Evaluating and improving syntactic lexica by plugging them within a parser
Elsa Tolone, Benoît Sagot, Éric Villemonte de la Clergerie

To cite this version:
Elsa Tolone, Benoît Sagot, Éric Villemonte de la Clergerie. Evaluating and improving syntactic lexica by plugging them within a parser. LREC 2012 - 8th International Conference on Language Resources and Evaluation, May 2012, Istambul, Turkey. electronic version (8 pp.). hal-00786883

HAL Id: hal-00786883
https://hal.archives-ouvertes.fr/hal-00786883
Submitted on 11 Feb 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Evaluating and improving syntactic lexica by plugging them within a parser

Elsa Tolone\(^1\), Benoît Sagot\(^2\), Éric Villemonte de La Clergerie\(^2\)

1. FaMAF, Universidad Nacional de Córdoba, Medina Allende s/n, Ciudad Universitaria, Córdoba, Argentina
2. Alpage, INRIA Paris-Rocquencourt & Université Paris 7, 175 rue du Chevaleret, 75013 Paris, France
elsa.tolone@univ-paris-est.fr, {benoit.sagot, eric.de_la_clergerie}@inria.fr

Abstract

We present some evaluation results for four French syntactic lexica, obtained through their conversion to the Alexina format used by the Leff\(^\) lexicon (Sagot, 2010), and their integration within the large-coverage TAG-based FRMG parser (de La Clergerie, 2005). The evaluations are run on two test corpora, annotated with two distinct annotation formats, namely EASY/Passage chunks and relations and CoNLL dependencies. The information provided by the evaluation results provide valuable feedback about the four lexica. Moreover, when coupled with error mining techniques, they allow us to identify how these lexica might be improved.

Keywords: syntactic lexica, parsing, error mining

1. Introduction

The development of a large-scale symbolic parsing system is generally achieved by co-developing its various components in a consistent way, in order to ensure good integration and consistency. This is the case, for example, concerning the development of the French syntactic lexicon Leff\(^\) (Sagot, 2010) together with that of the FRMG grammar and parser for French (de La Clergerie, 2005).

However, several syntactic lexica have been developed for French, sometimes for decades. An example thereof are Lexicon-Grammar tables (Gross, 1975; Boons et al., 1976b), although they were built with no or poor integration in large-scale NLP systems. Such a resource contains rich and valuable information, but its usefulness for real NLP applications has never been thoroughly assessed. This is also the case for the valency lexicon DICOVALENCE (van den Eynde and Mertens, 2006), although to a lesser extent. Therefore, after converting these resources in the same lexical model as the Leff\(^\), we performed a preliminary integration of these resources within FRMG. We also evaluated NewLeff\(^\), a new experimental version of the Leff\(^\) that benefits, among other things, from its merging with DICOVALENCE (Sagot and Danlos, 2012).

Following previous preliminary results (Tolone et al., 2011), we were able to evaluate these lexica on two reference corpora based on two different annotation schemes, namely the EasyDev corpus (Paroubek et al., 2009) and on the CONLL dependency version of the French TreeBank (Candito et al., 2010). As a side effect, these experiments also show that it is possible to switch lexica in a lexicalized parser like FRMG, at the cost of a relatively small decrease in performances.

2. Lexica

2.1. The Leff\(^\)

Our reference and baseline lexicon is Leff\(^\) (Sagot, 2010), a large coverage morphosyntactic and syntactic lexicon for French.\(^1\) As mentioned before, Leff\(^\) was specifically developed for NLP tasks, and in particular to be used in conjunction with the FRMG parser.

The Leff\(^\) relies on the Alexina framework for the acquisition and modeling of morphological and syntactic lexica. To represent lexical information, an Alexina lexicon relies on a two-level architecture:

- the intensional lexicon associates (among others) an inflection table and a canonical sub-categorization frame with each entry and lists all possible redistributions from this frame;
- the compilation of the intensional lexicon into an extensional lexicon builds different entries for each inflected form of the lemma and every possible redistribution.

The current version of the Leff\(^\) (version 3.1)\(^2\) contains only one entry for the lemma vérifier ‘verify’, ‘validate’. Here is a simplified version of this entry:

\[\text{vérifier}_1\]

|Lemma: v|\langle Suj: cln, sn\rangle|
|Obj: \langle cla|qcompl|scompl|sinf\rangle|s|
|@CtrlSujObj, @CompSubj &v:|\%
|\%ppp_employé comme adj, \%actif, \%passif, \%
|\%se_moyen_ impersonnel, \%passif_ impersonnel|

It describes a transitive verb entry whose arguments have the syntactic functions Suj and Obj listed between angle brackets.\(^3\) The subject might be realized as a nominative clitic (cln) or a noun phrase (sn), whereas the direct object can be realized as an accusative clitic (cla), a noun phrase (sn), an infinitive (sinf), with a control phenomenon expressed by @CtrlSujObj, a finite clause (scompl, in the subjunctive mood because of @CompSubj) or an indirect

---

\(^1\)Freely available at http://gforge.inria.fr/projects/alexina/

\(^2\)The Leff\(^\) 3.1 package includes v new files that are not considered as being part of the Leff\(^\) yet. In fact, replacing v\-phd files by v new files leads to what is called NewLeff\(^\) in this paper. See below for more details.

\(^3\)The different syntactic functions used in the Leff\(^\) are: Suj (subject), Obj (direct object), Obji (indirect object canonically introduced by preposition “à”), Objd (indirect object canonically introduced by preposition “de”), Loc (locative), Dloc (delocative), Att (attribute), Obl or Obl2 (other oblique arguments).
interrogative clause (qcompl). Finally, this verb entry allows for the functional redistributions past participle used as an adjective, active (the default distribution), impersonal middle-voice “se” construction, impersonal passive, and passive.

The Lefff 3.1 contains 7,108 verbal entries corresponding to 6,827 distinct lemmas, and 112,118 entries covering all other categories. Detailed figures are given in Table 1.4

### 2.2. Other syntactic lexica

Besides Lefff, we have considered three other lexica, whose verbal entries are combined with the non-verbal Lefff entries:

- **LGLex**: this lexicon results from a two-step conversion of the Lexicon-Grammar tables (Gross, 1975; Boons et al., 1976b), a rich syntactic lexical resource developed over several decades although not with an NLP orientation. A first conversion was made to get a fully electronic version of this lexicon into the LGLex format (Constant and Tolone, 2010), opening the way to a second conversion to the Alexina format (Tolone and Sagot, 2011). The result is a wide-coverage lexicon with, often, many entries for each verbal lemma, associated to several meanings and valency frames. LGLex contains 13,867 verbal entries corresponding to 5,738 distinct lemmas, as well 12,696 entries for predicative nouns corresponding to 8,531 distinct nominal lemmas;

- **DICOVALENCE**: this lexicon (van den Eynde and Mertens, 2006) follows the Pronominal Approach (Blanche-Benveniste et al., 1984) for characterizing verb valency frames and for defining fine-grained entries (several entries per lemma). This medium-coverage resource contains 8,313 verbal entries corresponding to 3,738 distinct lemmas;

- **NewLefff**: this experimental new version of Lefff targets more semantically-oriented finer-grained entries, while still preserving Lefff’s wide coverage. This lexicon is the result of two extension steps: (1) the automatic extraction, interpretation, conversion and integration or merging of denominal and deadjectival verbal entries in -iser and -isser from the LVF lexicon (Sagot and Fort, 2009); (2) the automatic merging of Lefff entries and DICOVALENCE entries through the comparison of their valency frames (Sagot and Daniel, 2012), then completed by a phase of manual validation on the 100 most frequent lemmas and on all dubious lemmas (those lemmas who got more entries than originally in both input lexica). NewLefff contains 12,613 verbal entries corresponding to 7,933 distinct lemmas.

Table 2 provides some figures about the coverage and granularity of each lexicon for verbal entries, showing that our four lexica actually cover a diverse spectrum of configurations (medium to large coverage, small to large granularity). For the experiments, all lexica use the non-verbal entries of Lefff in addition of their own verbal entries. In consequence, the differences between the lexica arise from the verbal entries.

<table>
<thead>
<tr>
<th>Lexica</th>
<th>#Entries</th>
<th>#Lemmas</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lefff</td>
<td>7,108</td>
<td>6,827</td>
<td>1.04</td>
</tr>
<tr>
<td>LGLex</td>
<td>13,867</td>
<td>5,738</td>
<td>2.41</td>
</tr>
<tr>
<td>DICOVALENCE</td>
<td>8,313</td>
<td>3,738</td>
<td>2.22</td>
</tr>
<tr>
<td>NewLefff</td>
<td>12,613</td>
<td>7,933</td>
<td>1.58</td>
</tr>
</tbody>
</table>

Table 2: All lexica at a glance (verbal entries)

### 3. FRMG

FRMG (de La Clergerie, 2005) is a large-coverage symbolic grammar and parser for French. In fact, the acronym FRMG denotes resources that cover several representation levels. The most abstract level corresponds to a linguistically motivated modular and hierarchical meta-grammar. This meta-grammar is used to generate a compact (feature-based) Tree Adjoining Grammar (TAG) (Joshi et al., 1975) containing around 300 factorized elementary trees, including only 35 trees anchored by verbs. Despite its compactness, the grammar exhibits a wide coverage, thanks to factorization operators, such as disjunction and guards, used in the trees to allow many possible tree traversals. The grammar is compiled into an efficient chart-based parser, also named FRMG, which is able to return both full parses (whenever possible) or sequences of partial parses (otherwise) as shared dependency forests. The forests may then be disambiguated using heuristic-based rules to get the best dependency trees. Finally, these trees may be converted to various output formats, including the EASy format and the CONLL format.

FRMG benefits from the extended domain of locality provided by TAG trees, with, for instance, the possibility to capture all the components of a verb valency frame through the nodes of a single elementary tree. However, it also implies that a TAG grammar like FRMG works best when coupled with a lexicon that provides such rich lexical information. There is also the need to propagate this information from the words to the trees. Concretely, each tree of FRMG is associated with an hypertag (Kinyon, 2000), a feature-structure resuming (in the case of a verbal tree) the various frames and argument realizations covered by the tree. Similarly, each verbal entry (but this is also true for other categories) has an hypertag derived from its lexical information. Anchoring a verbal tree by a verbal entry involves the unification of both hypertags. Figure 1 shows the

---

4The category “other” includes all kinds of conjunctions, determiners, interjections, punctuation marks, pronouns, prefixes and suffixes, as well as special entries for named entities and unknown words.

5Freely available at http://infolingu.univ-mlv.fr/english/ > Language Resources > Lexicon-Grammar > Download

6Freely available at http://bach.arts.kuleuven.be/dicovalence/

7Freely available at http://gforge.inria.fr/projects/alexina/. For obtaining NewLefff, v_new files in the Lefff 3.1 package must be compiled and used instead of v and v_phd files.

8505 verbal lemmas, corresponding to 986 entries.

9Freely available at http://imgkit.gforge.inria.fr/
### Table 1: Quantitative data about the Lefff

<table>
<thead>
<tr>
<th>Category</th>
<th>#Intensional Entries</th>
<th>#Unique Lemmas</th>
<th>#Extensional Entries</th>
</tr>
</thead>
<tbody>
<tr>
<td>verbs</td>
<td>7,108</td>
<td>6,827</td>
<td>363,120</td>
</tr>
<tr>
<td>verbal idioms</td>
<td>1,869</td>
<td>1,851</td>
<td>3,296</td>
</tr>
<tr>
<td>nouns</td>
<td>41,816</td>
<td>41,592</td>
<td>86,675</td>
</tr>
<tr>
<td>adjectives</td>
<td>10,556</td>
<td>10,517</td>
<td>34,359</td>
</tr>
<tr>
<td>adverbs</td>
<td>4,111</td>
<td>3,676</td>
<td>4,155</td>
</tr>
<tr>
<td>prepositions</td>
<td>260</td>
<td>259</td>
<td>728</td>
</tr>
<tr>
<td>proper nouns</td>
<td>52,499</td>
<td>52,202</td>
<td>52,571</td>
</tr>
<tr>
<td>other</td>
<td>1,007</td>
<td>854</td>
<td>1,589</td>
</tr>
<tr>
<td><strong>total</strong></td>
<td><strong>119,226</strong></td>
<td><strong>117,778</strong></td>
<td><strong>546,493</strong></td>
</tr>
</tbody>
</table>

**Figure 1: Hypertag for promettre ’promise’**

- **arg0**
  - fun: subj
  - pcas: -
- **arg1**
  - fun: obj
  - pcas: -
  - kind obj | scomp: |
- **arg2**
  - fun: objaj
  - pcas: a `|`
  - kind preobj: |
  - refl: -
  - ctrsubj: subj
  - imp: -

**Figure 1: Hypertag for promettre ’promise’**

### 4. EASY Evaluation

Our first evaluation was conducted on the EasyDev corpus, a small corpus of around 4000 sentences used during the first EASY French parsing evaluation campaign and covering various document styles (journalistic, literacy, medical, mail, speech, etc.). The corpus is annotated with the EASY format (Paroubek et al., 2006; Paroubek et al., 2009), a mix of 6 kinds of chunks and 14 kinds of dependencies between forms or chunks, as illustrated by Figure 2 (with ovals for chunks and diamonds for dependencies). Table 3 shows the performances of the various lexica, on this EasyDev corpus. The coverage column indicates the rate of full parses (keeping in mind that the almost all remaining sentences get partial parses), and shows a clear decrease for DICOVALENCE and smaller ones for LGLex and NewLefff.

We retrieve similar results in terms of F-measure on the chunks and dependencies. Finally, the fact that LGLex is both a wide-coverage and very fine-grained lexicon has a clear impact on parsing time. Figure 3 shows the F-measure for some of the EASY verbal dependencies, namely SUJ-V for the subject-verb relation, AUX-V for the auxiliary-verb relation, COD-V for the object-verb relation, CPL-V for the complement-verb relation (with no distinction between argument and adjuncts), and ATB-SO for the subject or object attributes. We have also added MOD-N for noun-modifiers, this relation being the most numerous one and being partly related to verbs through past and present participles on nouns.

Again, we observe a slight decrease for all new lexica versus Lefff, more marked on some relations like COD-V for LGLex and ATB-SO for LGLex and DICOVALENCE.

**Table 3: Overall performances on EasyDev**

<table>
<thead>
<tr>
<th>Lexicon</th>
<th>Cover. (%)</th>
<th>Chunks (%)</th>
<th>Rel. (%)</th>
<th>Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lefff</td>
<td>83.45</td>
<td>89.03</td>
<td>66.76</td>
<td>0.35</td>
</tr>
<tr>
<td>NewLefff</td>
<td>82.19</td>
<td>88.74</td>
<td>66.09</td>
<td>0.55</td>
</tr>
<tr>
<td>LGLex</td>
<td>80.61</td>
<td>87.89</td>
<td>63.19</td>
<td>1.10</td>
</tr>
<tr>
<td>DICOVALENCE</td>
<td>71.44</td>
<td>88.08</td>
<td>64.49</td>
<td>0.38</td>
</tr>
<tr>
<td>Old DICOVALENCE</td>
<td>65.69</td>
<td>87.06</td>
<td>62.72</td>
<td>0.42</td>
</tr>
</tbody>
</table>

### 5. CoNLL Evaluation

For our second evaluation, we used the version of the French TreeBank (journalistic style) (Abeillé et al., 2003) converted by Candito et al. (2010) into the CoNLL dependency format, a format now largely used in international parsing evaluation campaigns (Nivre et al., 2007). This version of the French TreeBank has already been used to train and compare several statistical parsers (Candito et al., 2010), thus providing us baselines to evaluate FRMG and the various lexica. Note however that our results are still preliminary.

The CoNLL format relies on a fine-grained set of verbal dependencies, with in particular for verbal dependencies:

- the distinction between several kinds of auxiliaries: aux-tps (temporal auxiliaries), aux-pass (passive constructions), aux-caus (causative constructions);
• suj and obj for the subjects and objects (but noting that the obj relation is also used in non-verbal cases);
• relations for the prepositional objects, with a_obj for those introduced by a (to), de_obj for those introduced by de (of), and p_obj for the remaining ones;
• relations for the attributes, with a_to for the attributes of objects and a_to for those of subjects;
• aff for affixes, actually verb clitics not covered by the above-mentioned relations;
• mod for verb modifiers such as adverbs (noting that again this relation is also used for non verbal cases).

Table 4 shows that all lexica got very good full parse coverage (on journalistic style) and emphasizes again the speed problems for LGLex (on relatively long and complex sentences, with a mean length average of 27 words vs 19.3 for EasyDev). Again, we note a slight decrease for the alternative lexica in terms of Labeled Attachment Score (LAS). We also note that all FRMG versions are still a few points below state-of-the-art statistical parsers, for instance MST (Candido et al., 2010). A finer analysis in terms of recall and precision at the dependency level shows a contrasted landscape (Figure 4), with LGLex or more specifically NewLeff being sometimes better either in recall (aff, a_obj) or precision (a_to, aux_caus). More generally, recall is relatively good but we observe precision problems. We conjecture that the finer granularity of LGLex and also of NewLeff tends to wrongly select rare valency frames for some medium to high frequency verbs, frames that are strongly favored by the heuristic-based FRMG disambiguation algorithm, leading to confusion between verb arguments (obj, a_obj, de_obj, p_obj) and modifiers (mod, dep).

Table 4: Overall evaluation on French Tree Bank (test part)
6. Error Mining

The evaluation already provides interesting feedback to identify the strong and weak points of a lexicon (as illustrated by DICOVALENCE with the ATB-S0 relation). However, to get information at the level of a verb or of an entry, we rely on error mining techniques (Sagot and Villemonte de La Clergerie, 2006). More precisely, the basic idea is to identify suspect lexical entries by mining the full parse failures on a large corpus, based on the following intuition:

A form is suspect if it occurs more often than expected in non-full-parsable sentences, in co-occurrence with non-suspect forms.

The mathematical formulation of this intuition leads to a fix-point iterative algorithm, close to EM (Expectation-Maximization), which may be used to also return, for each fix-point iterative algorithm, close to EM (Expectation-Maximization), which may be achieved through a slight rephrasing, as follows:

A verb is suspect for lexicon $L$ if it occurs more often than expected in sentences not full-parsable when using $L$ but that received a full parse when using LGLex, in co-occurrence with non-suspect verbs.

The modified algorithm was then used on a larger corpus of 100K sentences (1.6M words), named CPJ (Corpus Passage Jouet), and comprising various style of documents (encyclopedic with Wikipedia, literacy with Wikisource, news with AFP, and discourse with Europarl). We have started exploiting the results for (former versions of) LGLex and, to a lesser extent, NewLefff, and have identified several kinds of errors for some entries. The interest of this approach is that it can be applied to very large corpora, and we plan to do it, to overcome lexical data sparseness. Here, we provide some analysis of the data provided by the algorithm, with an emphasis on LGLex.

6.1. LGLex

We analyzed the first 15 suspicious verbs in LGLex in order to determine where the errors come from. We indicate the number of failed sentences for each verb between parentheses and we give one example. In total, there are 212 failed sentences for this selection of verbs:

- Some entries do not appear in the tables:
  - mixer ‘mix’ (7) in Mixé par Jimi Hazel, assisté de Bruce Calder, enregistré chez Jimi à l’“Electric Lady Studios” à New York: this entry is encoded in table 36S but with the meaning ’blend’ (Max mixe les carottes (et+avec) les navets dans un mixeur). We added this entry to table 32PL (Max a mixé les sons), which has the defining feature $<$N0 $\lor$ N1$>$, with an encoding similar to the entry mélanger ’blend’ (Max a mélangé les (étiquettes+cartes+couleurs));
  - zapper ‘omit’ (4) in Elle a également ”déploré” la mémoire de ”plus en plus sélective” de la jeune femme, ”qui zappe les détails qui font désordre”: this entry appear in table 35L but with the meaning ’channel hop’ (Max zappe de la 1ère et la 2ème chaîne). We added this entry to table 32R2 (Max a zappé un (repas+paragraphe)), which has the defining feature $<$N0 $\lor$ N1$>$, with an encoding similar to the entry sauter ’skip’, ‘miss’ (Max a sauté (un repas+une étape+une ligne));
  - réaffirmer ’reaffirm’ (28) in Nous réaffirmons la nécessité de consulter les sans-abri et leurs or-
organisations sur les programmes européens pertinents. We can add the feature <re-V> for the entry affirmer ‘affirm’ of table 9 (Max a affirmé à Luc qu’il viendrait)10, with the defining feature <N0 V N1 à N1>, which accepts the construction <N0 V N1>, in order to recognize the entry réaffirmer;
− réélire ‘reelect’ (10) in Helmut Kohl est réélu au poste de Chancelier fédéral. We can add the feature <re-V> for the entry élire ‘elect’ of table 39 (On a éludé Macron [E+comme] député), with the defining feature <N0 V N1 N2>, in order to recognize the entry réélire;
− the pronominal form se réimplanter ‘re-establish itself’ (5) in Celles-ci cherchent toujours à se réimplanter dans la zone”, a relevé M. Besson. The pronominal form s’implanter ‘establish itself’ does not appear either in the tables. We can add the feature <re-V> for the entry implanter ‘establish’ in table 38LD (On a implanté une usine dans cette région), with the defining feature <N0 V N1 Loc N2 destination>, in order to recognize the entry réimplanter ‘re-establish’. Then, we can add the feature <se V> (or more precisely, <N1 se V W>) to accept the pronominal forms s’implanter and se réimplanter (Une usine s’implante dans la zone)11.
• Some entries appear in the tables but are not encoded (~) or have been corrected:
− susciter ‘spark off’ (41) in A d’autres niveaux, les propositions sur la table suscitent de sérieuses objections and recruter ‘recruit’ (14) in 80 intérimaires ont déjà été recrutés, pour assurer les commandes: these two entries appear in table 38R (Ceci a suscité un émeute république+réaction) chez Luc and Max a recruté Luc (comme+sur un poste de) lecteur), which has the defining feature <N0 V N1 Loc N2 destination>, but are not encoded. This implies that no other construction are accepted, whereas the construction <N0 V N1> appear in the table and allows the erasure of the second complement. We encoded this construction as +;
− réprouver ‘reprove’, ‘reprobate’ (11) in Dieu ne réprouve donc personne: this entry is encoded in table 12 (Max réprouve qu’Ida boive), which has the defining feature <N0 V N1 de N2>. When we modified the defining features (Tolone, 2011), we replaced it by <N0 V N1>.
• Other entries are encoded in the tables but with obligatory complements which do not appear in the sentences of the corpus:
− délocaliser ‘relocate’ (9) in Ils ont fait le choix de délocaliser en Tunisie: this entry appear in table 38L (On a délocalisé ce service de Paris à Dax), which has the defining feature <N0 V N1 Loc N2 source Loc N3 destination>, but the entry is not encoded. We encoded the construction <N0 V N1 Loc N3 destination> as +, but no other construction allow the erasure of the first complement;
− kidnapper ‘kidnap’ (12) and revendre ‘retail’ (5) in sentences without second complement, such as Les deux Italiens ont été kidnappés le 18 décembre and Charles mangeait l’avoine des chevaux, doublant les fournitures, revendant par une porte de derrière ce qui entrait par la grande porte: these two entries are encoded in table 36DT (On a kidnappé son fils à Max and Max a revendu à Luc la télé gagnée au jeu), which has the defining feature <N0 V N1 Prep N2>, without allowing the erasure of the second complement introduced by the preposition à;
− écrouer ‘put behind bars’ (5) in Le lycéen de 18 ans soupçonné d’avoir poignardé vendredi un camarade, Hakim, dans leur lycée du Kremlin-Bicêtre (Val-de-Marne), a été mis en examen and écroué hier, alors que lycées et collèges sont invités à observer une minute de silence aujourd’hui à la mémoire de la victime: this entry is encoded in table 38LHD (On a écroué Luc dans un pénitentier), which has the defining feature <N0 V N1 Loc N2 destination>, without allowing the erasure of the locative complement;
− camper ‘camp’ (5) in Les troupes campent entre Harlem et Nimègue: this entry is encoded in table 38LHR (Le roi campe ses troupes dans la plaine), which has the defining feature <N0 V N1 Loc N2>, and it accepts the construction <N1 V W>. This means that an object N1 can become the subject of a sentence with conservation of its other objects. Therefore, this corresponds to the construction <N1 V Loc N2> (Ses troupes campent dans la plaine), without allowing the erasure of the locative complement.
• Last, some specific cases:
− rediriger ‘redirect’ (50) in wrong sentences, such as deux cent cinquante-troisredirige ici;
The previous examples show (a) that some entries appear in the tables but are not encoded and therefore we have to encode them, and (b) that some other entries are missing, with several cases to be distinguished:

1. the entry should be added as a new entry. It can be a new verb or a different meaning of an existing verb (cf. mixer and zipper);

2. the entry re-V (or ré-V) has a meaning which can be derived by a "simple" use of the verb V (cf. réaffirmer and réélire): we should add a column <re-V> to all tables and encode it for all entries. Indeed, those without a derivable meaning have been added as another entry (entries re-V which they do not mean faire une deuxième fois 'do twice'), such as revendre 'retail' that does not mean vendre une deuxième fois 'sell twice' or rembourser 'come down' (La balle retombe) which is not tomber une deuxième fois 'fall twice';

3. the entry se V has a meaning which can be derived by a transitive use of the verb V (cf. s’implanter); we have at least 5 different cases (Boons et al., 1976a) (p.120-163), so we should add 5 columns <se V> to all tables and encode it for all entries: for instance, se regarder 'look at oneself' (Paul se regarde dans la glace), se mentir 'lie to one another' (Paul et Marie se mentent), s’étonner 'be surprised' (Paul s’étonne de mon silence), se laver 'wash' (Paul se lave les pieds), se manger 'eat' 'be served' (Le roti se mange froid). Indeed, only the intrinsic pronominals, which are not linked by a transitive use, have been added as a new entry: for instance, s’évanouir 'faint' (Paul s’est évanouï);

4. the entry se V and re-V is a combination of the two previous cases (cf. se réimplanter);

5. the entry dé-V has a meaning which can be derived by a verb V (no example in this selection of verbs): we should add a column <dé-V> to all tables and encode it for all entries. Indeed, only the uses different from faire l’action inverse 'do the opposite action' have been added like another entry. For instance, dévisser has an entry for the meaning ‘fall’ (L’alpiniste a dévissé), but not for the meaning ‘unscrew’ (dévisser une vis), which is the opposite action of ‘screw on’ (visser une vis).

As we show, some features are also missing, including some that encode the erasure or the clitic pronominalization of certain complements. Indeed, we can allow as erasures the complements that are in the defining features, and we should add these features to the appropriate tables and encode them for all entries in all tables. For instance, the entry donner 'give' in table 36DT can accept the erasure of all complements if the context allows it:

- <N0 V N1 Prep N2>: Paul donne du fric aux associations
- <N0 V N1>: Paul donne du fric
- <N0 V Prep N2>: Paul donne aux associations
- <N0 V>: Paul donne souvent

In conclusion, error mining enables us to detect and correct many errors in LGLex but we should manually analyze all 613 suspicious verbs in all 2,623 concerned sentences and all corrections have to be done manually, which represents an important (but rewarding) effort.

6.2. NewLeff

Error mining results on parses produced by FRMG with NewLeff on the large 82.2% presented in this paper — and much closer to the indirect object), and others.

7. Conclusion

We have presented some preliminary but promising evaluation results for several lexica, obtained through their integration within a lexicalized deep TAG parser. Clearly, even if good, the results show that some efforts of adaptation remain to be done to improve the integration and to better exploit the richness of these lexica. Error mining techniques should help us to achieve this objective, and should also help us to identify the strong and weak points of each lexicon, which should lead to a new generation of better quality lexica, freely available and ready to use in large scale NLP systems.

We would like also to mention very recent results showing that partially supervised learning techniques may be used to boost the performance of FRMG disambiguation to reach a LAS of 85.1% when using Leff, to be compared with the 82.2% presented in this paper — and much closer to MST, a stochastic parser specifically trained on the French Treebank. It remains to be tested whether this improved disambiguator leads as such to similar gains when using the other lexica, or whether the learning phase has to be done for each of them.

In NewLeff, estimer has three entries, that corresponds to the meanings ‘consider’, ‘estimate’ and ‘esteem’.
8. References


