

A task-based evaluation of the ECM database. Effect on parsing performance.

Begoña Villada Moirón
University of Groningen
m.b.villada.moiron@rug.nl

STEVIN IRME DELIVERABLE 5.1B

Contents

1	Introduction	1
1.1	Preliminaries	2
2	Automatic conversion	3
3	Parsing accuracy	4
3.1	Introduction	4
3.2	Methodology	4
3.3	Expectations	7
3.4	Results	8
3.5	Error analysis	9
4	Final remarks	12

1 Introduction

One project aim was to establish a theory-neutral lexical representation of MWES so that the resulting lexical database could be used in various NLP systems in an effective way. We set up a small scale task-based evaluation to find out to what extent the wide-coverage Alpino parser can benefit from using the ECM lexical database.

In the remainder, we first explain some needed notions. Further, we deal with the automatic conversion evaluation and next describe the methodology used to evaluate parsing performance. The results and error analysis follow. We finalize this report by summarizing the implications and our conclusions.

1.1 Preliminaries

The automatic conversion of ECM MWEs into the Alpino lexicon has been broadly documented in the report STEVIN-IRME WP2, deliverable 5.1a by Nicole Grégoire. There, a detailed description of the **lexical entries** in both the ECM database and in the Alpino lexicon is given.

In our evaluation, we only take into account verbal MWEs because other types of MWEs existing in the ECM are not treated in the Alpino parser, yet. Among the MWEs we tested there were NP VERB, NP PP VERB, PP VERB and NP NP VERB.

Here, it is important to emphasize that Alpino views a verbal MWE as a special use of the head verb. The MWE shares the tense paradigm with the full verb lexeme. The MWE description is actually a (complementation) frame that enumerates¹: (i) the type of argument the complements make up: *fixed* or *fixed with a particle verb*; (ii) the required lexemes or phrases; (iii) the type of syntactic argument that a phrase projects: accusative, dative, verb phrase, that-clause, etc. and (iv), passive type (none, normal passive or imperative passive).

All the information within an Alpino lexical entry is crucial to the parser given that the parser's decisions are informed by the lexicon; eventually, the syntactic analysis of a MWE (parse) specifies not only the grammatical functions assigned to each constituent in a sentence but also dependency structures that to some extent mirror the semantic representation of the sentence meaning.

Robust and Wide-coverage parsing The Alpino parser is a robust wide-coverage parser. Roughly, this means that the parser always returns an analysis of the input sentence, though not necessarily a fully correct analysis.

Correct syntactic analysis A sentence that contains a MWE should be assigned a syntactic representation that 'denotes' the semantic interpretation of the meaning of the MWE use and not the meaning of a literal use. A robust wide-coverage parser can return a well-formed and grammatical analysis of a sentence with a MWE. However, the resulting analysis evokes a literal semantic interpretation. Thus, by correct syntactic analysis of a MWE we mean that analysis whose dependency structure corresponds to an event denoting a MWE interpretation. Here an example: *Iedereen moet zijn handen uit de mouwen steken* 'Everybody must get to work'. A robust parser probably returns a syntactic analysis where *zijn handen* is treated as a direct object of the head verb *steken*, *uit de mouwen* as a locative or directional PP and *iedereen* the subject. This analysis underlies a literal semantic interpretation with the meaning 'everybody stretch the hands out of the sleeves'. In the correct analysis *iedereen* is the subject, *steken* heads a complex predicate that requires a direct object *zijn handen* and the fixed

¹Grégoire documented all this in the above mentioned report.

phrase *uit de mouwen*. This complex predicate gets a dependency structure that underlies the semantic interpretation 'everybody (must) get to work'.

2 Automatic conversion

Refer to the report STEVIN-IRME WP2, deliverable 5.1a by Nicole Grégoire to know how the automatic conversion has been done. The output of the automatic conversion is basically a new lexicon that includes the original Alpino lexicon extended with the verbal MWES from the ECM. This lexicon file is named `new_verbs.pl`.

Before the Alpino parser can actually use the new lexicon, the file needs to be checked for potential syntax errors and compiled. If errors exist, Alpino, built on Sicstus Prolog, cannot compile.

Here, we enumerate a few errors that remained and needed to be manually corrected before the parser could run using the new lexicon:

1. A complement frame lists the required arguments selected by a head verb. The arguments are specified either as a syntactic type (`accusative`, `dative`, `sbar`) or as a lexeme or sequence of lexemes (`handen`, `uit de mouwen`). Three syntactic types in `new_verbs.pl` are unknown by the parser: `sbar_obj_no_het` (sbar object not being *het*), `sbar_obj` (sbar object) and `vp_obj` (verb phrase functioning as object). That is, the parser ignores that these syntactic arguments may be part of a MWE complementation frame. The first two types were replaced by `sbar` and the latter by `vp`.
2. The ECM verbal MWES do not specify what passive type a MWE may allow. All MWE lexical entries get as default the `norm_passive` type. The original Alpino lexicon does treat passive. As a consequence, the lexicon file compiles o.k, however, in an open domain, an input MWE in a passive construction other than the `norm_passive` type, will not be recognized as a MWE by the parser.
3. Automatic conversion also added two errors in two different verb entries:

```
v(positioneer,positioneert,positioneren,gepositioneerd,  
  positioneerde, positioneerden,  
    [h([transitive]))].  
refl_ld_pp,  
refl_ld_adv,
```

and

```
v(passeer ,passeert ,passeren ,gepasseerd ,passeerde ,passeerden ,
```

```

        [z([intransitive,
pc_pp(voor)])],
        h([np_pc_pp(voor)]),
        b([transitive]))].
fixed([[de,revue]],norm_passive)seerd

```

These were manually changed to:

```

v(positioneer,positioneert,positioneren,gepositioneerd,
positioneerde, positioneerden,
[h([transitive,
refl_ld_pp,
refl_ld_adv]))].

```

```

v(passeer,passeert,passeren,gepasseerd,passeerde,passeerden,
[z([intransitive,
pc_pp(voor)]),
h([np_pc_pp(voor)]),
b([transitive,
fixed([[de,revue]],norm_passive)))]).

```

Concerning the first error: Examples of verbal MWEs that show such sub-categorization requirements are easily found in the corpora we used to extract the morpho-syntactic information of the MWEs in the database. Thus, the argument types mentioned above are possible within the syntactic dependents' list in a verbal MWE. However, the current version of the Alpino grammar does not know about them.

3 Parsing accuracy

3.1 Introduction

Here, we address the following questions: What is the parsing accuracy with Alpino original lexicon? Does the accuracy improve or degrade with the extended lexicon? What did we expect? What errors are found with the original lexicon and with the extended lexicon? Are some errors corrected thanks to using the ECM database? What errors remain?

3.2 Methodology

Given two samples of sentences. A sample of 10 sentences that do not contain MWEs was selected from the Twente Nieuws Corpus. Additionally, we took a sample of 100 short sentences that contain MWEs (each sentence a different MWE). This second sample was extracted from the list of examples

Lexicon	# of verbal MWES
Alpino lexicon (<code>lexicon</code>)	1908
Extended lexicon (<code>lexicon+irme</code>)	9257

Table 1: Lexica – object of comparison

of MWES distributed with the ECM lexical database. Below, Table 6 shows a few examples of short sentences included in this second sample.

We compared the accuracy of the parser while parsing the two sets of sentences and using (a) the original Alpino lexicon and (b) the *extended* lexicon. By extended lexicon we mean the original lexicon expanded with the verbal MWES from the ECM database. Table 1 shows the number of verbal MWES in the two lexica. (Note a MWE that allows morpho-syntactic variation, e.g. various possessive determiners requires an entry per possessive determiner realization.)

For each input sentence we have two parses, one using the original lexicon and one using the extended lexicon. To establish whether parsing accuracy improves or degrades, one can compare the two analyses and figure out how they differ (if at all). In addition, we want to do the comparison of the syntactic analyses automatically and measure the differences in a systematic and objective way.

The analysis returned by the parser is a syntax tree that can be represented e.g. as a set of dependency structures. Using this notation, we have access to all the dependency relations between the constituents that make up the input sentence and that the parser found. Among other representations, Alpino can encode such dependency relations as dependency triples. For example, for a hypothetical English input sentence *John bought flowers* the parser would return the dependency triples (`buy,subject,John`) and (`buy,object1,flowers`). This notation facilitates the comparison between analyses returned by the parser. Furthermore, the notation facilitates measuring the differences between analyses.

We pursued the following approach:²

Parse sentences with original lexicon Parsing was done automatically by Alpino. From all possible parses we took the best parse only.

Parse sentences with extended lexicon We recompiled the parser using the new extended lexicon and re-parsed the two samples of sentences. Again, we took the best parse only.

Manually annotate sentences Using D-coi annotation tools, we manu-

²Gertjan van Noord kindly gave useful advice on setting up the evaluation and provided the necessary tools to examine the results.


```
end_hook=xml -parse 2>> $(suite).log \
```

Among the parsing parameters that one could experiment with, the number of solutions (parses) is important. As mentioned above, we considered only one parse per sentence produced by Alpino. This is the parse that received the highest probability, thus, the best parse for a given sentence. One could consider more than one parse solution during evaluation, however, we decided against this possibility in order to facilitate the comparison and the evaluation. It is possible that if we took into account more than one solution, the performance scores would vary; however, we do not expect a big difference in the scores. Regarding CPU time and sentence length, we did not experiment with these options either, given that the sample sentences were rather short. Because of this, the parser did not get close to exhausting the CPU time allocated during parsing.

3.3 Expectations

In general, if a MWE is described in the lexicon (this applies to both the Alpino original lexicon and the new lexicon) the dependency structure assigned by the parser to that MWE mirrors the semantic interpretation of the whole expression as a semantic unit. Thus, given that the extended lexicon contains more MWE lexical entries, it is expected that when Alpino uses the extended lexicon, more sentences with MWE are correctly analyzed than when Alpino uses the original lexicon. In short, using the extended lexicon the parsing accuracy should improve given that more MWEs are described in the lexicon.

While using the extended lexicon, we also expect that:

1. The phrase projected by the head verb within a MWE is correctly analysed as a verbal phrase with all its necessary syntactic dependents. This also means, that the local tree of the VP has all the elements needed to get the MWE interpretation and not just a literal or nonsense interpretation.
2. A verbal MWE may get at least two analyses: one with a regular productive structure and one with a fixed phrase structure. For example, *een gezin stichten* may be analyzed as a transitive VP or as a verbal MWE headed by *stichten* that requires a fixed (lexicalized) accusative object headed by *gezin*. Syntactically, any of the two analyses is correct. Semantically, only the second analysis would mirror a MWE interpretation and a special treatment as a lexicalized unit (relevant for machine translation e.g.).
3. The syntactic dependents that are *fixed* –i.e. they allow no morpho-syntactic variation– are labeled as such and not as regular and productive constituents (PP, NP).

4. If an argument in a MWE lexical entry is specified as a fixed string in the database and the argument happens to allow morpho-syntactic variation –i.e. examples to be parsed show other possible surface realizations– then, the MWE will be analyzed as a regular productive expression provided that the input is a well-formed Dutch sentence.
5. Families of MWEs exist where a PHRASE combines with a group of verbs (so-called support verbs). If the phrase is only specified as complement of one support verb, other instances of the phrase with other support verbs will not be recognized as a MWE. For example, *het recht hebben om . . .*, *het recht krijgen om*, etc.
6. Solving a PP-attachment site is difficult for a parser. A constituent PP within an MWE may be wrongly analyzed as a PP modifier of a preceding noun ((NP *oogje* (PP *in het zeil*)) *houden*) or as a modifier of the verb ((NP *oogje*) (PP *in het zeil*) *houden*). If the MWE is described in the lexicon, we expect the PP-attachment to be correctly analyzed: the PP is a required dependent of the VERB and labeled as a lexicalized part of the verb phrase. Again, this has consequences for the semantic interpretation.
7. Passive: the ECM lexical database does not encode the passivizability of verbal MWEs; by default, a MWE allows a normal passive. This means that the lexical description of MWEs in the database needs to be enriched with information about whether the MWE allows passive and, if it does, what sort of passive is possible. An NLP system that incorporates the ECM database and that needs to process a verbal MWE in a passive context will treat the verbal MWE as a productive verbal phrase (and not as a MWE).

3.4 Results

Table 2 provides the per sentence concept accuracy in the two ‘corpora’ with the two different lexica. The parsing accuracy of sentences that contain MWEs improves substantially when using the extended lexicon. On the other hand, the parsing accuracy of sentences without MWEs slightly improves with the extended lexicon.

Accompanying this deliverable, the evaluation material listed in Table 3 is made available. The material under the directory **Suites** includes the test sentences. The material under the directory **Treebank** consists of dependency structures stored in `.xml` format. These can be viewed with an `.xml` editor or any tree viewing program such as `dtview` that is distributed with Alpino annotation tools.

	Lexicon	Concept accuracy CA_S
MWES	Alpino lexicon	82.849
	Extended lexicon	94.088
Non-MWES	Alpino lexicon	95.833
	Extended lexicon	96.389

Table 2: Parsing accuracy scores

Files	Description
Suites directory	
<code>*.sents</code>	Ascii text files with sentences
<code>*.pl</code>	Prolog files with sentences
Makefile	<code>*.sents</code> to <code>*.pl</code> conversion script
Treebank directory	
GoldStandard/(no_)mwe	Manually annotated sentences
Original/(no_)mwe	Sentences annotated with Original lexicon
ExtendedLex/(no_)mwe	Sentences annotated with Extended lexicon
README	Brief description of this material

Table 3: Evaluation material

3.5 Error analysis

We wanted to establish whether the analyses of sentences that contain a verbal MWE and that are returned by the parser differ. If the analyses differ, we wanted to know what errors are found and how does the parser performance differ when using one or the other lexicon.

Using alpino treebanking tools, we extracted an error categorization. Three error types are checked to assess the parsing accuracy:

- **Wrong dependency:** An example: *Hij heeft naar adem gehapt*. The parser labels *naar adem* as a locative directive PP, whereas the gold standard considers it a fixed argument (**svp**) of the verb. Arguments annotated as fixed argument (**svp**) in the gold standard are labeled as **ld**, **obj1**, **predc** or **mod**, by the parser using the original lexicon. Other phrases annotated as **mwp**, **se**, **pc** in gold standard are treated as **obj1**, **mod** or **ld** by the parser.
- **Missing relation:** An example is *Hij heeft iets in bewaring gegeven*. There holds no dependency relation between the verb *geven* and the phrase *in bewaring*; instead, the parser treats *in bewaring* as post-modifier of the noun *iets*.

	Error type	Instances
MWES	Wrong dependency	54
	Missing relation	62
	System mistake	64
Non-MWES	Error type	Instances
	Wrong dependency	1
	Missing relation	6
	System mistake	6

Table 4: Error types using original lexicon

	Error type	Instances
MWES	Wrong dependency	18
	Missing relation	22
	System mistake	24
Non-MWES	Error type	Instances
	Wrong dependency	0
	Missing relation	6
	system mistake	6

Table 5: Error types using extended lexicon

- **System mistake:** In the above example, the system mistake is to label *in bewareng* as post-modifier of the noun *iets*.

Table 4 enumerates the error types and number of errors per type observed when Alpino uses the original lexicon. Table 5 gives similar information when Alpino uses the extended lexicon.

What errors are **corrected** when using extended lexicon? In general, the required arguments of the verb heading the MWE are labeled as such (either *svp*, *obj1*, reflexive *se* or *pc*) depending on the lexical specification. For example, in *de boot afhouden*, the NP is correctly labeled as a fixed object (*svp*) instead of an *obj1*. Furthermore, fixed arguments inside a MWE are labeled as *svp* and treated as a multiword unit, instead of a productive NP or PP.

What errors **remain** when using the extended lexicon?

- Expressions that are described in the lexicon as MWE do not get the expected MWE analysis. Table 6 lists some examples.
- Some PP-attachment decisions remain difficult: *Hij heeft veel interesse gehad in iets*. Here, the parser always chooses for noun attachment, no matter what lexicon we use.

Example	MWE in lexicon ?
<i>naar adem happen</i>	yes
<i>Hij heeft iets onder één noemer gebracht</i>	yes
<i>Hij heeft zijn ontslag ingediend</i>	yes
<i>Hij heeft achter het raam gezeten</i>	yes
<i>Hij heeft zijn naam ontleend aan iets</i>	yes
<i>iemand zijn vinger opsteken</i>	yes
<i>Hij heeft iemand op zijn gemak gesteld</i>	yes
<i>Hij heeft een klap in het gezicht gekregen</i>	yes
<i>Hij heeft de wind in de zeilen gehad</i>	yes

Table 6: Examples that are erroneously treated.

- Lexicon introduces ambiguity: the NP complement may be treated either as a fixed string or as a flexible NP headed by *strijd* in the MWE *zijn strijd staken*:

```
v(staak,staakt,staking,gestaakt,staakte,staakten,
  [h([intransitive,
      fixed([acc(strijd)],norm_passive),
      fixed([zijn,strijd]),norm_passive),
```

If present, the PP phrase is labeled as a modifier due to ambiguity in the lexical entry, and not as a PC complement:

```
v(vorm,vormt,vormen,gevormd,vormde,vormden,
  [h([np_np,
      fixed([acc(bedreiging),pc(voor)],norm_passive),
      fixed([acc(bedreiging)],norm_passive),
```

- Inside the fixed string, a noun seems to be missing:

```
fixed([[in,de],acc],norm_passive),+ hebben
```

- *Iets is volgens plan gegaan*. The MWE *iets volgens plan gaan* is not present in the lexicon. During error evaluation it became evident that numerous MWES with the verb *gaan* 'go' were not converted and added into the new Alpino lexicon.
- Undesirable proliferation of descriptions of those MWES that allow a possessive determiner within a required argument: *iets in zijn hoofd hebben*. This introduces a lot of redundancy in the lexicon and makes lexicon expansion more error-prone.

- A few annotation errors exist in the gold standard treebank. This has a small influence on the quantitative scores of the evaluation.

Finally a general remark. Structurally, the analysis of NP VERB MWEs whose NP allows insertion of variation and modification cannot be distinguished from a productive NP VERB expression. One only knows the parser treats the input sentence as a MWE when checking the verb frames.

The parses produced by Alpino when parsing the non-MWE corpus show almost the same errors no matter what lexicon Alpino uses (the original or the extended lexicon). The slight difference in concept accuracy scores was due to one sentence in which a comparative phrase (*als . . .*) was wrongly analyzed as a PP when using the original lexicon (and correctly analyzed as a comparative phrase when using the extended lexicon). We ignore how general this observation may be, given that the non-MWEs sample was very small.

4 Final remarks

Although this evaluation was performed on two small corpora, we did find out important issues to bear in mind when developing a MWE lexical database to be used in an NLP system.

The automatic conversion of the ECM lexical entries into the Alpino lexicon format was rather successful with a few errors left and manually corrected. A lexical entry in the ECM contains more information that has not been used in the conversion process (e.g. other syntactic arguments as part of a MWE, possible lexical realizations of variable open slots inside a MWE). This information is left out because using it, would require adjustments in the Alpino grammar. This could always be a future extension to this project.

The error categorization and analysis showed that an appropriate treatment of MWEs in the lexicon (at least for this type of parsers) can solve PP-attachment decisions as well as yielding the expected semantic interpretation of the whole MWE as a meaning unit. Care needs to be taken to minimize lexical ambiguity and avoid the proliferation of unnecessary or redundant lexical entries of those MWEs that allow morpho-syntactic variation.

Generally speaking, a robust wide-coverage parser such as Alpino will always produce a fairly good syntactic analysis of sentences with a verbal MWE; however, the parser needs detailed lexical descriptions of the MWE in order to provide an appropriate semantic representation.

References

Gertjan van Noord. At Last Parsing Is Now Operational. In Piet Mertens, Cedrick Fairon, Anne Dister, and Patrick Watrin, editors, *TALN06*

Verbum Ex Machina. Actes de la 13e conference sur le traitement automatique des langues naturelles., pages 20–42, 2006.