Building a treebank for French

Anne Abeillé*, Lionel Clément*, Alexandra Kinyon*◇

*TALaNa, Université Paris 7 ◇University of Pennsylvania
75251 Paris cedex 05 Philadelphia
FRANCE USA
abeille, clement, kinyon@linguist.jussieu.fr

Abstract

Very few gold standard annotated corpora are currently available for French. We present an ongoing project to build a reference treebank for French starting with a tagged newspaper corpus of 1 Million words (Abeillé et al., 1998), (Abeillé and Clément, 1999). Similarly to the Penn TreeBank (Marcus et al., 1993), we distinguish an automatic parsing phase followed by a second phase of systematic manual validation and correction. Similarly to the Prague treebank (Hajicova et al., 1998), we rely on several types of morphosyntactic and syntactic annotations for which we define extensive guidelines. Our goal is to provide a theory neutral, surface oriented, error free treebank for French. Similarly to the Negra project (Brants et al., 1999), we annotate both constituents and functional relations.

1. The tagged corpus

As reported in (Abeillé and Clément, 1999), we present the general methodology, the automatic tagging phase, the human validation phase and the final state of the tagged corpus.

1.1. Methodology

1.1.1. Choosing the corpus

The corpus consists of extracts from the daily newspaper Le Monde, ranging from 1989 to 1993, and covering a variety of authors and domains (economy, literature, politics, etc.), representative of contemporary written French. It comprises roughly 1M. tokens.

1.1.2. Choosing the tagset

We define a complete morphosyntactic tag as follows:

1. POS (ex Determiner)
2. subcategorization (ex possessive or cardinal)
3. inflection (ex masculine singular)
4. lemma (canonical form)

For parts of speech, we made traditional choices, except for weak pronouns that were given a POS of their own (clitic) according to the generative tradition (Kayne, 1975), and foreign words (in quotations) which receive a special POS (ET). Punctuations are divided between strong (clause markers) and weak (all the others). Most typographical signs (including %, numbers and abreviations) are assigned a traditional POS (usually Noun). We chose to annotate more than just parts of speech, for the following reasons: Some parts of speech are too inclusive (e.g. conjunctions or nouns) and further distinctions (called here subcategories) are needed (e.g. proper and common for nouns, subordinating or coordinating for conjunctions), if one wants to annotate linguistically motivated distributional classes. Some words are unambiguous for parts of speech but ambiguous for such subcatgegories, for example *neuf* which can either be a numeral adjective (= nine) or a predicative adjective (= new), lui which can either be a strong personal pronoun (= him) or a weak clitic pronoun (= to him or to her), *plus* can either be a negative adverb (= not any more) or a simple adverb (= more). Inflectional morphology also has to be annotated since morphological endings are important for gathering constituants (based on agreement marks) and also because lots of forms in French are ambiguous with respect to mode, person, number or gender. For example, the determiner *ces* can be either masculine or feminine, the verb form *mange* can be either indicative or subjunctive, and either first or third person, or even 2d person imperative. Compounds also have to be annotated since they may comprise words not existing otherwise (e.g. insu in the compound preposition *à l'insu de* = to the ignorance of) or exhibit sequences of tags otherwise agrammatical (e.g. *à la va vite* = Prep + Det + finite verb + adverb = in a hurry), or sequences with different grammatical properties than expected from those of the parts: *peut-être* is a compound adverb made of two verb forms, a *peau rouge* (= american indian) can be masculine (although *peau* (= skin) is feminine in French) and a *cordon bleu* (chief cook) can be feminine (although cordon (= lace) is masculine in French). Some sequences are ambiguous between compound and not compound interpretations, altough in corpora the compound interpretation often prevails :

(1) Paul veut bien que Marie vienne (Paul wants indeed that Marie comes).

(2) Paul pleure bien que Marie vienne (Paul is crying although Marie is coming)

(3) Paul en fait a raison (Paul in fact is right)

(4) Paul en fait trop (Paul is acting too much)

In (1), there is no compound : *bien* is an adverb (=well) and *que* a subordinating conjunction (=that); whereas in (2) the same sequence *bien que* is a compound subordinating conjunction (=although). In (3), the sequence *en fait* is a compound adverb (in fact), whereas in (4) the same sequence must be decomposed into *en* as a clitic and *fait* as a finite verb. Compounds are annotated with the same tagset as not compounds, plus tags for each of their parts. Since

where to draw the limit between compounds and free sequences is subject to much linguistic debate, we chose to also annotate the parts of the compounds. Tagging the parts as well is useful for specific studies on compounds, but also if a user wants to view our corpus without the compounds already amalgamated.

1.1.3. The tagging pipeline

The overall organisation of the tagging phase is more complex than just automatic tagging followed by human validation, because of the rich tagset we are using. Segmentation (for compounds) is done before tagging and lemmatization after. At each phase, we try to minimize the number of tags involved, so in practice we define three different tagsets: a reduced one for the tagger (in order to minimize its errors), an enriched one for the annotators (so that all possible ambiguities are resolved but without bothering the annotators with distinctions easy to make automatically), and the final tagset of the treebank. Mapping tools between these tagsets have thus been developed.

1.2. Automatic tagging of the corpus

Due to the lack of reusable annotation tools at the beginning of the project, we have developed a morphosyntactic tagger for French (Reyes, 1997), (Abeillé et al., 1998). Our tagger, which is intended to be used independently of the project, is based on Brill (1993) in that it has two phases (initial or dummy tagging, and context sensitive tag rewriting). The main difference with a true Brill tagger is that we have added an external lexicon and rely mainly on manually written contextual rules. The tagger uses a reduced tagset for POS and morphology (110 tags), mainly derived from existing lexicons, with only a few simplifications for distinctions difficult to handle automatically (for example between interrogative and relative pronouns which are ambiguous forms in French).

1.2.1. The tagger's lexicon

The lexicon of the tagger comprises over 360,000 forms including 36,000 compounds. It comes from the lexicon we had developed for our French parser (FTAG, (Abeillé, 1991), (Candito, 1999), plus some extracts from MULTEXT lexicon, from ABU lexicon (for proper names) and from INTEX for compounds (Silberztein, 1993). It has been extended with most forms from the corpus (excluding numbers and specific proper names).

1.2.2. Segmentation

Our tagger comprises a sentence splitter and a tokeniser. The sentence splitter uses lexical data to properly distinguish between capital letters for proper names or beginning of sentences, between dots for acronyms or end of sentence, between hyphenation and linking dashes, etc. (cf. (Silberztein, 1993). As in English, word segmentation is always a problem since lots of compounds show up as separate words in French (pomme de terre = potatoe, bien que = although etc). The tokenizer thus reads the lexicon for compound recognition and amalgamates the best compound candidates (choosing the longest one in case of several candidates; for example the compound adverb (or noun) face à face and not the compound preposition face à in the sequence face à face). Amalgamating compounds before tagging helps the tagger in most cases. It can trigger errors in the case of sequences ambiguous between compounds and simple words (*en fait* = compound adverb 'in fact' or clitic pronoun - Verb 'makes of-it'), but these cases are rare in real texts and can be solved by lexicon tuning.

1.2.3. Unknown words

As with Brill's tagger, our tagger uses lexical rules for unknown words. We currently have 198 such rules corresponding to common suffixes for verbs, nouns, and adjectives. They are somewhat similar to a morphological analyser. We also have regular expressions for numbers and propers nouns (acronyms). Since our lexicon has been extended as part of the project, most unknown words are foreign words and typos.

1.2.4. Initial tagging

The initial (dummy) tagging is important since more than 40% of the words in our corpus receive more than one tag with the tagger's lexicon (and more than 20% more than 2 tags). In order to assign the best possible tag for each word, we rely on the trigram method using genotypes as data (cf. (Tzoukermann et al., 1995)) and computed the probability of each tag for each word with a large unannotated corpus (of newspaper texts).

1.2.5. Contextual retagging

The initial tag assigned to each word (by lexical lookup) can be changed depending on the context. The form été has Verb (past participle) as its most probable tag, but must be rettaged as Noun after a Determiner for example. Contrary to Brill's automatic rule induction approach (which gave poor results, cf. (Reyes, 1997), (Abeillé et al., 1998), we prefered to develop most of the contextual rules by hand, based on linguistic knowledge and corpus lookup. In order to fit the linguist's need for expressivity, we have added compositionality to the rule formalism, as well as three operators (for negation, for testing whether a tagset contains a specific tag, and to force a transformation even if the tag is not in the tagset of the word in the lexicon). We also have added the possibility to have unifiable variables in the tag names (for morphological agreement). The tagger uses 322 contextual (retagging) rules.3 The contextual rules cannot deal with compound/not compound ambiguity (*carte bleue*: blue card or credit card) since they cannot modify the initial segmentation. The tagger does not handle lemmas, which are handled by a specific postprocessor. It performed on our corpus with an error rate of about 5%.

1.3. Validating the tagged corpus

Word segmentation (compound recognition) had to be validated by systematic human scrutiny of the tagged corpus, as well as for each form (simple or compound) POS and inflection. Some subcategorization information, most lemmas and all parts of compounds only depend on lexical information (independently of the context) and were added automatically by lexical lookup (once compound segmentation, POS and morphology had been manually validated). For the two main manual validation tasks (compound validation and tagging validation), very precise guidelines have

TAG	Subcategorization	Morphology	Description
N	Common, proper	f,m + s,p	Nouns
A	Card., ordinal, possessive, qualif., indef.	f,m + s,p + 1,2,3	Adjectives
Adv	-, inter, exclam, negative	-	Adverbs
P	-	-	Prepositions
D	card, dem, def, indef, excl., neg., poss	f,m + s,p + 1,2,3	Determiners
CL	subj, refl, obj, -	f,m + s,p + 1,2,3	Clitic pronouns
PRO	inter, pers, negative, poss, rel, indef	f,m + s,p + 1,2,3	Otherpronouns
C	Subord, Coord	-	Conjunctions
I	-	-	Interjections
V	-	W, G, K, P, I, J, F, T , C, S, Y + f,m + s,p + 1,2,3	Verbs
ET	-	-	Foreign words
PONCT	Strong, weak	-	Punctuation

Table 1: Tagset of the tagged corpus

been written (Abeillé and Clément, 1997) and updated during the project. The reference tagged corpus was checked and corrected by two annotators (one after another) reading (some part of) the text in a longitudinal way, then some tools were applied to check the most difficult cases (for some frequent grammatical words such as de or que). Weekly meetings were also necessary to ensure consistency between annotators (over 15 persons were involved altogether).

1.3.1. Validation of compounds

The automatic annotation for compounds was done by INTEX (Silberztein, 1993) and by our tagger. We asked the annotators to validate the compound interpretation in context, to add compounds missing from our lexicons (especially for proper names) and to add discontinuous instances of compounds that could not be automatically found. We gave them guidelines about what to consider a compound based on linguistic tests, using morphological tests (parts not existing otherwise: *fi* in *faire fi de* (ignore)), syntactic tests (no internal modification or determination :*carte bleue* (credit card) **carte très bleue*) and semantic criteria (opacity : *en revanche* (=on the contrary, lit in a revenge)). Lots of candidate compounds turned out not to be compounds at all. For example *sur ce* can be the compound adverb (on the spot) but was always the preposition (*sur*) followed by the determiner (*ce*). To our surprise, very few discontinuous compounds (*afin<justement>de* 'in order precisely to') were found in the corpus. Annotating the parts of the compounds was done automatically with our compound lexicon.

1.3.2. Validation of tags

Our complete tagset comprises 250 tags (see table 1). Since most subcategories can be assigned unambiguously to a word (once its POS is known), we chose to simplify the tagset for the annotators. Possessive determiners, for example, can be ambiguous with other POS but not with other determiner subcategories, and the same for possessive pronouns; so the subcategory Possessive can be eliminated from the annotators' tagset. The tagset for the annotators was thus reduced to 122 tags, and they were presented with subcategories only in case of possible ambiguity (*neuf* as cardinal or qualifying adjective ('nine' or 'new'), lequel ('which') as interrogative or relative pronoun etc.). Annotators had to validate the output of the tagger and to add subcategories when needed. Most of the subcategories were added automatically with lexical lookup afterwards. Difficult cases involved tagging numbers, tagging weak pronouns (clitics), choosing between adjective and past par-

ticiple, between proper and common Noun (for unknown words), between Prep and (indefinite or partitive) Det (for de). For numbers, we depart from Multext guidelines in choosing the same tagset as other words. The annotators had thus to choose between:

- determiner : Deux hommes sont venus (Two men came)
- pronoun : Il en a accueilli deux (He welcomed two of them)
- adjective : Les deux hommes sont venus (The two men came)
- noun : Le joueur a misé sur le deux (The player bet on the two)

For clitic pronouns, we simplified the usual case system and kept only nominative / objective / reflexive subcategories, since assigning the right case (or no case at all for uses as inherent clitics or mediopassive) is part of syntactic analysis and will be done (partly automatically) in the second phase of the project. Another difficulty is that most clitic forms in French are ambiguous with respect to gender (*je, leur, les*..) or number (*se*) or both (*y, en*). The annotator had thus to find their antecedent to properly annotate their morphosyntax.

Most difficult cases involved ambiguous grammatical words (such as *tous* 'all' or *que* 'that') the tagging of which is a matter of debate among linguists since it depends on the syntactic analysis of notoriously complex constructions (cleft sentences, comparatives etc). In such cases, we made obvioulsy debatable choices: our main goals was to be explicit (in the documentation), consistent (throughout the corpus) and theory neutral (so that our tagging is compatible with several syntactic analyses).

1.3.3. Validation of lemmas

The lemmas were not shown to the annotators. They were added automatically (using our lexicon) after tag correction. At this stage, very few lemma ambiguities remained (suis VP1s from *suivre* 'follow' or *être* 'be', *fils* NCmp from *fil* 'thread' or *fils* 'son' ...). They were resolved by hand. The well known ambiguities such as *savons* (= *savon* 'soap' or *savoir* 'know'), *portes* (= *porte* 'door' or *porter* 'carry'), which are problematic for most lemmatizers, do not arise once the corpus has been tagged with parts of speech.

1.3.4. Status of the tagged corpus

The automatically tokenized and tagged corpus has been manually validated and enriched (with longitudinal exhaustive checking by at least two different human annotators). The 1 Million tokens amount when annotated to 870,000 words, excluding punctuation signs, and clustering compounds into one word, for a total of 32,000 sentences with 17,000 different lemmas. There are no remaining ambiguities, nor unknown words (the original typos have been corrected). It is available in two versions :

- light version with a reduced tagset in a compact format,
- full version with a richer set of tags, lemmas, annotation for parts of compounds, all in SGML format

It is freely available for research purposes. The cost for 1 Million words was about 50 man month (including tagger development). The average correction rate was 500 words per hour. This is much lower than that of the Penn Treebank (2000 words per hour) because of the compounds, and because of our richer tagset (the annotators were presented only 36 tags in the Penn Treebank). Each text was validated twice by two different annotators in succession (one correcting or validating the work of the other). The coordination task included writing the documentation, writing tools for the annotators and for postchecking the annotators' work, and weekly meeting with the annotators.

2. The parsing phase

We first present our annotation choices, then our tools for automatic syntactic annotation, then the preliminary validation phase. Contrary to the tagged version of the corpus, which has been entirely validated (and should be error-free), the validation of the parsed version is not complete yet.

2.1. Syntactic annotation scheme

Contrary to tagging annotations, language specific guidelines are usually missing for syntactic annotations. In order to provide annotations reusable by researchers with various backgrounds, we chose to annotate both constituency and functional relations. We focus on surface and shallow annotations, compatible with various syntactic frameworks.

For constituency, we annotate only major phrases, with very few internal structure (we have determiners and modifying adjectives at the same level in the noun phrase for example). For verbal phrases, we only annotate the minimal verbal nucleus (clitics, auxiliaries, negation and verb), because the traditional VP is subject to much linguistic debate and is often discontinuous in French. In order to be as theory neutral as possible, we do not use empty categories, nor functional phrases (no DP or CP). For certain phrases, we annotate a subcategory, which is of importance for functional annoation, for example relative or subordinate for embedded sentences.

For functional relations, we annotate both surface function (for major phrases) and subcat frames (or valence information) for verbs (including the subcat aux or modal for auxiliaries and modals). We do not annotate ellipsis, nor pronoun-antecedent relations. We do not annotate deep functions (such as the deep subject of an infinitive).

The following information is contained in each syntactic tag :

1. Main category (e.g. VP, NP, ...)
2. Eventual subcategory (e.g. Rel for relative clauses)
3. Surface function (eg. Subj, Object for NPs)
4. Begin or end of phrase

Phrasalcategory	Subcategorization	Function	Description
<NP>, </NP>	-,coord	-, Subj, Obj, Loc-obj, Pred-obj, A-mod, P-mod	Noun phrases
<VN>, </VN>	-,coord	-	Verbal nucleus
<VPinf>,<.VPinf>	-, a, de, sub, coord	-, Subj, Obj-inf, A-obj, De-obj, A-mod, P-mod	Inf. and nonf. clauses
<PP>, </PP>	-,a, de, coord	-, A-obj, De-obj, Loc-obj, Agt-obj, A-mod, P-mod	Prep. Phrases
<AdP>, </AdP>	-, neg, coord	-, Loc-obj, Man-obj, A-mod, P-mod	Adv. Phrases
<AP>, </AP>	-,coord	-, Pred-obj, A-mod, P-mod	Adj. phrases
<S>, </S>	-, inter, sub, comp, rel, coord	-, Obj-comp, Subj, Obj-int, A-mod, P-mod	Sentences and finite clauses
<name>, </name>	-	-	Names (clusters)
<title>, </title>	-	-	Titles (clusters)
<date>, </date>	-	-	Dates (clusters)
<number>, </number>	-	-	Numbers (clusters)

Table 2: Syntactic Tagset

The set of grammatical functions associated with the phrases are surface functions derived from the French grammar FTAG (Abeillé et al., 1999), (Candito, 1996); it comprises the following functions :

subject, object, a-object, prep-object, de-object, agt-object, locative-obj, manner-obj, obj-infinitive, obj-comp, obj-interrogative, predicative-obj, premodifier, and post-modifier.

The set of valence frames are derived from the same project and currently comprise over 60 different subcat names (including auxiliary and modal).

2.2. Syntactic tools

We choose to use different tools for each task. We need a chunker for marking lexical clusters, a robust parser for marking major phrase boundaries and a functional tagger for marking syntactic functions (on major phrases) and valence (for each main verb). For marking constituency, we do not use a classical parser, but instead have adapted specific tools more robust and more suitable to our goal. For constituency, we use a rule-based shallow parser (Clement and Kinyon, 2000) (Kinyon, 2000). We proceed in 2 steps :

- marking special types of clusters (titles, numbers..) using a library of hand written regular expressions (cf (Senellart, 1999),
- marking major phrase boundaries (NP, PP, ...), with limited embedding (and no recursion) (Abney, 1990), (E., 1998).

The shallow parser is designed to minimize errors, so it does not try to attach PPs or relative clauses. These attachments have to be added by the human annotators.

2.2.1. The cluster marker

A markup tool has been developed by Lionel Clément with the help of M. Erenati and V. Nanta. The goal is to group items such as dates (e.g. *Mardi, de 9 à 12 heures et de 14h15 à 18 heures*), numbers, titles (*M. Dupond, président du comité d'organisation*). These items are recognized by hand written regular expressions (involving forms, lemmas categories and morphological endings), and delimited by SGML tags: each regular expression is transformed into a deterministic FSA (using FLEX). This marking phase allows to avoid looking further into the internal structure of these items during parsing or functional annotation. There are about 12 regular expressions for each type of cluster.6 A preliminary evaluation on 10,000 words give a sucess rate of about 80% for dates and numbers clusters. Wrong clus-

ters are about 3% and missingones are about 20%. So a task of the annotators is to add missingclusters.

2.2.2. The shallow parser

The shallow parser was developed by Alexandra Kinyon as part of the project (Kinyon, 2000). To our knowledge, few attempts have been made for chunking or shallow parsing French (contrary to English). Her goal was both to be efficient in practice, but also relevant from a psycholinguistic point of view. This is why a rule-based approach was chosen rather than a probabilistic one: rule-based systems are easier to develop and do not need a preexisting treebank (contrary to probabilistic ones) and are better motivated from the psycholinguistic point of view. Also, as argued in Tapanainen and Järvinen (1994) and as we will discuss infra, rule-based systems are not necessarily slow.

The shallow parser takes as input the tagged clustered text, slightly simplified (discarding the lemma and the morphological information, but retaining POS subcategories). It adds phrase boundaries in a left to right fashion. It was developed in java for portability and currently comprises approximately 50 rules. Each rule has access to a limited context : the previous, current and next tag plus the label of the constituent(s) currently being processed. The main underlying idea is to rely on function words as triggers of constituent boundaries (e.g. When encountering a determiner, start a NP phrase, or when encountering a clitic, start a Verbal nucleus). Although the idea to identify constituents by relying on function words is a very simple one, it has not been explicitly developed in practice to our knowledge9. Maybe this is due to the fact that most shallow parsers have been developed for English (where function words are often omitted).

In the psycholinguistic literature, little work has been carried out on the subject recently. But the role of function words in human sentence processing has been emphasized as early as (Kimball 1973), who, apart from introducing the well known "right association principle" formulates a "new node principle" which states that "The construction of a new node is signaled by the occurrence of a grammatical function word". Also, experimental evidence is presented in (Hakes 1972) showing that English sentences with complementizers are processed faster than those in which the complementizer is omitted. This result suggests that constituents which start by a non function word will eventually be identified, but not as readily as those who start with a function word. This indicates that our approach is psycholinguistically motivated.

The shallow parser uses a reduced tagset (compared to that of the final treebank) : NP, PP, VN, VPinf, AdP (for adverbial phrases), AP (for adjectival phrases), PONCT (for punctuation), S (for sentences) including Scoord (for coordination), Ssub (for sentential complements), Srel (for relative clauses), and INC (for unknown constituents).

The INC constituents are replaced in a postprocessing phase by a guesser, which tries to identify the head of the constituent to assign the correct label. If the guesser fails at guessing, it simply assigns the label AdP, since it is the most common unidentified label. Following the linguistic tradition, we consider as function words all words associated with a POS that labels a closed class i.e.: determiners, prepositions, clitics, pronouns (relative, demonstrative), conjunctions (subordination and coordination), auxiliaries, punctuation marks and adverbs that belong to a closed class (e.g. negation adverbs *ne, pas*). The general idea is that when one of these function words is encountered, an opening boundary for a new constituent is inserted in the text. Closing boundaries are added either naturally when a new constituent begins (e.g. NPs end when a new constituent starts), or triggered by a new function word (e.g. relatives and sentential complements end when a punctuation mark or a conjunction is encountered). Of course, some rules may refer to non function words (e.g. when encountering a proper noun, start an NP). Although inspired by the work on chunks presented in (Abney, 1990), the shallow parser bears 2 essential differences :

- it deals with syntactic information but do not establish any link between the constituent boundaries we introduce and any prosodic pattern in sentences
- it identifies non recursive constituents, but also do perform limited embedding (e.g. NPs embedded inside PPs , VN embedded inside a relative clause, itself embedded inside an NP) and limited attachment (e.g. coordination).

Thus, it is neither a chunker, nor a full parser (it does not attach PPs for example), hence the name shallow-parser. A sample of the raw output of the shallow parser (i.e. before postformatting and human validation) can be seen on figure 5 (light version i.e. non SGML format for the POS).

```
<S> <NP> La:Dfs proportion:NC </NP>
<PP> d':P <NP> étudiants:NC </NP> </PP>
<PP> par_rapport_à:P
<NP> la:Ddef population:NC</NP>
</PP>
<PONCT> ,:PONCT </PONCT>
<PP> dans:P <NP> notre:Dposs pays:NC</NP> </PP>
<PONCT> ,:PONCT</PONCT>
<VN> est:VP inférieure:Aqual </VN>
<PP> à:P <NP> ce:PROdem</NP> </PP>
<Srel> qu':PROR3ms
    <VN> elle:CL est:VP </VN>
<PP>à:P <NP> les:Ddef Etats-Unis:NP </NP> </PP>
<PPcoord>
        ou:CC à:P
        <NP> le:Ddef Japon:NP</NP>
    </PPcoord>
</Srel>
.:PONCT</S>
```

(the proportion of students compared to the population of our country is inferior to that in The United States or in Japan)

Table 3: Sample output of the shallow parser (light version)

The shallow parsers yields an output in linear time, since the input text is just scanned once, strictly from left to right, and constituent boundaries are added incrementally in a monotonic manner. It allows easy reusability : since it

uses few rules, and focus essentially on function words, the rules can be adapted to another tagset in very little time. To evaluate the shallow parser, we parsed the 1 million words of the tagged and hand corrected version of the corpus. On a home PC, it takes 3 minutes and 8 seconds to parse the whole tagged corpus. We put aside 1000 sentences for tuning our rules. Then, we picked at random 1000 sentences that were not in the tuning set and shallow parsed them manually, following the guidelines briefly discussed here. We then compared this to the output of the shallow parser on these same sentences. For opening brackets we obtain a recall of 94.3% (i.e. # of correct brackets in the parser's output / # of brackets in the manually chunked version) and a precision of 95.2% (i.e. # of correct brackets in the parser's output / total # of brackets in the parser's output). So, 5.7 % of the brackets are missing, while the output of the shallow parser has 4.8 % of spurious brackets. For closing brackets, we obtain a precision of 92.2 % and a recall of 91.4 %. If we now look at the labels of the brackets, 95.6% are assigned correctly. The 4.4 additional brackets are not strictly speaking assigned incorrectly, since they are labeled INC (i.e. unknown) These unknown constituents, rather then errors, constitute a mechanism of underspecification (the idea being to assign as little wrong information as possible). Half of these INC labels are reassigned the correct label by the guesser, the other half will need to be corrected manually by the annotators. To give an idea about coverage, sentences are on average 30 words long and comprise 20.6 opening brackets (and thus as many closing brackets). Fortunately, errors difficult to correct with access to a limited context involve mainly "missing" brackets (e.g. *comptez vous *ne pas le traiter* appears as a single constituent, while there should be 2), while "spurious" brackets can often be eliminated by adding more rules (e.g. for multiple prepositions: *de chez*). For closing brackets, most of the errors are due to misplaced clause boundaries. Overall, these results are very encouraging considering the simplicity of the tool, and sufficient for human validation.

2.2.3. Functional annotation

For functional annotation, we use a functional tagger currently being developed at Talana (Barrier 1999). It assigns a valence frame to each verb and a grammatical function to each major phrase.

The functional tagger uses a valence lexicon for verbs derived from the FTAG project (Abeillé et al 1999) and the LADL tables (cf. (Namer and Hathout, 1998)), and a list of regular expressions (automatically derived as the frontier nodes of the elementary trees for a given tree family in the FTAG grammar) as surface filters for assigning to each verb the most likely subcategorization frame (selecting the longest match in the context of the local clause). For each subcategorization frame, there is an average of 80 such expressions. They spot candidate arguments among the phrases in the same clause as the verb whose valence is to be tagged. Most subcategorization tools only consider the canonical realization of a given valence. The advantage of using a preexisting wide coverage grammar is to be able to also match non canonical realization of a given valence. The following filters are among those defined for the valence with nominal subject and nominal object for example table 4

2.3. Syntactic tools

It includes the inverted subject, the clitic preverbal realization of the object, or the passive variants for example. For the 1000 most frequent and most ambiguous verbs in French, we use a preexisting valence dictionary. So the task is only to choose among the possible valences for each such verb. The valence tagger uses the following ordered preferences principles:

1. prefer the grammatical valence (Aux) over the other ones,
2. prefer the longest valence match,
3. prefer the closest phrases as arguments.

Principle 1 is observed on our corpus : *avoir* and *être* are much more often the tense (or passive) auxiliaries than the main verbs (possesive or copula), and the same holds for verbs ambiguous between modal and full meaning (devoir = must or ow). The second principle implements a well know preference for arguments versus adjuncts: if a PP is a possible argument it is counted as an argument and not as an adjunct.

For the other verbs, transitive and intransitive valence are used as defaults (with the same contextual match). Only valence for verbs are considered, all PPs following adjectives or nouns (in a AP or NP) are marked as postmodifiers, since the criteria for distinguishing arguments and modifiers are less solid for non verbal categories. For phrases not marked as arguments, the general pre or post modifier function (A-mod or P-mod) is assigned looking at the position of the head of the current phrase.

2.4. Syntactic Validation

We have shallow-parsed the 1M. word tagged corpus and are in the process of validating it, with precise guidelines and regular meetings. The annotators' task consists in the following steps :

1. checking (and enriching) the names of the syntactic tags,
2. checking (and possibly moving) the position of the syntactic tags.

The first check is usually done manually only on opening boundaries and the second check usually involves moving closing boundaries.

For annotators, we use emacs tools for opening and matching closing brackets. Also, internal tools have been developed (based on unix shell scripts): this allows to extract information on the corpus (e.g. of the most frequent chunks ...) and also allows to insure consistency and error correction (e.g. finding non matching brackets, or crossed brackets, or chunks that have been assigned a non existing label because of typos)

Valence for X	Matching categories	Example
n0Vn1	NP:Subj X NP:Obj	le chat mange la souris
n0Vn1	NP:Subj Cl:Obj X	le chat la mange
n0Vn1	Cl:Subj Cl:Obj X	il la mange
n0Vn1	Cl:Obj X Cl:Subj	la mange-t-il
n0Vn1	Pro:Obj X NP:Subj	que mange le chat
n0Vn1	NP:Subj V:Aux XK	La souris sera mangée
n0Vn1	NP:Subj V:Aux XK PP:Agt-obj	La souris sera mangée par le chat
n0Vn1	Cl:Subj V:Aux XK	Elle sera mangée

Table 4: Surface filters for valence tagging

3. Conclusion

We have presented a project to build a reference treebank for French. We have developed a 1 M. word reference corpus for French (from newspaper texts) and tagged it for morphosyntax, lemmas, compounds, lexical clusters and phrase boundaries. The automatic segmentation and tagging have been validated by human annotators for the whole corpus. The reference tagged corpus is a resource to be distributed in two versions : lite (with a reduced tagset) or complete (with lemmas, internal constituants of compounds and full SGML marking). The whole corpus has been automatically annotated for clusters and phrases, and these syntactic marks are currently being validated.

As part of the project, we also have developed several syntactic annotation tools:

- a tagger inspired from (Brill 1993) but with mostly hand-written rules, an external sizable full form lexicon and a tokeniser (handling compounds).
- a cluster marker based on regular expressions for semi frozen sequences such as dates or titles,
- a shallow parser marking major phrase boundaries with limited embedding.

The next step will be to annotate our corpus for functional relations and valencies. A mid term perspective is to develop search tools in collaboration with the Loria team in Nancy. A longer term perspective could be to mark some anaphoric relations (for pronouns) or some word senses for verbs (for which the valence has been marked).

4. References

Abeillé, A., 1991. *Une grammaire lexicalisée d'arbres adjoints pour le français: application à l'analyse automatique*. Ph.D. thesis, University Paris 7.

Abeillé, A. and L. Clément, 1997. *Désambiguation morphosyntaxique; 1 Les mots simples; 2 Les mots composés*. TALANA, University Paris 7.

Abeillé, A. and L. Clément, 1999. A tagged reference corpus for french. In *Proceedings LINC-EACL'99*. Bergen.

Abeillé, A., L. Clément, and R. Reyes, 1998. Talana annotated corpus: the first results. In *Proceedings First Conference on Linguistic Resources*. Granada.

Abeillé, Anne, Marie-Hélène Candito, and Alexandra Kinyon, 1999. Ftag : current status and parsing scheme. In *VEXTAL'99*. Venise.

Abney, Steven, 1990. *Principle-based parsing*, chapter Parsing by chunks. Kluwer.

Brants, T., S. Skut, and H. Uszkoreit, 1999. Syntactic annotation of a germen newpaper corpus. In *Treebank Workshop*. Paris: ATALA.

Candito, Marie-Hélène, 1996. A principle-based hierarchical representation of ltags. In *Proceedings 19th COLING*. Kopenhagen.

Candito, Marie-Hélène, 1999. *Représentation hiérarchique de grammaires lexicalisées: application au français et à l'italien*. Ph.D. thesis, University Paris 7.

Clement, Lionel and Alexandra Kinyon, 2000. Chunking, marking and searching a morpho-syntactically annotated corpus for french. In *ACIDCA*. Monastir, Tunisia.

E., Giguet, 1998. *Méthodes pour l'analyse automatique de structures formelles sur documents multilingues*. Ph.D. thesis, Université de Caen.

Hajicova, E., J. Panevova, and P. Sgall, 1998. Language ressources need annotations to make them reusable: the prague dependency treebank. In *Proceedings First Conference on Linguistic Resources*. Granada.

Kayne, Richard S., 1975. *French syntax: the transformational cycle*. Cambridge, MA: MIT Press.

Kinyon, Alexandra, 2000. Shallow parsing french using function words. In *Colling'00*.

Marcus, M., M.-A. Marcinkiewicz, and B. Santorini, 1993. Building a large annotated corpus of english : the penn treebank. *Computational Linguistics*, 19(2):313–330.

Namer, F. and N. Hathout, 1998. Automatic construction and validation of french large lexical resources: reuse of verb theoretical descriptions. In *Proceedings First Conference on Linguistic Resources*. Granada.

Reyes, R., 1997. Un etiqueteur du français inspiré du taggeur de brill. Rapport de stage - TALaNa, Paris 7.

Senellart, J., 1999. *Localisation d'expressions linguistiques complexes dans de gros corpus*. Ph.D. thesis, University Paris 7.

Silberztein, M., 1993. *Dictionnaires électroniques et analyse automatique de textes: le système INTEX*. Paris: Masson.

Tzoukermann, E., D. Radev, and W. Gale, 1995. Tagging french without lexical probabilities – combining linguistic knowledge and statistical learning. In *Proceedings EACL SIGDAT Workshop*. Dublin.