Teaching English Verbs With Bilingual Corpora: Examples in the Computer Science Area

Natalie Kübler [1]
Pierre-Yves Foucou[2]

Abstract[3]

In French universities, most computer science syllabuses include compulsory teaching in English. However, English teachers are not necessarily experts in computing, and textbooks or dictionaries are not complete, and rapidly become obsolete, especially with regards to verbs. Yet it is precisely the English verb system which French-speakers have trouble mastering, particularly in technical areas.
We shall describe how using various types of corpora, such as technical English corpora, aligned English to French corpora, and « general » English corpora has allowed us to achieve two objectives : the
discovery and description of the authentic use of technical verbs; and
the preparation of teaching material. The resultant description will firstly help us to identify more appropriate pedagogic objectives for teaching a specialist's language ; it will then serve in a Web-based language teaching environment to generate learning activities.

0. Introduction

In French universities, English classes are very often included within specialised training, because English is nowadays the mostly used language in the technical and scientific world. English is particularly necessary in the computer science (CS) area because of the impressive and quick expansion of the domain. At the linguistic level, this is translated into a greater productivity in the coining of new terms or new uses of already existing terms. The technical documentation and terminology of most software packages or operating systems is first been written in English. Translating the documentation into other languages raises the issue of the double competence : users must have both linguistic and technical knowledge. This problem is becoming more acute in the teaching of English as a second language.

Observation of real language usage can invalidate conventional, and over-simplifying hypotheses. Let us consider the simple example of navigating on the Internet : different terms are used in the various browsers to describe the same function of memorizing addresses (URLs[4]) : the French notion of *signet* matches *bookmarks, hotlist* and *favorites,* in « Netscape » and « Internet Explorer » respectively. Students can easily acquire these uses, but closely related uses can present some difficulties :

(1) You should bookmark this page now !
(2) *You should favorite this page
(3) Bookmark this page in your favorites !

Furthermore, different translators will not always agree on the translation of a term or an expression : one French translation for *bookmark* is *marque-page*, but the following has also been found:

(4) Bookmarquez cette page ![5]

To allow students to find their way through this ever-changing jargon, it is necessary to teach CS English in a contrastive way by using authentic documents. This permits computer scientists – whatever their technical competence – to feel at ease in English, as well as in French. French translations lead beginners in computer science to a better understanding of technical documentation. More advanced computer scientists should be able to deal with the French terms, whilst they are already used to working with the English terms. That is why translators often give the English term at the beginning of a translated document, and subsequently use the French equivalent throughout. Thus, terms like *chipset* for *ensemble de composants, spool* for *queue* or *file d'attente*, or even *spreadsheet* for *tableur*, can usefully be given at the beginning of a French document because they are already known to French-speakers. In the present article, we describe the pedagogical experiences that

[1] Université de Paris 7 : kubler@ccr.jussieu.fr
[2] Université de Paris 13 : foucou@lli.univ-paris13.fr
[3] We would like to thank A. J. Renouf for her very helpful comments on an earlier version of this article.
[4] *URL* : *Uniform Resource Locator* : from the Free On-Line Dictionary of Computing http://www.foldoc.org
[5] We found around 100 occurrences of this form on Altavista.

took place at the Technology Institute of Villetaneuse at the University of Paris 13. We shall develop one of the most problematic issues for French-speaking learners : mastering CS English verbs. This point is particularly crucial, all the more so since it has often been overlooked in textbooks or specialised dictionaries.
We shall show how available corpora on the Internet can be used to present the students with varied examples, in contexts that are simple, yet encompass all possible structures. The contrastive analysis of bi- or multi-lingual technical documentation can lead to support a description of the same uses in different languages. Using authentic and constantly updated documents introduces a reality component in the description of usage : we aim at describing the verbs that are actually used by a scientific community, rather than the description of terms that have been standardized by an official body. We use the conventional corpus query tools that have been developed at the *Laboratoire de Linguistique Informatique* of the University of Paris 13. These tools have been adapted to the specific needs of language teaching : simple and bilingual concordances, the automated creation of learning activities, and so on.

1. Verbs and Corpora

A pedagogical choice

Confronting French-speakers with CS English can cause them some problems in comprehension and production. Very few verbs are presented in technical dictionary entries; they are often be introduced at the end of a noun entry, without any other information than the part-of-speech (POS) category. It is however these that pose the main problems. Once non-native speakers have acquired a technical term, be it simple, multi-word nouns, or adjectives, they seldom have further problems with it. The more they progress in computer science, the less this type of terms poses problems, because they have acquired the specific terms of their subject area. The difficulties that are encountered, be they on the level of comprehension or production, relate primarily to the verbs, as we noted among French-speaking students, whether they be beginners in English or more advanced.

In our project, we are currently developing a description of the English CS verbs and their equivalents in French. We have divided the verbs into three different categories, which are quite similar to pragmatic approaches to the definition of terms. Hoffman (1985) suggests that there are three categories of terms in a specialised vocabulary : subject-specific vocabulary, non subject-specific vocabulary and general vocabulary. For Trimble and Trimble (1978) , there are highly technical terms, a bank of technical terms, and sub technical terms. While the first two categories are the same as the first two described by Hoffman, the last one covers the terms coming from the general language, but that have taken on a specific meaning in specialised subject areas.

As our aim is slightly different from describing terms for native speakers, we chose an approach which takes into account the point of view of non native speakers, i.e. a pedagogical point of view. Examining the verbs, we noticed that the highly technical verbs (according to Hoffman's first category) are very often neologisms[6] which have to be acquired as such. The second category of verbs partly matches the first and second categories of Hoffman and Trimble and Trimble, since it consists of verbs that already exist in general English, but that have acquired a specialised use. The last group corresponds to both the third category of Hofman and of Trimble and Trimble : it consists of general English verbs that are used in CS English, particularly those that are extremely frequent and that are difficult to master for French-speakers in this subject area.

Our approach has potential for the creation of pedagogical material allowing teachers to present students directly with authentic data, as well as to automatically generate learning activities, such as drills for example. We have indeed a Web-assisted language learning (WALL) environment (Foucou & Kübler 2000), which generates learning activities allowing students to practice acquired knowledge.

1.1. Existing pedagogical material (dictionaries/textbooks ; online/offline)

A great number of textbooks offer descriptions of the specific characteristics of CS English, but these often remain basic. The verb/noun ambiguity, which is typical in technical English, and the great versatility in the creation of new terms are rarely mentioned. Very few indications are given about the sentence, i.e., the verbs structures and their distributional and transformational properties. As far as translations are concerned, CS English verbs and their equivalents in French are frequently described as lists that are unfortunately not always complete, and do not contain information about the different contexts of use, leaving the user to guess which translation must be used in which context.

[6] This is not surprising as computer science is producing new concepts almost everyday, especially with the development of the Internet.

General dictionaries are genrally sparing in their inclusion of CS terms (which is not their primary function, as they are not specialised dictionaries), and specialised dictionaries are often incomplete (for non native speakers) or become very quickly obsolete. The information provided by these two types of dictionaries is not very useful, given the real nature of texts. This explains why it is necessary to resort to more current reference sources. We agree with Pearson (1998), for whom the context is the only way of making the difference between a term and a word. This means here that we shall use corpora to decide whether a verb should be described or not.

CS dictionaries focus on nouns and their meanings, as well as their possible translations in French (in bilingual glossaries). Beginners and French-speaking students in computer science (such as French university students in the first two years) will find definitions, which are sometimes encyclopedic, in *FOLDOC* (*Free On-Line Dictionary Of Computing*[7]*)* or in other CS dictionaries. Students are faced with the same type of explanations and French translations of the terms in the various bilingual dictionaries that can be found on the Web[8].

- Numerous specialised acronyms are found in dictionary entries. Three types of acronyms can be found in bilingual dictionaries :

  - Acronyms that are translated into French, such as *ISDN* (*Integrated Service Digital Network*) translated in *RNIS* (*Réseau Numérique IntIntégré de Service*).
  - Acronyms of which only the expansion is used in French, such as *OS* (*Operating System*), which is translated by *système d'exploitation*, but for which the French acronym *SE* is very rarely used except among purists.
  - Finally, acronyms that do not have a translation in French, such as *SCSI* (*Small Computer Interface System*), or *MSDOS* (*Microsoft Disk Operating System*).

- Dictionaries also contain some very specialised modifiers, such as *controller-less, big or little endian*.

$$

1.2. Difficulties of French-Speakers

We have noticed among French-speaking learners several types of difficulties which are related to the verb system in English.

- Verb/noun ambiguity (nominal use of verbs and vice-versa) : It can be difficult for students to distinguish a verb from a noun ; for a native speaker of English the context alone is enough to make the difference, which is not the case for a non native speaker. This is all the more difficult since French-speakers often do not know how easily and frequently verbs can be created from nouns, (such as *to zip* out of *zip*, a program used to compress data) or nouns from verbs (such as *a login* based on the verb *to log in*). Moreover, some English verbs have no direct equivalent in French, and are translated by paraphrases, or support verbs and their predicate nouns (collocations).

- Polysemy : Some extremely polysemous English verbs can pose comprehension or structural problems for French-speakers. *To run* is a good example ; on one hand, its various uses are variously translated into French, on the other hand, some of its structures are determined by the possible arguments of the verb.

- Structural differences between French and English : Structure differences among very similar verbs in the two languages are often the cause of interference errors for French-speakers (Kübler 1995). This is also the case in CS English.

The teaching of CS English cannot be achieved without a description of verbs and their structures. Unfortunately, it is exactly this type of description that is missing in textbooks. It can however be extracted from corpora. A thorough description of CS verbs appears to be necessary, not only for teaching, but also for other applications, such as automatic error correction or automated translation systems.

---

[7] See footnote 4.

[8] http://www2.echo.lu.edic/EURODICAUTOM
http://web.culture.fr/culture/dglf/internet
http://www-rocq.inria.fr/qui/Philippe.Deschamp/CMTI/glossaire.html

2. Identifying problematic verbs

2.1. Specialised, general, and parallel corpora

The fast development of the World Wide Web opens up access to ever expanding ressources in terms of corpora. Using technical documentation which is exclusively related to the real world has the advantage of introducing an authentic component ; its importance has been highlighted for years in the literature on this subject (T. Johns 1988). In order to describe the reality of CS English, we chose as a working corpus the *Linux HOWTOs* (half a million words). The *HOWTOs* represent an easy to access and regularly updated technical documentation that has the advantadge of being multilingual. They have been translated into several languages, including into French.

In order to be thorough, we sampled other corpora. Texts relating to computer science offer a wide variety of styles and levels of language. We chose to use a representative sample of different possible styles. Our corpora have been extracted from the almost inexhaustible ressources offered by the *World Wide Web*, and divided into five categories :

i) Technical Documentation

- user's manual of the *UNIX* operating system (250 documents, 16 MB, 53300 types)
- the *Internet RFCs* which are the instructions for use of the *Internet* (2000 files, 85 MB, 161083 types)

ii) Specialised On-line Press

Wired : computer science magazine (1000 articles, 5MB, 38392 types)

iii) Newsgroups

Newsgroups deal with various aspects of computing ; the level of language is quite casual, and can be, at the same time, extremely casual, as shown in the following example, which has been extracted from the *comp.lang.perl.misc* newsgroup :

(5) You should either use double quotes or joins, but not both :
Either :$file = '../dir/dir/dir/'. $country.'_' $machine ;
Or, preferably (at least to me) :
$file = « ../dir/dir/dir/$country_$machine ;
should be :
$file = « ..dir/dir/dir/${country}_$machine » ;

Our newsgroups contains, for the time being, approximately a thousand articles (ca. 6500 types).

iv) FAQs (Frequently Asked Questions)

FAQs are often related to some *newsgroup* and consist of files that contain the most frequently asked questions on a given subject. For example *FAQs* about the following subject are available : *Y2K bug, Solaris OS,* or even *Windows*.

v) « General » English

To relativize the results and examine them from different angles, we use « general English » corpora, such as *The Times* (3'500'000 words), or *The Herald Tribune* (1'500'000 words). Other CS English corpora allow us to check specialised uses, « general English » corpora are used to verify the degree of specialisation of the selected verbs.

2.2. Frequencies

A first sampling of our corpus permitted us to obtain a list of the most frequent verbs. In the highest frequencies of the *HOWTOs*, the first three verbs (once auxiliaries and modals were discarded) are the following :

***Use* 3114 occurrences**	***run* 1565 occurrences**	***install* 1163 occurrences**
Using 1726 *Used* 1192 *Use* 196 (partly nominal occurrences)	*run* 886 *running* 523 *runs* 140 (i.e. a very low percentage of nouns)	*install* 662 *installed* 369 *installing* 132

The number of occurrences very quickly drops to a few hundreds (*to boot* has around 500 verbal occurrences), or even less than a hundred (*to download* has around 40).
These results can be compared with the frequencies in *The Times* where the three most frequent verbs are *use, run* (general English uses, and not CS English), and *call* :

***Use* 30324 occurrences**	***run* 26697 occurrences**	***call* 13771 occurrences**
Used 13333	*run* 12773	*called* 12445
Use 11363	*runs* 4517	*call* 5922
Using 4333	*running* 6541	*calls* 3601
Uses 1295	*ran* 2866	*calling* 1793

The frequencies in the French corpus, i.e. the French translations of the *HOWTOs*, are surprisingly different. The most frequent verb is *utiliser* with more than 2000 occurrences ; the next verb *fonctionner* plummets to around 300 occurrences, and the rest are even rarer. This shows that French translations of verbs are different depending on the uses ; among the various uses of *to run*, one is translated by *fonctionner*, which is also the translation of *to work*. Using several verbs to translate one only term reduces the frequencies of French verbs.

For this reason, describing and teaching the most frequent verbs is not satisfactory. Among the less frequent verbs in the references corpus are verbs that must be taught because they are especially difficult for French-speakers.

Our concordancer allows us to query the corpus on character strings or with perl-like regular expressions containing syntactical categories such as nouns, verbs, adjectives, etc. As shown in Figure 1, the perl-like regular expression *(have|has) \w+ed* looks for two sequences of words : either *have* or *has* followed by a word ending in *–ed*. This search string defines occurrences of present perfect verb forms :

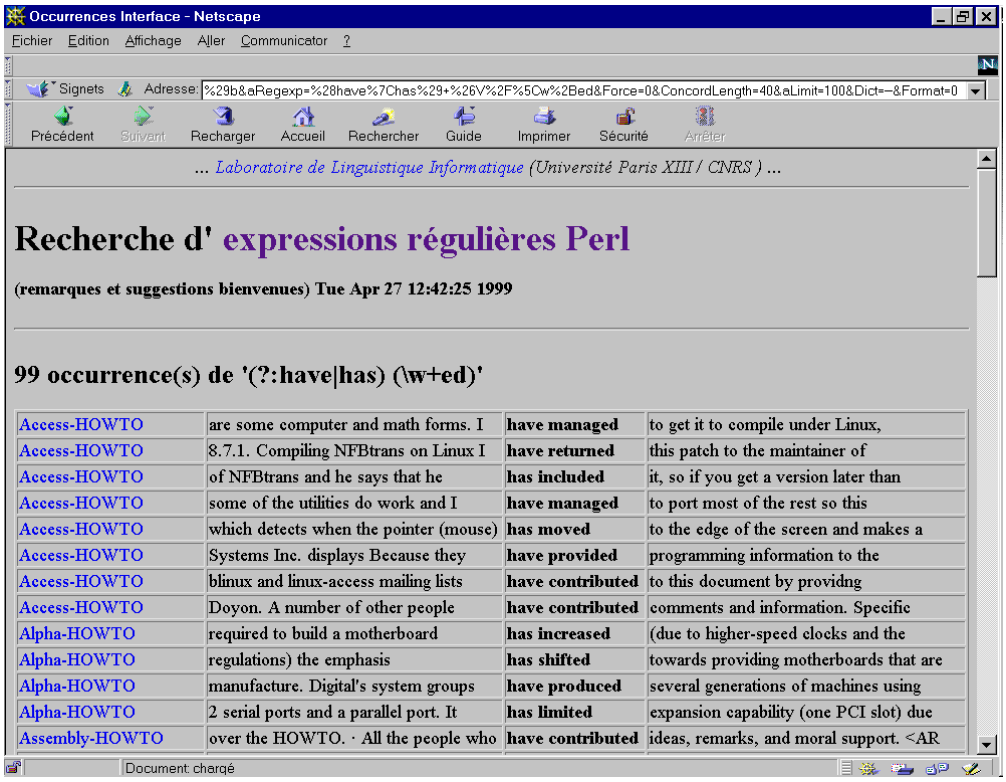


Figure 1 : Present perfect occurrences

A first query searching for all the terms that can be considered as verbs provided us with a more precise list than just the frequency list. This query is important because of the great differences existing between French and English. We picked out verbs like *to mirror* or *to cache* which are not frequent in the corpus (less than a hundred occurrences each), but which can cause difficulties, since there are no verbal equivalents in French. **Miroirer*[9] or *cacher* are not good candidates.

A second type of query dealt with the context in which each verb can be found individually, in order to extract their distributional and transformational properties. These examples of concordances were also edited for presentation to the students. What was at stake consisted in making the students aware of the verbs behaviour via the contact with authentic data. Data-driven approaches for language teaching often recommend comparing the examples extracted from a corpus with the descriptions that can be found in reference books (B. Dodd 1997 in Wichman et al. for example). This is not possible to achieve with CS English as there are no descriptions of CS English verbs. The comparison with the general English uses, however, can lead to extract specialised English verbs.

Our reference corpus *Linux HOWTOs* has been translated into different languages. Our English corpus can be aligned with its French translations. The French and English corpora were aligned, paragraph by paragraph by a perl script developed with, and included into, our series of tools (the Wall environment). Since the alignment is not always perfect (translators can decide to add or delete sections), the corresponding paragraph can then be manually searched for. Our tool allows the user to query either one of the corpora (cf. Figure 1) and then to search, for each occurrence of a verb, for its equivalent in the other language (cf. Figure 2).

[9] Words preceded by an asterisk do not exist in the given language.

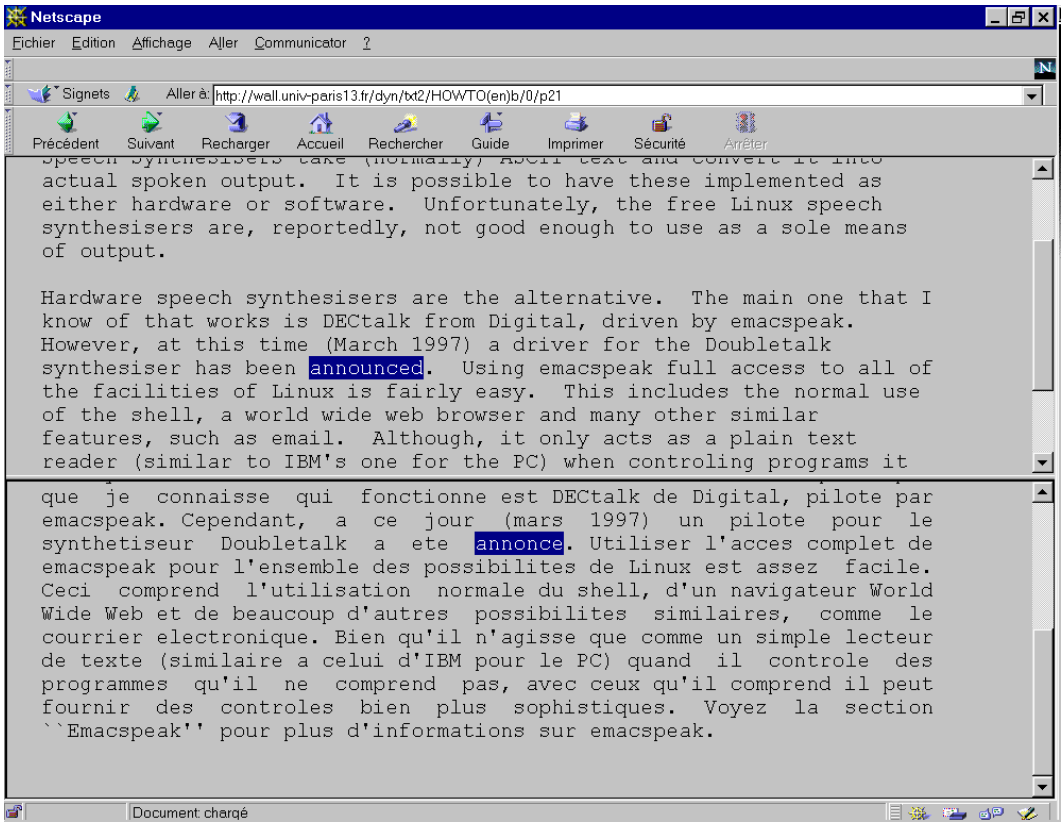


Figure 2 : Aligned paragraphs for « announce »

After examining concordances to discriminate between the different uses of each verb, we looked for the possible French translations for each use in the French translated corpus. Our aim consisted, on one hand, in refining the description of English verbs, and on the other hand in matching the different French equivalents. We did the same with the French corpus : analysis of the different uses, and searching for the English equivalents.

3. Illustration with a few verbs

We show here how querying corpora can reveal the diversity and variety of uses of verbs in CS English. Working on corpora allowed us to describe three types of verbs that are typical in English : neologisms, specialised uses of verbs that already exist in « general English », and « general English » verbs that are extremely frequent in CS English.

The results of corpus query can also reveal the potential difficulties that French-speakers have. Comparing CS English verbs with their French equivalents, but also with verbs in « general English » allowed us to highlight differences, especially in the first two types of verbs. The difficulties that French-speakers can have with verbs of the third type – general English – are common for all French-speakers in general. Basically, we postulate that two main factors are responsible for the errors that French-speakers make in English : interference from the mothertongue and overgeneralization of rules in the second language (Kübler 1995).

3.1. Verb/noun ambiguity in the neologisms of CS English

Neither the frequency list nor the list of terms tagged as verbs are enough to cover all the verbal neologisms that are created from technical or proper nouns. The terms we are looking for are not necessarily tagged as verbs in our working dictionary[10]. Reference books are of little benefit either. In textbooks for teaching CS English, these verbs are never clearly explained. English dictionaries of computing or bilingual glossaries of computing (be they hard-copy dictionaries or on-line glossaries that can be found on the Web) contain many nouns, but do not mention verbal uses. For example, although the *Dictionary of Computing,* published by Oxford University Press

[10] We use a specific dictionary to tag our corpous, e.g. a list of words with part of speech categories. Information extracted from our corpora allowed us to complete our dictionary.

and aimed at learners of English as a second language) is very complete, it does not offer any information of this type.

With this in mind, we looked for inflected forms of verbs, i.e. words ending in *–ed,-ing*, and *–(e)s*. This type of verb is regular because the simple past and past participle are built by simply adding *–ed* to the root. An even finer selection can be made by searching the concordances for more complex verb forms, such as *have, been*, or *being* followed by a word ending in *–ed* for instance. Verbs, such as *to ftp, to rlogin, to telnet, to gzip, to Mosaic* were extracted in this way. The verb *to zip* is derived from the nous *zip*, hence the inflected forms *zips, zipping, zipped*. In this case, the relationship between verb and noun is clear as is the syntactic structure of the verb :

(6) You can **zip** the file and attach it to your message

The term in use in French is as simple as in English :

(7) Vous pouvez **zipper** le fichier et le joindre à votre message

For other verbs, the relationship can be more opaque ; *to FTP* is derived from the acronym *FTP* (*File Transfer Protocol*), *to Mosaic* stems from *Mosaic* which is the name of the first browser of the World Wide Web :

(8) The latest source can be **FTPed** from the directory ftp…or **Mosaiced** from http

In this case, the English context alone is not enough to establish the basis syntactic structure of the verbs. Their meaning remains unclear to a layman. French-speakers can have comprehension problems and may even misinterpret the sentence. The possibility we have of verifying the French equivalent in exactly the same context is therefore extremely useful. The French translation of the above example is :

(9) On peut **charger** la dernière version sur **ftp** … et **sous Mosaic** depuis http …

In French, the creation of neologisms, such as **ftper* for example, is subject to more constraints than in English[11]. French translators of such technical texts often have recourse to paraphrase based on the noun from which the English verb has been derived. Describing structures in French and in English for the two verbs *to ftp* and *to Mosaic* for example, means describing very different structures. French uses *charger une version **sur ftp*** (*on ftp*), but ***sous Mosaic*** (*under Mosaic*). However, examining all the occurrences of *to FTP* in the corpus suggested other possible translations :

(10) a. You can ftp it from sunsite.unc.edu
b. Vous pouvez l'obtenir par sunsite.unc.edu

Working on bilingual corpora highlighted this diversity and showed that an English technical verb often has no stable translation in French ; that is why it is necessary to collect all possible equivalents. As we were checking the English equivalents of the French expressions, we also found an English paraphrase around the noun *FTP* :

(11) a. It can be obtained by anonymous FTP from sunsite.unc.edu
b. On peut l'obtenir en faisant un FTP anonyme à partir de …

When the rules of euphony allow it, some creations coexist with periphrastic equivalents :

(12) a. They must **telnet** to the firewall
b. Il faut se **connecter** au firewall par le **réseau**

(13) a. Only the administrator can **telnet** directly to the firewall via Port 24
b. ?Seul l'administrateur peut **télnéter** directement le firewall sur le port 24

[11] Here **ftper* does not exist probably for euphony reasons

The first translation represents an explanation of the *telnet* process ; the second one is quite surprising since from a prepositional verb in English (*Nhum telnet to Nmachine*[12]) a transitive verb (which is a loan translation) is created.

As there is only one occurrence of the French verb *télnéter* in the corpus, the acceptability of sentence (4F) is questionable, although all the rules concerning the coining of new words have been respected. In this case, combining frequency and structure can be useful to define the scope of the vocabulary to be taught : a structure which is both rare and doubtful should be discarded.

One of the major problems concerning verbs in computer science is the lack of regularity in translating them from English into French, and the divergences between norm and usage : standardized terms by an official body, such as the *Commission Ministérielle de Terminologie Informatique.* are not always used, while deprecated terms can be knowingly used because they are the ones that are used by the whole CS community[13].

The verb *to boot*, which is quite frequent (700 token occurrences), illustrates this issue. Here again, reference books, such as dictionaries or textbooks, are of little help. The on-line *Merriam-Webster's*[14] does not give any definition of *to boot* related to computer science : the given meanings are *to avail, to profit*. There is no verb entry for *to boot* in the *Collins-Cobuild*. Among the on-line dictionaries that are available on the Web, *Wordnet*[15] is a little more complete because there is a definition for the specialised use of this verb in computer science (n°2 below). However, the information concerning the arguments of the verb or its syntactic structure is not sufficiently full:

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

Another on-line dictionary that is specialised in computer science (*FOLDOC*) tells us that *to boot* comes from *to pull oneself up by one's own bootstraps* ; the original meaning of this expression (« to do something without help ») has been transferred to a verb *to bootstrap* :

(16) *Bootstrap : (From « to pull oneself up by one's bootstrap »)*
To load and initialise the operating system on a computer.
Normally abbreviated to « boot »

The original verb *to boostrap* is no longer used very often in CS English, according to our corpus evidence; only thirteen tokens, out of which only two verbal uses can be found in the corpus :

(17) a. This is useful to **bootstrap** Linux on a system with only one floppy drive
b. Ceci est utile pour **démarrer** Linux sur une machine qui ne possède qu'un lecteur de disquettes

In France, the translation standardized by the *Commission de Terminologie Informatique* of the Ministry of Culture is *amorce* for the noun, and *amorcer* for the verb ; these are specialised uses of already existing terms that roughly mean « start ». However, if the noun *amorce* can be found in our French corpus, the verb *amorcer* occurs very rarely. Looking for the French equivalents of the verb *to boot* in the French corpus reveals *démarrer, lancer*, and less often the anglicism *booter* :

(18) a. You can specify various hardware parameters before **booting** the Linux kernel.
b. Vous pouvez préciser différents paramètres matériels avant de **démarrer** le noyau Linux

(19) a. The system doesn't **boot** at all
b. Le système ne **boote** plus du tout

(20) a. LILO is a program that will allow you to **boot** Linux
b. LILO est un programme vous permettant de **lancer** Linux

[12] We use here the notation used in the theoretical and methodological frame of the lexicon-grammar, in which for example *Nhum* represents a human noun, i.e. all the nouns that can be considered as humans (*girl, driver, linguist, guy,* etc*)*. M. Gross, 1975 : *Méthodes en Syntaxe*, Klinsieck : Paris.
[13] This is particularly true in the GNU initiative and Linux community.
[14] http://www.m-w.com
[15] http://www.cogsci.princeton.edu/~wn/

Doing the job the other way round, i.e. analysing the English equivalents of *démarrer*, and *lancer*, not only allowed us to confirm *to boot*, but also to discover *to run, to launch, to type,* and *to issue* for the French *lancer.* Using English verbs can thus rapidly become quite complex for a French-speaker. Comparing English and French verb concordances shall allow students to find out in which context these verbs can be used.
The French *booter* and *amorcer* are unequivocally translated by *to boot* ; *booter* being a nonce borrowing, and *amorcer* a new use of the verb which has been especially created to give an official French equivalent to the English *to boot*. Analysing the concordances reveals precise indications of when to use the translations *démarrer* and *lancer* (or *se lancer* in some cases). Generally and with very few exceptions, *to boot* is used for *démarrer* and *lancer* when dealing with starting an operating system.

We show here what type of linguistic information can be extracted from the corpus. This information will be used in the preparation of pedagogical material, and for the automatic generation of exercises.

i) *To boot* is an ergative verb, i.e. the action can be described from the point of view of the agent or of the one that is affected by the action. The basis structure of this verb has three arguments and the subject is the agent of the action[16] :

N_0 *boots* N_1 *Prep* N_2 with the following arguments :

N_0 = :	*Nhum* or *Nbootappl* (= application software allowing the system to boot, such as *LILO*)
N_1 = :	*Nbootobj* (= all the objects that can be booted : *operating system, disk, bootdisk, hard disk, floppy disk, kernel*)
Prep = :	**With, from, off**
N_2 = :	*Nbootingobj* (= booting objects, e.g. *CD, CD-ROM, D :, C :, A :, file, emergency disk*)

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

The corpus allows us immediately to detect the variety of English prepositions and how they are translated into French. Analysing the sentences with a three-argument structure enabled us also to build up a list of arguments for each position.

ii) A simple transitive sub-structure is possible : N_0 *boots* N_1

N_0 = :	*Nhum + Nbootappl*
N_1 = :	*Nbootobj*

EN *LILO is a program that will allow you to boot Linux*
FR *LILO est un programme vous permettant de lancer Linux*

iii) The intransitive form in which the argument in the position of subject represents the element that is affected by the action is the following : N_0 *boots*, with N_0 = : *Nbootobj*

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

iv) A prepositional structure, in which the object in the N_1 (first object) position is assumed to have been deleted, is also quite common : N_0 *boots Prep* N_1, with N_0 = : *Nbootobj, Prep* = : *to* :

v)

EN *Your BIOS may not allow you to boot directly to a SCSI drive.*
FR *Votre BIOS ne vous permettra peut-être pas de démarrer directement à partir d'un disque SCSI*
EN *Your BIOS mau not allow you to boot to a Linux installed there*
FR *Votre BIOS peut ne pas vous permettre de démarrer un système Linux qui y serait installé*

[16] N_0 is the noun in the subject position, N_1 the nouns in the object position, and N_2 the nouns in the position of second object.

In this context, *lancer* can also very rarely be translated by *to launch*, which is a more general verb. In radically different contexts, such as *lancer une command*, *to run, to issue,* and *to type* can be found.

The structures and arguments described above show the difference between the general verb *to boot* and the highly subject-specific neologism *to boot*. Apart from the distinct etymological origin (which is however not very useful from a synchronic point of view), the neologism *to boot* presents structures, as well as arguments, that are very different from the general verb. This is illustrated by the two examples below, which have been extracted from a concordance on the *Herald Tribune* :

(21) In early 1988 the Saudis **booted out** Hume A. Horan
(22) …eating habits under control by **booting** the French chef and his staff. The next…

The next sub-section deals with the problem of verbs that already exist in general English, and that also have highly technical uses.

3.2. Specialised uses

Numerous verbs existing in general English can be found in the computer science subject area with specialised uses that are very different from the general English meaning. Comparing the candidates with their French equivalents, but also with their general English uses allowed us to isolate the subject-specific uses, as shown in the examples below :

To save

HOWTO	Herald Tribune
These settings will be **saved** for you Cette configuration sera **sauvegardée**	to **save** court time he turned to the church to **save** his skin the government hopes to **save** hundreds of millions of dollars

These example show that the arguments of the verbs are very different in CS English ; the French translation of *to save* in its specilized use is *sauvegarder*, whereas in the three general uses given above, the verb will be translated by *gagner, sauver*, and *épargner*, respectively.

As was already shown in the case of neologisms, comparing an English verb with its French equivalents allowed us to underscore uses that are unknown by French-speakers. The *a priori* meaning of *to post* in CS English is « to send a message by e-mail, especially to a newsgroup », which is confirmed by the French translation below :

(23) a. Everybody should have a look through this section before **posting** for help
b. Tout le monde devrait y jeter un coup d'œil avant d**'envoyer un message** demandant de l'aide

The meaning of the following example is completely different :

(24) a. Called by the kernel when the card **posts** an interrupt
b. Appelé par le noyau quand la carte **déclenche** une interruption

The distance between general use and specialised use is on a continuum between « almost general » and « completely specialised ». Command terms that are used with an operating system like *UNIX* and *Linux* can be integrated into sentences as verbs with very specialised meanings. The technical use of *to quit* for example, is close to its general meaning, i.e. « to get out of a session ». In the e-mail application running under *UNIX* or *Linux*, *quit* is a command whose function is to leave the application without saving deleted messages ; the meaning of verbs and the name of commands merge together when the name of a command is integrated into a sentence as a verb. In this case, the use of the technical verb is very different from its general use.
To kill which means « to suddenly stop a process » is not as close to its general use, although the French translation *tuer* can be found, as well as *détruire*. Finally, the relation between general and specialised for *to zip* (French : *compresser*) and *to unzip* (French : *décompresser*) is very distant.[17]

These verbs are quite numerous, and some of them are also very frequent, like *to run* for example. *To run* has various uses, and is a frequent verb in CS English (according to our corpus evidence) ; in our corpus of general newspaper (*The Times*) it is quite frequent as well (cf. 3.2. Frequencies), but with other meanings. However,

[17] The neologism *to gzip* has been created on the basis of *to zip*

very few indications about its specialised uses can be found in reference books. Computing dictionaries[18] do not mention it. Among the thirty or so uses given by the *Merriam-Webster's*, only one is related to computing : *to run a problem through a computer*, a use that is quite rare in CS English. This use can be found 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 gives the following result : there are 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 :

(25) a. *Dictionaries :* To run a problem through a computer
b. *Corpus :* If you run your file through TeX program

Scanning bilingual dictionaries gave us the following translations : *exécuter, passer, fonctionner, être en marche,* and *utiliser*. We then analyzed the occurrences of *to run* in the corpus. This showed us that the above translations are not the only ones in use, and gave us complete information about the phraseology of the different uses. We give here a few examples of the two basic uses of *to run* and the various translations that can be found in our corpus:

i) *to run* ⇔ *lancer, exécuter*

(26) a. You forgot to **run** LILO or system doesn't boot at all
b. Vous avez oublié de **lancer** LILO ou le système ne boote plus du tout
(27) a. It just **runs** a command…
b. Il ne fait **qu'exécuter** une commande…
(28) a. …32-bit code that runs in 16-bit mode…
b. …du code 32 bits qui **s'exécute** en mode 16 bits…

ii) *to run* ⇔ *faire tourner, tourner, fonctionner*

(29) a. You can **run** Linux on any Alpha-based machine
b. Vous pouvez **faire tourner** Linux sur n'importe quelle machine Alpha
(30) a. The ability of any **Alpha-based machine to run** Linux *(patient in subject position, active voice)*
b. La possibilité de **faire tourner** Linux sur une machine Alpha *(operator* faire => *introduction of a third argument in the subject position)*

(31) a. If the same program **is run** on a 21064… (passive voice, patient in subject position)
b. Si le même programme **tourne** sur un 21064… *(active, patient in subject position)*

The choice of the preposition on and under depends on the arguments in the subject and object positions : application software and operating systems run **on** a machine or an operating system, while application software runs **under** an operating system :

(32) a. VirtuFlex **runs on** standard UNIX Workstations
b. VirtuFlex **tourne sur** des stations UNIX standard
(33) a. ANSFORTH system that **runs under** Win3.2, Win95, WinNT
b. Le système ANSFORTH qui **tourne sous** Win3.2, Win95, WinNT

These examples show how corpus analysis can highlight the great variety of existing structures and arguments, as well as the relationship between structures and transformations. Extracting the left and right context of verbs enabled us to obtain a list of possible arguments which had to be checked with an expert in computer science. How can the layman know that *LILO* is a boot program, that *inetd* is the noun of a program, or that *Pentium*, which is the name of the microprocessor,is a metonymy for « computer » ?
The comparison with uses in general English can help isolate technical verbs. The uses described above cannot be found in general English ; on the contrary, it is possible to find structures that never appear in CS English :

(34) …become a presidential concern about **running for** re-election in 1996…
(35) …stamps, old coins, and odd documents, **run** around the square. Cafés and…

3.3. General English verbs

[18] *FOLDOC, A Glossary of Computing Terms, Dictionary of Computing For Learners of English*

Teaching CS English verbs cannot concentrate solely on highly subject specific or specialised verbs. Some general verbs are quite often used in CS English.
Comparing general corpus with specialised corpus for non-specialized verbs showed up differences in the frequency for different uses. While a general verb has several general uses, only one can be found in CS English. *To install* is more frequent in CS English, than in our general corpus ; in the computing field, it is used in only one type of context :

(36) You must configure and **install** an appropriate kernel and then **install** the AX.25

In the computing context, it is only programs which can be installed. In the *Herald Tribune*, in contrast, occurrences of *install* have been found in structures in which a human argument is in the position of direct object :

(37) the country's new president, who was **installed** in January. He was…

Technical uses of *install* occur much less frequently in general English :

(38) by having a catalytic converter installed in her old-fashioned Volkswagen Derby

Noun uses can be different in technical and general English : in general English, the noun is *installation,* while in CS English, the mostly used noun is *install*. Verb/noun ambiguity can thus be more difficult to resolve in CS English.

Another problem related to verb/noun ambiguity lies in the structural differences between a verb and a noun, in French and English. *Access* is an example of this difficulty. In English, the noun is followed by the preposition *to* ; the French noun *accès* is followed by the preposition *à*. *Access* however is also a transitive verb in English ; whilst, the French verb is followed by a preposition : *accéder à*.

(39) a. Postgress95 which provides simple **access to** any existing database
b. Postgress95 qui fournit un **accès à** n'importe quelle base de données existante

(40) a. The user can **access the system**
b. L'utilisateur peut **accéder au système**
⇨ *The user can **access to** the system

Adding a preposition after *to access* is a very common mistake among French-speakers[19].

This shows how useful it can be to look for general English verbs in specialised corpora.

4. Conclusion

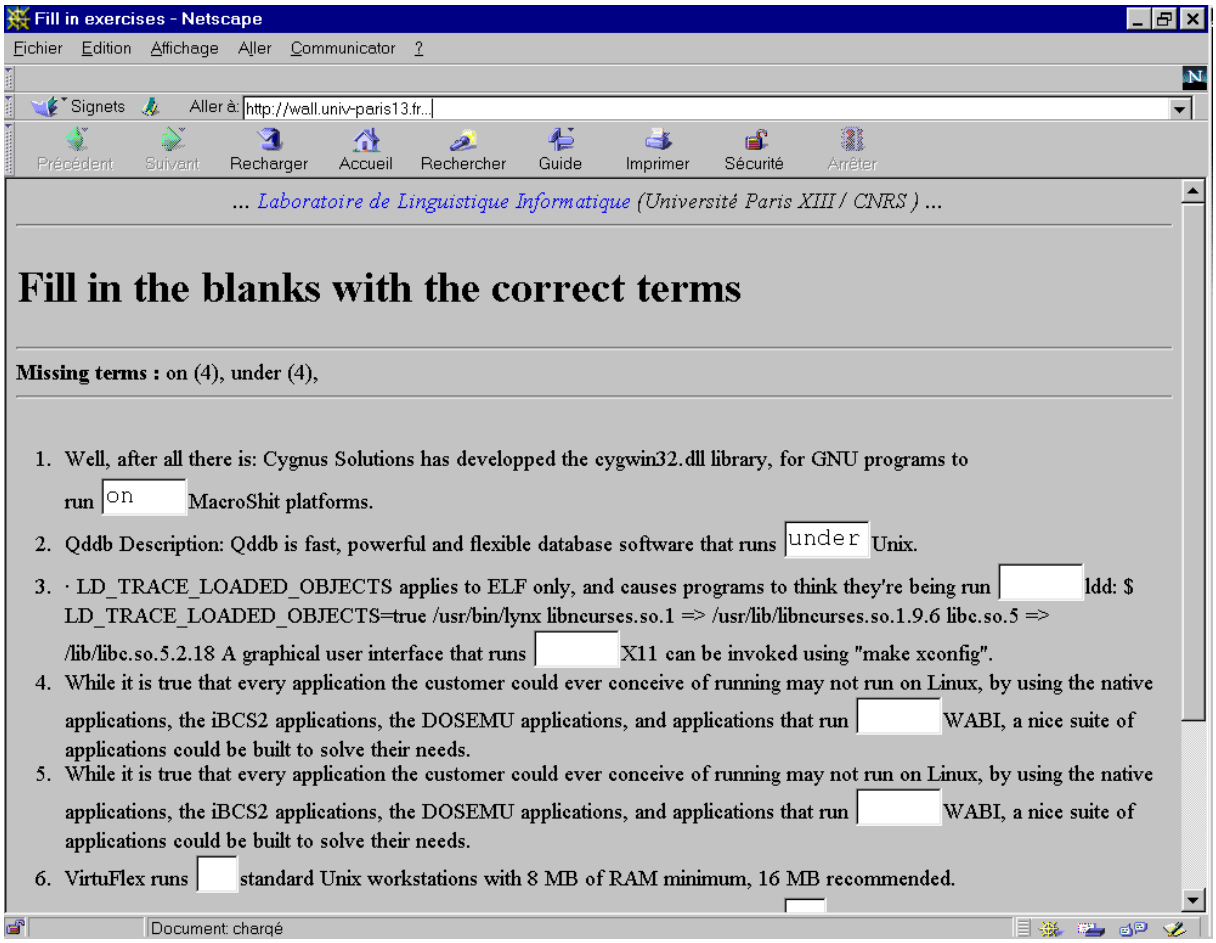
Developing a linguistic description is not an easy task in a highly technical subject area. The linguist cannot rely on intuition because s/he does not have the necessary technical knowledge ; information found in reference books is of little help.
Using and relying on authentic documents is therefore absolutely necessary ; contrastive work on bilingual corpora allowed us to list the characteritics of technical verbs. It has also enable us to identify differences in the use of specific structures between French and English. The observation of the English equivalents of the French verbs threw new light on the relationships between the different uses of a verb in English.
The current linguistic description needs to be refined : the description of the structures is not coupled with systematic statistic information.

Concerning the teaching of CS English, compiling a learner's corpus should help us complete our teaching objectives. A corpus-driven description of learner's English would lead to the description of linguistic reality. As stated in Granger and Tribble (1998) compiling and analysing a corpus of the non native learner allow the linguist to highlight the learner's difficulties, and therefore to decide what must be taught.
Working on corpora permitted us to achieve two aims : on the one hand, to show students the different verb structures from contrastive concordance samples and then to allow them to look for equivalences in the parallel

[19] We frequently noticed it among undergraduate students.

corpora ; on the other hand, it enabled us to make a description of the verbs focusing on differences and on potential problems. The description was then used to generate exercises automatically.

Gap-filling exercises can be produced on concordances. It is possible to ask students to find the correct preposition after *to run* for example :



The more precise the linguistic description is, the more sophisticated the exercises can be. Moreover combining a linguistic description with a corpus-based French-speakers' errors (see Cornu et al. 1993) will lead to the automatic correction of less restricted exercises (than gap-filling exercises) which need precise grammar-checking.

References

Bosworth-Gerome S., Ingrand C., Marret R., 1992 : *Comprendre l'anglais scientifique et technique*. Ellipse : Paris.

Brookes M., Lagoutte F., 1993 : *English for the Computer World*. Belin : Paris.

Cornu E., Kübler N., Bodmer F., Grosjean F., Grosjean L., Lewy N., Tschichold C., Tschumi C., 1997 : « Prototype of a Second-Language Writing Tool for French-Speakers Writing in English ». *Natural Language Engineering,* 2(3).

Granger S., Tribble C., 1998 : « Learner Corpus Data in the Foreign Language Classroom : form-focused instruction and data-driven learning » ; in : Granger S. (ed ;) *Learner English and the Computer*. Longman : London.

Foucou P.-Y., Kübler N., 1999 : « A Web-based Environment for Teaching English : General Architecture ». *ReCall*, special issue.

Foucou P.-Y., Kübler N., 2000 : « A Web-based Environment for Teaching Specialised English. ; in Lou Burnard and Tony McEnery (eds.) *Rethinking Language Pedagogy: papers from the third international conference on language and teaching.* Peter Lang GmbH : Frankfurt am Main.

Hoffman L. 1985 : *Kommunikationsmittel Fachsprache*. Günter Narr Verlag : Tübingen .

Johns T., 1988 : « Whence and Wither Classroom Concordancing » ; in : Bongaerts, T. et al. (eds) *Computer Applications in Language Learning*, 9-27. Foris : Dordrecht.

Kübler N., 1995 : *L'automatisation de la correction d'erreurs syntaxiques : application aux verbes de transfert en anglais pour francophones*. PhD thesis, Université de Paris 7, publications de l'Institut Gaspard Monge : Université de Marne La Vallée.

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

Trimble R. M. T., and Trimble L. 1978 : « The Development of EFL Materials for Occupational English : The Technical Manual ». In : R. M. T. Trimble, L. Trimble and K. Drobnic (eds) : *English for Specific Purposes*. *Science and Technology*. English Language Institute : Oregon State University.

Vance S., 1995 : « Concordances with Language Learners : Why ? When ? What ? », *CAELL Journal*, vol.6, n°2.

Wichmann A., S. Fligelstone, A. McEnery and G. Knowles (eds) 1997. *Teaching and Language Corpora.* Longman : London.