Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 7033–7037
Marseille, 11–16 May 2020

ReSiPC: a tool for complex searches in parallel corpora

Antoni Oliver, Bojana Mikelenić
Universitat Oberta de Cataluya, University of Zagreb
Barcelona (Catalonia-Spain), Zagreb (Croatia)
aoliverg@uoc.edu, bmikelen@ffzg.hr

Abstract

In this paper, a tool specifically designed to allow for complex searches in large parallel corpora is presented. The formalism for the queries is very powerful as it uses standard regular expressions that allow for complex queries combining word forms, lemmata and POS-tags. As queries are performed over POS-tags, at least one of the languages in the parallel corpus should be POS-tagged. Searches can be performed in one of the languages or in both languages at the same time. The program is able to POS-tag the corpora using the Freeling analyzer through its Python API. ReSiPC is developed in Python version 3 and it is distributed under a free license (GNU GPL). The tool can be used to provide data for contrastive linguistics research and an example of use in a Spanish-Croatian parallel corpus is presented. ReSiPC is designed for queries in POS-tagged corpora, but it can be easily adapted for querying corpora containing other kinds of information.

Keywords: parallel corpora, regular expressions, contrastive linguistics

1. Introduction

Bilingual or multilingual corpora can be parallel (also known as translation corpora) or comparable, where the former contain texts in one language and their translations into one or more others and the latter contain original texts in two or more languages. For this paper, we will be focusing only on parallel corpora. Johannson (2007, 301) lists some of the areas where multilingual corpora are used: contrastive linguistics, language typology, translation studies, including translator education, bilingual lexicography, foreign-language teaching and natural language processing, including machine translation. For the latter, parallel corpora are often used to train machine translation models, statistical (Koehn et al., 2007) or neural (Bahdanau et al., 2015) and also for the purposes of bilingual terminology extraction (Lefever et al., 2009). Here, we are especially interested in their application in contrastive linguistics.

Contrastive linguistics or contrastive analysis is a comparison of two or more languages in order to show structural similarities and differences between them. The contrastive approach has a long tradition in Croatian linguistics, dating from the works of Rudolf Filipović, most importantly his "Yugoslav Serbo-Croatian – English Contrastive Project" started in 1968 at the Institute of Linguistics of the University of Zagreb. For the purposes of this project and others to follow, one of the first parallel corpora ever was created, the so-called Zagreb Version of the Brown Corpus. In Filipović (1969), it is explained how the Brown Corpus was shortened, translated into Serbo-Croatian and grammatically tagged. This was done because the importance of translation for isolation of structures to be contrasted was observed from the very beginning (Filipović, 1985, 23-4). Considering the use of corpora in contrastive studies, Johannson (2007, 1) agrees, claiming that multilingual corpora are especially useful for observing how languages differ and what they share.

Corpora can be annotated for different information (e.g. morphological, syntactic, semantic, etc.) depending on the task they are used for. In this paper, we are interested in parallel corpora that contain POS-tagged segments for at least one of the languages, which allows for the extraction of specific morphosyntactic structures in one language, together with their translations. In this way, the structures we want to analyze and contrast can be easily identified automatically. For this purpose, we require an appropriate tool that would enable complex searches based on these POS-tags.

2. Similar tools

There are several tools available for corpus linguistic research.

- Antconc[1] (Anthony, 2013)
- CLaRK System[2] (Simov et al., 2001)
- #LancsBox (*Lancaster University corpus toolbox*)[3] (Brezina et al., 2015)
- WordSmith Tools[4] (Wilkinson, 2011)
- WordStatix[5]
- Coquery[6]
- DART[7] (Weisser, 2016)
- Sketch Engine[8] (Kilgarriff et al., 2014)
- NoSketch Engine[9] (Kocincová et al., 2015)

In table 1 several features of ReSiPC and similar programs are summarized. Namely, we show the following features:

[1] http://www.laurenceanthony.net/software/antconc/
[2] http://bultreebank.org/bg/clark/
[3] http://corpora.lancs.ac.uk/lancsbox/
[4] https://lexically.net/wordsmith/
[5] https://sites.google.com/site/wordstatix/
[6] https://www.coquery.org/
[7] http://martinweisser.org/ling_soft.html
[8] https://www.sketchengine.eu/
[9] https://nlp.fi.muni.cz/trac/noske

	Parallel corpora	Tagged corpora	Taggeer	Reg. exp.	n-grams	Concordances	Annotations	GUI	Free to use	Free license	Multiplatform
ReSiPC	X	X	X	X					X	X	X
AntConc		X			X	X		X	X		X
CLaRK System		X		X			X	X	X		X
#LancsBox		X	X	X	X	X		X	X		X
WordSmith Tool						X		X			
WordStatix						X		X	X	X	X
Coquery		X	X					X	X	X	X
DART		X	X	X	X	X	X	X	X	X	X
Sketch Engine	X	X		X	X	X	X	X	X		X
NoSketch Engine	X	X		X		X	X	X	X	X	X

Table 1: Comparison of ReSiPC features with other similar programs

- Parallel corpora: the tool includes support for queries over parallel corpora.
- Tagged corpora: the tool allows for queries in tagged corpora using word forms, lemmata or POS-tags.
- Tagger: the tool includes a tagger or it can automatically access an external tagger.
- Reg. exp: searches can be performed using standard regular expressions.
- n-grams: allows for n-gram calculation and performs some actions or statistical analysis over the n-grams.
- Concordance: allows for the performance of concordance searches.
- Annotations: includes some functionalities allowing for manual annotation of the corpora.
- GUI: the tool has a graphic user interface.
- Free to use: the tool can be freely used, at least for academic use.
- Free licence: the tool holds a free licence.
- Mutliplatform: the tool can be used on several operating systems (usually Windows, Linux and Mac).

There are still more corpora tools available. Most of the available tools provide limited or no support for parallel corpora. ReSiPC is designed bearing four ideas in mind: ease of use (despite the fact that for the moment no GUI interface is available), use of standard formats for the corpora, full support for parallel and POS-tagged corpora and the use of powerful standard regular expressions for queries.

3. Tool description

The tool is released under a free license (GNU-GPL) and can be freely downloaded from its SourceForge page [10].

3.1. Interface

The current version of ReSiPC works in a terminal and no visual user interface is available. We plan to develop an easy-to-use visual interface in the near future.

3.2. Accepted formats for parallel corpora

ReSiPC accepts the following parallel corpus formats:

- Moses format: separated files for each language where segments are aligned line by line.
- TSV (*Tab Separated Values*): a text file where each element is separated by a tabulator. The first field should be the source language segment, the second one the target language segment, the third field the POS-tagged source language segment, and the fourth, which is optional, the POS-tagged target language segment.
- TMX (*Translation Memory eXchange*): an XML-based format widely used for storing and sharing translation memories.

More than one file can be loaded and processed at the same time.

3.3. POS-tagging

Our tool can connect with Freeling[11] (Padró and Stanilovsky, 2012) to perform POS-tagging. This analyzer is available for the following languages: Asturian, Catalan, English, French, German, Galician, Italian, Norwegian, Portuguese, Spanish, Russian, Slovene and Welsh. Freeling can be used both directly through an API connection or importing the tagged corpus into ReSiPC. Other POS-taggers can also be used[12]. When importing tagged corpora, the output format should be converted. The required format for ReSiPC is one sentence per line, with each token containing a word form, lemma and POS-tag separated by a vertical bar (|), as it can be observed in figure 1. A set of scripts for converting the output of widely used POS-taggers is provided in the ReSiPC distribution. Namely, scripts for the following tagggers are provided: Stanford POS tagger[13] (Toutanova and Manning, 2000), TreeTagger[14] (Schmid, 2013) and Freeling.
The connection with Freeling can be done in two ways:

[10] https://sourceforge.net/projects/resipc/

[11] http://nlp.lsi.upc.edu/freeling

[12] A comprehensive list of taggers can be found at https://nlp.stanford.edu/links/statnlp.html#Taggers

[13] https://nlp.stanford.edu/software/tagger.shtml

[14] https://www.cis.uni-muenchen.de/ schmid/tools/TreeTagger/

- Using the Freeling API. Before using this option, the API should be installed and configured on the user's computer. Once this is done, the tagging can be performed using the following instructions (a Spanish corpus is used in the example):[15]

  ```
  python3 ReSiPC.py --st source-spa.txt
  --tag_freeling_api /path/to/freeling/api
  --output tagged-spa.txt
  --freeling_api_lang="es"
  ```

- Using a Freeling server that can be running on the user's computer or on a remote server. To tag the corpus, the following command should be used:[16]

  ```
  python3 ReSiPC.py --st original-spa.txt
  --tag_freeling_server 6543 --output
  etiquetado-spa.txt
  ```

After this command, we will have the original Spanish corpus both as a plain text and POS-tagged. We also have the target language corpus (Croatian in our example). We can see the format in figure 1.
As it can be seen in the example, the POS-tagged tokens are comprised of 3 fields separated by a vertical bar (|): word form, lemma and POS-tag. Freeling uses the EAGLE's[17] POS-tags, that can be found in the Freeling documentation[18]. As already mentioned, it is possible to use other analyzers and different POS-tags, provided we are able to adapt the output of the analyzer to the format required by ReSiPC.

3.4. Searches in the tagged corpora

Our program allows searching in the corpus for segments containing a given morphosyntactic pattern that will be looked up in the POS-tagged version of the source or target segment (in case it is also POS-tagged). The formalism allows for the specification of sequences of word forms, lemmata or POS-tags. For example, the pattern in (1) would select all the segments whose Spanish version contains a verb followed by a preposition followed by a noun.

```
(1)  ||V.* ||SP.* ||N.*
```

As we can see from the example, standard regular expressions can be used in the patterns, making this formalism very powerful.
Exact word forms or lemmata can also be specified in the patterns, as in (2) where the patterns will select all the segments that in Spanish have the verb *incitar* (in any form, as it is specified as a lemma, *to incite*), followed by an *a* and followed by a verb in the infinitive form.

```
(2)  |incitar| |a| ||VMN.*
```

The formalism also allows specifying a series of undefined elements between two defined elements. The expression in (3) would select any form of the verb *proporcionar* (*to provide*) followed by an *a* allowing from 0 to 5 undefined tokens between them. This expression would detect, for example, the following cases: *..proporcionó a..., ...proporcionó ayuda a..., ...proporcionó mucha ayuda a..., ...proporcionó una gran ayuda a. . . , ...proporcionó una ayuda muy valiosa a. . . , proporcionó una ayuda realmente muy valiosa a....*

```
(3)  |proporcionar| *{0,5} |a|
```

The regular expressions should be stored in a text file, with one element in each line (let's name this file patterns.txt in the examples, but any other name can be used). Provided we have our corpora in a separate file (source-spa.txt, target-hrv.txt and postagged-spa.txt) and that we want the results written in the file results.txt, we should use the following command:

```
python3 ReSiPC.py --st source-spa.txt
--tt target-hrv.txt --tst postagged-spa.txt
--patterns patterns.txt --output results.txt
```

If we work with a tab separated text file for the corpus (corpustab.txt in the example), we should use the following command:

```
python ReSiPC.py --tsv corpustab.txt
--patterns patterns.txt --output results.txt
```

We can use several corpora files together, separating them with a ":", as in the following example:

```
--tsv corpus1.txt:corpus2.txt:corpus3.txt
```

3.5. Output format

The program saves the results in a text file where the source and target language segments satisfying one of the patterns are shown, along with the matching pattern and the name of the corpus file, as shown in the example:

```
|pensar| *{0,5} |en|SP.*
corpus1-spa.txt |La verdad es que yo no
pienso mucho en eso |dije.
"Ja zapravo ne razmišljam mnogo o tome",
rekao sam.
```

The results can be saved, grouped and sorted by several criteria, for example, by corpus file name or pattern. We can also set a maximum number of results for each pattern in each corpus file.

4. Example of use: structures with the prepositional object in Spanish and their translation to Croatian

Although this tool allows for general searches of, for example, all verbs or all prepositions in a corpus, it is also possible to search for specific words or structures, as mentioned above. In the example described here, we have searched for combinations of specific verbs and prepositions in Spanish. Our interest was in the prepositional phrases (PPs) that

[15] The name of the files and the language code should be adapted to the specific situation.

[16] In this example the server is running on the user's computer. To use a server with IP 213.74.32.71 and port 8798 the following options should be added: –tag_freeling_server 213.73.32.71:8798

[17] http://www.ilc.cnr.it/EAGLES96/annotate/node9.html

[18] https://talp-upc.gitbooks.io/freeling-4-1-user-manual/content/tagsets/tagset-es.html

Mi padre y yo vivíamos en un pequeño piso de la calle Santa Ana, junto a la plaz a de la iglesia.
Mi|mi|DP1CSS padre|padre|NCMS000 y|y|CC yo|yo|PP1CSN0 vivíamos|vivir|VMII1P0 en|en|SP un|uno|DI0MS0 pequeño|pequeño|AQ0MS00 piso|piso|NC MS000 de|de|SP la|el|DA0FS0 calle|calle|NCFS000 Santa_Ana|santa_ana|NP00000 ,|,| Fc junto_a|junto_a|SP la|el|DA0FS0 plaza|plaza|NCFS000 de|de|SP la|el|DA0FS0 iglesia|iglesia|NCFS000 .|.|Fp Otac i ja živjeli smo u malenom stanu u Ulici svete Ane, odmah do crkvenog trga.

Figure 1: Corpus format example

(4) |acostumbrar| *0,5 |a|SP.* Me **acostumbré a** esperar el anochecer en el caserón vacío, al calor de su compañía invisible. Navikao sam se čekati sumrak u toj praznoj kućerini, u toplini njihove nevidljive nazočnosti.
(5) |acostumbrar| *0,5 |a|SP.* Pero bueno, ya empiezo a **acostumbrar**me **a** este baile de máscaras en el que los desplazados han convertido el pasado. Ali, dobro, počinjem se privikavati na ovaj ples pod maskama u kojem vremenski putnici mijenjaju prošlost.
(6) |acostumbrar| *0,5 |a|SP.* Se **acostumbró** entonces **a** reverenciar al canasto cada noche, dedicándole una larga y afectuosa mirada, y acariciando con sus dedos el firme trenzado de sus mimbres, un sencillo ritual que aún seguía practicando a escondidas de Jane, y que bastaba para inflamar su espíritu hasta el punto de hacerle sentir invencible, poderoso, capaz de cruzar el Atlántico a nado o de vencer a un tigre con sus propias manos. U to je doba stekao naviku svake večeri iskazivati štovanje košari, posvećujući joj dug i nježan pogled i prstima prelazeći preko čvrstog spleta pruća, u iskrenom obredu što ga je još uvijek obavljao pazeći da ga Jane ne vidi, i koji je bio dovoljan da mu uspali duh do te mjere da se osjeti nepobjedivim, moćnim, sposobnim preplivati Atlantski ocean ili vlastitim rukama savladati tigra.

Figure 2: Example of use: structures with the prepositional object in Spanish and their translation to Croatian

those prepositions make a part of, especially in the cases when the PPs are analysed as prepositional objects. The data obtained can then be used for a contrastive analysis with equivalent structures in Croatian. The search was conducted on a Spanish-Croatian parallel corpus, where only the Spanish subcorpus was POS-tagged, even though POS-tagging of the Croatian subcorpus will soon be available[19]. Because of that, in this example the Croatian part of the analysis has to be conducted manually.

Prepositional object or complement in Spanish (spa. *complemento de régimen preposicional* – CR) is identified as the third type of object, alongside the direct and the indirect object. It is a verbal prepositional argument which the verb selects semantically (R.A.E., 2009). Its preposition is governed by the verb, where some verbs govern only one preposition (e.g. *creer en algo – to believe in something*), while others can govern more than one (e.g. *hablar de algo*, *hablar sobre algo – to talk about something*). The most frequent governed prepositions are the most frequent and polysemous ones in general, namely *a*, *en*, *de* and *con*. In these structures, they are mostly grammaticalized and some of the functions they adopt can be related to those of the case suffixes in languages that retain them (e.g. Croatian). Because a specific verb governs one or more specific prepositions, these structures can be easily searched for in a Spanish corpus that is POS-tagged. In addition, these governed prepositions are usually positioned right after the verb they are connected to (in the first position to the right), even though they can appear before (to the left) of the verb and further to the right. Depending on the size of the corpus and the research we wish to conduct, a search of a specific verb lemma combined with a specific preposition in the first position to the right of it can be enough to obtain the necessary data, or we may need to widen the search. In this example, we will search for the verb lemma and the preposition which can appear up to six places to the right of it (i.e. there can be from 0 to 5 undefined tokens between them). The examples shown here are of the reflexive verb *acostumbrarse* (*to get used to*) that governs the preposition *a*. Reflexive verbs have an obligatory reflexive pronoun *se* (with other forms: *me*, *te*, *nos*, *os*), which can appear before or after the verb and is tagged separately, so that is why it is important to allow for the preposition to be further away from the verb.

In figure 2 (4) and figure 2 (6) the reflexive pronoun comes before the verb (*me*, *se*), and in figure 2 (5) after (*me*). The preposition *a* in (4) is in the first position to the right of the verb (*Me acostumbré a*), in (5) in the second position to the right, behind the reflexive pronoun (*acostumbrarme a*) and in (6) also in the second position to the right, behind the adverb *entonces* (*then*). We can also observe the Croatian equivalents of the structure *acostumbrarse + CR(a)*: (4) *naviknuti se + infinitive* (*Navikao sam se čekati*), (5) *privikavati se + na + Accusative* (*se privikavati na ovaj ples*), (6) *steći naviku + infinitive* (*je (...) stekao naviku (...) iskazivati*). For now, the identification of these structures in Croatian is done manually.

5. Future development

For the moment, ReSiPC can only be used in a terminal, as no user interface is available. We plan to develop a simple visual user interface, allowing the use of the most common options in an easy-to-use way. This interface will be intended for those users not keen on using the terminal.

[19] For more information about this corpus, see: Mikelenić and Tadić, in this same publication.

6. Conclusions

In this paper we have presented ReSiPC, a tool allowing for complex queries in parallel corpora with POS-tag information for at least one of the languages. This tool can be very useful for contrastive linguistic research. In the paper we have also shown a practical use case in a contrastive analysis involving Spanish and Croatian.

7. Bibliographical References

Anthony, L. (2013). Developing AntConc for a new generation of corpus linguists. In *Proceedings of the Corpus Linguistics conference (CL 2013)*, pages 14–16.

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations, ICLR*.

Brezina, V., McEnery, T., and Wattam, S. (2015). Collocations in context: A new perspective on collocation networks. *International Journal of Corpus Linguistics*, 20(2):139–173.

Filipović, R. (1969). The Choice of the Corpus for the Contrastive Anayisis of Serbo-Croatian and English. In Rudolf Filipović, editor, *YSCECP, B. Studies 1*, pages 37–46. Institute of Linguisics, Zagreb (Croatia).

Filipović, R. (1985). The Yugoslav Serbo-Croatian – English Contrastive Project: Theoretical and Methodological Considerations. In Rudolf Filipović, editor, *Chapters in Serbo-Croatian English Contrastive Gramma*, pages 9–36. Zagreb University Press – Liber, Zagreb (Croatia).

Johannson, S. (2007). *Seeing through Multilingual Corpora: On the use of corpora in contrastive studies*. John Benjamins Publishing Company.

Kilgarriff, A., Baisa, V., Bušta, J., Jakubíček, M., Kovář, V., Michelfeit, J., Rychlỳ, P., and Suchomel, V. (2014). The Sketch Engine: ten years on. *Lexicography*, 1(1):7–36.

Kocincová, L., Baisa, V., et al. (2015). Interactive visualizations of corpus data in Sketch Engine. In *Proceedings of the Workshop on Innovative Corpus Query and Visualization Tools at NODALIDA 2015, May 11-13, 2015, Vilnius, Lithuania*, number 111, pages 17–22. Linköping University Electronic Press.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the association for computational linguistics companion volume proceedings of the demo and poster sessions*, pages 177–180.

Lefever, E., Macken, L., and Hoste, V. (2009). Language-independent bilingual terminology extraction from a multilingual parallel corpus. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 496–504.

Padró, L. and Stanilovsky, E. (2012). Freeling 3.0: Towards wider multilinguality. In *LREC2012*.

R.A.E. (2009). *Nueva gramática de la lengua española*, volume 2. Espasa Libros.

Schmid, H. (2013). Probabilistic part-ofispeech tagging using decision trees. In *New methods in language processing*, page 154.

Simov, K., Peev, Z., Kouylekov, M., Simov, A., Dimitrov, M., and Kiryakov, A. (2001). CLaRK-an XML-based system for corpora development. In *Proc. of the Corpus Linguistics 2001 Conference*, pages 558–560.

Toutanova, K. and Manning, C. D. (2000). Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 13*, pages 63–70. Association for Computational Linguistics.

Weisser, M. (2016). DART–The dialogue annotation and research tool. *Corpus Linguistics and Linguistic Theory*, 12(2):355–388.

Wilkinson, M. (2011). WordSmith Tools: The best corpus analysis program for translators? *Translation Journal*, 15(3).