# Chapter 8

# Multilingual parsing and MWE detection

Vasiliki Foufi
University of Geneva

Luka Nerima
University of Geneva

Eric Wehrli
University of Geneva

Identifying multiword expressions (MWEs) in a sentence in order to ensure their proper processing in subsequent applications, like machine translation, and performing the syntactic analysis of the sentence are interrelated processes. In our approach, priority is given to parsing alternatives involving collocations, and hence collocational information helps the parser through the maze of alternatives, with the aim to lead to substantial improvements in the performance of both tasks (collocation identification and parsing), and in that of a subsequent task (machine translation).

## 1 Introduction

Multiword expressions (MWEs) are lexical units consisting of more than one word (in the intuitive sense of *word*). There are several types of MWEs, including idioms (*a frog in the throat*, *break a leg*), fixed phrases (*per se*, *by and large*, *rock'n roll*), noun compounds (*traffic lights*, *cable car*), phrasal verbs (*look up*, *take off*), etc. While easily mastered by native speakers, their detection and/or their interpretation pose a major challenge for computational systems, due in part to their flexible and heterogeneous nature.

Vasiliki Foufi, Luka Nerima & Eric Wehrli. 2019. Multilingual parsing and MWE detection. In Yannick Parmentier & Jakub Waszczuk (eds.), *Representation and parsing of multiword expressions: Current trends*, 217–237. Berlin: Language Science Press.
DOI:10.5281/zenodo.2579047

In our research, MWEs are categorized in five subclasses: compounds, discontinuous words, named entities, collocations and idioms. While the first three are expressions of lexical categories (N, V, Adj, etc.) and can therefore be listed along with simple words, collocations and idioms are expressions of phrasal categories (NPs, VPs, etc.). The identification of compounds and named entities can be achieved during the lexical analysis, but the identification of discontinuous words (e.g., particle verbs or phrasal verbs), collocations and idioms requires grammatical data and should be viewed as part of the parsing process.

In this chapter, we will primarily focus on collocations, roughly defined as arbitrary and conventional associations of two words (not counting grammatical words) in a particular grammatical configuration (adjective-noun, noun-noun, verb-object, etc.). Throughout this chapter, we will refer to words belonging to such associations as CONTENT WORDS. We will argue that the identification of collocations and parsing are interrelated processes – in the sense that one cannot precede the other – and we will show how this has been achieved in the Fips multilingual parser (Wehrli 2007; Wehrli & Nerima 2015).

Section 2 will give a brief review of MWEs and previous work. Section 3 will describe how Fips handles MWEs and the way they are represented in our lexical database. Section 4 will be concerned with the treatment of collocation types which present a fair amount of syntactic flexibility (e.g. verb-object). For instance, verbal collocations may undergo syntactic processes such as passivization, relativization, interrogation and even pronominalization, which can leave the collocation constituents far away from each other and/or reverse their canonical order. Section 5 will present the collocation extraction process, which will be evaluated in Section 6. Finally we will conclude in Section 7.

## 2 Multiword expressions: A brief review of related work

The standard approach in dealing with MWEs in parsing is to apply a "words-with-spaces" preprocessing step, which marks the MWEs in the input sentence as units which will later be integrated as single blocks in the parse tree built during analysis (Brun 1998; Zhang & Kordoni 2006). This method is not really adequate for processing collocations. Unlike other expressions that are fixed or semi-fixed, several collocation types do not allow a "words-with-spaces" treatment because they have a high morphosyntactic flexibility. On the other hand, Alegria et al. (2004) and Villavicencio et al. (2007) adopted a compositional approach to the encoding of MWEs, able to capture more morphosyntactically flexible MWEs. Alegria et al. (2004) showed that by using a MWE processor in the preprocessing

stage, a significant improvement in the POS tagging precision is obtained. Villavicencio et al. (2007) found that the addition of 21 new MWEs to the lexicon led to a significant increase in the grammar coverage (from 7.1% to 22.7%), without altering the grammar accuracy. However, as argued by many researchers (e.g., Heid 1994; Seretan 2011), collocation identification is best performed on the basis of parsed material. This is due to the fact that collocations are co-occurrences of lexical items in a specific syntactic configuration. For that reason, we have chosen the identification of collocations as soon as possible during parsing. Finkel & Manning (2009) have built a joint model of parsing and named entity recognition, based on a discriminative feature-based constituency parser. They tested their model on the OntoNotes annotated corpus[1] and they achieved a remarkably good performance on both parsing and recognition of named entities. Green et al. (2013) have developed two structured prediction models with the aim to identify arbitrary-length, contiguous MWEs in Arabic and French. The first is based on context-free grammars and the second uses tree substitution grammars, a formalism that can store larger syntactic fragments. They claim that these techniques can be applied to any language for which a syntactic treebank, a MWE list, and a morphological analyzer exist. Nasr et al. (2015) have developed a joint parsing and MWE identification model for the detection and representation of ambiguous complex function words. Constant & Nivre (2016) developed a transition-based parser which combines two factorized substructures: a standard tree representing the syntactic dependencies between the lexical elements of a sentence and a forest of lexical trees including MWEs identified in the sentence.

## 3 The Fips parser

Fips is a multilingual parser, available for several languages, i.e. French, English, German, Italian, Spanish, Modern Greek, Romanian and Portuguese. It relies on generative grammar concepts and is basically made up of a generic parsing module which can be refined in order to suit the specific needs of a particular language. Fips is a constituent parser that functions as follows: it scans an input string from left to right, without any backtracking. The parsing algorithm, iteratively, performs the following three steps:

- get the next lexical item and project the relevant phrasal category ($X \longrightarrow XP$);
- merge XP with the structure in its left context (the structure already built);

[1] http://www.gabormelli.com/RKB/OntoNotes_Corpus, last accessed 26 February 2019.

- (syntactically) interpret XP, triggering procedures
  - to build predicate-argument structures
  - to create chains linking preposed elements to their trace
  - to find the antecedent of (3rd person) personal pronouns
  - to identify collocations.

The parsing procedure is a one-pass (no pre-processing, no post-processing) scan of the input text, using rules to build up constituent structures and (syntactic) interpretation procedures to determine the dependency relations between constituents (grammatical functions, etc.), including cases of long-distance dependencies. One of the key components of the parser is its lexicon, which contains detailed morphosyntactic and semantic information, selectional properties, valency information, and syntactico-semantic features that are likely to influence the syntactic analysis.

### 3.1 The Fips lexicon

The lexicon was built manually and contains fine-grained information required by the parser. It is organized as a relational database with four main tables:

words, representing all morphological forms (spellings) of the words of a language, grouped into inflectional paradigms;

lexemes, describing more abstract lexical forms which correspond to the syntactic and semantic readings of a word (a lexeme corresponds roughly to a standard dictionary entry);

collocations, which describe multiword expressions combining two lexical items, not counting function words;

variants, which list all the alternatives written forms for a word, e.g. the written forms of British English vs American English, the spellings introduced by a spelling reform, presence of both literary and modern forms in Greek, etc.

### 3.2 Representation of MWEs in the lexicon

In the introduction we mentioned that in our research, MWEs are categorized in five subclasses, i.e. compounds, discontinuous words, named entities, collocations and idioms. We will now describe how they are represented in the lexical database.

Compounds and named entities are represented by the same structure as simple words. An entry describes the syntactic and (some) semantic properties of the word: lexical category (POS), type (e.g., common noun, auxiliary verb), subtype, selectional features, argument structure, semantic features, thematic roles, etc. Each entry is associated with the inflectional paradigm of the word, that is all the inflected forms of the word along with the morphological features (number, gender, person, case, etc.). The possible spaces or hyphens of the compounds are processed at the lexical analyzer level in order to distinguish those that are separators from those belonging to the compound.

Discontinuous words, such as particle verbs or phrasal verbs, are represented in the same way as simple words as well, except that the orthographic string contains the bare verb only, the particle being represented separately in a specific field. The benefit of such an approach is that the phrasal verb inherits the inflectional paradigm of the basic verb. For agglutination, a lexical analyzer will detect and separate the particle from the basic verb.

Collocations are defined as associations of two lexical units (not counting function words) in a specific syntactic relation (for instance adjective-noun, verb-object, etc.). A lexical unit can be a word or a collocation. The definition is therefore recursive and enables to encode collocations that have more than two words (Nerima et al. 2010). For instance, the French collocation *tomber en panne d'essence* ('to run out of gas') is composed of the word *tomber* (lit. 'fall') and the collocation *panne d'essence* (lit. 'failure of gas'). Similarly, the English collocation *guaranteed minimum wage* is composed of the word *guaranteed* and the collocation *minimum wage*.

In addition to the two lexical units, a collocation entry encodes the following information: the citation form, the collocation type (i.e., the syntactic relation between its two components), the preposition (if any) and a set of syntactic frozenness constraints.

Some examples of entries are given in (1), (2) and (3).

(1) *ein Schlaglicht werfen* (DE) 'to highlight'
type : verb-direct object
lexeme #1 : *Schlaglicht* 'spotlight', noun-noun collocation
lexeme #2: *werfen* 'throw', _ NP PP verb
preposition : ∅
features : {}

(2) *κινητό τηλέφωνο* (kinitó tiléfono) (MG) 'mobile phone'
type : adjective-noun
lexeme #1 : *κινητό* (kinitó) 'mobile', adjective

lexeme #2 : *τηλέφωνο* (tiléfono) 'phone', noun
preposition : ∅
features : {}

(3) *banc de poissons* (FR) 'shoal of fish'
type : noun-prep-noun
lexeme #1 : *banc* 'bench', noun
lexeme #2 : *poisson* 'fish', noun
preposition : *de* 'of'
features : {determiner-less complement, plural complement}

For the time being, we represent idioms as collocations with more restriction features (cannot passivize, no modifiers, etc.). They are, therefore, stored in the same database table. Reducing idioms to collocations with specific features though convenient and appropriate for large classes of idioms is nevertheless not general enough. In particular, it does not allow for the representation of idioms with fixed phrases, such as *to get a foot in the door*.

## 3.3 Fips and collocations

### 3.3.1 Collocation identification mechanism

The collocation identification mechanism is integrated in the parser. In the present version of Fips, collocations, if present in the lexicon, are identified in the input sentence during the analysis of that sentence, rather than at the end. In this way, priority is given to parsing alternatives involving collocations, and collocational information helps the parser through the maze of alternatives. To fulfill the goal of interconnecting the parsing procedure and the identification of collocations, we have incorporated the collocation identification mechanism within the constituent attachment procedure (see Section 3.3.2). The Fips parser, like many grammar-based parsers, uses left attachment and right attachment rules to build respectively left subconstituents and right subconstituents. The grammar used for the computational modelling comprises rules and procedures. Attachment rules describe the conditions under which constituents can combine, while procedures compute properties such as long-distance dependencies, agreement, control properties, argument-structure building, and so on.

### 3.3.2 Treatment of collocations

The identification of compounds and named entities can be achieved during the lexical analysis, but the identification of discontinuous words, collocations and

idioms requires grammatical data and are, therefore, part of the parsing process. The identification of a collocation occurs when the second lexical unit of the collocation is attached, either by means of a left attachment rule (e.g., adjective-noun, noun-noun) or by means of a right-attachment rule (e.g., noun-adjective, noun-prep-noun, verb-object), as shown in example (4).

(4) Paul took up a new challenge.
[$_{TP}$ [$_{DP}$ Paul][$_{VP}$ took up [$_{DP}$ a [$_{NP}$ [$_{AP}$ new] challenge]]]]

When the parser reads the noun *challenge* and attaches it (along with the prenominal adjective) as complement of the incomplete [$_{DP}$ a] direct object of the verb *take up*, the identification procedure considers iteratively all the governing nodes of the attached noun and checks whether the association of the lexical head of the governing node and the attached element constitutes an entry in the collocation database. The process stops at the first governing node of a major category (noun, verb or adjective). In our example, going up from *challenge*, the process stops at the verb *take up*. Since *take up - challenge* is an entry in the collocation database and its type (verb-object) corresponds to the syntactic configuration, the identification process succeeds.

In several cases the two constituents of a collocation can be very far apart, or do not appear in the expected order. We will turn to such examples in Section 4. To handle them, the identification procedure sketched above must be slightly modified so that not only the attachment of a lexical item triggers the identification process, but also the attachment of the trace of a preposed lexical item. In such a case, the search will consider the antecedent of the trace. This shows, again, that the main advantage provided by a syntactic parser in such a task is its ability to identify collocations even when complex grammatical processes disturb the canonical order of constituents.

## 4 Detection of collocations in free word-order languages

Just as other types of MWEs, collocations are problematic for NLP because they have to be recognized and treated as a whole, rather than compositionally (Sag et al. 2002). On the other hand, there is no systematic restriction on lexical forms which constitute a collocation, on the order of items in a collocation, or on the number of words that may intervene between these items especially in free word-order languages. In such languages, the direct object of a verbal collocation can be found either before or after the verb, with or without intervening material. This is illustrated in the following examples with the Greek verb-object collocation *κάνω*

*έκκληση* (káno éklisi) 'to make an appeal'. In (5a), the direct object follows the verb, while in (5b), it precedes the verb, with several intervening words between them:

(5) a. Ο Υπουργός Παιδείας *έκανε έκκληση* στους διοικητικούς
O Ipurgós Pedías *ékane éklisi* stus diikitikús
υπαλλήλους να σταματήσουν την απεργία.
ipalílus ná stamatísun tín aperyía
'The Minister of Education *made an appeal* to the administrative staff to stop the strike.'

b. *Έκκληση* στους διοικητικούς υπαλλήλους να σταματήσουν την
*Éklisi* stus diikitikús ipalílus ná stamatísun tín
απεργία *έκανε* ο Υπουργός Παιδείας.
aperyía *ékane* o Ipurgós Pedías
'*An appeal* to the administrative staff to stop the strike *made* the Minister of Education.'

### 4.1 Nominal collocations

Modifiers can often be attached within a nominal collocation, separating the two terms. For example, between the constituents of a nominal collocation in the form of adjective-noun, other lexemes may interfere. Table 1 shows a part of the analysis of a sentence where the possessive determiner *του* (tu) 'his' occurs between the adjective *παρθενικό* (parthenikó) 'maiden' and the noun *ταξίδι* (taxídi) 'voyage' of the collocation *παρθενικό ταξίδι* (parthenikó taxídi) 'maiden voyage'. Note that, for the POS tagset, we opted for the universal tagset (Petrov et al. 2012).

Table 1: Identification of the nominal collocation *παρθενικό ταξίδι* (parthenikó taxídi) 'maiden voyage'

| word | tag | position | collocation |
|---|---|---|---|
| *Το* (to) 'the' | DET | 1 | |
| *παρθενικό* (parthenikó) 'maiden' | ADJ | 4 | |
| *του* (tu) 'his' | PRON | 14 | |
| *ταξίδι* (taxídi) 'voyage' | NOUN | 18 | *παρθενικό ταξίδι* 'maiden voyage' |

## 4.2 Verbal collocations

Verb-object collocations may undergo syntactic processes such as passivization, relativization, interrogation and even pronominalization, which can leave the collocation constituents far away from each other and/or reverse their canonical order.

### 4.2.1 Passive

In passive constructions, the direct object is promoted to the subject position leaving an empty constituent in the direct object position. The detection of a verb-object collocation in a passive sentence is thus triggered by the insertion of the empty constituent in direct object position. The collocation identification procedure checks whether the antecedent of the (empty) direct object and the verb constitute a verb-object collocation. In example (6), the noun *απόφαση* (apófasi) 'decision' of the collocation *παίρνω απόφαση* (pérno apófasi) 'to make a decision' precedes the verb and is in the nominal case, the usual case for subjects.

(6) Η απόφαση πάρθηκε.
I apófasi párthike.
'The decision was made.'

### 4.2.2 Pronominalization

Another transformation that can affect some collocation types is pronominalization. In such cases, it is important to identify the antecedent of the pronoun which can be found either in the same sentence or in the context. Example (7) illustrates a phrase where the pronoun *it* refers to the noun *money*. Since the pronoun is the subject of the passive form *would be well spent*, it is interpreted as the direct object of the verb and therefore stands for an occurrence of the collocation *to spend money*.

(7) … though where the money would come from, and how to ensure that *it* would be well *spent*, is unclear.

In example (8) and Table 2, both the verb *να αναλάβουν* (na analávun) 'to take' of the verb-object collocation *αναλαμβάνω ευθύνη* (analamváno efthíni) 'to take responsibility' and the pronominalized object *τις* (tis) 'them' are found in another sentence.

(8) Ας αναλογιστούν τις ευθύνες τους. Να τις αναλάβουν.
As analogistún tis efthínes tus. Na tis analávun.
'Let them consider their responsibilities. Should they take them.'

Table 2: Identification of a verbal collocation

| word | tag | position | collocation |
|---|---|---|---|
| *Ας* (as) 'Let them' | PRT | 1 | |
| *αναλογιστούν* (analogistún) 'consider' | VERB | 4 | |
| *τις* (tis) 'the' | DET | 17 | |
| ***ευθύνες*** (efthínes) 'responsibilities' | NOUN | 21 | |
| *τους* (tus) 'their' | PRON | 21 | |
| . | PUNC | 33 | |
| *Να* (Na) 'Should' | CONJ | 35 | |
| ***τις*** (tis) 'them' | PRON | 35 | |
| *αναλάβουν* (analávun) 'take' | VERB | 42 | *αναλαμβάνω την ευθύνη* 'take responsibility' |
| . | PUNC | 51 | |

Example (9) and Table 3 concern French and show again two sentences. Each one of them contains a collocation with the noun *record*: *établir un record* 'to set up a record' in the first one, and *battre un record* 'to break a record' in the second one, where the noun is pronominalized in the form of a clitic pronoun (*le* 'it').

(9) Ce *record* a été *établi* l'été dernier. Paul espère *le battre* bientôt.
This record has been set up last summer. Paul hopes *it break* soon.
'This record was set up last summer. Paul hopes to break it soon.'

The parser detects collocations in which the nominal element has been pronominalized thanks to the anaphora resolution component incorporated in Fips (Wehrli & Nerima 2013).

### 4.2.3 Wh-constructions

Our parser can also cope with long-distance dependencies, such as the ones found in wh-questions.[2] In sentence (10) and Table 4, the direct object constituent

[2]wh-words are interrogative (or relative) words such as *who*, *what*, *which*, etc. For a general discussion of wh-constructions, see (Chomsky 1977).

Table 3: Identification of verbal collocations, one with pronominalized object

| word | tag | position | collocation |
|---|---|---|---|
| *Ce* | DET | 1 | |
| *record* | NOUN | 4 | |
| *a* | VERB | 11 | |
| *été* | VERB | 13 | |
| *établi* | VERB | 17 | *établir un record* |
| *l'* | DET | 24 | |
| *été* | NOUN | 26 | *été dernier* |
| *dernier* | ADJ | 30 | |
| *.* | PUNC | 37 | |
| *Paul* | NOUN | 1 | |
| *espère* | VERB | 6 | |
| *le* | PRON | 13 | |
| *battre* | VERB | 16 | *battre un record* |
| *bientôt* | ADV | 23 | |
| *.* | PUNC | 30 | |

occurs at the beginning of the sentence. Again, assuming a generative grammar analysis, we consider that such pre-posed constituents are connected to so-called canonical positions. The fronted element being a direct object, the canonical position is a post-verbal DP position immediately dominated by the VP node. The parser establishes such a link and returns the structure from (10), where $[\text{DP e}]_i$ stands for the empty category (the "trace") of the preposed constituent *Ποιο ρεκόρ* (Pxó rekór) 'Which record'.

(10) Ποιο ρεκόρ θέλει να σπάσει ο Μελισσανίδης? $[_{\text{CP}}\ [_{\text{DP}}$ Ποιο ρεκόρ$]_i]\ [_{\text{TP}}$ Pxó rekór théli na spási o Melisanídis θέλει$]\ [_{\text{CP}}$ να$]\ [_{\text{TP}}$ σπάσει$]\ [_{\text{VP}}\ [_{\text{DP}}$ e$]_i]\ [_{\text{DP}}$ ο Μελισσανίδης$]$

'Which record does Melissanidis want to break?'

In such cases, the collocation identification process is triggered by the insertion of an empty constituent in the direct object position of the verb. Since the empty constituent is connected to the pre-posed constituent, such examples can be easily treated as a minor variant of the standard case described in Section 3.3.1. All so-called wh-constructions are treated in a similar fashion, that is relative clause and topicalization.

Table 4: Identification of verbal collocation in a wh-question

| word | tag | position | collocation |
|---|---|---|---|
| *Ποιο* (Pio) 'Which' | DET | 1 | |
| *ρεκόρ* (rekór) 'record' | NOUN | 6 | |
| *θέλει* (théli) 'wants' | VERB | 12 | |
| *να* (na) 'to' | CONJ | 18 | |
| *σπάσει* (spási) 'break' | VERB | 21 | *σπάζω το ρεκόρ* 'break the record' |
| *ο* (o) 'the' | DET | 28 | |
| *Μελισσανίδης* (Melisanídis) 'Melisanidis' | NOUN | 30 | |

#### 4.2.4 *Tough*-movement constructions

In such constructions, the matrix subject is construed as the direct object of the infinitival verb governed by a *tough* adjective. Following Chomsky's (1977) analysis of such constructions, the parser will hypothesize an abstract wh-operator in the specifier position of the infinitival clause, which is linked to the matrix subject. Like all wh-constituents, the abstract operator will itself be connected to an empty constituent later on in the analysis, giving rise to a chain connecting the subject of the main clause and the direct object position of the infinitival clause. The structure as computed by the parser is given in (11), with the chain marked by the index *i*.

(11) $[_{TP}\ [_{DP}$ this record$]_i$ seems$[_{AP}$ difficult$[_{TP}\ [_{DP}\ e]_i$ to$[_{VP}$ break$[_{DP}\ e]_i]]]]$

### 4.3 Complex collocations

As observed by Heid (1994), among others, collocations can involve more than two content words. Such complex expressions can be described recursively as collocations of collocations. Our identification procedure has been extended to handle such cases. For example, the Greek noun-noun collocation *απεργία πείνας* (aperyía pínas) 'hunger strike', which combines with the verb *κάνω* (káno) 'to do', yields the larger verb-object collocation *κάνω απεργία πείνας* (káno aperyía pínas) 'to go on hunger strike', where the object is itself a noun-noun collocation. Given the strict left-to-right processing order assumed by the parser, the system will first identify the collocation *κάνω απεργία* (káno aperyía) 'to go on strike' when attaching the word *απεργία* (aperyía) 'strike'. Then, reading the last word, *πείνας* (pínas) 'hunger' (here in genitive case), the parser will identify the collocation

*απεργία πείνας* (aperyía pínas) 'hunger strike'. The search succeeds with the verb *κάνω* (káno) 'to do', and the collocation *κάνω απεργία πείνας* (káno aperyía pínas) 'to go on hunger strike' is identified.

Moreover, the Greek lexical database comprises nominal collocations formed by a simple noun and a collocation or by two collocations. For example, *δύναμη πολιτικής προστασίας* (dínami politikís prostasías) 'civil protection force' is formed by a simple noun, *δύναμη* (dínami) 'force', and a nominal collocation in genitive case, *πολιτικής προστασίας* (politikís prostasías) 'of civil protection'. The collocation *πυρηνικός σταθμός παραγωγής ενέργειας* (pirinikós stathmós paragoyís enéryias) 'nuclear power station' is formed by the collocations *πυρηνικός σταθμός* (pirinikós stathmós) 'nuclear station' and *παραγωγής ενέργειας* (paragoyís enéryias) 'of energy production'.

## 5 Collocation extraction

As already mentioned, the parser can only identify collocations that are part of its lexical database. Therefore, it is crucial to have as good a coverage of collocations as possible in the database. To help the linguist/lexicographer in the time-consuming task of inserting collocations, we have designed a collocation extraction tool (Seretan 2011), dubbed FipsCo. Applied to a corpus, FipsCo parses all the sentences, extracting all the pairs of lexical items which co-occur in predefined grammatical configurations (adjective-noun, noun-noun, subject-verb, verb-object, etc.). All those pairs are considered as potential collocations.

Once the corpus has been completely parsed, a statistical filter is used to rank the potential collocations according to their degree of association. By default, we use the log-likelihood ratio measure (LLR), since it was shown to be particularly suited to language data (Dunning 1993). In our extractor, the items of each candidate expression represent base word forms (lemmas) and they are considered in the canonical order implied by the given syntactic configuration (e.g., for a verb-object candidate, the object is postverbal in subject-verb-object (SVO) languages like Greek). Even if the candidate occurs in corpus in different morphosyntactic realizations, its various occurrences are successfully identified as instances of the same type thanks to the syntactic analysis performed by the parser.

Figure 1 displays a list of verb-object collocations extracted from an English corpus taken from the magazine *The Economist*. On the left, candidate collocations are listed and at the same time they are shown in their context.

Our system recognizes a large range of collocation types (more than 30 types), including several nominal and verbal ones. The most frequent types are:

- Adjective-noun, e.g. *nuclear war*;
- Noun-noun, e.g. *flower shop*;
- Noun-preposition-noun, e.g. *casco di banane* ('bunch of bananas');
- Verb-object where the object is a bare noun, e.g. *take part*;
- Verb-preposition-noun, e.g. *bring to light*;
- Verb-adverb, e.g. *put together*.

Once filtered and ordered by means of standard association measures, the candidate collocations are manually validated and added to the lexical database. The current content of the database for six European languages is shown in Table 5.

## 6 Evaluation and results

The Fips parser performs well compared to other "deep" linguistic parsers (Delphin,[3] ParGram,[4] etc.) in terms of speed. Parsing time depends on two main factors: (i) the type and complexity of the corpus, and (ii) the selected beam size (maximum number of alternatives allowed). By default, Fips runs with a beam size of 40 alternatives, which gives it a speed ranging from 150 to 250 tokens (word, punctuation) per second. At that pace, parsing a one million word corpus takes approximately 2–3 hours. We are going to present the experiments that were performed for Modern Greek and English in order to evaluate the performance of our parser.

### 6.1 Modern Greek

The evaluation measures the performance of our parser to identify collocations that are lexicalized (i.e. collocations that are present in the lexical database). We also measure the impact of the collocation knowledge on the performance of the parser (in percentage of complete analyses). To achieve the evaluation, we took a small newspaper corpus of about 20,000 words and we manually identified

[3]International consortium developing HPSG grammars and other tools, cf. http://www.delphin.net/wiki/index.php/Home.

[4]ParGram is an international consortium for the development of LFG-based grammars, see http://pargram.b.uib.no.

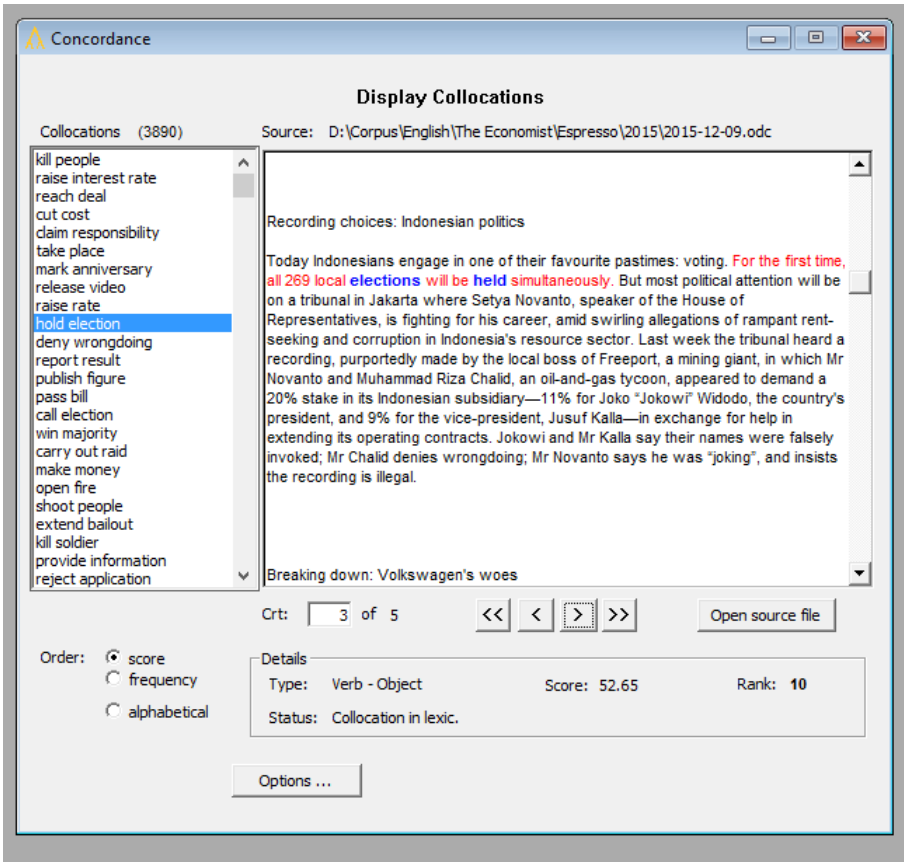


Figure 1: Extraction of verb-object collocations

Table 5: Number and types of collocations in the Fips lexical database

| collocation type | English | French | German | Italian | Spanish | Greek |
|---|---|---|---|---|---|---|
| Adjective-noun | 3,049 | 5,935 | 490 | 1,325 | 1,621 | 20,131 |
| Noun-noun | 5,671 | 454 | 2,476 | 131 | 66 | 471 |
| Noun-prep-noun | 555 | 7,846 | 22 | 1,246 | 988 | 11 |
| Verb-object | 850 | 1,560 | 197 | 250 | 1,098 | 382 |
| Others | 932 | 2,963 | 330 | 209 | 592 | 126 |
| Total | 11,057 | 18,758 | 3,515 | 3,161 | 4,365 | 21,122 |

638 collocations (both nominal and verbal). We ran the parser twice on the corpus: the first time before and the second time after enrichment of the collocation database. On the first run, the parser achieved 43.26% of complete analyses and identified 124 collocations. On the second run, after enrichment of the lexicon, the percentage of complete analyses increased to 44.33% and nearly three quarters of the corpus collocations were identified (482/638). Over this small corpus, the parser achieved a 100% precision in the collocation identification task, with a recall of 75.54% and an F-measure of 86%. The collocations that were not identified (156 out of 638) were part of sentences for which the parser did not achieve a complete analysis.

## 6.2 English

We have also conducted an evaluation over a corpus with approximately 6,000 sentences taken from *The Economist*. The research questions were specifically focused on the statistical significance of ambiguity resolution based on collocation knowledge and on how frequently, in a given corpus, the detection of a collocation helps the parser make the "right" decision. To answer those questions, we parsed the corpus twice, first with the collocation detection component turned on and then with the component turned off. We then compared the results of both runs. Since it was difficult to compare phrase-structure representations, we used the Fips tagger, that is the Fips parser with part-of-speech output. It is indeed much easier to compare POS-tags than phrase-structures. Tables 6 and 7 illustrate the Fips tagger output for the segment in boldface of the sentence *The researchers estimated* ***the total worldwide labour costs*** *for the iPad at $33, of which China's share was just $8.*

Table 6 gives the results obtained with the collocation detection component turned on, and Table 7 the results obtained with the component turned off.

Table 6: Parser output *with* collocation knowledge

| word | tag | position | collocation |
|---|---|---|---|
| *the* | DET | 27 | |
| *total* | ADJ | 31 | |
| *worldwide* | ADJ | 37 | |
| *labour* | NOUN | 47 | |
| *costs* | **NOUN** | 54 | *labour costs* |

Table 7: Parser output *without* collocation knowledge

| word | tag | position | collocation |
|---|---|---|---|
| *the* | DET | 27 | |
| *total* | ADJ | 31 | |
| *worldwide* | ADJ | 37 | |
| *labour* | NOUN | 47 | |
| *costs* | **VERB** | 54 | |

The sentence segment *the total worldwide labour costs* is displayed in both tables with the words in the first column, the part-of-speech tag in the second column and the position – expressed as position of the first character of each word starting from the beginning of the sentence – in the third column. As we can see, the word *costs* is taken as a noun in the first analysis, as a verb in the second. The (correct) choice of a nominal reading in the first analysis is due to the detection of the collocation *labour costs*. In the second run, given the absence of collocational knowledge, the parser opts for the verbal reading. Both output files could easily be manually compared using a specific user interface as illustrated in the screenshot in Figure 2, where POS differences are displayed in red.

Table 8: POS-tagging with and without collocation knowledge

| | with collocations | without collocations |
|---|---|---|
| complete analyses | 73.41% | 72.95% |
| POS-tag differences | 727 | - |
| better tags | 382 | 106 |
| number of collocations | 1668 | - |

A summary of the results of the evaluation is given in Table 8. The first line shows the number of complete analyses. Collocational knowledge increases the number of complete analysis by approximately 0.5%, or about 30 sentences for our corpus of 6,000 sentences. 727 tags are different between the two runs. Of those, excluding differences which do not really matter (some words can be analyzed either as predicative adjectives or as adverbs without much semantic differences, etc.), in 382 cases the tags were better in the first run (with collocational knowledge), and 106 cases better in the second run (without collocational

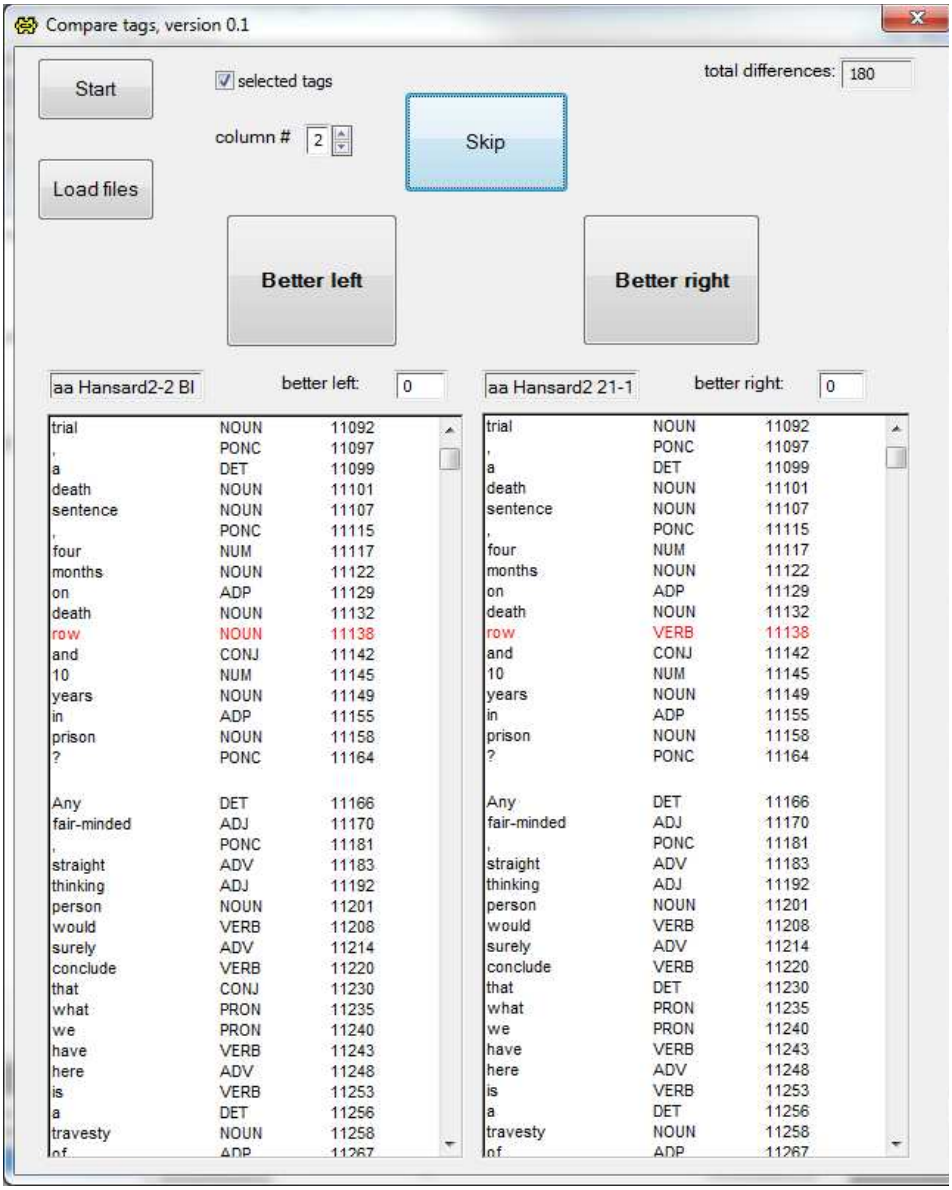


Figure 2: The evaluation user interface

knowledge). In other words, collocational knowledge helped the parser make the better decision four times more than it penalized it. Notice finally that 1,668 collocations were detected in the corpus (more than one in four sentences), which clearly stresses the high frequency of this phenomenon in natural language.

## 7 Conclusion

In this chapter, we have argued in favour of a treatment of collocations, and by extension of all MWEs, fully integrated in the parsing process. The argument is rather simple. On the one hand, we have shown that the identification of collocations must be based on analyzed data, and therefore cannot be performed before parsing. On the other hand, we have also shown that collocation identification can help the parser, for instance to solve lexical as well as syntactic ambiguities, provided that the identification is done before the end of parsing. The solution to this apparent paradox – collocation identification cannot be done before and cannot be done after parsing – is clear: collocation identification must be part of the parsing process and must be performed as early as possible, that is at the time the parser attaches the second constituent of the collocation, or inserts the trace of that constituent.

## Abbreviations

Tagset from Petrov et al. (2012).

| | | | |
|---|---|---|---|
| ADJ | adjective | NUM | numeral |
| ADP | adposition | PRON | pronoun |
| ADV | adverb | PRT | particle |
| CONJ | coordinating conjunction | PUNC | punctuation |
| DET | determiner | VERB | verb |
| NOUN | noun | X | other |

## References

Alegria, Iñaki, Olatz Ansa, Xabier Artola, Nerea Ezeiza, Koldo Gojenola & Ruben Urizar. 2004. Representation and treatment of multiword expressions in Basque. In Takaaki Tanaka, Aline Villavicencio, Francis Bond & Anna Korhonen (eds.), *Second ACL workshop on multiword expressions: Integrating processing*, 48–55. Barcelona, Spain: Association for Computational Linguistics.

Brun, Caroline. 1998. Terminology finite-state preprocessing for computational LFG. In *COLING 1998 volume 1: The 17th international conference on Computational Linguistics*, 196–200. Montreal, QC.

Chomsky, Noam. 1977. On Wh-movement. In P.W. Culicover, T. Wasw & A. Akmajian (eds.), *Formal syntax*. San Francisco, London: Academic Press.

Constant, Mathieu & Joakim Nivre. 2016. A transition-based system for joint lexical and syntactic analysis. In *Proceedings of the 54th annual meeting of the Association for Computational Linguistics*, vol. 1: *Long papers*, 161–171. Berlin, Germany: Association for Computational Linguistics. http://www.aclweb.org/anthology/P16-1016.

Dunning, Ted. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics* 19(1). 61–74. http://dl.acm.org/citation.cfm?id=972450.972454.

Finkel, Jenny Rose & Christopher D. Manning. 2009. Joint parsing and named entity recognition. In *Proceedings of Human Language Technologies: The 2009 annual conference of the North American chapter of the Association for Computational Linguistics*, 326–334. Boulder, Colorado: Association for Computational Linguistics. http://www.aclweb.org/anthology/N/N09/N09-1037.

Green, Spence, Marie-Catherine de Marneffe & Christopher D. Manning. 2013. Parsing models for identifying multiword expressions. *Computational Linguistics* 39(1). 195–227.

Heid, Ulrich. 1994. On ways words work together: Topics in lexical combinatorics. In Willy Martin et al. (ed.), *Proceedings of the VIth Euralex international congress (EURALEX'94)*, 226–257. Amsterdam. Eingeladener Hauptvortrag.

Nasr, Alexis, Carlos Ramisch, José Deulofeu & André Valli. 2015. Joint dependency parsing and multiword expression tokenisation. In *53rd annual meeting of the Association for Computational Linguistics*, 1116–1126. Beijing, China.

Nerima, Luka, Eric Wehrli & Violeta Seretan. 2010. A recursive treatment of collocations. In *Proceedings of the seventh international conference on Language Resources and Evaluation (lrec'10)*, 634–638. Valletta, Malta.

Petrov, Slav, Dipanjan Das & Ryan McDonald. 2012. A universal part-of-speech tagset. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Mehmet Uğur Doğan, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk & Stelios Piperidis (eds.), *Proceedings of the eighth international Language Resources and Evaluation Conference, LREC 2012*, 2089–2096. Istanbul, Turkey. http://www.lrec-conf.org/proceedings/lrec2012/summaries/274.html.

Sag, Ivan, Timothy Baldwin, Francis Bond, Ann Copestake & Dan Flickinger. 2002. Multiword expressions: A pain in the neck for NLP. In *Proceedings of the*

*3rd international conference on Computational Linguistics and Intelligent Text Processing* (Lecture Notes in Computer Science 2276), 1–15. Springer.

Seretan, Violeta. 2011. *Syntax-based collocation extraction* (Text, Speech and Language Technology 44). Dordrecht: Springer.

Villavicencio, Aline, Valia Kordoni, Yi Zhang, Marco Idiart & Carlos Ramisch. 2007. Validation and evaluation of automatically acquired multiword expressions for grammar engineering. In *Proceedings of the 2007 joint conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (emnlp-conll)*, 1034–1043. Prague, Czech Republic: Association for Computational Linguistics. http://www.aclweb.org/anthology/D/D07/D07-1110.

Wehrli, Eric. 2007. Fips: A "deep" linguistic multilingual parser. In *Proceedings of the ACL 2007 workshop on Deep Linguistic Processing*, 120–127. Prague, Czech Republic: Association for Computational Linguistics. http://www.aclweb.org/anthology/W/W07/W07-1216.

Wehrli, Eric & Luka Nerima. 2013. Anaphora resolution, collocations and translation. In J. Monti, R. Mitkov, G. Corpas Pastor & V. Seretan (eds.), *Proceedings of the workshop on Multiword Units in Machine Translation and Translation Technology (MUMTT'2013)*. Nice.

Wehrli, Eric & Luka Nerima. 2015. The Fips multilingual parser. In N. Gala, R. Rapp & G. Bel-Enquix (eds.), *Language production cognition, and the lexicon, Festschrift in honour of Michael Zock*, 473–489. Springer.

Zhang, Yi & Valia Kordoni. 2006. Automated deep lexical acquisition for robust open texts processing. In *Proceedings of the 5th internation conference on Language Resource and Evaluation (LREC-2006)*, 275–280. Genoa, Italy.