Verbs in specialised corpora: from manual corpus-based description to automatic extraction in an English-French parallel corpus

Natalie Kübler, CIEL, Université Paris 7–Denis Diderot, 2, place Jussieu,
F-75251 Paris Cedex 05, kubler@ccr.jussieu.fr
Cécile Frérot, ERSS, UMR 5610, Université de Toulouse-Le Mirail,
5, allées A. Machado, F-31058 Toulouse Cedex, frerot@univ-tlse2.fr

This paper tackles the issue of verbs in specialised corpora in the view of term extraction. Corpus-based manual descriptions to be used in various applications have highlighted the "deviant" uses of verbs in specialised corpora compared with general uses as well as the need for verb extraction. However, very few attention has been given to verbs both in the terminology theory and automatic term extraction. In the light of a manual corpus-based description, we investigate the status of verbs in an English-French (highly specialised) parallel corpus and advocate a verb-oriented analysis in the framework of a corpus-based parser adapted to verb extraction. Section 1 deals with the status of verbs in the terminology theory; section 2 introduces the framework of the experiment and focuses on the characterisation of the parallel corpus in the domain of Computer Science. Section 3 is dedicated to the corpus-based manual description. Finally, section 4 introduces a corpus-based automatic analysis.

1. Status of verbs

In the terminology theory, the status of verbs has been put aside for a long time. Only in recent years have terminologists and lexicographers started to study the issue of defining verbs as terms for lexicographic descriptions, term base creation, or ontology building. Therefore, studies of verbs as terms have been sparse until now. However, the need for categorising verbs as terms has slowly emerged with the growing use of electronic corpora in term extraction and phraseology description. Potential applications of term bases have raised the hypothesis that terms were not only nouns, as the rule was in the Wüster (and domineering) approach. A "textual terminology" approach, developed by (Bourigault and Slodzian 1999), and based on the use of electronic corpora, opened the way for such questioning.

Descriptions of Language(s) for Specific Purpose(s) (LSPs) hardly consider the status of verbs. This is particularly true of the Computer Science (CS) field - one of the most frequent LSP taught in France Numerous textbooks provide students with a description of CS English. However, the description focuses on noun terminology – multi-word nouns are for instance widely described - and on specific grammatical features used in CS English, such as the passive or comparison. Very few attention is dedicated to the sentence, i.e. to verbs and their distributional and transformational properties. Specialised dictionaries or bilingual glossaries mostly focus on nouns. When verbs are mentioned, they are usually described as derived from nouns, with no other information than part of speech. Bilingual glossaries are likely to give translations of verbs, but with very little syntactic and semantic information on how to use them in a sentence.

The need for describing verbs as terms has been experienced in various applications. (Kübler and Foucou (to appear)) show that French learners of CS English face major comprehension and production problems that are very often related to specialised verbs. Building corpus-based teaching applications of CS English requires a full description of the specialised verbs of the domain. (Kübler 2002) shows the need for verb description in specialised translation, especially when machine translation is used by non-programming translators. This need raises the issue of sorting out "verb terms"[1] from general verbs in specialised texts. In the lexicographic area, (L'Homme 1993, 1998) tried to forge definition criteria for verbs as terms in the French general CS domain. The first criterion consists in considering that if the arguments of the verb are terms, there is a high probability that the verb is a term too. This criterion is however based on the *a priori* characterisation of a term. (Pearson 1998) suggests that the context is the only way of making the difference between a term and a word. (Frérot 2001) has shown that arguments belonging to the general lexicon could indicate the presence of a verb term, by collocating with the verb, revealing a particular behaviour which is different from its general use. The second criterion defined by L'Homme considers verbs that are related to other lexical units that have already been identified as terms. The issue of the *a priori* term determination remains unsolved. Furthermore, verbs and nouns terms that are morphologically related are not always terminologically related. The analysis undertaken in this paper takes those criteria into account, but goes further in considering the syntactic differences the verb terms exhibit compared with their general use. The syntactic criterion has revealed to be most important as general language intermingles with LSPs in specialised texts. Most verbs that can be found in the general language have specific syntactic behaviours in LSPs and are subject to very specifically defined semantic restrictions (Frérot 2001).

[1] We will use the "verb term" expression to refer to verbs that can be considered as terms.

Consequently, both verb analysis and description are crucial and terminology acquisition tools are therefore expected to yield verb-related results. In designing Syntex, a corpus-based parser used to generate lexical resources from specialised corpora, (Bourigault and Fabre 2000) have taken it into consideration and have extended automatic extraction to verbs and verb phrases. Indeed, a verb-oriented analysis can improve terminology extraction as it helps to better identify syntactic dependencies in the sentences of a corpus as well as it enhances the distributional analysis (*i.e.* the grouping of words and phrases appearing in similar syntactic contexts) used for the construction of semantic classes.

2. Framework of the experiment

Corpus linguistics has now proven to yield reliable – and sometimes surprising – linguistic information. However, the need for automating linguistic information extraction has increased, as globalization raises the issue of multilingual document processing. Term extraction will thus become a key issue for the language industry, helping to build multilingual databases or ontologies for the semantic Web applications.

Term extraction not only deals with well-written texts, revised and corrected by language professionals. It will have to deal more and more with "naturally-occurring" texts, showing ill-formed language and various genres. For this reason, we decided to choose a corpus that was very specialised, and provided a good example of "naturally-occurring" texts: the Linux *HOWTOs*, which are the "user manual" of the Linux operating system. Those texts have not been written by language professionals and are highly specialised. They have been written in English and translated into many languages, among which, French. Aware of the needs for bilingual (but also multilingual) terminologies, we worked on the English and French versions. The sample we used consisted of 200,000 words in either language. To check for uses in case of doubt, we also used the full *HOWTO* corpus, which means ca. 500,000 words in each language; the Internet RFCs (ca. 8.5 million words) were also used to double-check some uncertain uses. To make sure a verb structure was "deviant", i.e. specific to the subject area compared with the "general" language, we also used French and English newspapers (one year of *The Herald Tribune* and of *Le Monde*).

2.1. Characterisation of the parallel corpus in Computer Science

The *HOWTOs* are highly specialised texts written by CS experts, who are not always English native speakers. The documents are aimed at CS experts[2], and therefore do not address a wide audience. The communication context can be that of an expert to expert communication and share the following parameters:

> ***"It is assumed that author and reader share a common language and that when certain words or phrases are used, each understands what is meant […]. Writer and reader, or speaker and hearer, are assumed to have the same level of expertise […]. This expert to expert communication context is likely to be the one with the highest density of terms" (Pearson 1998).***

The *HOWTOs* show therefore a high percentage of terms and present features that are specific to very specialised CS texts, such as command names, code lines, URLs,e-mail addresses, etc.. Below is an example of the type of highly specialised sentences that can be found :

EN: *Java API with c-tree Plus' ISAM functionality gives Java functionality through native methods/RMI.*
FR: *L'API Java associée aux fonctionnalités ISAM c-tree Plus permet des fonctionnalités Java au travers de méthodes natives/RMI.*

The English *HOWTOs* – the source texts – do not show the consistency and quality one can find in texts written by technical writers. The syntax is sometimes shaky, as is the idiomaticity of some documents written by non native speakers. No guideline, such as simplified English, has been used to avoid ambiguïties , such as e.g. *cable and ADSL connection*, instead of *cable connection and ADSL connection*. The French translations are not made by language professional either, but by experts in the field. The translations are not consistent with each other, providing thus several different translations for the same use of one term. As the translations are made by different people who are not professional translators, they can widely differ. As is often the case among French computer scientists inside their work environment, English terms are used instead of the recommended French translation (cf. expert FR: *disquette de boot* for EN: *boot disk,* instead of FR: *disquette d'amorce*). Another common feature consists in using the English verb and adding a French verbal suffix, instead of using the French equivalent, as the following examples show :

English verb term	French verb term equivalent	English verb with French suffix
to boot	*démarrer/amorcer*	*booter*
to telnet	*se connecter par telnet*	*telneter*

[2] One must be familiar with Linux, or at least UNIX systems, to really understand the *HOWTOs* and make best use of those.

This corpus is a good example of what must be really dealt with, i.e. texts that more closely reflect the performance aspect of the language, than the competence of the ideal hearer-speaker. Although analysing the *HOWTOs* do not really mean studying the performance of the "speakers" – the circumstancial characteristics of the communication situation are lost in written texts[3] – some typical performance features can be detected, such as typos[4]. However, ill-formed words or sentences, and the use of various types of translations for the English terms, cannot be explained only by performance mistakes, Hymes[5] approach of *communicative competence* is best adapted here to justify the non-standard linguistic structures and translations that are observed. Any application, based on the chomskyan definition of competence will not be able to correctly deal with this kind of texts.

2.2. From corpus-based manual description to automatic analysis

As mentioned above, term extraction has become a key issue in language industry, because of the increasing need for corpus-based multilingual databases, or ontologies. Corpus-based manual description have allowed linguists and language professionals to unveil actual language behaviour, and to dispose of real and statistical language data. However, corpus-based manual description requires an investment in time and energy that industry cannot afford. Hence the raising need for automating processes that have been tested by linguists. Section 3 describes the methodology applied for corpus-based manual description, highlighting the value of the resulting linguistic information, but also the weight of such a heavy and time-consuming task.

3. Corpus-based manual description

3.1. Methodology

Using a concordancer allowing perl-like regular expressions on a corpus that was not POS tagged, we extracted all entries that could be verbs. Heuristics were applied to extract verb candidates, such as words ending in *–ing* or *–ed*, preceded by auxiliaries, modals and *to*. A concordance was then processed for each verb in English. As the English and French *HOWTOs* are aligned in the concordancer interface, we studied the French equivalents for each English verbal occurrence. As will be shown, one English verb term may have several French translations that not always depend on different uses, but also vary depending on the translator. The concordances allowed us to analyse the syntactic contexts of the verbs identifying the arguments in the different syntactic positions, in order to build semantic classes.

The theoretical and methodological approach we used as a tool to analyse verb structures is based on the *lexique-grammaire*[6], which describes each verb via a basic sentence and divides them into classes, according to their basic structures and their common transformational and distributional features. However, corpus observations show that this approach does not take into account some particular features, such as syntactic constraints, which restrict the cooccurrence of arguments. Moreover, a general syntactic description using basic semantic features, such as *human, abstract, place* etc., is obviously unsatisfactory for the description of LSPs, whatever the use of the description is. The classes of arguments that take the different syntactic positions must be described extensively, hence the necessity of automatic term extraction. However, using regular expressions, we tried to extract lists of potential arguments for verbs. Below is an example of extraction for the verb *to run* that shows that there is still noise in the result:

run\w* \w+[^\.] \w+[^\.] \w*[^\.] (?:on|under|at) \w+ .{0,30}
run fdisk or cfdisk on it for you. Of the two, cfdisk is
runs to control everything on a machine, AND one is run per
run the following program on the client: #include <stdio.h> #
run the serial interface at a FIXED speed whilst allowing
run your NIS slaves on a Linux box? Or perhaps your

The manual description made use of comparison with general language (*The Herald Tribune + Le Monde*), in order to check the degree of specialisation of some verbs, using thus the contextual criterion to decide on the term status of verbs. Although newspapers are not completely representative of the language in general, they are general enough to give good hints on the degree of specificity of a term.

3.2. Some results

[3] such as hesitations, cuts, repeated segments of a sentence that are typical of spoken situations.
[4] such as (a) characters inversion, (b) missing character, or (c) replaced character, e.g. (a) **vebr, (b) *veb,* or (c) **vern* for *verb*.
[5] see Hymes D. (1972) On Communicative Competence. in J.B. Pridde and J. Holmes (eds) *Sociolinguitics: Selected Readings*. Baltimore: Penguin
[6] see Gross M. (1975) *Méthodes en syntaxe.* Paris: Hermann.

Let us take the description of the verb *to run*. Reference manuals give very few indications on its various uses. Dictionaries of computing [7] do not mention it. The *Merriam-Webster's* gives only one use which is related to computing : *to run a problem through a computer*. This use is also mentioned in the *Collins-Cobuild*, but along with another one : *You don't need a degree in mathematics to run (= operate) a computer*. A quick check in the *HOWTOs* and *RFCs* corpora yielded only four occurrences of *run something through* in the *HOWTOs*, and none in the *RFCs*. Moreover, the arguments of *to run* do not match with the ones found in the dictionaries :

Dictionaries : *To run a problem through a computer*
Corpus : *If you run your file through TeX program*

Bilingual dictionaries gave us the following translations : *exécuter, passer, fonctionner, être en marche,* and *utiliser*. Manually analysing the occurrences of *to run* in the corpus showed us that there are other translations in use. The information that could not be found in the dictionaries included syntactic and semantic properties. Translation inconsistencies are also noticeable in the corpus, as different French verbs are used for the same English term, without any linguistically motivated reason.

To run shows a basic syntactic structure with three arguments, that does not exist in general English. Two French equivalents are possible:

N_0 runs N_1 Prép N_2 <=> *N_0 (lance + exécute) N_1 Prép N_2*
N_0 =: Nhum + applications that boot, such as LILO
N_1 =: command name + programme
N_2 =: machine, platform + programme + operating system

(1) *as the ability to run different programs in different virtual terminals*
comme la possibilité de lancer des programmes différents dans différents terminaux virtuels

(2) *It just runs a command, which could be any Linux sound system*
Il ne fait qu'exécuter une commande qui pourrait être n'importe quel programme de son sous Linux

(3) *So you write 32-bit code that runs in 16-bit mode on a 32 bit CPU.*
vous écrivez donc du code 32 bits, qui s'exécute en mode 16 bits sur un processeur 32 bits.

Arguments can change depending on the preposition:

N_0 runs on N_1, with : *N_0 =: applications + operating system (Linux, Win95, X-Window), N_1 = : machine, platform (PC, 21066,) + operating system*

(4) *VirtuFlex runs on standard Unix Workstations*
VirtuFlex tourne sur des stations Unix standard

N_0 runs under N_1, with : *N_0 =: applications, N_1 = : operating system*

(5) *ANS FORTH system that successfully runs under Win32s, Win95, Win/NT*
système ANS FORTH 32 bit libre qui fonctionne sous Win32s, Win95, Win/NT

N_0 runs (at +with) N_1, with *N_0 = : Nhum, N_1 = : applications, Prep = :(at + with), N_2 = : N-hum*

(6) *Generally, the PCI runs at 33MHz*
En général, le PCI tourne à 33MHz

A causative construction is possible in French, with the introduction of the operator *faire* :

(7) *You can run Linux on any Alpha-based machine*
Vous pouvez faire tourner Linux sur n'importe quelle machine Alpha

Comparing those examples with general English uses can help isolate technical contexts. The structures described above do not exist in general English; on the other hand, there are structures in general English that cannot be used in CS English, as in *become a presidential concern about running for re-election in 1996* or *stamps, old coins and odd documents, run around the square*. The same does not apply to French *tourner* as in *quatre poules blanches tournant en rond sur une place de village,* since the structure is similar to the specialised use. There are other examples in general French that are not in use in CS French, such as *Quant au cachet de Barbra Streisand, il tourne autour de 20 millions de dollars*. The drawback of this kind of description is that it does not explicitely shows the different structure combinations that are possible, as will be shown in section 4.

Another significant example is the verb *boot. To boot* is quite frequent in the corpus (around 700 occurrences). However, general and even specialised dictionaries give little information on this verb. In the on-line *Merriam-Webster's*[8] only general uses of *to boot* can be found: *to avail, to profit*. The *Collins-Cobuild* offers

[7] *FOLDOC, A Glossary of Computing Terms, Dictionary of Computing For Learners of English*
[8] http://www.m-w.com

no verb entry for *to boot*. *Wordnet*[9] provides some information on the specialised use of the verb (n°2 below), but with very little syntactic semantic information:

1. Boot : *kick ; give a boot to*
2. boot *: cause to load (an operating system) and start the initial processes*

The basic structure of the verb, as analysed in the corpus, has three syntactic positions that can be filled in by specific arguments. The subject is the agent of the action:

N_0 boots N_1 Prep N_2, with the following argument classes :

- *N_0 =: Nhum* or applications such as *LILO* which work as a metaphor, as they can be attributed the agent role.
- *N_1* = : *operating system, system, disk, bootdisk, hard disk, floppy disk, kernel* => all bootable objects
- *N_2* = : *CD, CD-ROM, D :, C :, A :, file, emergency disk* => booting objects

Three prepositions are possible with that structure: *off, with, from*. An idiosyncratic use of the phrasal verb *to boot* off has been detected (4) :

(1) *To boot one of your old kernels off the hard drive…*
Pour lancer l'un de vos vieux noyaux à partir du disque dur…
(2) *A good idea might be to boot the notebook with a kernel*
Une bonne idée serait de démarrer le portable avec un noyau
(3) *In order to have LILO boot Linux from OS/2 Boot Manager,*
Afin que LILO lance Linux à partir du gestionnaire de démarrage d'OS/2,
(4) *You can boot off of a floppy disk* *Vous pouvez démarrer à partir d'une disquette*

PP deletion, allowing a transitive sub-structure with arguments restrictions can be found. As *to boot* is an ergative verb, an intransitive structure in which the subject argument is can be analysed as the patient affected by the action is allowed. In this case, the French equivalent is a pronominal structure, which is very often used to translate English passives:

(5) *When Linux boots, it is usually configured not to produce…*
Quand Linux se lance, il n'est habituellement pas configuré pour produire…

An intransitive structure with *to*, and *into* has been found in the corpus:

(6) *Your BIOS may not allow you to boot directly to a SCSI drive.*
Votre BIOS ne vous permettra peut-être pas de démarrer directement à partir d'un disque SCSI

The syntactic and semantic properties described above show the difference between the neologism *to boot* and the general verb, that has no etymological relationship with the specialised one. The general verb behaves very differently. Here are two examples extracted from one year of the *Herald Tribune* that speak for themselves:

In early 1988 the Saudis booted out Hume A. Horan
eating habits under control by booting the French chef and his staff.

French examples extracted from the "general" corpus of *Le Monde* show immediate differences in the use of *lancer* (as a translation of *to boot*):

et l hymne fraternel que lance à ce dernier Jérôme Garcin.
MCI se lance dans la bataille des " autoroutes de
Martine Aubry lance le débat sur le partage du temps de travail

The last verb we will exemplify for the corpus-based manual description here is *to dump. The Robert & Collins Super Senior English-French dictionary* gives the following examples and French translations *(*which are not in the corpus) *dump* (Comput) data, file, etc *vider* – to dump to the printer *transférer sur l'imprimante.*

To dump shows transitive locative constructions[10], such as:
N_0 dumps N_1 on N_2, with the following argument classes *:*

(1) *a fast computer […] can dump 32k of data on you*
qu'un ordinateur rapide […] pourra vous inonder de 32ko de données

N_0 dumps N_1 (to + onto) N_2, with the following argument classes :

[9] http://www.cogsci.princeton.edu/~wn/

[10] Guillet A. et Leclère Ch., 1992. *La structure des phrases simples en français*: *tome II: constructions transitives locatives* . Genève: Droz.

- N_0 =: *Nhum*.
- N_1 = : *data, content, memory*
- N_2 = : *disk, file*

(2) *dumps its memory image to disk in executable format*
écrit l'image de sa memoire sur le disque sous format binaire

(3) *you just dump the contents of one disk onto the other*
en copiant directement le contenu d'un disque sur un autre.

The different French translations account for two different uses depending on the preposition (*on* or (*in*)*to*). A query on the *Herald Tribune* corpus shows that the specialised uses are quite different from the general ones, i.e. no occurrence of the preposition *to* :

for defying his demands to dump her boyfriend and cut her long hair
overseas investors will dump Japanese stocks
The storm, which dumped 23 centimeters (9 inches) of snow Saturday in Tokyo,

The same applies to French. General uses of *écrire* and *copier* show significant differences as far as structures and argument classes are concerned:

Ecrire sur la mort d un ami est périlleux.
Réalisé avec de gros moyens, ce film copie Alien sans vergogne.
alors elles nous copient sur les spécialités!"

In this section, the corpus-based manual description highlighted idiosyncratic uses of verbs in our corpus. Even though we used aligned concordances to carry out the analysis, the procedure – as it is not fully automated – was time-consuming and quite hard-working. However, this methodology proved useful to highlight the uses that are specific to CS and do not belong to general language. Therefore, we investigated the possibility of using a corpus-based automatic tool - both for French and English - that would yield verb-related results, and above all, relies on the corpus as much as possible. Section 4 is thus dedicated the a corpus-based automatic analysis. We first point to a few arguments in favour of verb analysis and then give a brief overview of the automatic tool we used. Finally, we show - through Prepositional Phrase (PP) attachment - how endogenous techniques are well-adapted to our verb study.

4. Corpus-based automatic analysis

4.1. A few arguments in favour of verb analysis

As mentioned in Section 1., both verb analysis and description are crucial and terminology acquisition tools are therefore expected to yield verb-related results, though most tools do not go as far as verb output. Let us insist on three (at least) reasons accounting for a verb-oriented analysis.

Reason 1. From a terminological point of view, verbs may be terms just as nouns do, following (Bourigault and Jacquemin 2000) who postulate a verb terminology. Indeed, verbs play a major role in specialised texts and are likely to be terms on the basis that they can be: *(i)* morphologically related to a noun or noun phrase being itself a term such as the French and English examples[11]: *injecter un virus / injection d'un virus, to access the system / an access to the system*; *(ii)* specialised verbs (usually simple verbs), *i.e.* verbs whose use is restricted to a given domain and refer to a specific concept, such as the French verbs *transduire* or *transfecter* in the domain of gene therapy, or the English CS verbs *to telnet, to bufferize*; *(iii)* verbs exhibiting a "deviant" use in the terminological system in comparison with their expected use in the lexical system. In this context, "deviance[12]" refers to the unpredictable verb argument structure both in terms of syntactic and semantic behaviour (Frérot 2001). Let us illustrate that point with the following corpus-based examples : *construire des souris, recruter des cellules, the daemon listens to all the messages.*

Reason 2. A verb-oriented analysis improves automatic term extraction as it helps to better identify the constituents (frontiers) of sentences in a corpus, therefore increasing noun extraction accuracy. Let us look at the following examples:

[11] The corpus-based examples in section 4 are taken from the French-English HOWTO corpus and French corpora in the domain of geomorphology and gene therapy.

[12] « *[…] we postulated deviance as the linguistic characteristic of terms in relation to words. The deviance was described as being of several kinds ; 1. Unusually high frequency of compound verbs. 2. Coinage of new words. 3. Unusual syntactic behaviour : new or forbidden constructions. 4. Unusual semantic behaviour : appareance of new meanings which show themselves by unsual combinations* » (Condamines 1995).

(a) *to boot [a Linux kernel] [on a CD ROM]*

→ *[13] *Linux kernel on a CD ROM*
→ *to boot on a CD ROM / Linux kernel*

(b) *to give [the compiler] [hints] about how to optimize*

→ **compiler hints*
→ *to give hints / compiler*

(c) *enrober [de calcite] [des matériaux]*

→ * *calcite des matériaux*
→ *enrober de calcite / matériaux*

In the above examples, a noun focus alone will not allow a proper analysis of the sentence, as in *(a)* it leads to identify as a potential term *Linux kernel on a CD ROM,* though the preposition *on* depends on the verb *boot* ; the same applies to *(c)* : *des matériaux* is the direct object of *enrober,* the determiner *des* depends on the verb *enrober* (and not *calcite*). In this context, adopting a verb approach obviously reduces the generation of invalid terms (**Linux kernel on a CD ROM* , **calcite des matériaux)*, its correlate being the necessity to deal with verb structures.

Reason 3. Syntactic verb contexts are very productive for the distributional analysis as they enhance the grouping of words and phrases appearing in similar syntactic contexts, which is used in our tool for the construction of semantic classes.

to produce
→ *{basalt, crust, flow, lava, magma}*
to generate

In the above example, the nouns *basalt, crust, flow, lava, magma* are said to form a cohesive semantic class, on the basis that they share similar contexts with the verbs *produce* and *generate.*

In designing Syntex, a corpus-based parser used to generate lexical resources from specialised corpora, (Bourigault and Fabre 2000) have taken those three parameters into consideration and have extended automatic extraction to verbs and verb phrases.

4.2. Syntex : a corpus-based parser adapted to verb extraction

Syntex is a corpus-based parser[14] used to generate specialised lexical resources, such as lexicons for translation, ontologies or thesauri, and has been used in various « real world » applications (among the most recent are (Bourigault and Lame 2002, Le Moigno et al. 2002, Chodkiewicz et al. 2002)). Syntex first[15] identifies lexico-syntactic dependencies in the sentences of a given corpus (for instance, subjects, direct or indirect objects of verbs) and builds a network of words and phrases in which each phrase is linked to its syntactic heads and expansions[16]. The network is then used as a material for the construction of semantic classes on a distributional basis *i.e.* the grouping of words and phrases appearing in similar syntactic contexts (for an accurate description of the distributional analysis module, see (Bourigault 2002)). We will focus here on the extraction of lexico-syntactic dependencies - with an emphasis on PP attachment - as they are the starting point in the whole process and will give an overview of the general principles underlying the analysis.

Syntex's major specificity is to rely on endogenous techniques (Bourigault 1994) which allow the parser to acquire, for every new corpus analysed, the subcategorization information necessary to resolve syntactic attachment ambiguity. This strategy is based on in-depth studies of various domain corpora, highlighting idiosyncratic uses of lexico-syntactic structures compared with their general use and from one domain to another (Fabre and Bourigault 2001, Basili et al. 1997, Basili et al. 1999). In this context, using general linguistic knowledge tends to prove quite inefficient and irrelevant. Consequently, Syntex does not use any *a priori* linguistic resources. Let us illustrate the endogenous strategy on PP attachment, the very first procedure in delimitating phrases.

---

[13] * indicates a wrong analysis (invalid term).

[14] There is a French version of Syntex as well as an English version.

[15] Before Syntex is used, the corpus is morphosyntactically tagged (each word in the corpus is assigned a lemma and grammatical tag).

[16] Example : in the French noun phrase *plan de faille, plan* is the head and *faille* the extension.

- English Example : Verb Noun Preposition

Ambiguous case : *to run programs in virtual terminals*

→ potential governors for the preposition *in :*

program
run } *? in*

In order to solve this ambiguous case, in other words find the preposition's governor, our tool relies on unambiguous cases, *i.e.* cases containing only one governor, such as :

Unambiguous cases : *Dosemu has to be run in another terminal*
compile and run in double precision mode
why not run in a Linux box your NIS slaves

Syntex relies on those unambiguous cases to compute productivity measures used to perform the right attachment :

(run, in {terminal, mode, box}) : corpus-based occurrences
(run, in) : productivity = 3
→ *to run different programs in virtual terminals*

We now illustrate this procedure with a French example.

- French example **:** → *introduire du matériel génétique dans les cellules*

ce qui permet de les introduire dans une cible
a introduit dans une cellule eucaryote de l'ADN
du matériel génétique a été introduit dans l'organisme

(introduire, dans {cible, cellule, organisme}) : corpus-based occurrences
(introduire, dans) : productivity = 3

→ *introduire du matériel génétique dans les cellules*

4.3. First experiment on the HOWTO corpus

4.3.1 A few remarks on pre-processing

As mentioned earlier Section 2.1., the HOWTO corpus is characterised by a vast number of features such as URLs, e-mail addresses, code names, command lines or scripts - among others - that make the task of pre-processing difficult (by pre-processing, we mean sentence and word segmentation as well as morphosyntactic tagging). For instance, the high number of enumerations, in the form of listing, adds to the difficulty of sentence segmentation which becomes even more complex with the "naturally-occurring" dimension – referring to the lack of homogeneity in punctuation, typography, due in part to different people writing the user guide (be it in English or for the French translations).

As far as the morphosyntactic tagging is concerned, the tagset used covers part of the corpus specificities - the *NomMail* and *NomUrl* tags are well-suited to analyse phenomena such as URLs or e-mail addresses (examples : NomMail|timbo@moshpit.air.net.au / NomUrl|http://www.eleves.ens.fr) - though it does not cover all of them. Let us mention also a few more general[17] tagging errors, regarding for instance, ambiguous *ing* forms in English. Taken as a whole, it should be pointed out that the pre-processing quality - from sentence segmentation to morphosyntactic tagging - obviously impacts on the lexico-syntactic analysis.

4.3.2 Verb parsing output

We will now more closely look at the verb analysis - through PP attachment - performed by the parser and emphasise on an English verb sample {*boot, compile, configure, dump, mount, run*} which we believe to best represent the endogenous procedures as well as the type of verbs found in specialised corpora. In Section 3., we showed that the manual corpus-based description highlighted idiosyncratic uses of those verbs compared with their general uses, thus implying the need to use automatic tools that adapt to corpora. Therefore, what we intend to show here is that endogenous procedures are particularly well-adapted to corpora as they "respect" their specificities and use no other information but that of the corpus.

[17] General, as opposed to specific to the HOWTO corpus.

As mentioned earlier., our tool relies on endogenous techniques and more precisely on the corpus productivity, implying lexico-syntactic redundancy. Let's analyse the verb *run* in the corpus, which gets constructed with a wide variety of prepositions as shown below:

ability to run different programs in different virtual terminals at the same time
other subsystems (DRAM, for example) will run asynchronously at lower clock rates
you can run any other program from within emacs
can run the PCI at any frequency
the above tests were run with some of the special Cyrix
Run dosemu with partition access
your application which would probably run under the IBCS2 emulator
to run dosemu inside a color xterm
→ *(a) and runs on all platforms*
→ *(b) VirtuFlex runs on standard Unix workstations with 8 MB of RAM minimum*
→ *(c) why not run you NIS slaves on a Linux box*
→ *(d) to run popular Windows applications on Linux based system software*

Among all the verb occurrence frequency, *run* is one of the highest in the corpus, with approximately 400 occurrences. The more occurrences in the corpus, the more chances for our tool to find unambiguous cases (a) (b), used to resolve ambiguous cases (c) (d). The same remarks apply to the following verbs {*compile, configure, dump, mount*} which get constructed with various prepositions and whose number of occurrences (respectively, 224, 190, 50, 115) is high enough to allow the parser to resolve PP attachment ambiguity. The table below shows the verb-preposition associations and highlights some idiosyncratic uses and missing dictionary-based descriptions, such as *dump back to, compile in support, compile in support for, cross compile from-to*, which the manual verb description had indeed revealed.

Verb	Preposition	Occurrences
compile	against, as, from-to, into, with, in, in support, in support for, out of	*to compile things against this library, to compile as ELF, to cross compile from Linux to Dos, the steps to compile into the kernel, the driver was compiled with debugging enabled, to compile in your own personal values, to compile in support to use the program selection, you need to compile in support for your CDROM drive, it compiles right out of the box on Linux*
configure	as, for, in, in support for, on, to, under, with,	*configuring your machine as an NCP server, the card is configured for shared memory operation, allow the card to be configured in software, do not configure in support for the 82C710, devices to be configured on a Linux machine, the card is not configured to one of the addresses, configure CDROM drive under Linux, you have configured dosemu with a command like $*
dump	on, into, onto, to, back to	*you can dump 32k of data on you without stopping, it will dump everything into a ftape-2., when you just dump the contents of one disk onto the other, dump the image to the disk, dumping its memory image back to disk*
mount	as, for, from, into, on, through, under, with,	*name of the directory to mount as root, to mount a CDROM for read/write, a CDROM is mounted from Linux, to mount Novell volumes into your Linux filesystems, mounting the CDROM on bootup, mount the empty files through the loopback devices, you dos partition is assumed to be mounted under Linux, the root filesystem is mounted with write access*

The automatic analysis of the French translations for {*boot, compile, configure, dump, mount, run*} showed very frequent verb-preposition associations - be they long distance or adjacent - , such as for *boot* : *lancer/démarrer {à partir de, dans, sur}* or *run* : *faire tourner/exécuter {dans, sous, depuis}*. Our endogeous procedures proved efficient[18] to resolve the PP attachment ambiguities. Syntactico-lexical redundancy in the corpus was sufficient for the parser to correctly attach the preposition to the verb. Finally, it should be pointed out that so far, no distinction is made in the parser between arguments and adjuncts of verbs. Whatever the status of PPs - not as clear-cut as often claimed -, they are attached to the verb. This choice is based on the assumption that non argumental relations between verbs and PPs highly contribute to word semantic neighbouring, hence to word semantic class. More generally speaking, corpus-based tools may highlight and bring new insight to linguistic phenomena - as is the case here for the argument/adjunct distinction – that an intuition-based manual description, strongly influenced by categorical models of grammar would not show.

5. Conclusion and future work

[18] We exclude wrong analyses due to tagging errors.

This first experiment has shown the necessity of using corpus-based manual description as an incentive for automatic term extraction. Manual descriptions in LSPs highlight the "deviant" uses that are not found in the general language. However, manual corpus work being time-consuming and since the need for exhaustive linguistics description has increased, automating linguistic information extraction will enable linguists and language professionals to go further in research and applications. The approach implemented in Syntex indeed takes into account the LSPs specificities highlighted by corpus-based manual descriptions. As the use of machine translation has developed in the recent years, bilingual term extraction will be more and more needed to build MT specialised dictionaries, in order to improve translation results. Verb term extraction can also prove very useful in computer-assisted language learning (CALL), leading to automated exercise generation and helping in the correction process. Future work will deal with the complete bilingual extraction of verb terms from the *HOWTO* corpus, in order to test the creation and validity of MT dictionaries.

References

Basili R, Pazienza M-T, Vindigni M 1999 Adaptative Parsing and Lexical Learning. Actes de *VEXTAL'99*, Venise.

Basili R, Pazienza M-T, Vindigni M 1997 Corpus-driven Unsupervised Learning of Verb Subcategorization Frames, Actes du *5ème congrès AI*IA 97*, M. Lenzerini (ed), Lecture Notes in Artificial Intelligence, 1321, pp. 159-170.

Bourigault D 1994 Lexter, un logiciel d'extraction de terminologie. Application à l'acquisition de connaissances à partir de textes, Thèse de l'Ecole des Hautes Etudes en Sciences Sociales, Paris.

Bourigault D 2002 Upery : un outil d'analyse distributionnelle étendue pour la construction d'ontologies à partir de corpus. Actes de la conférence *TALN,* Nancy, 75-84.

Bourigault D, Jacquemin C 2000 Construction de ressources terminologiques. *In :* J-M. Pierrel, *Ingenierie des langues*, Paris : Hermès Sciences Publications, Chap. 9 : p. 215-233.

Bourigault D, Fabre C 2000 Approche linguistique pour l'analyse syntaxique de corpus. *Cahiers de grammaire*, Vol.25, pp.131-151.

Bourigault D, Slodzian M 1999 Pour une terminologie textuelle. *Terminologies nouvelles* 19: 29-32.

Chodkiewicz C, Bourigault D, Humbley J 2002 Making a workable glossary out of a specialised corpus: Term extraction and expert knowledge, in Altenberg B & Granger S. (eds), *Lexis in contrast, corpus-based approaches*, John Benjamins Publishing Company, Amsterdam/Philadephia,, pp. 249-267.

Condamines A 1995 Terminology : new trends, new perspectives. *Terminology* 2:2, 219-238.

Fabre C, Bourigault D 2001 Linguistic clues for corpus-based acquisition of lexical dependencies. Actes de *Corpus Linguistics Conference*, Lancaster, pp. 176-184.

Frérot C 2001 Caractérisation du verbe en terminologie. Application au domaine de la thérapie génique en cancérologie. Mémoire de DEA, Université Paris 7.

Kübler N 2002 Teaching Commercial MT to translators: Bridging the Gap between human and machine. In H. Somers (ed.) *Proceedings of the EAMT workshop on MT*, Manchester, UMIST. pp. 155-162.

Kübler N, Foucou P-Y. (to appear) Teaching English Verbs With Bilingual Corpora : Examples in the Computer Science Area. in S. Granger &S. Petch-Tyson (ed) : *Contrastive Linguistics and Translation Studies*, Rodopi, Amsterdam.

L'Homme M-C 1993 Le verbe en terminologie : du concept au contexte. *L'actualité terminologique* 26(2) 17-19.

L'Homme M-C 1998 Définition du statut du verbe en langue de spécialité et sa description lexicographique. *Cahiers de lexicologie* 73(2) 125-148.

Le Moigno S, Charlet J, Bourigault D, Jaulent M-C 2002 Terminology extraction from text to build an ontology in surgical intensive care, in *Proceedings of the AMIA 2002 annual symposium* (American Medical Informatics Association), San Antonio, USA,

Pearson J 1998 *Terms in Context,* Amsterdam: John Benjamins Publishing Company.