulté"/>
</DEPENDENCY>
<DEPENDENCY name="OBJ">
<PARAMETER ind="0" num="36" word="avoir"/>
<PARAMETER ind="1" num="40" word="enfant"/>
</DEPENDENCY>
<DEPENDENCY name="OBJ">
<PARAMETER ind="0" num="20" word="aider"/>
<PARAMETER ind="1" num="24" word="couple"/>
</DEPENDENCY>
<DEPENDENCY name="DEEPSUBJ">
<PARAMETER ind="0" num="8" word="mener"/>
<PARAMETER ind="1" num="14" word="ESIS"/>
</DEPENDENCY>
<DEPENDENCY name="VMOD">
<FEATURE attribute="POSIT1" value="+"/>
<PARAMETER ind="0" num="8" word="mener"/>
<PARAMETER ind="1" num="14" word="ESIS"/>
</DEPENDENCY>
<DEPENDENCY name="VMOD">
<PARAMETER ind="0" num="16" word="être"/>
<PARAMETER ind="1" num="20" word="aider"/>
</DEPENDENCY>
<DEPENDENCY name="NMOD">
<FEATURE attribute="POSIT1" value="+"/>
<PARAMETER ind="0" num="6" word="projet"/>
<PARAMETER ind="1" num="8" word="mener"/>
</DEPENDENCY>
<DEPENDENCY name="NMOD">
<FEATURE attribute="POSIT1" value="+"/>
<PARAMETER ind="0" num="2" word="but"/>
<PARAMETER ind="1" num="6" word="projet"/>
</DEPENDENCY>
<DEPENDENCY name="NMOD">
<PARAMETER ind="0" num="32" word="difficulté"/>
<PARAMETER ind="1" num="36" word="avoir"/>
</DEPENDENCY>
<DEPENDENCY name="PREPOBJ">
<PARAMETER ind="0" num="14" word="ESIS"/>
<PARAMETER ind="1" num="10" word="par"/>
</DEPENDENCY> <DEPENDENCY name="PREPOBJ">
<PARAMETER ind="0" num="6" word="projet"/>
<PARAMETER ind="1" num="4" word="de"/>
</DEPENDENCY> <DEPENDENCY name="PREPOBJ">
<PARAMETER ind="0" num="36" word="avoir"/>
<PARAMETER ind="1" num="34" word="à"/>
</DEPENDENCY> <DEPENDENCY name="PREPOBJ">
<PARAMETER ind="0" num="20" word="aider"/>
<PARAMETER ind="1" num="18" word="de"/>
```

```
</DEPENDENCY>
<DEPENDENCY name="DETERM">
<PARAMETER ind="0" num="40" word="enfant"/>
<PARAMETER ind="1" num="38" word="un"/>
</DEPENDENCY>
<DEPENDENCY name="DETERM">
<PARAMETER ind="0" num="32" word="difficulté"/>
<PARAMETER ind="1" num="30" word="un"/>
</DEPENDENCY>
<DEPENDENCY name="DETERM">
<FEATURE attribute="DEF" value="+"/>
<PARAMETER ind="0" num="24" word="couple"/>
<PARAMETER ind="1" num="22" word="le"/>
</DEPENDENCY>
<DEPENDENCY name="DETERM">
<FEATURE attribute="DEF" value="+"/>
<PARAMETER ind="0" num="14" word="ESIS"/>
<PARAMETER ind="1" num="12" word="le"/>
</DEPENDENCY>
<DEPENDENCY name="DETERM">
<FEATURE attribute="DEF" value="+"/>
<PARAMETER ind="0" num="2" word="but"/>
<PARAMETER ind="1" num="0" word="le"/>
</DEPENDENCY>
</LUNIT>
</XIPRESULT>

C:\xip-9.59-5\sample\grammar\sample8>if exist errors.err del error.err
```

Appendix B. Example Memodata File

Memodata output for word *approche*. the file has been reduced from its original size of about 7 pages; various headers of information on the content are set in bold character.


```
<RESULTS><DEF><E L='fr'><W>approche</W><SW
ENC='n'>006100700070007200 6F006300680065</SW><P>n.f.</P></E><DF N='1'><W>fait
de se rapprocher de quelque chose.</W></DF></DEF><DEF><E
L='fr'><W>approché</W><SW
ENC='n'>006100700070007200 6F0063006800E9</SW><P>adj.</P></E><DF
N='1'><W>approximatif, proche d'une valeur exacte.</W></DF></DEF><TRA><E
L='fr'><W>approche</W><SW
ENC='n'>006100700070007200 6F006300680065</SW><P>n.f.</P></E><WD L='nl'
W='1'><W L='nl'>aanpak</W><SW ENC='n'>00610061006E00700061006B</SW></WD><WD
L='nl' W='1'><W L='nl'>aanvalsplan</W><SW
ENC='n'>00610061006E00760061006C00730070006C0061006E</SW></WD><WD L='ro'
W='1'><W L='ro'>abordare</W><SW
ENC='n'>00610062006F00720064006100720065</SW></WD><WD L='es' W='1'><W
L='es'>acceso</W><SW ENC='n'>006100630063006500730 06F</SW></WD><WD L='it'
W='1'><W L='it'>accesso</W><SW
ENC='n'>006100630063006500730073006F</SW></WD><WD L='no' W='1'><W
L='no'>ankomst</W><SW ENC='n'>0061006E006B006F006D00730074</SW></WD><WD
L='sv' W='1'><W L='sv'>annalkande</W><SW
ENC='n'>0061006E006E0061006C006B0061006E00640065</SW></WD><WD L='en' W='5'><W
L='en'>approach</W><SW ENC='n'>006100700070007200 6F006100630068</SW></WD><WD
L='nl' W='1'><W L='nl'>approach</W><SW [ … ]
[ … ]
</TRA><SYN><E L='fr'><W>approche</W><SW
ENC='n'>006100700070007200 6F006300680065</SW><P>n.f.</P></E><WD L='fr'
W='3'><W L='fr'>abord</W><SW ENC='n'>00610062006F00720064</SW></WD><WD L='fr'
W='2'><W L='fr'>accès</W><SW ENC='n'>006100630063006300E80073</SW></WD><WD L='fr'
W='1'><W L='fr'>accouplement</W><SW
ENC='n'>006100630063006F00750070006C0065006D0065006E0074</SW></WD><WD L='fr'
W='1'><W L='fr'>alentours</W><SW
ENC='n'>0061006C0065006E0074006F007500720073</SW></WD><WD L='fr' W='1'><W
L='fr'>apparition</W><SW
ENC='n'>006100700070006100720069007400690 06F006E</SW></WD><WD L='fr' W='1'><W
L='fr'>approchement</W><SW
ENC='n'>006100700070007200 6F006300680065006D0065006E0074</SW></WD><WD L='fr'
W='2'><W L='fr'>arrivée</W><SW
ENC='n'>006100720072006900760 0E90065</SW></WD><WD L='fr' W='1'><W
L='fr'>attaque</W><SW ENC='n'>006100740074006100710 0750065</SW></WD><WD
L='fr' W='1'><W L='fr'>avance</W><SW
ENC='n'>006100760061006E00630065</SW></WD><WD L='fr' W='3'><W
L='fr'>cheminement</W><SW
ENC='n'>006300680065006D0069006E0065006D0065006E0074</SW></WD><WD L='fr'
W='1'><W L='fr'>contact</W><SW
ENC='n'>0063006F006E0074006100630074</SW></WD><WD L='fr' W='3'><W
```

L='fr'>démarche</W><SW ENC='n'>006400E9006D00610072006300680065</SW></WD><WD
L='fr' W='1'><W L='fr'>entrée</W><SW
ENC='n'>0065006E0074007200E90065</SW></WD><WD L='fr' W='1'><W
L='fr'>environs</W><SW ENC='n'>0065006E007600690072006F006E0073</SW></WD><WD
L='fr' W='1'><W L='fr'>fréquentation</W><SW
ENC='n'>0066007200E900710075006500 6E00740061007400 69006F006E</SW></WD><WD
L='fr' W='3'><W L='fr'>imminence</W><SW
ENC='n'>0069006D006D0069006E0065006E00630065</SW></WD><WD L='fr' W='1'><W
L='fr'>manière d'aborder</W><SW
ENC='n'>006D0061006E006900E80072006500200064002700610062006F0072006400650072<
/SW></WD><WD L='fr' W='1'><W L='fr'>manœuvre</W><SW
ENC='n'>006D0061006E015300750076007200 65</SW></WD><WD L='fr' W='1'><W
L='fr'>méthode</W><SW ENC='n'>006D00E900740068006F00640065</SW></WD><WD
L='fr' W='1'><W L='fr'>ombre</W><SW ENC='n'>006F006D006200720065</SW></WD><WD
L='fr' W='1'><W L='fr'>parages</W><SW
ENC='n'>0070006100720061006700650073</SW></WD><WD L='fr' W='1'><W
L='fr'>procédé</W><SW ENC='n'>00700072006F006300E9006400E9</SW></WD><WD
L='fr' W='1'><W L='fr'>procédure</W><SW
ENC='n'>00700072006F006300E90064007500720065</SW></WD><WD L='fr' W='3'><W
L='fr'>proximité</W><SW
ENC='n'>00700072006F00780069006D0069007400E9</SW></WD><WD L='fr' W='1'><W
L='fr'>rapprochement</W> […]
[…]
</SYN><**PHR**><W>APPROCHE</W><WD L='fr'><W>**chasse à l'approche**</W><SW
ENC='n'>006300680061007300730065002000E00020006C00270061007000700072006F00630
0680065</SW></WD><WD L='fr'><W>coup d'approche</W><SW
ENC='n'>0063006F007500700020006400270061007000700072006F006300680065</SW></WD
><WD L='fr'><W>**lunette d'approche**</W><SW
ENC='n'>006C0075006E006500740074006500200064002700610070007000 72006F006300680
065</SW></WD><WD L='fr'><W>niveau d'approche</W><SW
ENC='n'>006E006900760065006100750020006400270061007000700072006F006300680065<
/SW></WD></PHR><**DEF**><E L='fr'><W>**approcher**</W><SW
ENC='n'>006100700070007 2006F0063006800650072</SW><P>v.</P></E><DF
N='1'><W>aller plus près.</W><L>figuré</L></DF></DEF><TRA><E
L='fr'><W>approcher</W><SW
ENC='n'>006100700070007 2006F0063006800650072</SW><P>v.</P></E><WD L='ro'
W='2'><W L='ro'>[se] apropia</W><SW
ENC='n'>005B00730065005D00200061007000 72006F007000690061</SW></WD><WD L='nl'
W='1'><W L='nl'>aankomen</W><SW
ENC='n'>00610061006E006B006F006D0065006E</SW></WD><WD L='it' W='1'><W
L='it'>accagliarsi</W><SW
ENC='n'>0061006300630061006700 6C006900610072007300 69</SW></WD><WD L='it'
W='2'><W L='it'>accalcare</W><SW
ENC='n'>0061006300630061006C0063006100720065</SW></WD><WD L='it' W='3'><W
L='it'>accostare</W><SW
ENC='n'>006100630063006F0073007400610072006 5</SW></WD><WD L='pt' W='3'><W
L='pt'>acercar</W><SW ENC='n'>00610063006500720063006100 72</SW></WD><WD
L='es' W='1'><W L='es'>acercar</W><SW
ENC='n'>0061006300650072006300610072</SW></WD><WD L='es' W='1'><W
L='es'>acercarse a</W><SW
ENC='n'>006100630065007200630061007200730065002000 61</SW></WD><WD L='de'
W='1'><W L='de'>annähern</W><SW
ENC='n'>0061006E006E00E400680065 0072006E</SW></WD><WD L='it' W='1'><W
L='it'>appressare</W><SW
ENC='n'>00610070007000720065007300730061007200 65</SW></WD><WD L='en' W='2'><W
L='en'>**approach**</W><SW ENC='n'>00610070007000 72006F006100630068</SW></WD><WD
L='en' W='2'><W L='en'>**approximate**</W> […]

```
[ … ]
</TRA><SYN><E L='fr'><W>approcher</W><SW
ENC='n'>00610070007000720006F0063006800650072</SW><P>v.</P></E><WD L='fr'
W='4'><W L='fr'>aborder</W><SW
ENC='n'>00610062006F0072006400650072</SW></WD><WD L='fr' W='1'><W
L='fr'>accoster</W><SW ENC='n'>006100630063006F0073007400650072</SW></WD><WD
L='fr' W='1'><W L='fr'>aller vers</W><SW
ENC='n'>0061006C006C006500720020007600650072 0073</SW></WD><WD L='fr' W='3'><W
L='fr'>arriver</W><SW ENC='n'>00610072007200690076006500720073</SW></WD><WD
L='fr' W='2'><W L='fr'>avancer</W><SW
ENC='n'>006100760061006E006300650072</SW></WD><WD L='fr' W='1'><W
L='fr'>conduire vers</W><SW
ENC='n'>0063006F006E006400750069007200650020007600650072 0073</SW></WD><WD
L='fr' W='1'><W L='fr'>confiner à</W><SW
ENC='n'>0063006F006E00660069006E0065007200200 0E0</SW></WD><WD L='fr' W='1'><W
L='fr'>contacter</W><SW
ENC='n'>0063006F006E00740061006300740065 0072</SW></WD><WD L='fr' W='2'><W
L='fr'>côtoyer</W><SW ENC='n'>006300F40074006F007900650072</SW></WD><WD
L='fr' W='1'><W L='fr'>donner</W><SW
ENC='n'>0064006F006E006E00650072</SW></WD><WD L='fr' W='1'><W
L='fr'>effleurer</W><SW
ENC='n'>006500660066006C0065007500720065 0072</SW></WD><WD L='fr' W='1'><W
L='fr'>égaler</W><SW ENC='n'>00E900670061006C00650072</SW></WD><WD L='fr'
W='1'><W L='fr'>égaliser</W><SW
ENC='n'>00E900670061006C006900730065 0072</SW></WD><WD L='fr' W='1'><W
L='fr'>entourer</W><SW ENC='n'>0065006E0074006F007500720065 0072</SW></WD><WD
L='fr' W='1'><W L='fr'>être proche</W><SW
ENC='n'>00EA007400720065002000700072006F006300680065</SW></WD><WD L='fr'
W='1'><W L='fr'>fréquenter</W><SW
ENC='n'>0066007200E900710075006500 6E007400650072</SW></WD><WD L='fr' W='1'><W
L='fr'>friser</W><SW ENC='n'>006600720069007300650072</SW></WD><WD L='fr'
W='1'><W L='fr'>gagner de vitesse</W><SW
ENC='n'>006700610067006E0065007200200064006500200076006900740065007300730065<
/SW></WD><WD L='fr' W='2'><W L='fr'>joindre</W><SW
ENC='n'>006A006F0069006E006400720065</SW></WD><WD L='fr' W='4'><W
L='fr'>rapprocher</W><SW
ENC='n'>0072006100700070007200 6F0063006800650072</SW></WD><WD L='fr' W='1'><W
L='fr'>regarder</W><SW ENC='n'>00720065006700610072006400650072</SW></WD><WD
L='fr' W='1'><W L='fr'>ressembler</W><SW
ENC='n'>0072006500730073006500 6D0062006C00650072</SW></WD><WD L='fr' W='1'><W
L='fr'>rivaliser</W><SW
ENC='n'>0072006900760061006C006900730065 0072</SW></WD><WD L='fr' W='1'><W
L='fr'>s'apparenter</W><SW
ENC='n'>0073002700610070007000610072006500 6E007400650072</SW></WD><WD L='fr'
W='1'><W L='fr'>s'approcher</W><SW
ENC='n'>0073002700610070007000720 06F0063006800650072</SW></WD><WD L='fr'
W='2'><W L='fr'>s'avancer</W><SW
ENC='n'>007300270061007600610 06E006300650072</SW></WD><WD L='fr' W='1'><W
L='fr'>se dessiner</W><SW
ENC='n'>0073006500200064006500730073006900 6E00650072</SW></WD><WD L='fr'
W='1'><W L='fr'>se porter</W><SW
ENC='n'>007300650020007000 6F007200740065 0072</SW></WD><WD L='fr' W='2'><W
L='fr'>se rapprocher</W><SW
ENC='n'>0073006500200072006100700070007200 6F0063006800650072</SW></WD><WD
L='fr' W='1'><W L='fr'>tenir de</W><SW
ENC='n'>00740065006E006900720020006400 65</SW></WD><WD L='fr' W='1'><W
L='fr'>toucher</W><SW ENC='n'>0074006F007500630068006500 72</SW></WD><WD
```

L='fr' W='2'><W L='fr'>venir</W><SW
ENC='n'>00760065006E00690072</SW></WD></SYN><**PHR**><W>**APPROCHER**</W><WD
L='fr'><W>**approcher de**</W><SW
ENC='n'>006100700070007200 6F006300680065007200200064006500 20006C0027006100750
0740065006C</SW></WD><WD L='fr'><W>s'approcher</W><SW
ENC='n'>00730027006100700070007200 6F006300680065007200</SW></WD><WD
L='fr'><W>**s'approcher de**</W><SW
ENC='n'>00730027006100700070007200 6F006300680065007200200064006500 65</SW></WD></P
HR><DER><E L='fr'><W>approcher</W><SW
ENC='n'>006100700070007200 6F006300680065007200</SW><P>v.</P></E><WD L='fr'
K='DOWN'><W L='fr'>approchable</W><SW
ENC='n'>006100700070007200 6F006300680061006200 6C0065</SW><P>adj.</P></WD><WD
L='fr' K='DOWN'><W L='fr'>approchant</W><SW
ENC='n'>006100700070007200 6F006300680061006E0074</SW><P>adj.</P></WD></DER><C
ONTRIB><CTB LNK='http://www.memodata.com'>Memodata</CTB><CTB
LNK='http://wordnet.princeton.edu/'>WordNet 2.0 Copyright © 2003 by Princeton
University. All rights reserved.</CTB><CTB
LNK='http://www.ceid.upatras.gr/Balkanet/'>Balkanet</CTB></CONTRIB></RESULTS>

Appendix C. Processing Example

1.

Here, the program stops after having detected a break, which is correctly analyzed as non-necessary function word.

```
coming into generate_dependancies for sentence : le but du projet mené
par l'ESIS est d'aider les couples qui ayant des difficultés à avoir un
enfant.


Phrase Ref =>
00_rechercher 02_un 04_difference 06_de 08_tendance 10_de 12_trouble
14_de 16_fecondite 18_entre 20_le 22_pays 24_en 26_utiliser 28_un
30_methode 32_standardiser
```

Follows : XIP analysis formatting

```
Dependances Ref =>

OBJ(00_rechercher,04_difference)
OBJ(26_utiliser,30_methode)
VMOD(00_rechercher,22_pays)
VMOD(26_utiliser,00_rechercher)
NMOD(04_difference,08_tendance)
NMOD(04_difference,12_trouble)
NMOD(12_trouble,16_fecondite)
NMOD(30_methode,32_standardiser)
PREPOBJ(08_tendance,06_de)
PREPOBJ(12_trouble,10_de)
PREPOBJ(16_fecondite,14_de)
PREPOBJ(22_pays,18_entre)
PREPOBJ(26_utiliser,24_en)
DETERM(04_difference,02_un)
```

```
DETERM(22_pays,20_le)
DETERM(30_methode,28_un)


Lemmes Ref =>

00_VERB_rechercher
02_DET_un
04_NOUN_difference
06_PREP_de
08_NOUN_tendance
10_PREP_de
12_NOUN_trouble
14_PREP_de
16_NOUN_fecondite
18_PREP_entre
20_DET_le
22_NOUN_pays
24_PREP_en
26_VERB_utiliser
28_DET_un
30_NOUN_methode
32_VERB_standardiser


Phrase S =>
00_le 02_but 04_de 06_projet 08_mener 10_par 12_le 16_etre 18_de
20_aider 22_le 24_couple 26_qui 28_avoir 30_un 32_difficulte 34_a
36_avoir 38_un 40_enfant


Dependances S =>

SUBJ(16_etre,02_but)
DEEPSUBJ(08_mener,14_ESIS)
OBJ(20_aider,24_couple)
OBJ(28_avoir,32_difficulte)
OBJ(36_avoir,40_enfant)
VMOD(08_mener,14_ESIS)
VMOD(16_etre,20_aider)
NMOD(02_but,06_projet)
NMOD(06_projet,08_mener)
NMOD(32_difficulte,36_avoir)
PREPOBJ(06_projet,04_de)
PREPOBJ(14_ESIS,10_par)
PREPOBJ(20_aider,18_de)
PREPOBJ(36_avoir,34_a)
DETERM(02_but,00_le)
DETERM(14_ESIS,12_le)
DETERM(24_couple,22_le)
DETERM(32_difficulte,30_un)
DETERM(40_enfant,38_un)


Lemmes S =>

00_DET_le
```

```
02_NOUN_but
04_PREP_de
06_NOUN_projet
08_VERB_mener
10_PREP_par
12_DET_le
16_VERB_etre
18_PREP_de
20_VERB_aider
22_DET_le
24_NOUN_couple
26_PRON_qui
28_VERB_avoir
30_DET_un
32_NOUN_difficulte
34_PREP_a
36_VERB_avoir
38_DET_un
40_NOUN_enfant


Phrase Q =>
00_quel 02_etre 04_le 06_but 08_de 10_projet 12_mener 14_par 16_le
18_ESIS


Dependances Q =>

SUBJ(02_etre,00_quel)
DEEPSUBJ(12_mener,18_ESIS)
OBJ(02_etre,06_but)
VMOD(12_mener,18_ESIS)
NMOD(06_but,10_projet)
NMOD(10_projet,12_mener)
PREPOBJ(10_projet,08_de)
PREPOBJ(18_ESIS,14_par)
DETERM(06_but,04_le)
DETERM(18_ESIS,16_le)


lemmes en LemmaToIndex : PRON_quel
VERB_etre
DET_le
NOUN_but
PREP_de
NOUN_projet
VERB_mener
PREP_par
DET_le
NOUN_ESIS
SENT_?

Lemmes Q =>

00_PRON_quel
02_VERB_etre
04_DET_le
```

```
06_NOUN_but
08_PREP_de
10_NOUN_projet
12_VERB_mener
14_PREP_par
16_DET_le
18_NOUN_ESIS
```

Select admissible connectors to answer the question

```
pour la question, prepo admissible : nil

pas de fautes d'orthographe
```

Detect breaks and excluded words

```
l'element 0 0 est 00_le
l'element 0 1 est 02_but
l'element 0 2 est 04_de
l'element 0 3 est 06_projet
l'element 0 4 est 08_mener
l'element 0 5 est 14_ESIS
l'element 0 6 est 16_etre
l'element 0 7 est 18_de
l'element 0 8 est 20_aider
l'element 0 9 est 22_le
l'element 0 10 est 24_couple

rupture au point 10 : 24_couple|28_avoir

l'element 1 0 est 28_avoir
l'element 1 1 est 30_un
l'element 1 2 est 32_difficulte
l'element 1 3 est 34_a
l'element 1 4 est 36_avoir
l'element 1 5 est 38_un
l'element 1 6 est 40_enfant

point de rupture en premiere sortie : 24_couple|28_avoir

exclus de l'analyse de dependance : 26_qui
```

Keep track of the break in the tree analysis

```
apres traitement phrasesMotif 2 :
@NP___02@PP___04@NP___06___08@PP___10@NP___14@FV___16@IV___18___20@NP__
_24@@@@@BG___26@GV___28@NP___32@IV___34___36@NP___40
```

Synthesis of analysis information; this will be used for:

- long distance attachments
- break and holes resolution

```
categories all_cats pour S :
0@@@@NP_DETERM02_00_le_SUBJ16_02_but@PP_PREP04_04_du@NMOD02_06_projet_N
MOD06_08_mene@PP_PREP10_10_par@DETERM14_12_l'_VMOD08_14_ESIS@FV_16_est@
IV_PREP18_18_d'_VMOD16_20_aider@NP_DETERM24_22_les_OBJ20_24_couples@BG_
PRON26_qui@GV_28_ayant@NP_DETERM32_30_des_OBJ28_32_difficultes@IV_PREP3
4_34_a_NMOD32_36_avoir@NP_DETERM40_38_un_OBJ36_40_enfant@SENT_.


categories all_cats de Ref :
0@@@@IV_VMOD26_00_rechercher@NP_DETERM04_02_une_OBJ00_04_difference@PP_
PREP06_06_de@NMOD04_08_tendance@PP_PREP10_10_des@NMOD04_12_troubles@PP_
PREP14_14_de@NMOD12_16_fecondite@PP_PREP18_18_entre@DETERM22_20_les_VMO
D00_22_pays@GV_PREP24_24_en_26_utilisant@NP_DETERM30_28_une_OBJ26_30_me
thode_NMOD30_32_standardisee@SENT_.
```

After Heuristics, Agreement and Tense errors detection

```
le pronom relatif "qui" doit etre sujet du verbe

erreur d'accord : pas d'erreur d'accord

erreur de temps : pas d'erreur de temps
```

After Syntactic errors resolution

```
il est possible de connecter NP_DETERM24_22_les_OBJ20_24_couples_NOUN à
GV_28_ayant_VERB sans connecteur spécifique
```

Finally, the program stops

```
on va analyser $phrasesMotif :
@NP___02@PP___04@NP___06___08@PP___10@NP___14@FV___16@IV___18___20@NP___24@@@
@@BG___26@@@GV___28@NP___32@IV___34___36@NP___40

rupture d'analyse apres le mot : NOUN_couple

rupture d'analyse apres le mot : PRON_qui
```

2.

The program correctly identifies S as a paraphrase of REF. But, it still outputs some elements of S divergent from REF: the fact that the student cut the complement of the `ADJ_semblable` (REF) / `VERB_approcher` (S) governor(s) and attached this complement's modifier instead.

```
Phrase Ref =>
00_le 02_rat 04_etre 06_un 08_animal 10_qui 12_posseder 14_un 16_systeme
18_cardio-vasculaire 20_tres 22_semblable 24_a 26_celui 28_de 30_le 32_homme


Dependances Ref =>

SUBJ(04_etre,02_rat)
```

```
SUBJ(12_posseder,10_qui)
OBJ(04_etre,08_animal)
OBJ(12_posseder,16_systeme)
COREF(08_animal,10_qui)
NMOD(16_systeme,18_cardio-vasculaire)
NMOD(16_systeme,22_semblable)
NMOD(26_celui,32_homme)
ADJMOD(22_semblable,20_tres)
ADJMOD(22_semblable,26_celui)
PREPOBJ(26_celui,24_a)
PREPOBJ(32_homme,28_de)
DETERM(02_rat,00_le)
DETERM(08_animal,06_un)
DETERM(16_systeme,14_un)
DETERM(32_homme,30_le)
CONNECT(12_posseder,10_qui)


Lemmes Ref =>

00_DET_le
02_NOUN_rat
04_VERB_etre
06_DET_un
08_NOUN_animal
10_PRON_qui
12_VERB_posseder
14_DET_un
16_NOUN_systeme
18_ADJ_cardio-vasculaire
20_ADV_tres
22_ADJ_semblable
24_PREP_a
26_PRON_celui
28_PREP_de
30_DET_le
32_NOUN_homme


Phrase S =>
00_le 02_systeme 04_cardio-vasculaire 06_de 08_un 10_rat 12_se 14_approcher
16_a 18_un 20_personne 22_humain


Dependances S =>

SUBJ(14_approcher,02_systeme)
OBJ(14_approcher,12_se)
VMOD(14_approcher,20_personne)
NMOD(02_systeme,04_cardio-vasculaire)
NMOD(02_systeme,10_rat)
NMOD(20_personne,22_humain)
PREPOBJ(10_rat,06_de)
PREPOBJ(20_personne,16_a)
DETERM(02_systeme,00_le)
DETERM(10_rat,08_un)
DETERM(20_personne,18_un)
```

```
REFLEX(14_approcher,12_se)


Lemmes S =>

00_DET_le
02_NOUN_systeme
04_ADJ_cardio-vasculaire
06_PREP_de
08_DET_un
10_NOUN_rat
12_PRON_se
14_VERB_approcher
16_PREP_a
18_DET_un
20_NOUN_personne
22_ADJ_humain


Phrase Q =>
00_pourquoi 02_pouvoir 04_on 06_tirer 08_le 10_meme 12_conclusion 14_pour
16_le 18_homme 20_et 22_pour 24_le 26_rat


Dependances Q =>


SUBJ(02_pouvoir,04_on)
OBJ(02_pouvoir,06_tirer)
OBJ(06_tirer,12_conclusion)
VMOD(06_tirer,18_homme)
VMOD(06_tirer,26_rat)
NMOD(12_conclusion,10_meme)
COORDITEMS(18_homme,26_rat,20_et)
PREPOBJ(18_homme,14_pour)
PREPOBJ(26_rat,22_pour)
DETERM(12_conclusion,08_le)
DETERM(18_homme,16_le)
DETERM(26_rat,24_le)
CONNECT(02_pouvoir,00_pourquoi)


Lemmes Q =>

00_ADV_pourquoi
02_VERB_pouvoir
04_PRON_on
06_VERB_tirer
08_DET_le
10_ADJ_meme
12_NOUN_conclusion
14_PREP_pour
16_DET_le
18_NOUN_homme
20_COORD_et
22_PREP_pour
24_DET_le
```

```
26_NOUN_rat

pour la question, prepo admissible : PREP_acausede

pour la question, prepo admissible : PREP_dufaitde

pour la question, prepo admissible : CONJ_dufaitque

pour la question, prepo admissible : CONJ_parceque

pour la question, prepo admissible : PREP_pour


pas de fautes d'orthographe


l'element 0 0 est 00_le
l'element 0 1 est 02_systeme
l'element 0 2 est 04_cardio-vasculaire
l'element 0 3 est 06_de
l'element 0 4 est 08_un
l'element 0 5 est 10_rat
l'element 0 6 est 12_se
l'element 0 7 est 14_approcher
l'element 0 8 est 16_a
l'element 0 9 est 18_un
l'element 0 10 est 20_personne
l'element 0 11 est 22_humain


apres traitement phrasesMotif 2 :
@NP___02@AP___04@PP___06@NP___10@FV___12___14@PP___16@NP___20@AP___22


categories all_cats pour S :

0@@@@AP_NMOD02_04_cardio-vasculaire@PP_PREP06_06_d'@DETERM10_08_un_NMOD
02_10_rat@FV_PRON12_REFLEX14_12_s'_14_approche@PP_PREP16_16_a@DETERM20_18_une
_VMOD14_20_personne@AP_NMOD20_22_humain@SENT_.


categories all_cats de Ref :

0@@@@NP_DETERM02_00_le_SUBJ04_02_rat@FV_04_est@NP_DETERM08_06_un_OBJ04_08_ani
mal@SC@BG_PRON10_CONNECT12_10_qui@FV_12_possede@NP_DETERM16_14_un_OBJ12_16_sy
steme@AP_NMOD16_18_cardio-vasculaire@AP@ADV20_ADJMOD22_20_
tres_NMOD16_22_semblable@PP_PREP24_24_a@PRON26_ADJMOD22_26_celui@PP_PREP28_28
_de@DETERM32_30_l'_NMOD26_32_homme@SENT_.


erreur d'accord : |
pas d'accord de genre entre personne et humain


erreur de temps : pas d'erreur de temps
```

Synonymy Detection and Replacement in the items of analysis: lemma and dependencies

```
pas d'antonymes

pas de synonymes entre Q et S

synonymes : VERB_approcher(de)|ADJ_semblable

motif syntaxique de reformulation : V-->ADJ

synonymes : NOUN_personne|NOUN_homme


apres remplacement des syns :
SUBJ(14_approcher,02_systeme)
OBJ(14_approcher,12_se)
VMOD(14_approcher,20_homme)
NMOD(02_systeme,04_cardio-vasculaire)
NMOD(02_systeme,10_rat)
NMOD(20_homme,22_humain)
PREPOBJ(10_rat,06_de)
PREPOBJ(20_homme,16_a)
DETERM(02_systeme,00_le)
DETERM(10_rat,08_un)
DETERM(20_homme,18_un)
REFLEX(14_approcher,12_se)

et lemmes :
00_DET_le
02_NOUN_systeme
04_ADJ_cardio-vasculaire
06_PREP_de
08_DET_un
10_NOUN_rat
12_PRON_se
14_VERB_approcher
16_PREP_a
18_DET_un
20_NOUN_homme
22_ADJ_humain
```

Phrase Mapping

```
resultat de mapPhrase Ref pour SUBJ(04_etre,02_rat) :
SUBJ(04_etre,02_rat)||OBJ(04_etre,08_animal)


resultat de mapPhrase Ref pour SUBJ(12_posseder,08_animal) :
SUBJ(12_posseder,08_animal)||OBJ(12_posseder,16_systeme)|CONNECT(12_posseder,
08_animal)
OBJ(12_posseder,16_systeme)||NMOD(16_systeme,18_cardio-
vasculaire)|NMOD(16_systeme,22_semblable)
NMOD(16_systeme,22_semblable)||ADJMOD(22_semblable,20_tres)|ADJMOD(22_semblab
le,26_celui)
ADJMOD(22_semblable,26_celui)||NMOD(26_celui,32_homme)|PREPOBJ(26_celui,24_a)
NMOD(26_celui,32_homme)||PREPOBJ(32_homme,28_de)


resultat de mapPhrase S pour SUBJ(14_approcher,02_systeme) :
SUBJ(14_approcher,02_systeme)||OBJ(14_approcher,12_se)|VMOD(14_approcher,20_h
omme)|REFLEX(14_approcher,12_se)
SUBJ(14_approcher,02_systeme)||NMOD(02_systeme,04_cardio-
vasculaire)|NMOD(02_systeme,10_rat)
VMOD(14_approcher,20_homme)||NMOD(20_homme,22_humain)|PREPOBJ(20_homme,16_a)
NMOD(02_systeme,10_rat)||PREPOBJ(10_rat,06_de)


resultat de mapPhrase Q pour SUBJ(02_pouvoir,04_on) :
SUBJ(02_pouvoir,04_on)||OBJ(02_pouvoir,06_tirer)|CONNECT(02_pouvoir,00_pourqu
oi)
OBJ(02_pouvoir,06_tirer)||OBJ(06_tirer,12_conclusion)|VMOD(06_tirer,18_homme)
|VMOD(06_tirer,26_rat)
OBJ(06_tirer,12_conclusion)||NMOD(12_conclusion,10_meme)
VMOD(06_tirer,18_homme)||PREPOBJ(18_homme,14_pour)
VMOD(06_tirer,26_rat)||PREPOBJ(26_rat,22_pour)


theme : CODE1@02_NOUN_rat@Etat@08_NOUN_animal

theme : CODE3@08_NOUN_animal@Attr@16_NOUN_systeme

theme : CODE4@12_VERB_posseder@Attr@08_NOUN_animal

theme : CODE4@08_NOUN_animal@Attr@12_VERB_posseder

theme : CODE16@16_NOUN_systeme@Attr@18_ADJ_cardio-vasculaire

theme : CODE16@16_NOUN_systeme@Attr@22_ADJ_semblable

theme : CODE17@22_ADJ_semblable@Attr@20_ADV_tres

theme : CODE17@22_ADJ_semblable@Attr@26_PRON_celui

theme : CODE13@32_NOUN_homme@Attr@26_PRON_celui



theme 2  : CODE5@02_NOUN_systeme@Attr@14_VERB_approcher
```

theme 2 : CODE5@14_VERB_approcher@AttrOBJ@12_PRON_se

theme 2 : CODE6@14_VERB_approcher@AttrMOD@20_NOUN_homme

theme 2 : CODE16@02_NOUN_systeme@Attr@04_ADJ_cardio-vasculaire

theme 2 : CODE13@10_NOUN_rat@Attr@02_NOUN_systeme

theme 2 : CODE16@20_NOUN_homme@Attr@22_ADJ_humain

theme 3 : CODE5@04_PRON_on@Attr@02_VERB_pouvoir

theme 3 : CODE5@02_VERB_pouvoir@AttrOBJ@06_VERB_tirer

theme 3 : CODE9@06_VERB_tirer@AttrOBJ@12_NOUN_conclusion

theme 3 : CODE11@06_VERB_tirer@AttrMOD@18_NOUN_homme

theme 3 : CODE11@06_VERB_tirer@AttrMOD@26_NOUN_rat

theme 3 : CODE16@12_NOUN_conclusion@Attr@10_ADJ_meme

theme 3 : CODE9@18_NOUN_homme@AttrOBJ@14_PREP_pour

theme 3 : CODE9@26_NOUN_rat@AttrOBJ@22_PREP_pour

Semantic Generalization

themes en relate : 02_NOUN_rat@Attr@16_NOUN_systeme

themes en relate : 16_NOUN_systeme@Attr@18_ADJ_cardio-vasculaire

themes en relate : 22_ADJ_semblable@Attr@20_ADV_tres

themes en relate : 32_NOUN_homme@Attr@26_PRON_celui

themes en connect : 02_NOUN_rat@Attr@12_VERB_posseder

themes en connect : 12_VERB_posseder@Attr@08_NOUN_animal

themes en connect : 16_NOUN_systeme@Attr@22_ADJ_semblable

themes en connect : 22_ADJ_semblable@Attr@26_PRON_celui

themes 2 en relate : 02_NOUN_systeme@Attr@04_ADJ_cardio-vasculaire

```
themes 2 en relate : 10_NOUN_rat@Attr@02_NOUN_systeme

themes 2 en relate : 20_NOUN_homme@Attr@22_ADJ_humain

themes 2 en connect : 02_NOUN_systeme@Attr@14_VERB_approcher

themes 2 en connect : 14_VERB_approcher@AttrMOD@20_NOUN_homme

themes 2 en connect : 14_VERB_approcher@AttrOBJ@12_PRON_se

themes 3 en relate : 12_NOUN_conclusion@Attr@10_ADJ_meme

themes 3 en connect : 02_VERB_pouvoir@AttrOBJ@06_VERB_tirer

themes 3 en connect : 04_PRON_on@Attr@02_VERB_pouvoir

themes 3 en connect : 06_VERB_tirer@AttrMOD@18_NOUN_homme

themes 3 en connect : 06_VERB_tirer@AttrMOD@26_NOUN_rat

themes 3 en connect : 06_VERB_tirer@AttrOBJ@12_NOUN_conclusion
```

Try to overlap Ref on S : here it does not work

```
themes en relateS apres traitement Ref : 20_NOUN_homme@Attr@22_ADJ_humain

themes en connectS apres traitement Ref :
02_NOUN_systeme@Attr@14_VERB_approcher

themes en connectS apres traitement Ref :
14_VERB_approcher@AttrMOD@20_NOUN_homme

themes en connectS apres traitement Ref :
14_VERB_approcher@AttrOBJ@12_PRON_se
```

From the above, retrieve dependency relations necessary for paraphrase analysis

```
en tree1 : REFLEX(14_VERB_approcher,12_PRON_se)

en tree1 : SUBJ(14_VERB_approcher,02_NOUN_systeme)
```

```
en tree1 : VMOD(14_VERB_approcher,20_NOUN_homme)

en tree1 : OBJ(14_VERB_approcher,12_PRON_se)

en tree1 : NMOD(20_NOUN_homme,22_ADJ_humain)

en tree2 : OBJ(12_VERB_posseder,16_NOUN_systeme)

en tree2 : NMOD(16_NOUN_systeme,22_ADJ_semblable)

en tree2 : OBJ(04_STATE_etre,08_NOUN_animal)

en tree2 : ADJMOD(22_ADJ_semblable,20_ADV_tres)

en tree2 : ADJMOD(22_ADJ_semblable,26_PRON_celui)

en tree2 : NMOD(26_PRON_celui,32_NOUN_homme)

en tree2 : SUBJ(04_STATE_etre,02_NOUN_rat)
```

Prepare the dependency relations for paraphrase recognition

```
pour S :

|REFLEX(SYN0,12_PRON_se)|SUBJ(SYN0,02_NOUN_systeme)|VMOD(SYN0,20_NOUN_homme)|
OBJ(SYN0,12_PRON_se)|NMOD(20_NOUN_homme,22_ADJ_humain)

pour Ref :

|OBJ(12_VERB_posseder,16_NOUN_systeme)||NMOD(16_NOUN_systeme,SYN0)|OBJ(04_STA
TE_etre,08_NOUN_animal)|ADJMOD(SYN0,20_ADV_tres)|ADJMOD(SYN0,26_PRON_celui)|N
MOD(26_PRON_celui,32_NOUN_homme)|SUBJ(04_STATE_etre,02_NOUN_rat)


en @toParaphrase :
|REFLEX(SYN0,12_PRON_se)|SUBJ(SYN0,NOUN1)|VMOD(SYN0,NOUN4)|OBJ(SYN0,12_PRON_s
e)|NMOD(NOUN4,22_ADJ_humain)

en @toParaphrase :

|OBJ(12_VERB_posseder,NOUN1)||NMOD(NOUN1,SYN0)|OBJ(04_STATE_etre,08_NOUN_anim
al)|ADJMOD(SYN0,20_ADV_tres)|ADJMOD(SYN0,26_PRON_celui)|NMOD(26_PRON_celui,NO
UN4)|SUBJ(04_STATE_etre,NOUN2)
```

Paraphrase Recognition : the paraphrase patterns are selected according to the syntactic

reformulation pattern recorded at synonymy detection time : `motif syntaxique de reformulation` (here, `V-->ADJ`)

```
[…]

on entre dans la boucle matchReformulation avec
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@VMOD\(z0\,(z1|\d+_PRON_[a-z]+)\)--
>NMOD\(z1\,z0\)@ADJMOD\(z0\,(z1|\d+_PRON_[a-z]+)\)


paraphrase est vide


match de paraphrase
```

Here, it is a success, although some dependencies remain. Following the paraphrase pattern, some dependencies can be cut, which explains that some material may remain eventually.

```
reste en S : REFLEX(14_VERB_approcher,12_PRON_se)

reste en S : VMOD(14_VERB_approcher,20_NOUN_homme)

reste en S : OBJ(14_VERB_approcher,12_PRON_se)

reste en S : NMOD(20_NOUN_homme,22_ADJ_humain)
```

3.

The program stops after a break – it cannot identify the cause of the break, and so, it just provides a position signal.

```
Phrase Ref =>
```

```
00_le 02_tabagisme 04_, 06_parexemple 08_, 10_gener 12_le 14_passage 16_de
18_sang 20_atravers 22_le 24_placenta 26_, 28_le 30_structure 32_qui
34_assurer 36_le 38_communication 40_entre 42_le 44_circulation 46_de 48_le
50_mere 52_et 54_celui 56_de 58_le 60_enfant


Dependances Ref =>

SUBJ(10_gener,02_tabagisme)
SUBJ(34_assurer,32_qui)
OBJ(10_gener,14_passage)
OBJ(34_assurer,38_communication)
VMOD(10_gener,06_parexemple)
VMOD(10_gener,24_placenta)
VMOD(34_assurer,50_mere)
COREF(30_structure,32_qui)
NMOD(14_passage,18_sang)
NMOD(38_communication,44_circulation)
NMOD(38_communication,54_celui)
NMOD(44_circulation,50_mere)
NMOD(54_celui,60_enfant)
COORDITEMS(44_circulation,54_celui)
PREPOBJ(18_sang,16_de)
PREPOBJ(24_placenta,20_atravers)
PREPOBJ(44_circulation,40_entre)
PREPOBJ(50_mere,46_de)
PREPOBJ(60_enfant,56_de)
DETERM(02_tabagisme,00_le)
DETERM(14_passage,12_le)
DETERM(24_placenta,22_le)
DETERM(30_structure,28_le)
DETERM(38_communication,36_le)
DETERM(44_circulation,42_le)
DETERM(50_mere,48_le)
DETERM(60_enfant,58_le)
CONNECT(34_assurer,32_qui)


Lemmes Ref =>

00_DET_le
02_NOUN_tabagisme
_PUNCT_,
06_ADV_parexemple
_PUNCT_,
10_VERB_gener
12_DET_le
14_NOUN_passage
16_PREP_de
18_NOUN_sang
20_PREP_atravers
22_DET_le
24_NOUN_placenta
_PUNCT_,
28_DET_le
30_NOUN_structure
32_PRON_qui
```

```
34_VERB_assurer
36_DET_le
38_NOUN_communication
40_PREP_entre
42_DET_le
44_NOUN_circulation
46_PREP_de
48_DET_le
50_NOUN_mere
52_COORD_et
54_PRON_celui
56_PREP_de
58_DET_le
60_NOUN_enfant


Phrase S =>
00_le 02_tabac 04_arret 06_le 08_circulation 10_de 12_sang 14_a 16_organs
18_noble 20_de 22_foetus


Dependances S =>

NMOD(02_tabac,04_arret)
NMOD(08_circulation,12_sang)
NMOD(08_circulation,16_organs)
NMOD(16_organs,18_noble)
NMOD(16_organs,22_foetus)
PREPOBJ(12_sang,10_de)
PREPOBJ(16_organs,14_a)
PREPOBJ(22_foetus,20_de)
DETERM(02_tabac,00_le)
DETERM(08_circulation,06_le)


Lemmes S =>

00_DET_le
02_NOUN_tabac
04_NOUN_arret
06_DET_le
08_NOUN_circulation
10_PREP_de
12_NOUN_sang
14_PREP_a
16_NOUN_organs
18_ADJ_noble
20_PREP_de
22_NOUN_foetus


Phrase Q =>
00_comment 02_le 04_tabac 06_pouvoir 08_il 10_affecter 12_le 14_foetus


Dependances Q =>
```

```
SUBJ(06_pouvoir,04_tabac)
SUBJ(10_affecter,04_tabac)
SUBJCLIT(06_pouvoir,08_il)
OBJ(06_pouvoir,10_affecter)
OBJ(10_affecter,14_foetus)
DETERM(04_tabac,02_le)
DETERM(14_foetus,12_le)
CONNECT(06_pouvoir,00_comment)


Lemmes Q =>

00_ADV_comment
02_DET_le
04_NOUN_tabac
06_VERB_pouvoir
08_PRON_il
10_VERB_affecter
12_DET_le
14_NOUN_foetus


pour la question, prepo admissible : PREP_par

pour la question, prepo admissible : PREP_en


pas de fautes d'orthographe


l'element 0 0 est 00_le
l'element 0 1 est 02_tabac
l'element 0 2 est 04_arret

rupture au point 2 : 04_arret|06_le

l'element 1 0 est 06_le
l'element 1 1 est 08_circulation
l'element 1 2 est 10_de
l'element 1 3 est 12_sang
l'element 1 4 est 14_a
l'element 1 5 est 16_organs
l'element 1 6 est 18_noble
l'element 1 7 est 20_de
l'element 1 8 est 22_foetus

point de rupture en premiere sortie : 04_arret|06_le


apres traitement phrasesMotif 2 :
@NP___02@NP___04@@@NP___08@PP___10@NP___12@PP___14@NP___16@AP___18@PP___20@NP
___22


categories all_cats pour S :
0@@@@NP_DETERM02_00_le_02_tabac@NP_NMOD02_04_arret@NP_DETERM08_06_la_08_circu
```

```
lation@PP_PREP10_10_du@NMOD08_12_sang@PP_PREP14_14_aux@NMOD08_16_organs@AP_NM
OD16_18_nobles@PP_PREP20_20_du@NMOD16_22_foetus@SENT_.
```

Here, the synthesis of syntactic analysis is spread as the phrases are separated by commas. This will be used to define attachments: the third segment of REF attaches to the second

```
categories all_cats de Ref : 0@@@@@NP_DETERM02_00_le_SUBJ10_02_tabagisme@PUNCT

categories all_cats de Ref : 1@@@@ADV06_VMOD10_06_parexemple@PUNCT

categories all_cats de Ref :
2@@@@FV_10_gene@NP_DETERM14_12_le_OBJ10_14_passage@PP_PREP16_16_du@NMOD14_18_
sang@PP_PREP20_20_atravers@DETERM24_22_le_VMOD10_24_placenta@PUNCT

categories all_cats de Ref :
3@@@@NP_DETERM30_28_la_30_structure@SC@BG_PRON32_CONNECT34_32_qui@FV_34_assur
e@NP_DETERM38_36_la_OBJ34_38_communication@PP_PREP40_40_entre@DETERM44_42_la_
NMOD38_44_circulation@PP_PREP46_46_de@DETERM50_48_la_NMOD44_50_mere@COORD52_e
t@NP_PRON54@COORDITEMS44_54_celle@PP_PREP56_56_de@DETERM60_58_l'_NMOD54_60_en
fant@SENT_.


resultat attachement Ref 3@@@le structure qui assurer le communication entre
le circulation de le mere et celui de le enfant :
2@@@@FV_10_gene@NP_DETERM14_12_le_OBJ10_14_passage@PP_PREP16_16_du@NMOD14_18_
sang@PP_PREP20_20_atravers@DETERM24_22_le_VMOD10_24_placenta@PUNCT


erreur d'accord : pas d'erreur d'accord


erreur de temps : pas d'erreur de temps
```

Here the program is not able to detect that the student mistook a noun for a verb. So it simply outputs the break position

```
rupture d'analyse apres le mot : arret
```

Appendix D. Paraphrase Patterns

The sets of paraphrase patterns are presented in this appendix. The first set gathers paraphrase patterns based on synonymic reformulation of a syntactic head. The second set contains paraphrases which do not rest on synonymic reformulation.

File `paraphrase.dat` :

```
### Base paraphrase

### format :
###   indice de reformulation
###   reformulation index
###   #####
###   correspondance
###
NOUN-->ADJ
#####
1*NMOD\((SYN[0-9])\,(NOUN[0-9])\)
-->NMOD\(z1\,z0\)@ADJMOD\(z0\,(z1|\d+_PRON_[a-z]+)\)
1*NMOD\((SYN[0-9])\,(NOUN[0-9])\)-->NMOD\(z1\,z0\)
1*NMOD\((NOUN[0-9])\,(SYN[0-9])\)-->NMOD\(z0\,z1\)
1*NMOD\((SYN[0-9])\,(SYN[0-9])\)-->NMOD\(z1\,z0\)
1*OBJ\((VERB[0-9])\,(SYN[0-9])\)-->OBJ\(z0\,z1\)
##########
ADJ-->NOUN
#####
1*NMOD\((NOUN[0-9])\,(SYN[0-9])\)@ADJMOD\(z1\,(z0|\d+_PRON_[a-z]+)\)
-->NMOD\(z1\,z0\)
1*NMOD\((NOUN[0-9])\,(SYN[0-9])\)-->NMOD\(z1\,z0\)
1*NMOD\((NOUN[0-9])\,(SYN[0-9])\)-->NMOD\(z0\,z1\)
1*NMOD\((SYN[0-9])\,(SYN[0-9])\)-->NMOD\(z1\,z0\)
1*OBJ\((VERB[0-9])\,(SYN[0-9])\)-->OBJ\(z0\,z1\)
##########
VERB-->ADJ
#####
2*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@COREF_REL\(z1\,(z1|\d+_PRON_[a-
z]+)\)@CONNECT_REL\(z0\,(z1|\d+_PRON_[a-z]+)\)@VMOD\(z0\,(ADV[0-9])\)
-->NMOD\(z2\,z0\)
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@COREF_REL\(z1\,(z1|\d+_PRON_[a-
z]+)\)@CONNECT_REL\(z0\,(z1|\d+_PRON_[a-z]+)\)-->NMOD\(z2\,z0\)
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@VMOD\(z0\,(z1|\d+_PRON_[a-z]+)\)
-->NMOD\(z1\,z0\)@ADJMOD\(z0\,(z1|\d+_PRON_[a-z]+)\)
# « voir » --> « visible »
1*SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,(NOUN[0-
9])\)@VMOD\(z0\,(ADV[0-9])\)
-->SUBJ\(\d+_STATE_[a-z]+\,z1\)@OBJ\(\d+_STATE_[a-
z]+\,z0\)@VMOD\(\d+_STATE_[a-z]+\,z2\)
1*SUBJ\((SYN[0-9])\,\d+_NOUN_[a-z]+\)@OBJ\(z0\,(NOUN[0-
9])\)@VMOD\(z0\,(ADV[0-9])\)
```

```
-->SUBJ\(\d+_STATE_[a-z]+\,z1\)@OBJ\(\d+_STATE_[a-
z]+\,z0\)@VMOD\(\d+_STATE_[a-z]+\,z2\)
1*SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-
z]+\)@VMOD\(z0\,(ADV[0-9])\)
-->SUBJ\(\d+_STATE_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_STATE_[a-
z]+\,z0\)@VMOD\(\d+_STATE_[a-z]+\,z2\)
1*SUBJ\((SYN[0-9])\,\d+_NOUN_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-
z]+\)@VMOD\(z0\,(ADV[0-9])\)
-->SUBJ\(\d+_STATE_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_STATE_[a-
z]+\,z0\)@VMOD\(\d+_STATE_[a-z]+\,z2\)
1*SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,(NOUN[0-9])\)
-->SUBJ\(\d+_STATE_[a-z]+\,z1\)@OBJ\(\d+_STATE_[a-z]+\,z0\)
1*SUBJ\((SYN[0-9])\,\d+_NOUN_[a-z]+\)@OBJ\(z0\,(NOUN[0-9])\)
-->SUBJ\(\d+_STATE_[a-z]+\,z1\)@OBJ\(\d+_STATE_[a-z]+\,z0\)
1*SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-z]+\)
-->SUBJ\(\d+_STATE_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_STATE_[a-z]+\,z0\)
1*SUBJ\((SYN[0-9])\,\d+_NOUN_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-z]+\)
-->SUBJ\(\d+_STATE_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_STATE_[a-z]+\,z0\)
# « Observer » --> « observe » (Adj)
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,\d+_CONJQUE_que\)@COREF\((NOUN[0-
9])\,\d+_CONJQUE_que\)@CONNECT\(z0\,\d+_CONJQUE_que\)
-->NMOD\(z2\,z0\)
1*SUBJ\((SYN[0-9])\,\d+_PRON_[a-
z]+\)@OBJ\(z0\,\d+_CONJQUE_que\)@COREF\((NOUN[0-
9])\,\d+_CONJQUE_que\)@CONNECT\(z0\,\d+_CONJQUE_que\)
-->NMOD\(z2\,z0\)
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,(NOUN[0-
9])\)@COREF\(z2\,z2\)@CONNECT\(z0\,z2\)
-->NMOD\(z2\,z0\)
1*SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,(NOUN[0-
9])\)@COREF\(z2\,z2\)@CONNECT\(z0\,z2\)
-->NMOD\(z2\,z0\)
# l'inverse - the contrary
NMOD\((NOUN[0-9])\,(SYN[0-9])\)
-->SUBJ\(z1\,d\+_NOUN_[a-
z]+\)@OBJ\(z1\,\d+_CONJQUE_que\)@COREF\(z0\,\d+_CONJQUE_que\)@CONNECT\(z1\,\d
+_CONJQUE_que\)
NMOD\((NOUN[0-9])\,(SYN[0-9])\)
-->SUBJ\(z1\,d\+_NOUN_[a-z]+\)@OBJ\(z1\,z0\)@COREF\(z0\,z0
\)@CONNECT\(z1\,z0\)
NMOD\((NOUN[0-9])\,(SYN[0-9])\)
-->SUBJ\(z1\,d\+_PRON_[a-
z]+\)@OBJ\(z1\,\d+_CONJQUE_que\)@COREF\(z0\,\d+_CONJQUE_que\)@CONNECT\(z1\,\d
+_CONJQUE_que\)
NMOD\((NOUN[0-9])\,(SYN[0-9])\)
-->SUBJ\(z1\,d\+_PRON_[a-z]+\)@OBJ\(z1\,z0\)@COREF\(z0\,z0
\)@CONNECT\(z1\,z0\)
##########
ADJ-->VERB
#####
1*NMOD\((NOUN[0-9])\,(SYN[0-9])\)
-->SUBJ\(z1\,z0\)@COREF_REL\(z0\,(z0|\d+_PRON_[a-
z]+)\)@CONNECT_REL\(z1\,(z0|\d+_PRON_[a-z]+)\)@VMOD\(z1\,ADV_[a-z]+\)
1*NMOD\((NOUN[0-9])\,(SYN[0-9])\)
-->SUBJ\(z1\,z0\)@COREF_REL\(z0\,(z0|\d+_PRON_[a-
z]+)\)@CONNECT_REL\(z1\,(z0|\d+_PRON_[a-z]+)\)
1*NMOD\((NOUN[0-9])\,(SYN[0-9])\)@ADJMOD\(z1\,(z0|\d+_PRON_[a-z]+)\)
```

```
-->SUBJ\(z1\,z0\)@VMOD\(z1\,(z0|\d+_PRON_[a-z]+)\)
# « visible » --> « voir »
1*SUBJ\(\d+_STATE_[a-z]+\,(NOUN[0-9])\)@OBJ\(\d+_STATE_[a-z]+\,(SYN[0-
9])\)@VMOD\(\d+_STATE_[a-z]+\,(ADV[0-9])\)
-->SUBJ\(z1\,\d+_NOUN_[a-z]+\)@OBJ\(z1\,z0\)@VMOD\(z1\,z2\)
1*SUBJ\(\d+_STATE_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_STATE_[a-z]+\,(SYN[0-
9])\)@VMOD\(\d+_STATE_[a-z]+\,(ADV[0-9])\)
-->SUBJ\(z0\,\d+_NOUN_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-z]+\)@VMOD\(z0\,z1\)
1*SUBJ\(\d+_STATE_[a-z]+\,(NOUN[0-9])\)@OBJ\(\d+_STATE_[a-z]+\,(SYN[0-9])\)
-->SUBJ\(z1\,\d+_NOUN_[a-z]+\)@OBJ\(z1\,z0\)
1*SUBJ\(\d+_STATE_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_STATE_[a-z]+\,(SYN[0-
9])\)
-->SUBJ\(z0\,\d+_NOUN_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-z]+\)
1*SUBJ\(\d+_STATE_[a-z]+\,(NOUN[0-9])\)@OBJ\(\d+_STATE_[a-z]+\,(SYN[0-9])\)
-->SUBJ\(z1\,\d+_PRON_[a-z]+\)@OBJ\(z1\,z0\)
1*SUBJ\(\d+_STATE_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_STATE_[a-z]+\,(SYN[0-
9])\)
-->SUBJ\(z0\,\d+_PRON_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-z]+\)
##########
NOUN-->VERB
#####
# « respect » --> « respecter »
1*SUBJ\(\d+_VERB_[a-z]+\,(NOUN[0-9])\)@OBJ\(\d+_VERB_[a-z]+\,(SYN[0-
9])\)@NMOD\(z1\,(NOUN[0-9])\)
-->SUBJ\(z1\,(z0|\d+_PRON_[a-z]+)\)@OBJ\(z1\,(z2|\d+_PRON_[a-z]+)\)
1*SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_VERB_[a-z]+\,(SYN[0-
9])\)@NMOD\(z0\,\d+_PRON_[a-z]+\)
-->SUBJ\(z0\,\d+_PRON_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-z]+\)
1*SUBJ\(\d+_VERB_[a-z]+\,(NOUN[0-9])\)@OBJ\(\d+_VERB_[a-z]+\,(SYN[0-
9])\)@NMOD\(z1\,\d+_PRON_[a-z]+\)
-->SUBJ\(z1\,(z0|\d+_PRON_[a-z]+)\)@OBJ\(z1\,\d+_PRON_[a-z]+\)
1*SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_VERB_[a-z]+\,(SYN[0-
9])\)@NMOD\(z0\,(NOUN[0-9])\)
-->SUBJ\(z0\,\d+_PRON_[a-z]+\)@OBJ\(z0\,(z1|\d+_PRON_[a-z]+)\)
# « son analyse se deroule selon »
--> « analyse les echantillons selon ... »
1*SUBJ\(\d+_VERB_[a-z]+\,(SYN[0-9])\)@VMOD\(\d+_VERB_[a-z]+\,(NOUN[0-
9])\)@NMOD\(z0\,(NOUN[0-9])\)
-->SUBJ\(z0\,\d+_NOUN_[a-z]+\)@OBJ\(z0\,z2\)@VMOD\(z0\,z1\)
1*SUBJ\(\d+_VERB_[a-z]+\,(SYN[0-9])\)@VMOD\(\d+_VERB_[a-z]+\,(NOUN[0-
9])\)@NMOD\(z0\,(NOUN[0-9])\)
-->SUBJ\(z0\,\d+_PRON_[a-z]+\)@OBJ\(z0\,z2\)@VMOD\(z0\,z1\)
# « analyse » --> « analyser »
1*OBJ\((VERB[0-9])\,\d+_VERB_[a-z]+\)@OBJ\(\d+_VERB_[a-z]+\,(SYN[0-
9])\)@NMOD\(z1\,(NOUN[0-9])\)
-->OBJ\(z0\,z1\)@OBJ\(z1\,(z2|\d+_PRON_[a-z]+)\)
1*OBJ\((VERB[0-9])\,\d+_VERB_[a-z]+\)@OBJ\(\d+_VERB_[a-z]+\,(SYN[0-
9])\)@NMOD\(z1\,\d+_PRON_[a-z]+\)
-->OBJ\(z0\,z1\)@OBJ\(z1\,\d+_PRON_[a-z]+\)
1*OBJ\((VERB[0-9])\,\d+_VERB_[a-z]+\)@VMOD\(\d+_VERB_[a-z]+\,(SYN[0-
9])\)@NMOD\(z1\,(NOUN[0-9])\)
-->OBJ\(z0\,z1\)@OBJ\(z1\,(z2|\d+_PRON_[a-z]+)\)
1*OBJ\((VERB[0-9])\,\d+_VERB_[a-z]+\)@VMOD\(\d+_VERB_[a-z]+\,(SYN[0-
9])\)@NMOD\(z1\,\d+_PRON_[a-z]+\)
-->OBJ\(z0\,z1\)@OBJ\(z1\,\d+_PRON_[a-z]+\)
# « en reponse » --> « repondre »
```

```
1*SUBJ\((VERB[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,(NOUN[0-9])\)@VMOD\(z0\,(SYN[0-
9])\)
-->SUBJ\(z3\,z1\)@OBJ\(z0\,z2\)@VMOD\(z3\,z0\)
1*SUBJ\((VERB[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,(NOUN[0-
9])\)@VMOD\(z0\,(SYN[0-9])\)
-->SUBJ\(z2\,\d+_PRON_[a-z]+\)@OBJ\(z0\,z1\)@VMOD\(z2\,z0\)
# « soumettre a analyse » --> « analyser »
1*SUBJ\(\d+_VERB_[a-z]+\,(NOUN[0-9])\)@OBJ\(\d+_VERB_[a-z]+\,(NOUN[0-
9])\)@VMOD\(\d+_VERB_[a-z]+\,(SYN[0-9])\)
-->SUBJ\(z2\,z0\)@OBJ\(z2\,z1\)
1*SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_VERB_[a-z]+\,(NOUN[0-
9])\)@VMOD\(\d+_VERB_[a-z]+\,(SYN[0-9])\)
-->SUBJ\(z1\,\d+_PRON_[a-z]+\)@OBJ\(z1\,z0\)
1*SUBJ\(\d+_VERB_[a-z]+\,(NOUN[0-9])\)@OBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-
z]+\)@VMOD\(\d+_VERB_[a-z]+\,(SYN[0-9])\)
-->SUBJ\(z1\,z0\)@OBJ\(z1\,\d+_PRON_[a-z]+\)
1*SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-
z]+\)@VMOD\(\d+_VERB_[a-z]+\,(SYN[0-9])\)
-->SUBJ\(z0\,\d+_PRON_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-z]+\)
# « sommeil » --> « dormir »
1*SUBJ\(\d+_STATE_[a-z]+\,(SYN[0-9])\)@OBJ\(\d+_VERB_[a-z]+\,(SYN[0-
9])\)@NMOD\(z0\,(NOUN[0-9])\)
-->SUBJ\(\d+_STATE_[a-z]+\,z2\)@VMOD\(z0\,z1\)
1*SUBJ\(\d+_STATE_[a-z]+\,(SYN[0-9])\)@OBJ\(\d+_VERB_[a-z]+\,(SYN[0-9])\)
-->SUBJ\(\d+_STATE_[a-z]+\,\d+_PRON_[a-z]+\)@VMOD\(z0\,z1\)
# « augmentation des prix » --> « les prix augmentent »
1*NMOD\((SYN[0-9])\,(NOUN[0-9])\)@NMOD\(z0\,(NOUN[0-9])\)
-->SUBJ\(z0\,z1\)@VMOD\(z0\,z2\)
1*NMOD\((SYN[0-9])\,(NOUN[0-9])\)
-->SUBJ\(z0\,(z1|\d+_PRON_[a-z]+)\)@VMOD\(z0\,(z1|\d+_PRON_[a-z]+)\)
1*NMOD\((SYN[0-9])\,(NOUN[0-9])\)
-->SUBJ\(z0\,(z1|\d+_PRON_[a-z]+)\)@OBJ\(z0\,(z1|\d+_PRON_[a-z]+)\)
1*NMOD\((SYN[0-9])\,(NOUN[0-9])\)
-->SUBJ\(z0\,(z1|\d+_PRON_[a-z]+)\)
1*SUBJ\(\d+_VERB_[a-z]+\,(NOUN[0-9])\)@REFLEX\(\d+_VERB_[a-z]+\,\d+_PRON_[a-
z]+\)@VMOD\(\d+_VERB_[a-z]+\,(SYN[0-9])\)
-->SUBJ\(z1\,\d+_PRON_[a-z]+\)@OBJ\(z1\,z0\)
1*SUBJ\(\d+_VERB_[a-z]+\,(NOUN[0-9])\)@REFLEX\(\d+_VERB_[a-z]+\,\d+_PRON_[a-
z]+\)@VMOD\(\d+_VERB_[a-z]+\,(SYN[0-9])\)@NMOD\((SYN[0-9])\,(NOUN[0-9])\)
-->SUBJ\(z1\,z2\)@OBJ\(z1\,z0\)
# « recevoir des conseils » --> « conseiller »
SUBJ\(\d+_VERB_[a-z]+\,(NOUN[0-9])\)@OBJ\(\d+_VERB_[a-z]+\,(SYN[0-
9])\)@VMOD\(\d+_VERB_[a-z]+\,(NOUN[0-9])\)
-->SUBJ\(z1\,z2\)@OBJ\(z1\,z0\)
SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_VERB_[a-z]+\,(SYN[0-
9])\)@VMOD\(\d+_VERB_[a-z]+\,(NOUN[0-9])\)
-->SUBJ\(z0\,z1\)@OBJ\(z0\,\d+_PRON_[a-z]+\)
SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_VERB_[a-z]+\,(SYN[0-
9])\)@VMOD\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)
-->SUBJ\(z0\,\d+_PRON_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-z]+\)
SUBJ\(\d+_VERB_[a-z]+\,(NOUN[0-9])\)@OBJ\(\d+_VERB_[a-z]+\,(SYN[0-
9])\)@VMOD\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)
-->SUBJ\(z1\,\d+_PRON_[a-z]+\)@OBJ\(z1\,z0\)
# « prise » --> « avoir pris »
NMOD\((SYN[0-9])\,(NOUN[0-9])\)
-->OBJ\(z0\,z1\)@AUXIL\(\d+_VERB_avoir\,z0\)
NMOD\((SYN[0-9])\,(NOUN[0-9])\)
```

```
-->OBJ\(z0\,z1\)
#
SUBJ\(\d+_VERB_[a-z]+\,(SYN[0-9])\)@OBJ\(\d+_VERB_[a-z]+\,(NOUN[0-
9])\)@NMOD\(z0\,(NOUN[0-9])\)
-->SUBJ\(z0\,z2\)@VMOD\(z0\,z1\)
#########
VERB-->NOUN
#####
1*SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,z1\)@REFLEX\(\d+_VERB_[a-z]+\,\d+_PRON_[a-
z]+\)@VMOD\(\d+_VERB_[a-z]+\,z0\)
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,z2\)@REFLEX\(\d+_VERB_[a-z]+\,\d+_PRON_[a-
z]+\)@VMOD\(\d+_VERB_[a-z]+\,z0\)@NMOD\(z0\,z1\)
#
# « respecter » --> « respect »
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,(z1|\d+_PRON_[a-z]+)\)@OBJ\(\d+_VERB_[a-
z]+\,z0\)@NMOD\(z0\,(z2|\d+_PRON_[a-z]+)\)
1*SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-z]+\)
-->SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_VERB_[a-
z]+\,z0\)@NMOD\(z0\,\d+_PRON_[a-z]+\)
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,\d+_PRON_[a-z]+\)
-->SUBJ\(\d+_VERB_[a-z]+\,(z1|\d+_PRON_[a-z]+)\)@OBJ\(\d+_VERB_[a-
z]+\,z0\)@NMOD\(z0\,\d+_PRON_[a-z]+\)
1*SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_VERB_[a-
z]+\,z0\)@NMOD\(z0\,(z1|\d+_PRON_[a-z]+)\)
# « conseiller » --> « recevoir des conseils »
SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,z2\)@OBJ\(\d+_VERB_[a-z]+\,z0\)@VMOD\(z0\,z1\)
SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,z1\)@OBJ\(\d+_VERB_[a-
z]+\,z0\)@VMOD\(z0\,\d+_PRON_[a-z]+\)
SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-z]+\)
-->SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_VERB_[a-
z]+\,z0\)@VMOD\(z0\,\d+_PRON_[a-z]+\)
SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,\d+_PRON_[a-z]+\)
-->SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_VERB_[a-
z]+\,z0\)@VMOD\(z0\,z1\)
# « analyse les echantillons selon » --> « son analyse se deroule selon ... »
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,(NOUN[0-9])\)@VMOD\(z0\,(NOUN[0-
9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,z0\)@VMOD\(\d+_VERB_[a-z]+\,z3\)@NMOD\(z0\,z2\)
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,(NOUN[0-9])\)@VMOD\(z0\,(NOUN[0-
9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,z0\)@VMOD\(\d+_VERB_[a-
z]+\,z3\)@NMOD\(z0\,z2\)@NMOD\(z0\,z1\)
1*SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,(NOUN[0-
9])\)@VMOD\(z0\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,z0\)@VMOD\(\d+_VERB_[a-z]+\,z2\)@NMOD\(z0\,z1\)
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,\d+_PRON_[a-
z]+\)@VMOD\(z0\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,z0\)@VMOD\(\d+_VERB_[a-z]+\,z2\)@NMOD\(z0\,z1\)
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,\d+_PRON_[a-
z]+\)@VMOD\(z0\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,z0\)@VMOD\(\d+_VERB_[a-z]+\,z2\)
```

```
1*SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-
z]+\)@VMOD\(z0\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,z0\)@VMOD\(\d+_VERB_[a-z]+\,z1\)
# « repondre » --> « en reponse »
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@OBJ\((VERB[0-9])\,(NOUN[0-
9])\)@VMOD\(z0\,z2\)
-->SUBJ\(z2\,z1\)@OBJ\(z2\,z3\)@VMOD\(z2\,z0\)
1*SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)@OBJ\((VERB[0-9])\,(NOUN[0-
9])\)@VMOD\(z0\,z1\)
-->SUBJ\(z1\,\d+_PRON_[a-z]+\)@OBJ\(z1\,z2\)@VMOD\(z1\,z0\)
# « analyser » --> « soumettre a analyse »
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,z1\)@OBJ\(\d+_VERB_[a-z]+\,z2\)@VMOD\(\d+_VERB_[a-
z]+\,z0\)
1*SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_VERB_[a-
z]+\,z1\)@VMOD\(\d+_VERB_[a-z]+\,z0\)
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,\d+_PRON_[a-z]+\)
-->SUBJ\(\d+_VERB_[a-z]+\,z1\)@OBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-
z]+\)@VMOD\(\d+_VERB_[a-z]+\,z0\)
1*SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-z]+\)
-->SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_VERB_[a-
z]+\,\d+_PRON_[a-z]+\)@VMOD\(\d+_VERB_[a-z]+\,z0\)
# « augmenter » --> « augmentation »
SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@VMOD\(z0\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,z0\)@OBJ\(\d+_VERB_[a-z]+\,z2\)@NMOD\(z0\,z1\)
# « dormir » --> « sommeil »
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@VMOD\(z0\,(SYN[0-9])\)
-->SUBJ\(\d+_STATE_[a-z]+\,z0\)@OBJ\(\d+_STATE_[a-z]+\,z2\)@NMOD\(z0\,z1\)
1*SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)@VMOD\(z0\,(SYN[0-9])\)
-->SUBJ\(\d+_STATE_[a-z]+\,z0\)@OBJ\(\d+_STATE_[a-z]+\,z1\)
# « augmenter » --> « augmentation »
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@VMOD\(z0\,(NOUN[0-9])\)
-->NMOD\(z0\,z1\)@NMOD\(z0\,z2\)
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@VMOD\(z0\,(z1|\d+_PRON_[a-z]+)\)
-->NMOD\(z0\,(z1|\d+_PRON_[a-z]+)\)
1*SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)@VMOD\(z0\,(z1|\d+_PRON_[a-z]+)\)
-->NMOD\(z0\,(z1|\d+_PRON_[a-z]+)\)
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,(z1|\d+_PRON_[a-z]+)\)
-->NMOD\(z0\,(z1|\d+_PRON_[a-z]+)\)
1*SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,(z1|\d+_PRON_[a-z]+)\)
-->NMOD\(z0\,(z1|\d+_PRON_[a-z]+)\)
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)-->NMOD\(z0\,(z1|\d+_PRON_[a-z]+)\)
1*SUBJ\((SYN[0-9])\,\d+_PRON_[a-z]+\)-->NMOD\(z0\,(z1|\d+_PRON_[a-z]+)\)
# « analyser » --> « analyse »
1*OBJ\((VERB[0-9])\,(SYN[0-9])\)@OBJ\(z1\,(NOUN[0-9])\)
-->OBJ\(z0\,\d+_VERB_[a-z]+\)@OBJ\(\d+_VERB_[a-
z]+\,z1\)@NMOD\(z1\,(z2|\d+_PRON_[a-z]+)\)
1*OBJ\((VERB[0-9])\,(SYN[0-9])\)@OBJ\(z1\,\d+_PRON_[a-z]+\)
-->OBJ\(z0\,\d+_VERB_[a-z]+\)@OBJ\(\d+_VERB_[a-
z]+\,z1\)@NMOD\(z1\,\d+_PRON_[a-z]+\)
1*OBJ\((VERB[0-9])\,(SYN[0-9])\)@OBJ\(z1\,(NOUN[0-9])\)
-->OBJ\(z0\,\d+_VERB_[a-z]+\)@VMOD\(\d+_VERB_[a-
z]+\,z1\)@NMOD\(z1\,(z2|\d+_PRON_[a-z]+)\)
1*OBJ\((VERB[0-9])\,(SYN[0-9])\)@OBJ\(z1\,\d+_PRON_[a-z]+\)
-->OBJ\(z0\,\d+_VERB_[a-z]+\)@VMOD\(\d+_VERB_[a-
z]+\,z1\)@NMOD\(z1\,\d+_PRON_[a-z]+\)
```

```
# « avoir pris » --> « la prise »
1*OBJ\((SYN[0-9])\,(NOUN[0-9])\)@AUXIL\(\d+_VERB_avoir\,z0\)
-->NMOD\(z0\,z1\)
1*OBJ\((SYN[0-9])\,(NOUN[0-9])\)-->NMOD\(z0\,z1\)
#########
ADV-->VERB
#####
1*SUBJ\((VERB[0-9])\,(NOUN[0-9])\)@VMOD\(z0\,(SYN[0-9])\)
-->SUBJ\(z2\,z1\)@VMOD\(z2\,z0\)
#########
VERB-->ADV
#####
1*SUBJ\((SYN[0-9])\,(NOUN[0-9])\)@VMOD\(z0\,(VERB[0-9])\)
-->SUBJ\(z2\,z1\)@VMOD\(z2\,z0\)
#########
ADV-->ADJ
#####
1*OBJ\((VERB[0-9])\,(SYN[0-9])\)@VMOD\(z0\,(SYN[0-9])\
)-->OBJ\(z0\,z1\)@NMOD\(z1\,z2\)
#########
ADJ-->ADV
#####
1*OBJ\((VERB[0-9])\,(SYN[0-9])\)@NMOD\(z1\,(SYN[0-9])\)
-->OBJ\(z0\,z1\)@VMOD\(z0\,z2\)
```

File `paraphrase2.dat` :

```
### Base paraphrase 2 : pas de synonymes - no synonyms

### format :
###   indice de reformulation
###   reformulation index
###   #####
###   correspondance
###
VERB
#####
# « depart de P surprendre » --> « P surprendre par depart »
1*SUBJ\((VERB[0-9])\,(NOUN[0-9])\)@VMOD\(z0\,(NOUN[0-9])\)
-->SUBJ\(z0\,z2\)@NMOD\(z2\,z1\)
1*SUBJ\((VERB[0-9])\,\d+_PRON_[a-z]+\)@VMOD\(z0\,(NOUN[0-9])\)
-->SUBJ\(z0\,z1\)
# l'inverse - the contrary
1*SUBJ\((VERB[0-9])\,(NOUN[0-9])\)@NMOD\(z1\,(NOUN[0-9])\)
-->SUBJ\(z0\,z2\)@VMOD\(z0\,z1\)
1*SUBJ\((VERB[0-9])\,(NOUN[0-9])\)-->SUBJ\(z0\,\d+_PRON_[a-
z]+\)@VMOD\(z0\,z1\)
# « boiter » --> « en boitant »
1*SUBJ\(\d+_VERB_[a-z]+\,(NOUN[0-9])\)@VMOD\(\d+_VERB_[a-z]+\,(VERB[0-9]+)\)-
->SUBJ\(z1\,z0\)
1*SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)@VMOD\(\d+_VERB_[a-z]+\,(VERB[0-
9]+)\)-->SUBJ\(z0\,\d+_PRON_[a-z]+\)
# idem inverse - the contrary
1*SUBJ\((VERB[0-9])\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,z1\)@VMOD\(\d+_VERB_[a-z]+\,z0\)
```

```
1*SUBJ\((VERB[0-9])\,\d+_PRON_[a-z]+\)
-->SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)@VMOD\(\d+_VERB_[a-z]+\,z0\)
# « voir » --> « se voir »
1*SUBJ\((VERB[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,(NOUN[0-9])\)
-->SUBJ\(z0\,z1\)@OBJ\(z0\,\d+_PRON_[a-z]+\)@REFLEX\(z0\,\d+_PRON_[a-z]+\)
1*SUBJ\((VERB[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-z]+\)
-->SUBJ\(z0\,\d+_PRON_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-
z]+\)@REFLEX\(z0\,\d+_PRON_[a-z]+\)
1*SUBJ\((VERB[0-9])\,\d+_NOUN_[a-z]+\)@OBJ\(z0\,(NOUN[0-9])\)
-->SUBJ\(z0\,z1\)@OBJ\(z0\,\d+_PRON_[a-z]+\)@REFLEX\(z0\,\d+_PRON_[a-z]+\)
1*SUBJ\((VERB[0-9])\,\d+_NOUN_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-z]+\)
-->SUBJ\(z0\,\d+_PRON_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-
z]+\)@REFLEX\(z0\,\d+_PRON_[a-z]+\)
# idem inverse - the contrary
1*SUBJ\((VERB[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,\d+_PRON_[a-
z]+\)@REFLEX\(z0\,\d+_PRON_[a-z]+\)
-->SUBJ\(z0\,\d+_PRON_[a-z]+\)@OBJ\(z0\,z1\)
1*SUBJ\((VERB[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,\d+_PRON_[a-
z]+\)@REFLEX\(z0\,\d+_PRON_[a-z]+\)
-->SUBJ\(z0\,\d+_NOUN_[a-z]+\)@OBJ\(z0\,z1\)
1*SUBJ\((VERB[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-
z]+\)@REFLEX\(z0\,\d+_PRON_[a-z]+\)
-->SUBJ\(z0\,\d+_PRON_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-z]+\)
1*SUBJ\((VERB[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-
z]+\)@REFLEX\(z0\,\d+_PRON_[a-z]+\)
-->SUBJ\(z0\,\d+_NOUN_[a-z]+\)@OBJ\(z0\,\d+_PRON_[a-z]+\)
# « etonnant » --> « etonne » (PastPart) SYMETRICAL
1*SUBJ\((VERB[0-9])\,(NOUN[0-9])\)@VMOD\(z0\,(NOUN[0-9])\)
-->SUBJ\(z0\,z2\)@VMOD\(z0\,z1\)
1*SUBJ\((VERB[0-9])\,\d+_PRON_[a-z]+\)@VMOD\(z0\,(NOUN[0-9])\)
-->SUBJ\(z0\,z1\)@VMOD\(z0\,\d+_PRON_[a-z]+\)
1*SUBJ\((VERB[0-9])\,(NOUN[0-9])\)@VMOD\(z0\,\d+_PRON_[a-z]+\)
-->SUBJ\(z0\,\d+_PRON_[a-z]+\)@VMOD\(z0\,z1\)
# « en analysant » --> « analyser »
1*SUBJ\(\d+_VERB_[a-z]+\,(NOUN[0-9])\)@OBJ\(\d+_VERB_[a-z]+\,(NOUN[0-
9])\)@VMOD\(\d+_VERB_[a-z]+\,(VERB[0-9])\)@OBJ\(z2\,(NOUN[0-9])\)
-->SUBJ\(z2\,z0\)@OBJ\(z2\,z3\)@VMOD\(z2\,z1\)
1*SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_VERB_[a-z]+\,(NOUN[0-
9])\)@VMOD\(\d+_VERB_[a-z]+\,(VERB[0-9])\)@OBJ\(z2\,(NOUN[0-9])\)
-->SUBJ\(z1\,\d+_PRON[a-z]+\)@OBJ\(z2\,z3\)@VMOD\(z2\,z1\)
# Idem inverse - the contrary
1*SUBJ\((VERB[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,(NOUN[0-9])\)@VMOD\(z0\,(NOUN[0-
9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,z1\)@OBJ\(\d+_VERB_[a-z]+\,z3\)@OBJ\(z0\,z2\)
1*SUBJ\((VERB[0-9])\,\d+_PRON_[a-z]+\)@OBJ\(z0\,(NOUN[0-
9])\)@VMOD\(z0\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)@OBJ\(\d+_VERB_[a-
z]+\,z2\)@OBJ\(z0\,z1\)
# GV1 + FV2 --> FV1 + GV2 - SYMETRICAL
1*SUBJ\((VERB[0-9])\,(NOUN[0-9])\)@VMOD\(z0\,(VERB[0-9])\)
-->SUBJ\(z2\,z1\)@VMOD\(z2\,z0\)
# « qui viennent » --> « venant »
1*SUBJ\((VERB[0-9])\,\d+_PRON_qui\)@COREF\((NOUN[0-
9])\,\d+_PRON_qui\)@CONNECT\(z0\,\d+_PRON_qui\)-->SUBJ\(z0\,z1\)
1*SUBJ\((VERB[0-9])\,(NOUN[0-9])\)@COREF\(z1\,z1\)@CONNECT\(z0\,z1\)
-->SUBJ\(z0\,z1\)
```

```
1*SUBJ\((VERB[0-9])\,\d+_PRON_qui\)@COREF\(\d+_PRON_[a-
z]+\,\d+_PRON_qui\)@CONNECT\(z0\,\d+_PRON_qui\)
-->SUBJ\(z0\,\d+_PRON_[a-z]+\)
1*SUBJ\((VERB[0-9])\,\d+_PRON_[a-z]+\)@COREF\(\d+_PRON_[a-z]+\,\d+_PRON_[a-
z]+\)@CONNECT\(z0\,\d+_PRON_[a-z]+\)
-->SUBJ\(z0\,\d+_PRON_[a-z]+\)
# Idem, inverse - the contrary
1*SUBJ\((VERB[0-9])\,(NOUN[0-9])\)
-->SUBJ\(z0\,z1\)@COREF\(z1\,\d+_PRON_qui\)@CONNECT\(z0\,\d+_PRON_qui\)
1*SUBJ\((VERB[0-9])\,(NOUN[0-9])\)
-->SUBJ\(z0\,z1\)@COREF\(z1\,z1\)@CONNECT\(z0\,z1\)
1*SUBJ\((VERB[0-9])\,\d+_PRON_[a-z]+\)
-->SUBJ\(z0\,\d+_PRON_[a-z]+\)@COREF\(\d+_PRON_[a-z]+\,\d+_PRON_[a-
z]+\)@CONNECT\(z0\,\d+_PRON_[a-z]+\)
1*SUBJ\((VERB[0-9])\,\d+_PRON_[a-z]+\)
-->SUBJ\(z0\,\d+_PRON_[a-z]+\)@COREF\(\d+_PRON_[a-
z]+\,\d+_PRON_qui\)@CONNECT\(z0\,\d+_PRON_qui\)
#########
NOUN
#####
# NP + PP --> NP + FV + PP
1*NMOD\(\d+_PRON_[a-z]+\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_qui\)@COREF\(\d+_PRON_[a-
z]+\,\d+_PRON_qui\)@VMOD\(\d+_VERB_[a-z]+\,z0\)@CONNECT\(\d+_VERB_[a-
z]+\,\d+_PRON_qui\)
1*NMOD\((NOUN[0-9])\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-
z]+\,\d+_PRON_qui\)@COREF\(z0\,\d+_PRON_qui\)@VMOD\(\d+_VERB_[a-
z]+\,z1\)@CONNECT\(\d+_VERB_[a-z]+\,\d+_PRON_qui\)
# IDEM, mais compte-tenu de la resolution anaphorique du PronPers
# the same, but with anaphoric resolution of the PronPers
1*NMOD\((NOUN[0-9])\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,z0\)@COREF\(z0\,z0\)@VMOD\(\d+_VERB_[a-
z]+\,z1\)@CONNECT\(\d+_VERB_[a-z]+\,z0\)
1*NMOD\(\d+_PRON_[a-z]+\,(NOUN[0-9])\)
-->SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)@COREF\(\d+_PRON_[a-
z]+\,\d+_PRON_[a-z]+\)@VMOD\(\d+_VERB_[a-z]+\,z0\)@CONNECT\(\d+_VERB_[a-
z]+\,\d+_PRON_[a-z]+\)
# Idem inverse - the contrary
1*SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_qui\)@COREF\((NOUN[0-
9])\,\d+_PRON_qui\)@VMOD\(\d+_VERB_[a-z]+\,(NOUN[0-
9])\)@CONNECT\(\d+_VERB_[a-z]+\,\d+_PRON_qui\)-->NMOD\(z0\,z1\)
1*SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_qui\)@COREF\(\d+_PRON_[a-
z]+\,\d+_PRON_qui\)@VMOD\(\d+_VERB_[a-z]+\,(NOUN[0-
9])\)@CONNECT\(\d+_VERB_[a-z]+\,\d+_PRON_qui\)
-->NMOD\(\d+_PRON_[a-z]+\,z0\)
# IDEM inverse, mais compte-tenu de la resolution anaphorique du PronPers
# the same, but with anaphoric resolution of the PronPers
1*SUBJ\(\d+_VERB_[a-z]+\,(NOUN[0-9])\)@COREF\(z0\,z0\)@VMOD\(\d+_VERB_[a-
z]+\,(NOUN[0-9])\)@CONNECT\(\d+_VERB_[a-z]+\,z0\)-->NMOD\(z0\,z1\)
1*SUBJ\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)@COREF\(\d+_PRON_[a-
z]+\,\d+_PRON_[a-z]+\)@VMOD\(\d+_VERB_[a-z]+\,(NOUN[0-
9])\)@CONNECT\(\d+_VERB_[a-z]+\,\d+_PRON_[a-z]+\)
-->NMOD\(\d+_PRON_[a-z]+\,z0\)
# N+ que + V --> N + Adj(VPartPas)
1*SUBJ\((VERB[0-9])\,(NOUN[0-9])\)@OBJ\(z0\,\d+_CONJQUE_que\)@COREF\((NOUN[0-
9])\,\d+_CONJQUE_que\)@CONNECT\(z0\,\d+_CONJQUE_que\)-->NMOD\(z2\,z0\)
```

```
1*SUBJ\((VERB[0-9])\,\d+_PRON_[a-
z]+\)@OBJ\(z0\,\d+_CONJQUE_que\)@COREF\((NOUN[0-
9])\,\d+_CONJQUE_que\)@CONNECT\(z0\,\d+_CONJQUE_que\)-->NMOD\(z2\,z0\)
# l'inverse - the contrary
1*NMOD\((NOUN[0-9])\,(VERB[0-9])\)
-->SUBJ\(z1\,d\+_NOUN_[a-
z]+\)@OBJ\(z1\,\d+_CONJQUE_que\)@COREF\(z0\,\d+_CONJQUE_que\)@CONNECT\(z1\,\d
+_CONJQUE_que\)
1*NMOD\((NOUN[0-9])\,(VERB[0-9])\)
-->SUBJ\(z1\,d\+_NOUN_[a-z]+\)@OBJ\(z1\,z0\)@COREF\(z0\,z0
\)@CONNECT\(z1\,z0\)
1*NMOD\((NOUN[0-9])\,(VERB[0-9])\)
-->SUBJ\(z1\,d\+_PRON_[a-
z]+\)@OBJ\(z1\,\d+_CONJQUE_que\)@COREF\(z0\,\d+_CONJQUE_que\)@CONNECT\(z1\,\d
+_CONJQUE_que\)
1*NMOD\((NOUN[0-9])\,(VERB[0-9])\)
-->SUBJ\(z1\,d\+_PRON_[a-z]+\)@OBJ\(z1\,z0\)@COREF\(z0\,z0
\)@CONNECT\(z1\,z0\)
```