On-line Multilingual Linguistic Services

Eric Wehrli LATL / University of Geneva
Yves Scherrer LATL / University of Geneva
Luka Nerima LATL / University of Geneva
{Eric.Wehrli, Yves.Scherrer, Luka.Nerima}@unige.ch

Abstract

In this demo, we present our free on-line multilingual linguistic services which allow to analyze sentences or to extract collocations from a corpus directly on-line, or by uploading a corpus. They are available for 8 European languages and can also be accessed as web services by programs.

1 Introduction

Linguistic information is useful for a wide-range of applications dealing with natural language. In a large number of cases, lexical disambiguation and part-of-speech (POS) assignment is all that is needed; in some other cases, additional information, such as phrase-structure representations or dependency structures, grammatical functions or multiword expressions may also prove useful.

To satisfy such needs, we have developed an on-line platform of linguistic services offering a multilingual parser/tagger for 8 European languages[1] (English, French, German, Greek, Italian, Portuguese, Romanian, Spanish), as well as a collocation extraction tool for the same languages. Those services can be freely accessed either directly on a dedicated webpage (`http://latlapps.unige.ch`), or (in the case of the parser/tagger) by programs interacting with the services (an example of a Python script is given below). While several open systems are available for POS-tagging and dependency parsing[2] or terminology extraction[3], their integration into an application requires some – sometimes non-trivial – computational competence. Furthermore, none of the parsers/taggers handles MWEs very satisfactorily, in particular when the two terms are distant from each other or in reverse order. Our tools, on the other hand, are specifically designed for users with no particular computational literacy. They do not require from the user any download, installation or adaptation if used on-line, and their integration in an application, using one the scripts described below is quite easy. Furthermore, by default, the parser handles collocations and other MWEs, as well as anaphora resolution (limited to 3rd person personal pronouns). When used in the tagger mode, it can be set to display grammatical functions and collocations (see below for details).

The following sections give a short description of the Fips parser, which is at the core of all the tools, some specific details and descriptions of the parser/tagger tool, and finally a description of the collocation extraction tool.

2 The Fips parser/tagger

The Fips multilingual parser (Wehrli, 2007; Wehrli & Nerima, 2015) is a grammar-based constituency parser using both attachment rules (to build phrase-structure representations) and specific procedures

[1] The parsing quality is not identical for all languages. The best results are achieved with English and French, then German, Spanish, Italian, then Portuguese and Greek, and finally Romanian.

[2] For instance, the Stanford parser (Klein & Manning, 2003; Chen & Manning, 2014), the MaltParser (Nivre et al. 2007), TreeTagger (Schmidt, 1995), Mate Tools (Bohnet et al., 2013), SyntaxNet (Andor et al, 2016), Marmot (Mueller et al, 2013).

[3] The Sketch engine (Kilgarriff et al., 2014), mwetoolkit (Ramisch, 2015).

Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations, pages 89–92, Osaka, Japan, December 11-17 2016.

to compute properties such as long-distance dependencies, argument-structure building, coordination structures, and so on. It uses an information-rich lexical database containing inflected words, lexemes and collocations.

The Fips parser/tagger is a powerful tool to analyse textual corpora. It can display results in several modes, ranging from phrase-structure representation (along the lines of Chomskyan generative grammar), to easier to read or to process part-of-speech representations, which can be optionally augmented with grammatical functions, dependency relations and collocations. By default, we use the universal tagset, but a richer tagset is also available, displaying number, gender, case, tense, modality, etc. Fips computes several analyses in parallel, but only the best analysis is dispalayed in the on-line service.

Figure 1, below shows a screenshot of the results returned by the Fips service for the short German example *Türkische Panzer rücken nach Syrien vor. 'Turkish tanks move forward towards Syria'* . In this example, Fips is selected with the Tagger output and rich POS tagset. The results show the words in column 1, the rich tags in column 2, the position of the first letter of each word with respect to the beginning of the sentence in column 3, the lexeme[4] in column 4. Column 5 displays the grammatical function associated with the syntactic head of each constituent (SU for subject, PO for prepositional object) and the argument structure of the predicate (the particle verb *vorrücken* 'move forward') with the grammatical function labels and the (semantic) head of each argument.

Türkische	ADJ-PLU-MAS-NOM-ACC	1	türkisch		
Panzer	NOM-PLU-MAS-NOM-ACC-GEN	11	Panzer	SU	
rücken	VER-IND-PRE-3-PLU	18	vor|rücken		SU:Panzer PO:Syrien
nach	PRE	25	nach	PO	
Syrien	NOM-SIN-NEU-DAT	30	Syrien		
vor	PART	37	vor		

Figure 1: Fips German analysis in Tagger mode

3 Collocation extraction

The collocation extraction tool is based on the Fips parser. In a nutshell, the input corpus is first parsed, sentence by sentence. For each parsed tree, all the word pairs in a given syntactic configuration (eg. adjective-noun, noun-noun, noun-preposition-noun, verb-object, subject-verb, etc.) are extracted as potential collocations and stored in a database. At the end of the process, the database is filtered by means of an association measure – by default log-likelihood (cf. Dunning, 1993)– and the results can be displayed[5]. As pointed out by Seretan (2011), the main advantage of this syntax-based method is (i) a much better precision than other systems and (ii) better recall with collocations likely to have the two terms separated by several words and/or in reverse order, such as verb-object, subject-verb or particle verb (for instance in German).

Figure 2 shows the web page for collocation extraction. The user selects a language and uploads the desired corpus, either in ANSI or UTF-8 format. Optionally the user can choose another association measure, a minimal score for association measure and the minimal number of occurrences. As the treatment of a large corpus can take several minutes or more, the user can also leave an e-mail address to receive a notification when processing is completed, along with the link to the results.

Figure 3 shows the results obtained by the extraction process on a small sample of the Europarl corpus (0.5 MB) for collocations of type verb-object. By clicking on a collocation type, the user will see all the occurrences of that collocation in the corpus.

[4]The lexeme associated with the word *rücken* is the particle verb *vorrücken* ("to move forward"). We inserted a vertical bar to make it explicit.

[5]See Seretan (2011) for a thorough description of the extraction method and comparison to other extraction tools.

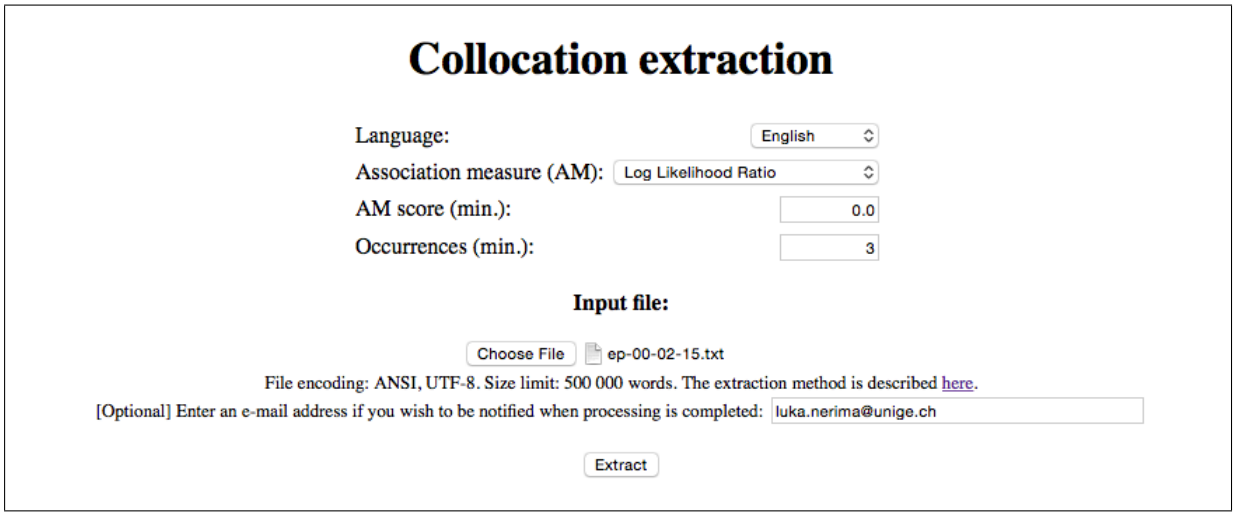

Figure 2: The Collocation extraction web page

Figure 3: Verb-object collocations

4 Accessing the online services programmatically

We provide both a Python and a PHP scripts to integrate the linguistic services into existing pipelines[6]. The Python script accesses the parser/tagger tool and provides the same parameters as the web version. Its usage is as follows:

```
python latlapps.py application language inputfilepath outputfilepath
```

where the *application* parameter accepts the same values as the web version. The *language* parameters specifies the language of the input data in the form of the two-letter ISO code. The third and fourth parameters specify the path to the file to be analyzed, and to the file to be created with the results of the analysis. Both files are expected to be in UTF-8 encoding. On Unix systems, these two parameters can be replaced by standard input and standard output pipes.

The script sends the input text line by line to the linguistic service. Therefore, it is important that each line corresponds to a linguistically meaningful entity such as a sentence or a paragraph. Figure 4 shows an example of the use of the Python script for our German sentence.

```
$ echo "Türkische Panzer rücken nach Syrien vor" | python latlapps.py Tagger de
```

Figure 4: Usage example of the latlapps.py script

The input sentence is communicated to the script by standard input, and the result –same as the one

[6]Both scripts are available on the site `http://latlapps.unige.ch`.

given in Figure 1 above– is written on standard output (the terminal). The application is 'Tagger' and the language code is 'de', which stands for German.

For PHP, two scripts are provided: one to be used from a command line with the same parameters as the Python script, while the second is designed to be used in an HTML file, as in the example below.

```
<form name="form1" id="form1" method="post" action="latlapps4html.php" >
```

The link to the script is done through the `action` attribute of the `form` tag. In the definition of the form, the application field name must be `ap` and the language field name must be `ln`.

References

Andor, D., Ch. Alberti, D. Weiss, A. Severyn, A. Presta, K. Ganchev, S. Petrov & M. Collins, 2016. "Globally Normalized Transition-Based Neural Networks", *Proceedings of ACL 2016*, 2442-2452.

Bonnet, B., J. Nivre, I. Boguslavsky, R. Farkas, F. Ginter & J. Hajic, 2013. "Joint Morphological and Syntactic Analysis for Richly Inflected Languages", *Proceedings of TACL* 1, 415-428.

Chen, D. and Ch. Manning, 2014. "A Fast and Accurate Dependency Parser using Neural Networks", *Proceedings of EMNLP 2014.*

Kilgarriff, A., V. Baisa, J. Busta, M. Jakubicek, V. Kovar, J. Michelfeit, P. Rychly, V. Suchomel, 2014. "The Sketch Engine: ten years on", *Lexicography*, vol. 1, Issue 1, 7-36.

Klein, D. & Ch. Manning, 2003. "Accurate Unlexicalized Parsing" *Proceeding of ACL 2014*, 423-430.

Mueller, T., H. Schmid & H. Schütze, 2013. "Efficient Higher-Order CRFs for Morphological Tagging", *Proceedings of EMNLP 2013*, 322-332.

Nivre, J., J. Hall, J. Nilsson, A Chanev, G. Eryigit, S. Kubler, S. Marinov & E. Marsi, 2007. "MaltParser: language-independent system for data-driven dependency parsing", *Natural Language Engineering* 13.2, 95-135.

Petrov, S., D. Das & R. McDonald, 2012. "A Universal Part-of-Speech Tagset", in *Proceeding of LREC 2012*, 2089-2096.

Ramisch, C. 2015. *Multiword Expressions Acquisition: A Generic and Open Framework*, Theory and Applications of Natural Language Processing, Springer.

Schmidt, H. 1995. "Improvements in Part-of-Speech Tagging with an Application to German", *Proceedings of the ACL SIGDAT Workshop*, Dublin.

Seretan, V. 2011. *Syntax-based Collocation Extraction*, Springer.

Wehrli, E. 2007. "Fips, a "deep" linguistic multilingual parser" in *Proceedings of the ACL 2007 Workshop on Deep Linguistic processing*, 120-127.

Wehrli, E. & L. Nerima, 2015. "The Fips Multilingual Parser", in N. Gala, R. Rapp, and G. Bel-Enguix (eds.), *Language Production, Cognition and The Lexicon.* Text, Speech and Language Technology 48, Springer, 473-490.