Frequencies of occurrence of entries and subcategorization frames in *LGLex* lexicon with IRASUBCAT

Elsa Tolone\(^1\) & Romina Altamirano\(^1\)

1. FaMAF, Universidad Nacional de Córdoba (Argentina)

LGC2013
Universidade do Algarve, Faculdade de Ciências Humanas e Sociais, Campus de Gambelas (Portugal)
September 10-14, 2013
Using IRASubcat with the converted lexicon and the relevant information extracted of the processed corpus we can complete the lexicon with the frequencies of occurrence for each verb and each syntactic function.
1. Lexicon-Grammar tables for French

2. IRASUBCAT

3. Experiment with IRASUBCAT and the LGLex lexicon of French

4. Results
1. Lexicon-Grammar tables for French

2. IRASUBCAT

3. Experiment with IRASUBCAT and the *LGLex* lexicon of French

4. Results

---

**1. Lexicon-Grammar tables for French**

Elsa Tolone & Romina Altamirano

Frequencies of occurrence of entries and subcategorization frames
Example: Table V_33

<table>
<thead>
<tr>
<th>(N_0 = \text{Nnum} )</th>
<th>(N_0 = \text{N-hum} )</th>
<th>(N_0 = \text{Nnr} )</th>
<th>(Ppv = \text{se figé} )</th>
<th>(Ppv = \text{en figé} )</th>
<th>(Ppv = \text{les figé} )</th>
<th>(Nég )</th>
<th>(\langle \text{ENT} \rangle )</th>
<th>(N_0 \text{ V} )</th>
<th>(N_0 \text{ être V-ant} )</th>
<th>(N_1 = \text{Nnum} )</th>
<th>(N_1 = \text{N-hum} )</th>
<th>(N_1 = \text{le fait QuP} )</th>
<th>(Ppv = \text{y} )</th>
<th>(Ppv = \text{y} )</th>
<th>(\text{Lnum V W sur ce point extr} )</th>
<th>(\langle \text{OPT} \rangle )</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>Max renait au bonheur de vivre</td>
</tr>
<tr>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>Max s'est rendu à mon opinion</td>
</tr>
<tr>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>Le caporal s'est rendu à l'ennemi</td>
</tr>
<tr>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>Max renonce à son héritage</td>
</tr>
</tbody>
</table>

Defining feature in table of classes: \(N_0 \text{ V à } N_1\)

[Gross 1975 ; 1994 ; LADL since 1970s ; LIGM since late 1990s]
The improvement of the tables enables the extraction of a syntactic lexicon for each categories from Lexicon-Grammar tables [Constant & Tolone 2010]:

- named LGLex lexicon
- generated from the original Excel or CSV tables by the LGExtract tool
- exchange format with the same linguistic concepts of the tables
- text or XML format
1. Lexicon-Grammar tables for French
2. IRASUBCAT
3. Experiment with IRASUBCAT and the *LGLex* lexicon of French
4. Results

The conversion towards the Alexina format enables the integration of them in a real-life **symbolic parser** [Tolone & Sagot 2011; Tolone *et al.* 2012]

- NLP tools used:
  - parser: FRMG [Thomasset & de La Clergerie 2005]
  - lexical formalism: Alexina, formalism used by the *Lefff* lexicon [Sagot 2010] used by FRMG

- named *LGLex-Lefff* lexicon → this allows a comparison between FRMG_{Lefff} and FRMG_{LGLex}
Conversion of Lexicon-Grammar tables

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

[Tolone 2012]
2. IRASUBCAT

Frequencies of occurrence of entries and subcategorization frames in the LGLex lexicon with IRASUBCAT.
IRASUBCAT

- a tool that acquires subcategorization information about the behaviour of any tag class (e.g., part of speech, syntactic function, etc.) or combination of them, from corpora
- takes as input a corpus in XML format
- the output is a lexicon, also in XML format, where each of the verbs under inspection is associated to a set of subcategorization patterns. The lexicon also provides information about frequencies of occurrence for verbs, patterns, and their co-occurrences in corpus
- allows to integrate the output lexicon with a preexisting one, merging information about verbs and patterns with information that had been previously extracted, possibly from a different corpus or even from a hand-built lexicon

[Altamirano & Alonso Alemany 2010]
Adding frequencies with IRASUBCAT

Lexicon-Grammar tables
- Electronic inventory of lexical & syntactic information

Conversion with LGExtract

LGLex
- NLP oriented syntactic lexicon

Conversion and integration in FRMG parser

LGLex-Lefff used with FRMG-LGLex
- Syntactic lexicon integrated in a parser

IRASUBCAT

LGLex with frequencies
- Syntactic lexicon in IRASUBCAT format
3. Experiment with IRASUBCAT and the $LGLex$ lexicon of French
The experiment

We want to use the results of FRMG parser on a big corpus with IRASubcat in order to improve the *LGLex* lexicon of French, adding the frequencies of occurrence for each entry and each subcategorization frame. To do this, we must:

- choose a corpus with millions of words, also we just only need a small part of this corpus for the experiment
- parse the corpus with the FRMG parser, with and without the *LGLex* lexicon (i.e. only with the *Lefff* lexicon) – results with FRMG_{LGLex} and with FRMG_{Lefff}
- convert both the processed corpus and the *LGLex* lexicon into XML format, required by IRASubcat;
- use IRASubcat in order to add the frequencies of occurrence extracted from the big corpus into the *LGLex* lexicon
The corpus

The processed corpus with FRMG_{LGLex} to see how we use the FRMG parser with the LGLex lexicon) used for the experiment is the CPJ (Corpus Passage Jouet) with 100K sentences of AFP (Agence France-Presse), Europarl, Wikipedia and Wikisources, extracted from the corpus of the evaluation campaign (in 2009) for French parsers Passage [Hamon et al. 2008]
Conversion into XML format

We created 2 programs in Python:

- one to convert the verbal LGLex lexicon in the same format as IRASubcat output lexicon
- another to convert the processed corpus CPJ with the FRMG parser in a format directly readable by IRASubcat
The input is the verbal *LGLex* lexicon, or more precisely, the *extensional lexicon* of *LGLex*-Lefff lexicon, which contains each inflected form of the lemma and every possible redistribution.

In the output lexicon converted into XML format as IRASubcat output lexicon (named *lglex-lefff-IRASubcat.xml*), each lemma is associated to a set of subcategorization patterns. For example:

```xml
<pattern id="['Suj:cln|sn', 'Obj:sn']"> </pattern>
<pattern id="['Suj:(cln|sn)', 'Obl:de-sinf']"> </pattern>
```
Conversion of the verbal *LGLex* lexicon: An example

We have in total 14,068 distinct lemmas.

Here is a complete example of *lglex-lefff-IRASubcat.xml*:

```xml
<dictionary>
  <entry verb="achever_ _V_1_1" count_oc_verb="0">
    <tag name="fs" different_patterns="6">
      <pattern id="['obj', 'suj']" count_w_verb="0" total_count="0"
        rejected_patterns_freq_test="NO"> </pattern>
      <pattern id="['obl', 'suj']" count_w_verb="0" total_count="0"
        rejected_patterns_freq_test="NO"> </pattern>
      <pattern id="['obl2', 'suj']" count_w_verb="0" total_count="0"
        rejected_patterns_freq_test="NO"> </pattern>
      <pattern id="['obl', 'obl2']" count_w_verb="0" total_count="0"
        rejected_patterns_freq_test="NO"> </pattern>
    </tag>
  </entry>
</dictionary>
```
Conversion of the processed corpus with the FRMG parser (1)

The input is the processed corpus CPJ with the FRMG parser, more precisely, with FRMG_{\text{LGLex}}, i.e. the FRMG parser with the \text{LGLex-Lefff} lexicon. In the processed corpus CPJ, we represent in XMLDep format a graph of dependencies with nodes (lemmas), grouped in clusters (forms), with arcs describing the syntactic dependencies between nodes. So, we want to extract only the useful information in a format directly readable by IRASubcat.
Conversion of the processed corpus with the FRMG parser (2)

In the output in XML format (named CPJ-IRASubcat.xml), for each sentence of the corpus (for example, <sentence ID="12" corpus="frwikipedia_012" s="12" >), we extracted the verbs (cat="v") with their identifiers (for example, lemmaid="achever__V_1_1"). For each verb, we extracted the syntactic functions and we indicated the number of arguments (nb_fs="2") and then, each syntactic function (fs) one by one (for example, fs="suj" for subject, and fs="obl2" for oblique).
Conversion of the processed corpus with the FRMG parser: An example

Here is a complete example of CPJ-IRASubcat.xml:

```
<sentence ID="12" corpus="frwikipedia_012" s="12">
  <word lexica="achevée" lemma="achever" lemmaid="achever__V_1_1"
    cat="v" nb_fs="2">achevée</word>
  <word fs="suj"></word>
  <word fs="obl2"></word>
</sentence>
```
Using IRASubcat with *LGLex*

We changed the information in the configuration file to execute IRASubcat with our lexicon `lglex-lefff-IRASubcat.xml` and our corpus `CPJ-IRASubcat.xml` (in UTF-8):

- VERB LIST = NO
- EXISTING DICTIONARY = lglex-lefff-IRASubcat.xml
- LENGTH OF VERBAL CONTEXT = 3
- COMPLETE WITH EMPTY WORD = NO
- KEEP ORDER = NO
- TARGET TAGS = fs
- USE LEXICAL FORM OF WORDS = NO
- INTRODUCE VERBAL MARK = NO
- COLLAPSE PATTERNS = NO
- MAX ITERATION TO COLLAPSE PATTERNS = FALSE
- MIN FREQUENCY OF VERBS = 0
- MIN REL FREQUENCY OF PATTERNS = 0
- USE LIKEHOOD RATIO TEST = NO
4. Results
The execution create:

- the file `OutputDictionaryOrd.xml` with the lexicon
- the file `info_file` with the statistics of execution
- the file `IdsSentencesOrigenDictionary.xml` with the ID’s of sentences that give origin of the patterns in `OutputDictionaryOrd.xml`
Here is the previous example of `lglex-lefff-IRASubcat.xml` as it appears in `OutputDictionaryOrd.xml`:

```xml
<dictionary>
  <entry verb="achever__V_1_1" count_oc_verb="1">
    <tag name="fs" different_patterns="4">
      <pattern id="['obj', 'suj']" count_w_verb="0" total_count="1001"
        rejected_patterns_freq_test="NO"></pattern>
      <pattern id="['obl', 'suj']" count_w_verb="0" total_count="214"
        rejected_patterns_freq_test="NO"></pattern>
      <pattern id="['obl2', 'suj']" count_w_verb="1" total_count="325"
        rejected_patterns_freq_test="NO"></pattern>
      <pattern id="['obl', 'obl2']" count_w_verb="0" total_count="0"
        rejected_patterns_freq_test="NO"></pattern>
    </tag>
  </entry>
</dictionary>
```
The result lexicon (2)

- We can see that the number of occurrences of the verb `achever__V.1.1` in the corpus is 1 and the pattern is `[‘obl2’, ‘suj’]`. For this pattern, we have in total 325 occurrences in the corpus for all verbs.

- We can see in the example of `IdsSentencesOrigenDictionary.xml` (see below) that the occurrence of `verb=”achever__V.1.1”` with the pattern `[‘obl2’, ‘suj’]` is in the sentence `[‘12’]`
The result lexicon (3)

```xml
<ids_from>
  <entry verb="achever__V_1.1" total_count="1">
    <tag name="fs">
      <pattern id="['obj', 'suj']">
        <s_list>[]</s_list>
      </pattern>
      <pattern id="['obl', 'suj']">
        <s_list>[]</s_list>
      </pattern>
      <pattern id="['obl2', 'suj']">
        <s_list>['12']</s_list>
      </pattern>
      <pattern id="['obl', 'obl2']">
        <s_list>[]</s_list>
      </pattern>
    </tag>
  </entry>
</ids_from>
```

Elsa Tolone & Romina Altamirano

Frequencies of occurrence of entries and subcategorization frames
The frequencies: Number of occurrences of patterns

The frequencies indicated in *OutputDictionaryOrd.xml* allow us to know the total number of occurrences of each pattern in the corpus. We don’t indicate the patterns which never appear.

<table>
<thead>
<tr>
<th>pattern</th>
<th>total_count</th>
</tr>
</thead>
<tbody>
<tr>
<td>['obj', 'suj']</td>
<td>1001</td>
</tr>
<tr>
<td>['obl2', 'suj']</td>
<td>325</td>
</tr>
<tr>
<td>['obl', 'suj']</td>
<td>214</td>
</tr>
<tr>
<td>['att', 'suj']</td>
<td>142</td>
</tr>
<tr>
<td>['loc', 'suj']</td>
<td>92</td>
</tr>
<tr>
<td>['objà', 'suj']</td>
<td>91</td>
</tr>
<tr>
<td>['suj']</td>
<td>62</td>
</tr>
<tr>
<td>['objde', 'suj']</td>
<td>55</td>
</tr>
<tr>
<td>['obj']</td>
<td>26</td>
</tr>
<tr>
<td>['dloc', 'suj']</td>
<td>11</td>
</tr>
<tr>
<td>others</td>
<td>0</td>
</tr>
</tbody>
</table>
The frequencies indicated in *IdsSentencesOrigenDictionary.xml* allow us to calculate the number of verbs associated with each total number of occurrences of this verb. We indicate the verb when there is only one verb.

<table>
<thead>
<tr>
<th>verb or nb of verbs</th>
<th>total_count</th>
<th>nb of verbs</th>
<th>total_count</th>
</tr>
</thead>
<tbody>
<tr>
<td>être</td>
<td>63</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>pouvoir</td>
<td>60</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>devoir</td>
<td>37</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>faire</td>
<td>22</td>
<td>12</td>
<td>6</td>
</tr>
<tr>
<td>dire</td>
<td>19</td>
<td>14</td>
<td>5</td>
</tr>
<tr>
<td>vouloir</td>
<td>17</td>
<td>30</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>16</td>
<td>63</td>
<td>3</td>
</tr>
<tr>
<td>avoir</td>
<td>13</td>
<td>192</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
<td>740</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>13 043</td>
<td>0</td>
</tr>
</tbody>
</table>
Conclusions and perspectives

- The processed corpus is the results of the FRMG parser with LGLex lexicon, so it could find wrong sense
- The next step is to consider the information on realizations, that we must extract from processed corpus, but it is not a straightforward task
- Then we have to use the FRMG parser with Lefff lexicon only, without the LGLex lexicon influences the results
- We could also use IRASubcat with another parser which is statistical, such as MaltParser, MSTParser, or Berkeley Parser
  Candito et al. 2010
- And we could do a comparison using the original lexicon and the enlarged lexicon with that different parsers to verify that the accuracy is better using more information
References (1)


References (2)

References (3)


