AUTOMATIC TERM DETECTION: A REVIEW OF CURRENT SYSTEMS[*]

M. Teresa Cabré Castellví, Rosa Estopà Bagot, Jordi Vivaldi Palatresi
Institut Universitari de Lingüística Aplicada. Universitat Pompeu Fabra
La Rambla, 30-32
Barcelona – Spain - 08002
teresa.cabre@trad.upf.es, rosa.estopa@trad.upf.es, jorge.vivaldi@info.upf.es

Abstract

In this paper we account for the main characteristics and performance of a number of recently developed term extraction systems. The analysed tools represent the main strategies followed by researchers in this area. All systems are analysed and compared against a set of technically relevant characteristics.

1. Introduction

In the late 80s there was an acute need, from different disciplines and goals, to automatically extract terminological units from specialised texts. In the 90s large computerised textual corpora have been constructed resulting in the first programs for terminology extraction[1] (henceforth TE) which have showed encouraging results.

Throughout the current decade computational linguists, applied linguists, translators, interpreters, scientific journalists and computer engineers have been interested in automatically isolating terminology from texts. There are many goals that have led these different professional groups to design software tools so as to directly extract terminology from texts: building of glossaries, vocabularies and terminological dictionaries; text indexing; automatic translation; building of

[*] In Bourigault, D.; Jacquemin, C.; L'Homme, M-C. (2001) *Recent Advances in Computational Terminology*, 53-88.

[1] In order to give a broader view of TE we use both *extractor* and *detector* to refer to the same notion. However, we are aware of the fact that some scholars attribute different meanings to these words.

knowledge databases; construction of hypertext systems; construction of expert systems and corpus analysis.

From the appearance of TERMINO (the first broadly known term detector) in 1990 until today a number of projects to design different types of automatic terminology detectors have been carried out to assist terminological work. However, despite the large number of studies in progress, the automatisation of the terminological extraction phase is still fraught with problems. The main problems encountered by term extractors are: (1) identification of complex terms, that is, determining where a terminological phrase begins and ends; (2) recognition of complex terms, that is, deciding whether a discursive unit constitutes a terminological phrase or a free unit; (3) identification of the terminological nature of a lexical unit, that is, knowing whether in a specialised text a lexical unit has a terminological nature or belongs to general language and (4) appropriateness of a terminological unit to a given vocabulary (this has scarcely been addressed from the point of view of automatization).

Systems for TE are based on three types of knowledge: (a) linguistic; (b) statistical; (c) hybrid (statistical and linguistic). Hence, there are different approaches to automatic term detection. All systems analyse a corpus of specialised texts in electronic form and extract lists of word chunks (i.e. candidate terms) that are to be confirmed by the terminologist. To make the terminologist's task easier the candidate term is provided with its context and, when available, with any other further information (frequency, relationship between terms, etc.)

Two relevant aspects regarding the nature of terms are termhood and unithood[2]; TE systems may be designed based on only one of these two aspects. Some practical experiments following each scheme for ranking a set terms extracted from Japanese texts are presented in (Nakagawa & Mori, 1998). They show that results in precision and recall are very close but the set of terms extracted are a somewhat different. This is still a research issue.

Alongside term detection we find the task of automatic document indexing (i.e. information retrieval, IR). This applied field of natural language processing (NLP) techniques has an interesting common point with automatic term detection, that is, word chunks that index a given document are often terminological units. This same goal explains why many extraction systems are rooted on IR as well as on the analysis of a specific IR system with no application whatsoever to TE.

[2] (Kageura & Umino, 1996) refer to *unithood* as the degree of stability of syntagmatic combinations (collocations) and *termhood* as the degree in that a linguistic unit is related to a domain-specific concept.

The difference between these two approaches lies in the fact that a tool for TE should extract *all* terminological units from a text, whereas IR focuses on the extraction of only words or word sequences that better describe the contents of the document regardless of their grammatical features.

The standard approach to IR consists in processing documents so as to extract the so-called *indexing terms*. These terms are usually isolated words containing enough semantic load to provide information about its *goodness* when describing documents. Queries are processed in a similar fashion to extract *query terms*. With regard to queries the relevance of documents is based exclusively on their representing terms. This is the reason why their choice is crucial.

Often these indexing terms are single words although it is known that isolated words are seldom relevant enough to decide the semantic value of a document with regard to the query. This fact has given rise to the ever-growing appearance, in the TREC[3] assessments, of word and word-sequence indexing systems using NLP techniques.

Statistically based systems function by detecting two or more lexical units whose occurrence is higher than a given level. This is not a random situation, but it is related to a particular usage of these lexical units. This principle, called *Mutual Information*, also applies to other science domains such as telecommunications and physics. Term detectors based on hybrid knowledge tend to use this idea prior to a linguistic-based processing.

The problem with this kind of approach is that there are low-frequency terms difficult to be managed by extraction systems. Here it is important to note that these systems use basically numerical information and thus are prone to be language-independent. The two most frequently used measures in the assessment of these systems are found in IR: *recall* and *precision*. Recall is defined as the relationship between the sum of retrieved terms and the sum of existing terms in the document that is being explored. In contrast precision accounts for the relationship between those extracted terms that are really terms and the aggregate of candidate terms that are found. These measures can be interpreted as the capacity of the detection system to extract all terms from a document (*recall*) and the capacity to discriminate between those units detected by the system which are terms and those which are not (*precision*). The fact that recall accounts for all terms from a document implies that it is a figure much more difficult to estimate and improve than precision.

In contrast with this traditional approach, other approaches attempt to solve the problem by using linguistic knowledge, which may include two types of information:

[3] TREC (*Text Retrieval Engineering Conference*) refers to a series of conferences supported by NIST and DARPA (U.S. agencies). Further information can be found at: *http://trec.nist.gov/*.

a) Term specific: it consists in the detection of the recurrent patterns from complex terminological units such as noun-adjective and noun-preposition-noun. This calls for the use of *regular expressions* and techniques of *finite state automata*.

b) Language generic: it consists in the use of more complex systems of NLP that start with the detection of more basic linguistic structures: noun phrase (NP), prepositional phrase (PP), etc.

In both approaches each word is associated to a morphological category. In order to do so different strategies are proposed: from coarse systems that do not make use of any dictionary to complex systems that have an extremely detailed morphological analysis and a final phase of disambiguation.

Systems that harness structural information resort to techniques of partial analysis to detect potentially terminological phrasal structures. There are also systems that benefit from their understanding of what is a non-term so they are at some point in between those systems already mentioned. Other systems try to reutilize current terminological databases to find terms, variants or new terms.

Systems based on linguistic knowledge tend to use *noise* and *silence* as a measure of its efficiency. Noise attempts to assess the rate between discarded candidates and accepted ones; silence attempts to assess those terms contained in an analysed text that are not detected by the system. Noise is common problem of those systems using this approach. Errors in the assignation of morphological category are also shared by these systems.

The type of knowledge used leads to language-specific systems and therefore it requires a prior linguistic analysis and probably a redesign of many parts of the system. Knowledge in artificial intelligence has been traditionally obtained from experts in each domain. This has yielded several difficulties so that some scholars have focused on automatization and systematisation in knowledge acquisition. This strategy seems to show the benefits of a terminological approach. Thus some researchers (e.g. Condamines, 1995) have proposed the construction of terminological knowledge databases so as to include linguistic knowledge in traditional databases. Although this is a recent approach, there is no database yet containing all the features that could be used in TE, i.e. there is hardly any semantic information. Thus closed lists of words containing sparse semantic information within a given specialised domain have been proposed.

In this paper we attempt to analyse the main systems of terminology extraction in order to describe its current status and thus be able to enrich them. This paper is divided up into two main parts: firstly, the largest part is devoted to describe various systems of terminology extraction together with a short evaluation in which weak and strong points have been outlined. Secondly, the terminology extraction systems have been classified according to some parameters.

2. Description of some terminology extraction systems

In the following sections we offer a critical description of number of semiautomatic terminology extraction systems. In all cases, the following information is given:

a) The reference data of the system, that is, the author and the publication where the tool is first mentioned and the system goal.

b) A brief description of the system.

c) A short evaluation of the most relevant aspects. This evaluation is mainly based on papers, oral presentations in congresses and working papers, etc.

2.1. *ANA*

Reference publication: Enguehard and Pantera (1994)

Main goal: Term extraction

ANA (Automatic Natural Acquisition) has been developed in accordance with the following design principles: non-utilisation of linguistic knowledge, dealing with written and oral texts (interview transcripts) and non-concern about syntactic errors.

According to the current trend of harnessing statistical techniques in the study of natural language, scholars use Mutual Information as a measure of lexical association[4]. In order to avoid the involvement of linguistic knowledge the concept of "flexible string recognition" is created, which generates a mathematical function so as to determine the degree of similarity between words. Thus, no tool for morphological analysis is needed. For instance, the string *colour of painting* represents other similar strings like: *colour of paintings*, *colour of this painting*, *colour of any painting*, etc. The system has neither a dictionary nor a grammar.

The architecture of ANA is composed of 2 modules: a familiarity module and a discovery module. The first module determines the following 3 groups of words, which constitute the only required knowledge for term detection:

a. function words (i.e. empty words): *a*, *any*, *for*, *in*, *is*, *of*, *to*...

b. scheme words (i.e. words establishing semantic relationships) such as *box of nails*, where the preposition shows some kind of relationship between *box* and *nails*.

c. bootstrap (i.e. set of terms that constitutes the kernel of the system and the starting point for term detection).

The second module consists in a gradual acquisition process of new terms from existing ones. Further, links between detected terms are automatically generated

[4] Remarkable examples of the use of these techniques are the works of Church & Hanks (1989) on word association and Smadja (1991) on collocation extraction from large corpora.

to build a semantic network. This module is based on word co-occurrence that can have 3 types of interpretations:

- *expression*: high-frequency existing terms (T_{EXP}) in the same window. The new word is considered a new term and thus is included in the semantic network. For instance if the system has *diesel* and *engine* as a known terms and finds sequences like: *... the diesel engine* is... or *... this diesel engine has...* Then the sequence *diesel engine* is accepted as a new term and is included in the semantic network as a new node with links to *diesel* and *engine* (see figure below).
- *candidate*: an existing term appears frequently (T_{CAND}) together with another word and a scheme word as in: ... any *shade* of wood... or ... this *shade* of colour... Here *shade* becomes a new term and is placed in a new node of the semantic network (see figure below).
- *expansion*: an existing term appears frequently (T_{EXPA}) in the same word sequence, without including any scheme word: ... use any *soft* woods to... or ... this *soft* woods or... As a result, *soft wood* is incorporated into the term list and the semantic network as a new node with a link to *woods* (see fig. 1 below).

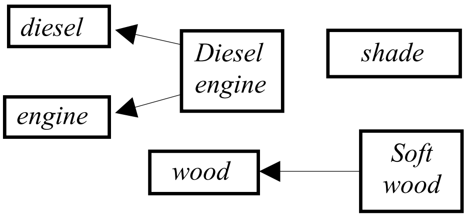


Figure 1 Term candidates interpretation

The system keeps on recursively seeking elements with the three interpretations already mentioned until a new term is found. Enguehard and Pantera (1994) tested it by processing a document in English of around 25,000 words and 29 reference terms. The system managed to extract 200 terms with an error rate of 25%.

Evaluation

Minimising linguistic resources is an extremely interesting issue, since it is difficult to compile them. Likewise *flexible string recognition* may well apply to actual texts.

A negative aspect of the system is that those terminological units added to the list of valid terms after each cycle are not validated. Thus ANA allows for the inclusion of non-valid terms that add up to the term list. However, no data about the efficiency of this proposal are reported.

2.2. *CLARIT*[5]

Reference publication: Evans and Zhai (1996)

Main goal: Document indexing

[5] Further information can be found at: *http://www.clarit.com*.

Document indexing for IR is an important field of application of NLP techniques. This branch holds common points with term detection since the word sequences that help in document indexing are normally terminological units too.

CLARIT belongs to the group of systems that advocate an elaborated textual processing to detect complex terms in order to reach a more appropriate description of documents. This is the reason why we have included this system amongst terminology detectors.

Evans and Zhai (1996) propose the following kind of phrases for indexation purposes:

1. lexical atoms (*hot dog*, *stainless steel*, *data base*, *on line*, ...)
2. head modifier pairs (*treated strip*, ...)
3. subcompounds (*stainless steel strip*, ...)
4. cross-preposition modification pairs (*quality surface* vs. *quality of surface*)

The methodology starts with the morphological analysis of words and the detection of noun phrases (NPs). The system distinguishes simplex noun phrases from cross-preposition simplex phrases.

What is behind this is the introduction of statistics to corpus linguistics. Statistics here focuses on documents, that is, there is no prior training corpus. Linguistic knowledge facilitates the calculation weeding out irrelevant structures, improves the reliability of statistical decisions and adjusts the statistical parameters.

The whole process is showed in the figure below:

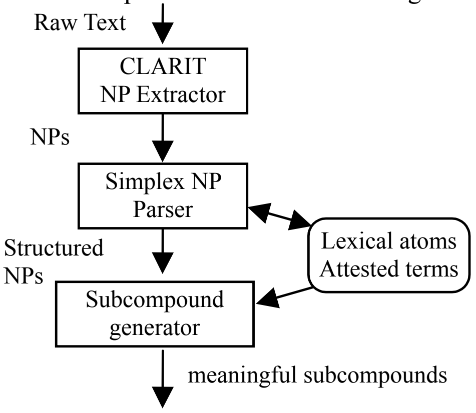


Figure 2 Whole process in CLARIT

First, the raw text is parsed so as to extract NPs. Then each NP is recursively parsed with the purpose of finding the most secure groupings. In this phase lexical atoms are also detected and NPs are structured. Finally at the generation phase the remaining compounds are obtained.

Lexical atoms are defined as sequences of two or more words constituting a single semantic unit such as *space shuttle*, *part of speech* and *hot dog*. Since the detection of these units is fraught with problems two heuristical rules are proposed:

a. The words that constitute a lexical atom establish a close relationship and tend to lexicalise as if they were a single-word lexical unit.

b. When acting as NP, lexical atoms hardly allow the insertion of words.

The first condition takes place if the frequency of the target pair W_1W_2 is higher than any other pair from the NP that is being processed. In the second condition the frequencies of grouped and separated occurrences are compared and there is a threshold beyond which the association is weeded out. This threshold is variable according to the function of sentence morphological category. In English texts, the most favoured sequence is that of noun-noun.

NP analysis is also a recursive process. At every new phase the most recent lexical atoms are used for finding new associations that will be used in the following phase. The process keeps going until the whole NP is analysed. Let us consider the example below:

general purpose high performance computer
general purpose ***[high performance]*** *computer*
[general purpose] *[high performance] computer*
[general purpose] ***[[high performance] computer]***
[[general purpose] [[high performance] computer]]

The grouping order shows those sequences with a more reliable association score. In order to determine the association score a number of rules are taken into account:

- Lexical atoms are given score 0 as well as adverb combination with adjective, past participles and progressive verbs,
- Syntactically impossible pairs are given score 100 (noun-adjective, noun-adverb, adjective-adjective, etc.).
- As to the remaining pairs, there is a formula that account for the frequency of each word, the association score of this word with other words from the NP and of two random parameters.

To increase its reliability the association score is recomputed after every assignation association. The system has been tested in an actual retrieval task of document indexing substituting the default NLP module in the CLARIT system. The corpus and the queries were the standards used in the TREC conferences. There have been noticed some improvements in recall as well as in precision, which, in the author's opinion, justifies the use of these techniques. Then in the TREC-5 report a more detailed evaluation of the system is made (Zhai *et al.* 1996). All in all it is concluded that the use of these techniques is effective, which enforces the similarities between term indexing and terminology extraction.

Evaluation

This seems to be an interesting system and the applicability of some basic ideas to terminology detection appears to be feasible. Actually CLARIT holds similarities with the Daille's (1994) proposal (a linguistically-driven statistics).

It should be borne in mind, however, that problems of terminology extraction and document indexing are similar but no identical so that many decisions should be re-considered strictly from the point of view of term detection. It is also noteworthy that this system only extracts NP terminological units and the data provided about how this system works are related to the application for which it has been designed.

2.3. *Daille-94 (ACABIT)*

Reference publication: Daille (1994)
Main goal: Term extraction
The main idea behind this system is to combine linguistic knowledge with statistical measures. Here the corpus should contain all the morphological information. Then a list of candidate terms is created according to text sequences that provide syntactic patterns of term formation. This information uses statistical methods to filter out this list. This final process is different from other systems in that it only uses linguistic resources.

Assuming the fact that all terminological banks are basically composed of compound nouns, the program focuses on the detection of *binary compound nouns* and disregards other co-occurring categories. This assumption lies in the fact that there is a large number of this kind of nouns in specialised languages. Further, most of these compounds of 3 or more constituents can be treated in a binary form.

Those patterns considered relevant for French are N_1 PREP (DET) N_2 and N ADJ PREP *à* (DET) N_2, together with right and left coordination. Statistical algorithms are applied to these patterns. The author is aware of the fact that the application of statistical measures leads to some noise rate, that is, low-frequency terms will not be recognised.

The technique used for pattern recognition is that of finite state automata. Automata are represented by a subset of grammatical tags to which some lemmas, inflected forms and a punctuation mark are added. Thus we can regard automata as linguistic filters that select defined patterns and also determine their occurrence frequency, distance and variation. Each morphosyntactic pattern is associated with a specific finite automaton.

The corpus is given a statistical treatment based on a large number of statistical measures, which are grouped in the following classes: frequency measures, association criteria, diversity criteria and distance measures. The starting point is considering the two lemmas that constitute a pair within a pattern as two variables on which the dependence degree is measured. Data are represented in a standard contingency table:

	L_2	L_n
L_1	A	b
L_m	C	d

where
a = L_1L_2 occurrences
b = L_1+L_n ($n\neq2$) occurrences
c = L_m+L_2 ($m\neq1$) occurrences
d = L_m+L_n ($m\neq1$ and $n\neq2$)

Eighteen measures are applied with the aim of establishing the degree of independence of the variables in the contingency table. The analysis of the results shows that only four of these measures are relevant to the purpose: frequency, cubed association criterion[6] (IM^3), likelihood criterion, Fager/MacGowan criterion.

Evaluation

Unlike in other systems, in ACABIT frequency has turned out to be one of the most important measures for term detection from a given area. However, the classification resulting from the application of this frequency shows an important number of frequent sequences that are not terms and, in contrast, does not suggest the low-frequency terms.

Daille (1994) believes that the best measure is the likelihood criterion, since it is a real statistical test, it proposes a classification that accounts for frequency, it behaves adequately with large and medium size corpora and it is not defined in those cases that are not to be considered. In any case, this measure yields some noise due to several reasons:

a. Errors in the morphological mark-up.

b. Some combinations that are never of a compounding nature: *ko bits* (kilobits), *à titre d'exemple* (as an example)... ...

c. Combinations of 3 or more elements, related to the problems of composition and modification: *bande latérale -unique-* (-single- side band), *service fixe -par satellite-* (-satellite- fixed service), etc.

2.4. *FASTR*[7]

Reference publication: Jacquemin (1996)

Main goal: Term variation detection

The aim of this tool is to detect terms variants from a set of previously known terms. These terms may be available from a reference database or a term acquisition software. What is crucial in this system that it is not needed to start from scratch every time. Optionally Fastr can also be used for TE.

The first step for applying Fastr is to obtain and analyse a set of existing terms and thus having a set of rules of a given grammar. The FASTR grammatical

[6] The formula was experimentally obtained by the autor from the association number described in Brown et al. (1988) in the aim of favouring the most frequent pairs: $IM^3=\log_2 (a^3/(a+b)(a-b))$

[7] Further information can be obtained at *http://www.limsi.fr/Individu/jacquemi/FASTR/index.html*

formalism is an extension of that of PATR-II (Shieber, 1986). A partial parser based on the unification mechanism is responsible for the application of these rules. Term variants are obtained through a metarule mechanism that is dynamically calculated.

For instance, the term *serum albumin* corresponds to the Noun-Noun sequence and is associated with the following rule:

rule 1: $N_1 \rightarrow N_2\ N_3$
<N_1 lexicalization>= 'N_2'
<N_2 lemma>=serum
<N_3 lemma>=albumin.

The value indicated by the feature "lexicalization" will be use just before partial parsing to selectively activate the target rules. Thus the above rule is linked to the word *serum* and so is activated when this word occurs in the sentence that is being parsed.

At a different level several metarules generate new rules in order to describe all possible variations of each term from the reference list. Each metarule presents a particular structure and a specific pattern type. For instance, the following metarule can be applied to the previous rule:

Metarule Coor($X_1 \rightarrow X_2\ X_3$) = $X_1 \rightarrow X_2\ C_4\ X_5\ X_3$

which leads to the new rule: $N_1 \rightarrow N_2\ C_4\ X_5\ N_3$

This latter rule allows new constructions that substitute C_4 for a conjunction and X_5 for an isolated word such as *serum and egg albumin*. The candidate term is not the whole new construction but the *coordinated* term (i.e., *egg albumin*). The words that have given way to the new rule (*egg* and *albumin*) maintain their function of constricted equations of the original rule. Further, they are the anchoring point for the application of the metarule. A metarule can be associated with specific restrictions, as for instance: (*<C_4 lemma>≠but*) or (*<X_5 cat>≠Dd*). In this way, those sequences with no lexical relationship such as *serum and the albumin* are rejected.

The above rule is a coordination rule and it should be noted that there are also other types of rules that account for different kinds of variations:

1. insertion rules: *medullary carcinoma* ➔ *medullary thyroid carcinoma*
2. permutation rules: *control center* ➔ *center for disease control*

The FASTR metagrammar for English contains 73 metarules altogether: 25 coordination rules, 17 insertion rules and 31 permutation rules. In any case, for efficiency reasons the new rules are dynamically generated. Each rule is linked to a pattern extractor that permits a very quick acquisition of information. As has been pointed out, the FASTR grammatical formalism is a PATR-II extension (Shieber, 1986). This language allows to write grammars using feature structures. The rules describing terms are composed of a free-context part ($N_1 \rightarrow N_2\ N_3$) and a number of restriction equations (e.g. <N_2 lemma>=*serum*). First, the system filters the rules that are to be applied according to the given text and then an analysis take place.

When Fastr is applied for term acquisition the process is gradual: from a given set of terms the system detects new ones, which allows the beginning of a new cycle and the detection of new candidates. The loop goes on until new terms cannot be detected. The author presents an experiment carried out on a medicine corpus of 1,5 million words and a reference list of 70,000 terms from different specialised domains. After 15 cycles 17,000 terms were detected of which 5,000 were new. The text was processed at a 2,562 word/minute speed.

However, the number of recognised terms decreases when the reference list has fewer items. For instance, if the reference sublist of medicine drops to 6,000 terms, then only 3,800 new terms are recognised.

The author also postulates the existence of a conceptual relation. between the new terms and the term that has led to their recognition. This relationship is variable in accordance with the type of rule that is applied i.e., insertion or coordination rule. Permutation does not allow any relationship due to the phrasal nature of the relationship.

All the language dependent data used by Fastr is stored in separated text files. This feature facilitates the use of the system in other languages as showed by the recent application of Fastr to Japanese, German and Spanish/Catalan.

Recently Jacquemin has developed the detection of semantic variation using resources like WordNet or the Microsoft Word97 thesaurus (Jacquemin, 1999).

Evaluation

The main characteristic of FASTR is its ability for detecting term variants, an aspect often not considered by other systems. The fact of using already recognised and accepted terms is very useful, although, as the author admits, it places restrictions on the acquisition of new terms that are not related to the source terms.

TE in Fastr implies that terms that are added to the list of valid terms after each cycle are not validated. Thus, a non-valid term may be added to the list so it is likely that in forecoming cycles more non-valid terms are added. Jacquemin (1996) believes that this is not an important error source because the system, in some way, corrects itself since “normally” non-correct candidates do not give way to new potential candidate terms.

Actually this technique should not be isolately applied. Rather, it should be coordinated with other strategies as in (Jacquin & Liscouet 1996) and (Daille 1998).

2.5. *HEID*

Reference publication: Heid et *al.* (1996)

Main goal: Term extraction

Heid *et al.* (1996) believe that automatic TE has various applications and dictionary or glossary construction would be the major one. In dictionary construction from computerized corpora two phases are distinguished: linguistic pre-analysis and a term identification tool. Each of these phases requires specific computer tools.

In the linguistic pre-processing phase the following processes are required[8]:

a) tokenizing, which identifies word and sentence boundaries.
b) morphosyntactic analysis, which identifies grammatical categories as well as distributional and morphosyntactic features.
c) POS tagging, which disambiguate morphosyntactic hypotheses.
d) lemmatization, which identifies the lemma candidates.

For term identification the system has a general corpus retrieval interface that includes a corpus query processor (CQP), a macroprocessor for the CQP query language and a key word in context (KWIC) program, to extract and sort concordances and lists of absolute and relative frequency of search items.

TE is linked to a complex query language. The queries will be different according to the types of candidate terms searched for. Thus, for instance, queries about single-word terms are made from morphemes or typical components of compound or derived words (derivatives). In these queries it is assumed that NP affixed terms from specialised languages use more specific affixes and/or prefixes than others. All the word sequence extracted (N-A, N-N, N-V), are based on POS patterns.

Heid *et al.* (1996) have applied these tools to technical texts on automobile engineering in German, which amounts to 35,000 words. The sample has been manually analysed before the application of the above procedures. The results are as follows:

- With regard to single-word terms, there has been found a 90% of candidate terms and a 10% of silence. This rate varies from one scheme to another.
- With regard to multiword terms, there are no concluding results. The results are less satisfactory and that the same problems as linguistic based are found: POS patterns do not constrain enough the context and produce too much noise. Heid *et al.* (1996) believe that by using a syntactic parser, as it is the case in English, noise would diminish.

[8] Heid *et al.* (1996) note that a broad coverage morphosyntactic parser for German is not attained. Thus parser results are simulated using POS patterns.

- Finally, collocation extraction is shown to produce noise but not silence, since Heid *et al.* (1996) consider the frequency criterion.

The Ahmad's statistical measure (Ahmad *et al*, 1992) of relative frequency in corpora of specialised and general language is applied to this corpus of 35,000 words. They show that the results produced by linguistic corpus query are included in the output of statistical methods. However noise in statistical methods is higher than in linguistic methods.

Evaluation

To tackle this system it should be taken into account the morphosyntactic features of the German language. Unlike Romance languages, German prefers to form compounds in a synthesising manner. It means that what other languages express via terminological phrases in German is expressed with a single-word term (by word is meant any segment found between two gaps). Thus it can be seen that in German automatic term detection does not depend much on term delimitation but on the terminological nature of a word. This is the reason why we need parameters to distinguish a term from a word of the general language, both having the same morphosyntactic structure.

Like most of the reviewed programs, Heid focuses on NP terms although it can also extract collocations combining nouns and verbs. In this case Heid *et al.* (1996) note that the results are much worse. We do not have specific data about the performance and the results of this system.

2.6. *LEXTER*

Reference publication: Bourigault (1994)

Main goal: Term extraction

This system has been developed in the need of the EDF (*Electricité de France*) society for improving their indexation system. LEXTER aims at locating boundaries among which potentially terminological NPs could be isolated. LEXTER carries a superficial analysis and makes use of the text heuristics in order to obtain those NPs of maximum length that it regards as candidate terms.

The program is composed of several modules and works as follows:

1. Morphological analysis and disambiguation module. Texts receive information about the POS and the lemma assigned to every word.

2. Delimitation module. At this stage a local syntactic analysis is carried so as to split the text into maximal-length NPs. For example: *alimentation en eau* (water supply), *pompe d'extraction* (extraction pump), *alimentation electrique de la pompe de refoulement* (electric supply of the forcing back supply). Here the system takes advantage of the negative knowledge about the parts of complex terms. Thus those patterns of a potential term –finite verbs, pronouns and conjunctions– that will never become part of a term are identified and considered

as boundaries. Some of these patterns are simple whereas others are complex (sequences of preposition + determiner).

A French example of the latter would be *SUR* (prep) + *LE* (definite article): the most common analysis is to propose that this sequence establishes a boundary between NPs as in: *on raccorde le câble d'alimentation du banc sur le coffret de décharge batterie.* However there is a rate (10%) in which this sequence is part of the term: *action sur le bouton poussoir de réarmement* or *action sur le systeme d'alimentation de secours*

To solve this and other similar situations, the system uses an endogenous learning strategy of the patterns sub-categorisation. This strategy consists in looking at the corpus to find those sequences of *(noun) + sur + le* having different contexts on the right hand side. Then non-productive nouns are weeded out. Then sequences such as s*ur + le* are considered sentence boundaries, except for those cases wherein sequences are preceded by the productive noun located in the learning phase. To see how this system works let us suppose that at a first analysis the sequences below are found.

Le protection ***contre*** *le gel est assurée par*
Protection ***contre*** *les grands froids*
il s'agit de maintenir la teneur en oxygène de cette eau ***dans*** *les limites fixées*
on procède à l'injection d'eau ***dans*** *les limites fixées*
on procède à l'injection d'eau ***dans*** *les générateurs de vapeur*
le système permet l'aiguillage des automates ***sur*** *le prélèvement effectué*

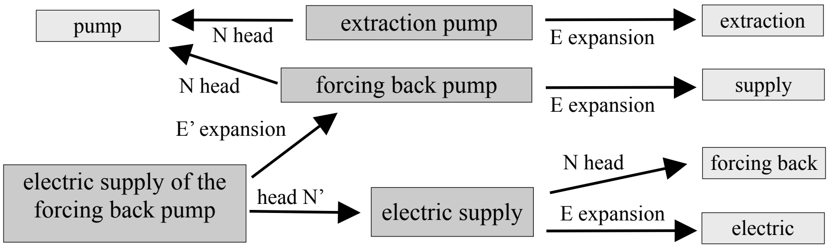
Then productive sequences are not regarded as term boundaries whereas non-productive sequences are viewed as external boundaries of the candidate term. In the example above *protection contre* and *eau dans* do not become boundaries whereas *automates sur* does.

This strategy permits to detect a considerable amount of complex nouns which otherwise would have been lost. Unfortunately it also allows a great deal of undesirable material (between 10% and 50%).

3. Splitting module. NPs are analysed and their constituents are divided into head and expansion. For example the term candidate *pompe d'extraction* (extraction pump) is splitted into*: pompe* –head– (pump) + *extraction* –expansion– (extraction).

At this point the system may find ambiguous situations such as "Noun Adj$_1$ Adj$_2$" and "Noun$_1$ Prep Noun$_2$ Adj" whose analysis is uncertain. To solve these cases an endogenous learning process is followed which is similar to that presented in the delimitation module.

4. Structuring module. The list of term candidates is organised in a terminological network. This network can be produced only by looking at a list of candidate terms and recognising the different parts of each candidate term, like in the following example:



Additionally Lexter calculate some productivity figures based on links type occurrences. These coefficients do not become filters, but are passed on to the terminologist as a piece of data so as to facilitate the evaluation of candidate terms.

5. Navigation module. A consulting interface is built (called *terminological hypertext*) from the source corpus, the candidate term network and the above-mentioned coefficients and lists.

Although LEXTER is exclusively based on linguistic techniques it produces highly satisfying results and is currently used to exploit different corpora from EDF and different research projects. Besides it has been proved helpful in: text indexation, hypertextual consulting of technical documentation, knowledge acquisition and construction of terminological databases.

LEXTER is also used as a terminology extractor in the terminological knowledge base designed by the Terminologie et Intelligence Artificielle (Terminology & Artificial Intelligence) terminology group. SYCLADE (Habert, 1996), a tool for word classification also makes use of LEXTER.

Evaluation

LEXTER was born in an industrial environment and from the very beginning it sought a robust, accurate and domain-independent tools. These objectives were basically attained although mark-up and disambiguation errors weaken the capacity of the system. Some scholars note that this system (like those which make use of symbolic techniques) produce a considerable amount of noise. Thus of a corpus of 200,000 words there are obtained 20,000 candidate terms which, after the validation stage, amount to 10,000. Also, Bourigault stresses the silence problem, which he estimated around 5% of the total valid terms. Like the vast majority of systems, LEXTER only focuses on NPs since verbs are believed to be term boundary and so they are never part of candidate terms.

One of the most remarkable achievements of this system is the endogenous learning mechanism that allow to work autonomously and so there is no need for a complex and large dictionary. In a similar vein it should be highlighted the usefulness of presenting the results hypertextually, since it facilitates the terminologist's task.

2.7. *NAULLEAU*

Reference publication: Naulleau (1998)

Main goal: Noun phrase filtering system

The model designed by Naulleau is a NP extraction system that proposes as term candidates those sequences that comply with certain user tailored profile. The whole process can be divided in two main stages: profile acquisition and profile application.

To define its own profile the user chooses the set of phrases that s/he considers relevant for his task and discards the ones that s/he does not consider useful at that time. The data collected in such way is generalised according to their morphological, syntactical and semantic characteristics dynamically creating a set of positive and negative filters. A simple example of positive and negative filters is the following:

(1) positive filter: *metallic/automatic/nuclear/industrial taps*
(2) negative filter: *important/recent/necessary/unreliable taps*

Then, those filters produced in the learning stage are applied to new sequences analysed. As a result, some noun arguments and/or PPs can be eliminated. Thus a NP can be divided or reduced and the resulting sequences are passed on to an expert to be evaluated.

In doing so the author acknowledges the sociolinguistic nature of the term. It implies that there is no linguistic model that can tell whether a NP is a term or not beyond the scope of a field or even the application. Also, this procedure introduces the idea of how relevant a phrase is in relation to the interest profile of the user and assumes that such relevance may be evaluated on linguistic grounds.

This is a fully symbolic approach that uses the AlethIP engine that produces sentences fully lemmatised, tagged and syntactically parsed. Then nouns and adjectives are semantically tagged according to both suffix information and semantic data from AlethIP and using a set of contextual rules for the more frequent and ambiguous words. The whole strategy is based on the evaluation of the relevance of simple syntactic dependencies. Such relevance is only based on the data provided by the user.

According to the author, the results are encouraging. However it is difficult to evaluate due to the practical problem posed by such a detailed evaluation. Some additional experiments are described in (Nalleau, 1999).

Evaluation

This system may be considered the first one to use semantic data as a specific resource for proposing term candidates. Also, as far as we know, is the first time since the very beginning in the design of a TE systems that the *user* and the *idea of relevance to an application* are taken into account.

In this way the user may adapt the system to its specific needs but also its intervention may crucially affect the performance of the system. The loss of specific data makes difficult to evaluate the tool behaviour in an actual context.

2.8. *NEURAL*

Reference publication: Frantzi and Ananiadou (1995)
Main goal: Term extraction
Neural is a system for TE of a hybrid nature, that is it uses both linguistic (morphosyntactic patterns and a list of suffixes specific to the domain) and statistically knowledge (frequency and mutual information). Frantzi & Ananiadou (1995) pays special attention to two different problems: detection of nested terms and detection of low frequency terms using statistical methods.

The test bench is a corpus of 55,000 words in the domain of medicine (ophthalmology). The structures analysed are Noun-Noun and Adjective-Noun that are identified using a standard tagger. The list of suffixes includes those frequently found in terminological units in the field of ophthalmology like *-oid*, *-oma*, *-ium*. The system is implemented using a Back-Propagation (BP) two layers neural network. The threshold has been set to .5 but this may vary. The BP neural network has been trained with a set of 300 compounds and the tests were made with another set of 300 words. It obtained a success rate of 70 %.

The author and other scholars from the Manchester Metropolitan University have been active since 1995 developing specific statistical figures for TE. In this way it is necessary to mention those tasks related to the adding of context information (Frantzi, 1997, Maynard & Ananiadou, 1999). Usually the context is discarded or, alternatively, considered as a bag of words although its relevance is signalled by many scholars. Here the basic assumption is that terms tend to appear grouped in real text, so the termhood figure of a candidate would increase if there are other terms (or candidates highly ranked) in the context.

Both Frantzi, 1997 and Maynard & Ananiadou, 1999 propose a similarity figure based on the distance between the candidate and the context words (nouns, adjectives and verbs). This figure is calculated by Frantzi (1997) using statistical and syntactic information while Maynard & Ananiadou (1999) include also semantic information from a specialised thesaurus (UMLS semantic Network).

In Maynard & Ananiadou (1999) this similarity figure may also be used to take into account some kind of semantic disambiguation for the sense that gets a better value. A context factor (CF) is added to the figure already used to rank the candidates (Cvalue) and thus reordering the set of candidates as follows: SNCvalue(a) = 0.8*Cvalue(a) + 0.2*CF(a). The authors report improvements in the ranking of term candidates from his eye pathology corpus.

Evaluation

The original system can be seen as a standard hybrid system. The linguistic knowledge includes Greek and Latin affixes and morphosyntactic patterns. The incorporation of this kind of suffixes should be highly productive. However the chosen patterns may well apply to English but not to Romance Languages.

The incorporation of the context as part of the data available for evaluating the termhood of a candidate is a very interesting contribution to the behaviour of terms in real texts. It should also serve to increase the relevance of low frequency candidates but no specific figure is given.

It is necessary to mention the use of semantic information as a kind of resource that is increasingly used in the TE field.

2.9. *NODALIDA-95*

Reference publication: Arppe (1995)

Main goal: Term extraction

NODALIDA, a product designed by the Lingsoft firm, is based on an enhanced version of NPtool that is a program developed at the Department of General Linguistics at the Helsinki University (Finland). NPtool (Voutilanen 1993) generates lists of NPs occurring in the sentences of a text and provides an assessment about whether these phrases are candidates terms or not (ok/?). From these lists all the acceptable sub-chains are obtained. Besides, the source list is multiplied. Let us see an actual example, for the sentence: "*exact form of the correct theory of quantum gravity*" NPtool proposes the following additional NPs:

form of the correct theory of quantum gravity
exact form of the correct theory
form of the correct theory
form
exact form
theory
correct theory
gravity
quantum gravity

Simultaneously there are a number of premises that become the first filter like in the following: "Those NPs preceded by a determiner, adjective or prefixed sentence (*kind of, some, one, ...*) are weeded out."

As for the remaining NPs, their occurrence frequency is calculated. Further, they are ordered and grouped according to their grammatical head and are presented to the terminologist together with their context. The NPtool module (Voutilanen, 1993) is at the heart of the system. It is a NP detector largely based on the *constraint grammar* formalism (Karlsson, 1990). Its main features are: (1) Morphological/syntactical descriptions are based on a large set of hand-coded linguistic rules, (2) both the grammar and the lexicon allow a corpus analysis with non-controlled text and (3) disambiguation is made according to only linguistic criteria. As a result, between 3% and 6% of the words remain ambiguous.

The text goes through a previous process so as to determine sentence boundaries, idiomatic expressions, compound forms, typographical signs, etc. Then it is morphologically analysed and a result like this is obtained[9]:

```
(“<*the>”      (“the” DET CENTRAL ART SG/PL (@>N)))
(“<inlet>”     (“inlet” N NOM SG))
(“<and>”       (“and” CC (@CC)))
(“<exhaust>”   (“exhaust” <SVO> V SUBJUNCTIVE VFIN (@V))
               (“exhaust” <SVO> V IMP VFIN (@V))
               (“exhaust” <SVO> V INF)
               (“exhaust” <SVO> V PRES -SG3 VFIN (@V))
               (“exhaust” N NOM SG))
               (“<manifold>”   (“manifold” N NOM PL))
```

At this moment disambiguation takes place. For example in the sentence: ”*The inlet and exhaust manifolds are mounted on opposite sides of the cylinder head*“ two analyses are obtained:

(1) on/@AH opposite/@N sides/@NH of/@N< the/@>N cylinder/@NH **head/@V**
(2) on/@AH opposite/@N sides/@NH of/@N< the/@>N cylinder/@>N **head/@NH**

What distinguishes these two analyses is the consideration of whether the final sequence (*cylinder head*) is a NP or not. The ongoing process gives only two possible analyses for each sentence. First, those NPs of a maximal length are preferred (*NP-friendly*) and, second, those NPs of a minimal length are preferred (*NP-hostile*). Then the system compares both strategies and labels each NP as ok/? by considering whether the analysis is shared or not by both strategies

Thus the last sentence gets this analysis below:

(3) ok : *inlet and exhaust manifolds* ?: *opposite sides of the cylinder*
ok: *exhaust manifolds* ?: *opposite sides of the cylinder head*

In order to validate this additional information the terminologist is provided with a list of candidate terms. The results reported by the NPtool module are pretty good (precision=95-98% and recall=98.5-100%) with a text of about 20 Kwords.

Evaluation

NODALIDA is based on the use of linguistic knowledge through a structural approach (i.e., detection of phrasal structures and structural disambiguation). Arppe (1995) presents high-quality results. However, the corpus should be enlarged, since so far tests have been made on quite small corpora. It is not clear how precision and recall figures are calculated, particularly how to determine which terms are deemed to be correct (i.e., those which have the ok signal or all of them). Also it should be stressed that NODALIDA has not been tested using the NPtool enhanced version in an actual situation of terminology problems.

[9] The meaning of the syntactic function tags are: @>N = pre-modifier; @<N = post-modifier; @CC and @CS= coordination and subordination conjunction; @V = Verb; @NH = nominal head. Finally, “>” and “<” indicate the direction of the phrasal head.

Taking into account that the disambiguator is one of the main error sources in this kind of systems, Arppe (1995) believes that a high-degree quality is achieved despite the fact that there are no data about terminology extraction in real situations. Besides, to achieve this quality NODALIDA proposes a great deal of rules, which yields management and control overhead.

The list that is passed on to the terminologist to be validated comprises those candidates signalled with **ok** and **?**. The way in which potential NPs are obtained by the system leads us to suspect that there are many candidate terms in the validation list that the terminologist has to analyse.

2.10. *TERMIGHT*

Reference publication: Dagan and Church (1994)

Main goal: Translation aid

Termight is currently used by *A&T Business Translation Systems*. It was created to be a tool for automating some stages of the professional translator terminological research.

To do so it starts with a tagged and disambiguated text as well as a list of predetermined syntactic patterns that could be adjusted to every document. Thus, a list of candidate terms is obtained comprising one or more words. Single-word candidates are defined as all those words that are not included in a previously determined list of empty words (i.e. stop list). Multiword terms are referred to one of the predetermined syntactic patterns via regular expressions. Dagan and Church (1994) considered only noun sequences patterns.

Candidate terms are grouped and classified according to their lemma (i.e. the right hand side noun) and frequency. Those candidates sharing the same lemma are classified alphabetically in accordance with the inverse order of their compounding words. Thus it is showed the order of changes of the English simple NPs.

For each candidate term the corresponding concordances are obtained, which are alphabetically classified according to their context. This information enables the terminologist to evaluate whether each candidate is appropriate or not.

Dagan and Church (1994) note that the rate of term list construction is of 150 and 200 terms per hour, which is twice faster than the average. As for the extraction quality, they state that, unlike exclusively statistical methods, Termight permits to extract low-frequency terms.

Moreover this system has a bilingual module which, via statistical methods, obtains a word-level alignment from texts. Thus terms found in language A are referred to their counterparts in language B. This well-ordered list of candidate terms is again passed on to the terminologist to be evaluated.

The Termight bilingual module does not seem to be developed and tested as the basic one. Tests have been made on 192 terms from a technical manual in English and German. The correct translation is found in the first suggested solution in 40% of the cases, whereas only 7% corresponds to the correct translation suggested in the second place. As for the remaining, the correct translation was in other places of the proposal list.

Evaluation

Termight is a remarkable system in that there is an accurate classification and presentation of candidate terms and it does not attempt to become an automatic system. Rather, it helps the translator.

However, it presents a number of shortcomings: (1) The only syntactic pattern considered is very simple: noun sequences. This pattern may well be valid for English but not for Romance languages and (2) no numerical information about the recognition quality is given. The type of pattern considered may suppose high precision but low recall

2.11. *TERMINO*

Reference publication: Plante and Dumas (1989)

Main goal: Facilitation of the term extraction terminographer's task.

The TERMINO program is composed of several tools to facilitate TE in French. It is a help for the terminologist insofar as the identification of those discourse units that denominate notions or objects. Besides it provides every unit with the immediate context from which data relevant to the notions denominated by theses units can be obtained. There are a number of TERMINO versions which improve in some ways previous ones.

This tool is based mainly on linguistic knowledge and it comprises 3 sub-systems: a pre-editor, which separates texts into words and sentences and identifies proper nouns, a morphosyntactic parser and a record-drafting facility.

The text does not get any special treatment: it is only required to be codified in ASCII form.

With regard to term delimitation and extraction the more interesting sub-system is the morphosyntactic parser. It consists of 3 modules: a morphological parser; a syntactic parser and a synapsy detector.

The morphological parser has two functions: *a)* automatic categorisation; *b)* lemma and tag identification. According to Plante and Dumas, 30% of words in French can be attributed to more than one category. This has led to the tagging of all the possible categories for each word. As a result, there is an overproduction of words with different tags. Categorisation and lemmatisation are obtained from the application of the LCML program, it is not a dictionary but a morphological parser of lexical forms so it can correctly lemmatise and tag new lexical forms.

The syntactic parser is responsible for weeding out the vast majority of ambiguities generated in previous stages. It is managed through the construction of a syntactic structure for each sentence.

Finally, the synapsy detector (MRSF) selects, among the syntactic units from the parser, those lexical noun units that are likely to be terms. S. David (David and Plante, 1991) created MRSF especially for TERMINO. MRSF is based on principles of noun group construction. David's understanding of synapsy is that of a polylexical unit of a syntactic nature that is the head of the NP. Thus, synapsies are only NPs groups: some of them will become terms and some of them will not. Further, some of them will only be "topics" that will enable the terminologist to know different concepts or grasp an overview of the text topics. The MRSF module comprises 5 sub-modules: (1) head hunter module; (2) expansion recogniser module; (3) categorisation module; (4) synapsy generator module and (5) representation and evaluation module.

TERMINO has a set of software tools, which is much larger and comprises different modules that allow to manipulate terminological data. These tools help the terminologist decide whether a synapsy is a term or not, elaborate terminological filing forms and create terminological databases.

TERMINO recognises between 70% and 74% of the complex terms. The fact that 30% of terms are not recognised by TERMINO can be explained by coordination (it is a signal of segment breaking), acronyms and common nouns in capital letters. Moreover, there is 28% of noise, of which 47% is due to a wrong mark-up and a 53% is due to synapsies belonging to general language.

Evaluation

TERMINO is one of the first candidate term extractors that worked and it is a linguistically-based extractor, composed of different independent modules. This system is based on the concept of *synapsy*. The synapsy detector is based on the establishment of a number of heuristic rules that may well be increased provided the corpus is delimited.

There is a need to improve this system taking into account that it is still too noisy (28%), which could be improved, for example, with a different treatment of capital letters and acronyms.

2.12. *TERMS*

Reference publication: Justeson and Katz (1995)

Main goal: Term extraction

Justeson and Katz (1995) hold the following views about terms:

a) Terminological noun phrases (TNP) are different from non-terminological noun phrases (nTNP) in that the modifiers of the first ones are much shorter than those of the second ones.

b) An entity introduced by a nTNP can be later referred to only by the head of the NP and often by other NP (synonyms, hyponyms, hyperonyms). By contrast, an entity introduced by a TNP is normally repeated identically in a given document, as a single omission of a modifier could yield a change of the referred entity.
c) In technical texts lexical NPs are almost exclusively terminological.
d) Multiword technical terms are nearly always composed of nouns and adjectives (97%) and some prepositions (3%) between two NPs.
e) The average length of a TNP is of 1.91 words.
The proposed filter finds strings with a frequency equal or higher than two. These strings follow with this regular expression: ((A|N)+ | ((A|N)*(N P)?)(A|N)*N. Those candidate terms of a length of 2 (2 patterns: AN and NA) and 3 (5 patterns: AAN, ANN, NAN, NNN and NPN) are by far the most commonly encountered.
The purpose of this algorithm is to combine good coverage of the usual terminology from technical texts with high quality in the extraction phase. The algorithm prefers quality to coverage, since if it only made use of the grammatical constraints then the system would propose many irrelevant NPs. The vast majority of relevant NPs overcome the frequency constraint.
Selection of grammatical patterns also affects quality. If prepositions are admitted within the pattern many candidates are introduced, although few will be valid. As a result, quality decreases whereas quantity increases and, accordingly, Justeson and Katz (1995) prefer not to take prepositions into consideration.
The implementation of grammatical patterns also affects the quality/coverage trade-off. There are two ways in which a given linguistic unit is attributed to a grammatical category: disambiguation and filtering. The first one is rejected because disambiguators are not totally reliable yet.
Filtering consist in parsing and lemmatising each word of the text. Then those sequences following the pattern are considered. If a word is not identified as a noun, adjective or preposition, it is discarded. Thus each word maintains its nominal, adjectival and prepositional values and in this order. The chain is weeded out if more than one word can be identified as a preposition or if it does not follow the pattern (e.g. if the pattern ends with a noun and there is more than one preposition then the word following the preposition is not a noun).
Filtering has a coverage at least as good as what can be attained by a standard tagger. However, quality is not that good (e.g. *fixed* is only identified as an adjective –*bug fixed*–, but it can also become a verb: *fixed disk drive*). In contrast, filtering is much faster than parsing.
In any case, Justeson and Katz (1995) suggest to control the patterns, the list of grammatical words and the frequency to adjust the performance of the system to each type of text.

This system has been applied to different domains (metallurgy, spatial engineering and nuclear energy) and it is used at IBM Translation Center. The TERMS results are presented on the basis of 3 technical texts (statistical classification of patterns, lexical semantics and chromatography). Coverage has only been estimated for one of the text and it is of 71%. Quality has been estimated between 77% and 96% of the instances.

Evaluation

Although Justeson and Katz (1995) present a detailed study on the performance of terminological units (wherein there are some overstatements), the proposed filter does not seem to take advantage of these previous analyses of terms. Further, it should be noted that this type of filtering based on quite simple patterns would not be so efficient if they were applied to languages other than English such as Romance languages. Also, this kind of patterns produces a lot of noise.

3. Contrastive Analysis

Here we will contrast the systems' main features, according to six relevant aspects when designing a new detection system of terminological units: linguistic resources, strategies of term delimitation, strategies of term filtering, classification of recognised terms and obtained results. For some of these criteria we have created a table containing the most significant data so as to make the system comparison easier.

3.1. *Linguistic resources*

It has been observed that the vast majority of the reviewed systems make use of some sort of linguistic information, at least a list of empty words taken as boundaries. The standard process includes a morphological analysis followed by some kind of disambiguation system. The systems altering this procedure are the following:

a. ANA: does not use any linguistic resource, just a list of auxiliary words
b. TERMS: use its own disambiguation system: POS filtering
c. Naulleau: introduces semantic information

Additionally, for the systems that use an incremental strategy, like ANA and Fastr, it is necessary a set of initial terms to bootstrap the process.

3.2. *Strategies of term delimitation*

All systems of terminology extraction have to determine at some point the beginning and the end of the candidate term, that is, delimit the potential terminological unit. The reviewed programs have different strategies to delimit terms: word-boundary elements, structural patterns, syntactic parser, text distribution, typographical elements, term lists, structure disambiguation. Below we show a summary of the different options adopted by each system:

Table 1: *Strategies of term telimitation*

System		term delimitation				structure disamb.	
	Name /Author	boundaries	Patterns	Parser	Other	learning	Other
1	ANA				X		-
2	CLARIT			X			statistical
3	Daille		X				-
4	FASTR		X	X	X		-
5	Heid		X				-
6	LEXTER	X				X	
7	Naulleau				X		-
8	NEURAL		X				-
9	NODALIDA-95				X		-
10	Termight		X				-
11	TERMINO			X			-
12	TERMS		X				-

3.3. *Strategies of term filtering*

Term filtering is a key stage of any term detection system. This means that the list of candidates is reduced as much as possible. The following table shows the strategies found in all the reviewed systems:

Table 2: *Strategies of term filtering*

System		Term Filtering					
	Name /Author	Freq.[10]	Linguistic	statistical + linguistic	linguistic + statistical	reference terms	user defined
1	ANA					X	
2	CLARIT			X	X		
3	Daille				X		
4	FASTR					X	
5	Heid		X				
6	LEXTER		X				
7	Naulleau						X
8	NEURAL				X		
9	NODALIDA-95		X				
10	Termight	X	X				
11	TERMINO		X				
12	TERMS	X	X				

[10] The technique of term filtering through frequency terms has been considered something in between those methods based on linguistic knowledge and those methods based on extralinguistic knowledge.

3.4. *Classification of recognised terms*

Some of the analysed systems classify recognised terms by grouping them according to some criteria. Thus, the related terms stay close to each other. Even FASTR attempts to infer an ontology from the recognised terms. Those systems which show some classification of recognised terms are the following:

a. ANA: it builds a semantic network from the detected terms.

b. FASTR: it builds a graph to relate recognised terms. Also it proposes the construction of partial ontologies for some terms.

c. LEXTER: it builds a terminological network splitting terms into head and expansion.

3.5. *Results*

The table below summarises for each system the type of corpus used for the tests and the results attained:

Table 3: *Results*

System		Test corpora			Terms %	
	Name /Author	Domain	Language	Size.[Kw.]	precision	recall
1	ANA	Aviation engineering Acoustics	French English	120 25	 ?	? ?
2	CLARIT[11]	News	English	240 Mb	-	81.6
3	Daille	Telecommunications	French	800	?	?
4	FASTR	Medicine (abstracts)	French	1.560	86.7	74.9
5	Heid	Engineering	German	35	?	?
6	LEXTER	Engineering	French	3.250	95	?
7	Naulleau	Technical	French	?	?	?
8	NEURAL	Medicine	English	55	?	70
9	NODALIDA-95	Cosmology Technical text	English	20	95-98	98.5-100
10	Termight	Computer science	English	?	?	?
11	TERMINO	Medicine	French	?	72	70-74
12	TERMS	Statistics Semantics Chromatography	English	2.3 6.3 14.9	77 86 96	

4. Conclusions

We can reach some conclusions after having analysed and evaluated some of the main systems of TE designed in the last decade:

[11] The system has been intensively tested with regard to the indexing frequency, but not in relation to the quality of the extracted terms.

a) The efficiency of the extraction presents a high degree of variation from one to another. Broadly speaking, there is neither clear nor measurable explanation of the final results. Besides, we have to bear in mind that these systems are tested with small and highly specialised corpora. This lack of data makes it difficult to evaluate and compare them. However, it does not prevent pinpointing those solutions, which are considered valid to solve specific problems.

b) None of the systems is entirely satisfactory due to two main reasons. First, all systems produce too much silence, especially statistically-based systems. Second, all of them generate a great deal of noise, especially linguistically-based systems.

c) Taking into account the noise generated, all systems propose large lists of candidate terms, which at the end of the process have to be manually accepted or rejected.

d) Most of the TE systems are related to only one language: French or English. Usually the language specific data is embedded in the tool. This makes difficult to use the system in a language other than the original.

e) As has been already pointed out, training corpora tend to be small (from 2.3 to 12 Kwords) and highly specialised with regard to the topic as well as the specialisation degree. This allows for a quite precise patterns and lexicosemantic, formal and morphosyntactic heuristics albeit this only applies to highly specialised corpora.

f) All systems focus entirely on NPs and none of them deals with verbal phrases. This is because there is a high rate of terminological NPs in specialised texts. This rate can vary according to the topic and the specialisation degree. Despite what has just been noted, it is noteworthy that all specialised languages have their own verbs (or specific combinations of a verbal nature), no matter how low the ratio is in comparison with nouns.

g) As a result, none of the systems refers to the distinction between nominal collocations and nominal terminological units of a syntactic nature. Nor do they refer to phraseology.

h) Many of the systems make use of a number of morphosyntactic patterns to identify complex terms. However they account for most of the terminological units they are still too few and also not very constraining. Thus, for English are AN and NN, for French NA and N prep N. Some terms present structures other than these ones and they are never detected. Those systems based only on these types of linguistic techniques generate too much noise.

i) It is generally agreed that frequency is a good criterion to indicate that a candidate term is actually a terminological unit. However, frequency is not on its own a sufficient criterion, as it yields a great deal of noise.

j) Only a few recent systems use semantic information to recognise and delimit terminological units although its use takes place at different levels.

k) None of the systems uses extensively the combinatory features of terms from specialised languages in relation to a given domain. It is needed more studies about the type of constraints that terminological units present with regard to conceptual field and text type.
l) Only one of the analysed systems take profit of the possibilities given by the alignment of specialised text.
i) Most of the authors consider the POS disambiguation as one of the most important error sources. However, they do not provide exact figures about its incidence degree.
To improve these systems of terminology extraction and lessen the noise and silence that are generated, two type of studies should be encouraged. First, it is required more linguistic oriented studies on the semantic relationships among terms, the semantic relationships among constituents of a terminological unit, semantico-lexical representation, constraints of terminological units within a given specialised domain and in a given text type, all the grammatical categories that are likely to become terms in specialised domains, the influence of the syntactic function of terminological phrases on texts, the relationships between terms and their arrangement in texts.
Second, we should focus on software systems that: combine in a more active manner statistical and linguistic methods; improve statistical measures; combine more than one strategy; are easily applicable to more than one language; improve interfaces to facilitate the machine-user interaction. Also it should be very useful, as suggested in Kageura *et al.* (1998), the development of a common test bench for aiding the evaluation/comparison of extracting methods.
In sum, should we progress in the field of automatic terminology extraction, statistical and linguistic methods have to actively be combined. It means that they are not either-or approaches but complementary ones. The final goal is to reduce the amount of silence and noise so that the process of terminological extraction becomes as automatic and precise as possible. In the future, we believe that any current terminology extractor, apart from accounting for the morphological, syntactic and structural aspects of terminological units, has to necessarily include semantic aspects if the efficiency of the system is to be improved with regard to the existing ones.

Acknowledgements

This paper has been written within the research project PB-96-0293, financially supported by the Spanish government.

We would like to thank Dr. H. Rodríguez (Universitat Politècnica de Catalunya), Dr. C. Jacquemin (Université de Nantes), Dr. L. de Yzaguirre and Mrs. J. Freixa (Universitat Pompeu Fabra) their valuable comments on preliminary versions of this paper. The remaining shortcomings are ours. We thank J. Morel for translating this paper from Catalan into English and C. Rodríguez its revision.

References

Arppe, A. 1995. “Term extraction from unrestricted text”. *Lingsoft Web Site: http://www.lingsoft.com*

Ahmad, K., Davies, A., Fulford, H. and Rogers, M. 1992. “What is a term? The semiautomatic extraction of terms from text”. *Translation Studies – an interdiscipline*. Amsterdam: John Benjamins.

Bourigault, D. 1994. *LEXTER, un Logiciel d'EXtraction de TERminologie. Application à l'acquisition des connaissances à partir de textes.* PhD Thesis. Paris: École des Hautes Études en Sciences Sociales.

Bourigault, D., Gonzalez-Mullier, I. and Gros, C. 1996. “LEXTER, a Natural Language Processing Tool for Terminology Extraction”. *Proceedings of the 7th EURALEX International Congress.* Göteborg.

Brown, P. F., Cocke, F., Pietra, S., Felihek. F., Merces, R. and Rossin, P. (1988) A statistical approach to language translation. *Procedings of 12th International Conference of Computational Linguistic (Coling-88).* Budapest, Hungary.

Cabré, M.T. 1999. *Terminology. Theory, methods and applications.* Amsterdam: John Benjamins.

Church, K. 1989. “Word association norms, mutual information and lexicography”. *Proceedings of the 27th annual meeting of the ACL.* Vancouver, 76-83.

Condamines, A. 1995. “Terminology: new needs, new perspectives”. *Terminology,* 2, 2: 219-238.

Dagan, I. and Church, K. 1994. “Termight: Identifying and translating technical terminology”. *Proceedings of the Fourth Conference on Applied Natural Language Processing,* 34-40.

Daille, B. 1994. *Approche mixte pour l'extraction de terminologie: statistique lexicale et filtres linguistiques*. PhD dissertation. Paris: Université Paris VII.

Daille, B. and Jacquemin, C 1998. “Lexical database and information access: a fruitfull association?”. *First International Conference on LREC.* Granada.

David, S. and Plante, P. 1991. “Le progiciel TERMINO: de la nécessité d'une analyse morphosyntaxique pour le dépouillement terminologique des textes”. *Proceedings of the Montreal Colloquium Les industries de la langue: perspectives des années 1990,* 1: 71-88.

Enguehard, C. and Pantera, L. 1994. “Automatic Natural Acquisition of a Terminology”. *Journal of Quantitative Linguistics,* 2, 1: 27-32.

Estopà, R. 1999. *Extracció de terminologia: elements per a la construcció d’un SEACUSE (Sistema d’extracció automàtica de candidats a unitats de significació especialitzada).* PhD thesis, Barcelona: Universitat Pompeu Fabra.

Estopà, R. and Vivaldi, J. 1998. “Systèmes de détection automatique de (candidats à) termes: vers une proposition intégratrice”. *Actes des 7èmes Journées ERLA-GLAT, Brest*, 385-410

Evans, D.A. and Zhai, C. 1996. “Noun-phrase Analysis in Unrestricted Text for information retrieval”. *Proceedings of ACL, Santa Cruz, University of California,* 17-24.

Frantzi, K. and Ananiadou, S. 1995. *Statistical measures for terminological extraction.* Working paper of the Department of Computing of Manchester Metropolitan University.

Frantzi, K. T. 1997. “Incorporating context information for extraction of terms”. *Proceedings* of ACL/EACL, Madrid, 501-503.

Habert, B., Naulleau, E. and Nazarenko, A. 1996. *“*Symbolic word clustering for medium-size corpora”. *Proceedings of Coling’96*: 490-495.

Heid, U., Jauss, S., Krüger, K. and Hohmann, A. 1996. “Term extraction with standard tools for corpus exploration. Experience from German”. In: *TKE ‘96: Terminology and Knowledge Engineering,,* 139-150. Berlin: Indeks Verlag.

Jacquemin, C. 1994. “Recycling Terms into a Partial Parser”. *Proceedings of ANLP’94*, 113-118.

Jacquemin, C. 1999. “Syntagmatic and paradigmatic representations of term variation”. *Proceedings of* ACL'99, University of Maryland, 341-348.

Jacquin, C. and Liscouet, M. 1996. “Terminology extraction from texts corpora: application to document keeping via Internet”. In: *TKE ‘96: Terminology and Knowledge Engineering, 74-83.* Berlin: Indeks Verlag.

Justeson, J. and Katz, S. 1995. “Technical terminology: some linguistic properties and an algorithm for identification in text”. *Natural Language Engineering,* 1, 1: 9-27.

Kageura, K. and Umino, B. 1996. “Methods of Automatic Term Recognition”. Papers of the *National Center for Science Information Systems*, 1-22.

Kageura, K., Yoshioka, M., Koyama, T. and Nozue, T. 1998. "Towards a common testbed for corpus-based computational terminology". *Proceedings of Computerm '98*, Montreal, 81-85.

Karlsson, F. 1990. "Constraint grammar as a framework for parsing running text". *Proceedings of the 13th International conference on computational linguistic*, 3: 168-173.

Lauriston, A. 1994. "Automatic recognition of complex terms: Problems and the TERMINO solution". *Terminology*, 1, 1: 147-170.

Maynard, D. and Ananiadou, S. 1999. "Identifying contextual information for multi-word term extraction". In: *TKE '99: Terminology and Knowledge Engineering, 212-221.* Vienna: TermNet.

Nakagawa, H. and Mori , T. 1998. "Nested collocation and Compound Noun for Term Extraction". *Proceedings of Computerm '98,* Montreal, 64-70.

Naulleau, E 1998. *Apprentissage et filtrage syntaxico-sémantique de syntagmes nominaux pertinents pour la recherche documentaire.* PhD thesis. Paris: Université Paris 13.

Naulleau, E. 1999. "Profile-guided terminology extraction". In: *TKE'99: Terminology and Knowledge Engineering*. 222-240. Vienna: TermNet.

Plante, P. and Dumas, L. 1998. "Le Dépoulliment terminologique assisté par ordinateur". *Terminogramme*, 46, 24-28.

Shieber, S.N. 1986. "An Introduction to Unification-Based Approaches to grammar*". CSLI Lecture Notes of* University Press, 4.

Smadja, F. 1991. *Extracting collocations from text. An application : language generation*. Columbia: Columbia University. Department of Computer Science. [Unpublished doctoral dissertation]

Voutilainen, A. 1993. "NPtool, a detector of English noun phrases". *Proceedings of the Workshop on Very Large Corpora*.

Zhai, C., Tong, X., Milic-Frayling, N. and Evans, D.A. 1996. "Evaluation of syntactic phrase indexing CLARIT. NLP track report". *Proceedings of the TREC-5.* TREC Web Site: *http://trec.nist.gov/pubs/trec5/t5_proceedings.html*