Automatic Annotation of Verbal Collocations in Modern Greek

Vasiliki Foufi1[0000-0002-8625-0734], Luka Nerima2[0000-0003-4247-7096] and Eric Wehrli3

1 University of Geneva, Switzerland
Vasiliki.Foufi@unige.ch
2 University of Geneva, Switzerland
Luka.Nerima@unige.ch
3 University of Geneva, Switzerland
Eric.Wehrli@unige.ch

Abstract. Identifying multiword expressions (MWEs) in a sentence 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 (automatic annotation). In this paper, we are going to present our system and the procedure that we have followed in order to proceed to the automatic annotation of Greek verbal multiword expressions (VMWEs) in running texts.

Keywords: Multiword expressions, verbal collocations, annotation, parsing.

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), noun compounds (traffic light, cable car), discontinuous words (look up, take off), collocations (take a decision, mobile phone), 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.

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 category (N, V, Adj, etc.) and can therefore be listed along with simple words, collocations and idioms are expressions of phrasal category (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 paper, we will primarily focus on verbal multiword expressions, and especially on verbal collocations in Modern Greek. Section 2 will give a brief review of MWEs and previous work. Section 3 will describe how our system handles MWEs, the way they are represented in its lexical database and will also be concerned with the treatment of collocation types which present a fair amount of syntactic flexibility (e.g. verb-object) especially in free word-order languages like Modern Greek. 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 4 will present the modifications made in order to adapt our system to the gold standard output provided in the framework of the PARSEME shared task\(^1\) so as to evaluate our system and provide results (section 5).

2 Multiword Expressions: Related Work

In one of the first studies in automatic corpus-based collocation extraction, Church and Hanks [1] proposed the association ratio, a metric based on the information theoretic concept of mutual information. In parsing, the standard approach in dealing with MWEs 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 [2, 3]. This method is not really adequate for processing collocations because they have a high morphosyntactic flexibility. Alegria et al. [4] and Villavicencio et al. [5] adopted a compositional approach to the encoding of MWEs, able to capture more morphosyntactically flexible MWEs. Alegria et al. [4] showed that by using a MWE processor in the preprocessing stage, a significant improvement in the POS tagging precision is obtained. However, as argued by many researchers [6, 7], 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. Additionally, Nasr et al. [8] have developed a joint parsing and MWE identification model for the detection and representation of ambiguous complex function words. Constant and Nivre [9] 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 MWE identified in the sentence.

Many studies have focused on the automatic identification of Modern Greek collocations. Frantzi [10] applied the C-value method on a specialized corpus in the aim to extract collocations and to enrich the specialized dictionary of Modern Greek. A hybrid method for the automatic extraction of nominal MWEs based on grammar rules and word lists but also on statistical evaluation of structures was constructed in the framework of the project ‘Ekfrasis’ [11]. Linardaki et al. [12] have created a dictionary of MWEs based on automatic extraction and manual validation of candidate

MWEs. Finally, Samaridi and Markantonatou [13] have integrated fixed verbal MWEs in an LFG grammar of Modern Greek.

3   The Fips Parser

Fips [14, 15] 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.

The parsing procedure is a one pass (no preprocessing, 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 Lexicon

The lexicon is 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 MWE combining two lexical items, not counting function words;
- variants, which list all the alternative 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   Collocations in the Lexicon

In the introduction, we mentioned that in our research the MWEs are categorized in five subclasses, i.e. compounds, discontinuous words, named entities, collocations and idioms. Let’s see how collocations are represented in the lexical database.

Collocations are defined as associations of two lexical units (not counting function words) in a specific syntactic relation (for instance adjective - noun, verb - noun (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. For in-
instance, the Greek collocation κάνω απεργία πείνας (‘to make a hunger strike’, literal translation) is composed of the word κάνω ‘make’ and the nominal collocation απεργία πείνας ‘hunger strike’.

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. Verbal collocations mainly occur in the following morpho-syntactic types:

- Verb-object where the object is a bare noun, e.g. κάνω βόλτα ‘take a walk’;
- Verb-object where the object is a nominal collocation, e.g. έχω τηλεφωνική επικοινωνία ‘have a phone conversation’ (literal translation);
- Verb-preposition-noun, e.g. φέρω εις πέρας ‘carry out’;
- Verb-adverb, e.g. λαμβάνω υπόψη ‘take into account’;
- Verb-adjective, e.g. κάνω (κάτι) γνωστό ‘make (something) known’.

3.3 Parsing and Collocations

Collocation Identification Mechanism. The collocation identification mechanism is integrated in the parser. Collocations, if present in the lexicon, are identified in the input sentence during the analysis of that sentence. In this way, priority can be given to parsing alternatives involving collocations. Thus collocational information helps the parser through the maze of alternatives. To fulfil the goal of interconnecting the parsing procedure and the identification of collocations, we have incorporated the collocation identification mechanism within the constituent attachment procedure. Our parser, like many grammar-based parsers, uses left attachment and right attachment rules to build respectively left 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.

Treatment of Collocations. 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). In the example Η κυβέρνηση έκανε λάθος ‘The government made a mistake’, when the parser reads the noun λάθος ‘mistake’ and attaches it as complement of the incomplete direct object of the verb έκανε ‘made’, 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 λάθος ‘mistake’, the process stops at the verb έκανε ‘made’. Since έκανε ‘made’- λάθος ‘mistake’ is an entry in the collocation
database and its type (verb-object) corresponds to the syntactic configuration, the identification process succeeds.

As already pointed out, in several cases the two constituents of a collocation can be very far apart, or do not appear in the expected order. For instance, 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.

In passive constructions, the direct object is promoted to the subject position leaving a trace, i.e. 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 the example Πάρθηκε η απόφαση ‘It was made the decision’, the noun απόφαση ‘decision’ of the collocation παίρνω μια απόφαση ‘to make a decision’ precedes the verb.

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. The example cited below illustrates a sentence where the pronoun τις ‘them’ found in the second sentence refers to the noun ευθύνες ‘responsibilities’ found in the first sentence. Since the pronoun is the object of the verb αναλάβουν ‘take on’, it stands for an occurrence of the collocation η αναλαμβάνω ευθύνη ‘take on responsibility’: Ας αναλογιστούν τις ευθύνες τους. Να τις αναλάβουν. ‘Let them consider their responsibilities. Should they take them on.’

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 Setup for the Evaluation of the System

In this section, we are going to present the experiment that was performed for Greek and the modifications that were made to our parser in order to fulfill this task. The evaluation (see section 5) was made on the gold standard annotated corpus constructed in the framework of the PARSEME shared task on automatic identification of verbal multiword expressions (VMWEs) [16, 17]. We have focused on the annotation of Greek light verb (or support verb) constructions (παίρνω απόφαση ‘make a decision’). Based on the assumption that verbal collocations are formed by a light (or support) verb [18, 19, 20, 21], we first made a list of light verbs such as κάνω ‘make’, δίνω
‘give’, παίρνω ‘take’, έχω ‘have’ etc. Then, we proceeded to the necessary modifications to our system.

4.1 Implementation

As the parser already includes a collocation identification module and produces full syntactic trees for the constituents of the sentence, including the verbal constructions, we only had to develop a transformation code between the PARSEME and the parser’s input output formats. There were three kinds of transformation needed: (i) the reconstitution of the raw text from the tokenized one that was already provided (ii) the alignment of the provided tokens with the tokens generated by the parser and (iii) the copy of the parser detected VMWE to the tokenized parsemetsv file, i.e. the annotation of the identified VMWEs.

Raw Text. The parser requires raw text input. This led us to develop a pre-processor that reconstructs the original text from the tokenized data provided for the shared task. The pre-processor consisted in concatenating the tokens, taking into account the ns field indicating the presence or absence of a space character.

Tokens Alignment. The shared task evaluation measures being token-based, we had to produce the results using strictly the same tokenization as those given in the data sets. Although, the parsemetsv and the parser’s tokenization of words are identical in general, in numerous cases they differ. The trend in parsemetsv tokenization is to consider two words separated by a space as two different tokens. On the other hand, the parser’s tokenization procedure is based on linguistic criteria, i.e. a token is a significant lexical unit. Thus, the parser groups together two or more words if they form a complex lexical unit, for instance the Greek compound φακός επαφής ‘contact lens’, the preposition σύμφωνα με ‘according to’ or complex fixed adverbial phrases such as λίγο-πολύ ‘more or less’. The parsemetsv format exhibits some special treatment for some tokens, e.g. the contracted determiner στο ‘into’ in Greek that generates three lines of data.

Annotation of Collocations. The parser can produce several output formats: syntactic tree, tagger, XML/TEI, etc. We chose the tagger output because it gives all the necessary information for the annotation and, like in parsemetsv, it outputs one token per line. In short, each token is displayed on one line, divided in six columns: the token, the Universal POS tag, the richer tag, the lemma, the grammatical function / valency (if any), the collocation (if any). The annotation is processed sentence by sentence as follows: the parser’s output (aligned with the parsemetsv data file) is sequentially traversed line by line. For each verb token, the following tests are performed (in the following priority order). Note that in every case the annotations are fulfilled in the parsemetsv (aligned) data file:
— if the collocation is lemmatized in the lexicon and its main verb is listed as a light verb, it is flagged (353 constructions were annotated);
— if the verb is a light verb and the grammatical function displays a direct object or a prepositional object, it is flagged (427 constructions were annotated); the parser’s output is then traversed forward until the object is encountered; if the object is not encountered, a backward traversal is performed (in order to deal with the passive forms).

It should be noted that both the canonical and the inflected forms of the verbs and nouns are taken into account.

5 Evaluation and Results

In order to measure the performance of our parser, we used the test data file that contains the reference gold annotations against which system outputs were compared for evaluation and that was made available online after the evaluation phase was over. Evaluation metrics are precision, recall and F1, both strict (per VMWE) and fuzzy (per token, i.e. taking partial matches into account). The token-based F1 takes into account:

— discontinuities (λαμβάνω κάτι υπόψη ‘take something into account’);
— overlapping (έχω τη διαίσθηση και την ψευδαίσθηση ‘have the intuition and the illusion’);
— embeddings both at the syntactic level and at the level of lexicalized components.

However, VMWE categories (e.g., LVC, ID) were ignored by the evaluation metrics. In the evaluation per MWE, our system achieved 0.2913 precision (178/611) with a recall of 0.3560 (178/500) and F-measure of 0.3254. In the evaluation per token, our system achieved 0.4341 (520/1198) precision with a recall of 0.4266 (520/1219) and F-measure of 0.4303.

6 Conclusion

The performance achieved by the system confirms that deep syntactic information helps to identify MWEs and especially VMWEs. Although the VMWE annotation would be more accurate if it was based on the syntactic tree, the “flat” rich tagger output chosen for the alignment ease with the required parsemetsv tokenization was a good solution. An enhancement of this output would be to implement a token identification scheme so as to establish explicit links between the verbs and their arguments (instead of sequentially traverse the sentence and rely on the orthographic form of the word).

The system ranked first in the competition achieved 0.3612 precision, 0.4500 recall and 0.4007 F-measure.
References