STRING: An Hybrid Statistical and Rule-Based Natural Language Processing Chain for Portuguese

Nuno Mamede[1,2], Jorge Baptista[3,2], Cláudio Diniz[2], and Vera Cabarrão[2]

[1] Instituto Superior Técnico, Universidade Técnica de Lisboa / Lisboa, Portugal
[2] Spoken Language Lab, INESC-ID Lisboa / Lisboa, Portugal
[3] Universidade do Algarve - CECL / Faro, Portugal
Nuno.Mamede@l2f.inesc-id.pt, cdiniz@l2f.inesc-id.pt jbaptis@ualg.pt
veracabarrao@gmail.com

Abstract. STRING is an hybrid statistical and rule-based natural language processing chain for Portuguese. STRING has a modular structure and performs all basic text processing tasks, namely tokenization and text segmentation, part-of-speech tagging, morphosyntactic disambiguation, shallow parsing (chunking) and deep parsing (dependency extraction). STRING performs Named Entity Recognition, Information Retrieval, Anaphora Resolution and other NLP tasks. A web-interface makes STRING available to the community and the general public.

Keywords: Natural Language Processing, Portuguese, Statistical and Rule-based POS disambiguation, Parsing

STRING is an hybrid statistical and rule-based natural language processing (NLP) chain for Portuguese, that has been developed by L2F-Spoken Language Laboratory, at INESC-ID Lisboa. STRING has modular structure and performs all the basic NLP tasks in four steps: (1) *Preprocessing*, (2) *Lexical analysis*, (3) statistical and rule-based POS *Disambiguation* and (4) *Parsing*.

In the *Preprocessing* stage, the system tokenizes the input text, performs a text segmentation into sentences, and applies lexical resources to the tokens adding them all possible part-of-speech (POS) tags. The tokenization module identifies words (both simple and compound); numbers and numerals (both simple and compound; cardinal, ordinal, or fractional; and roman numerals); the most common abbreviations, e-mail, IP and URL addresses; punctuation and other symbols.

The next module is the lexical analyser LexMan [1]. This is responsible for according to each token its part-of-speech and any other relevant morphosyntactic features (gender, number, tense, mood, case, degree, etc.). The rich tag set has a high granularity featuring 12 POS categories and 11 fields.

The system then proceeds with RuDriCo2 [2],[3], a rule-driven converter. This module is responsible for the word-splitting (i.e. solving contractions, (*e.g.*

comigo = *com*/Prep + *eu*/Pron). It also applies a considerably large set of disambiguation rules; finally, it identifies many unambiguous compound words and joins them in a single token. This new version RuDriCo2 is significantly (10 times) faster that the previous version, uses a more expressive language (allowing negation and disjunction, the use of regular expressions both in the lemma and in the surface form) and constitutes an approach to the XIP parser syntax (see below). RuDriCo2 also validates the input data, displays error messages, warns for potential problems.

Then, the statistical part-of-speech disambiguator Marv3 takes over. Its function is to choose the most likely POS tag for each word, using the Viterbi algorithm. The language model used by Marv 3 is trained on a 250K Portuguese corpus originally produced under projects. The current implementation of Marv3 features over 97% precision and it is significantly (9 times) faster that the previous version . Furthermore, it does not discard rejected tags and uses the same DTD than RuDriCo.

The last module of STRING is the XIP parser[4], a finite-state incremental parser developed by XeroxRCE[4] that uses a Portuguese rule-base grammar, initially developed in collaboration with Xerox, since 2004. XIP makes use of a rich set of lexical resources, which add linguistic (syntactic and semantic) information to the output of the POS tagger. XIP parses the text by dividing it into chunks, that is, elementary phrases such as `NP`, `PP`, and identifies their heads. Then, it extracts the syntactic relations between the heads of those chunks. These dependencies correspond to the major deep parsing relations of *Subject*, *Direct Object*, *Modifier*, etc., but they also include auxiliary dependencies between different chunks and words, such as those necessary to link verbal chains formed of strings of auxiliaries[5].

Two external modules use the STRING output for: (i) anaphora resolution; and (ii) temporal expressions normalization.

Since its initial assembly in 2007, the STRING NLP chain has been subject to continuous improvement in several of its modules, and particularly the conversion between them, yielding a 4 ms/word debit. Using the L2F 100 CPU GRID, it is now possible to process the entire CetemPúblico under 7 hours.

Currently, STRING is used to: selection of stems for cloze question generation in the REAP.PT computer-assisted language learning platform[5], based on linguistic analysis[6]; multiword indentification[7]; Named Entity Recognition[6] [8][9]; the identification, classification and normalization of temporal expressions [10][11]; and the extraction of relations between named entities

Current, on-going and future work with STRING include: the slot-filling, relation extraction and events identification; the development of a statistical-based dependency post-parser to improve the XIP dependency extraction. Recently, a web interface/service has been created to provide access to the general public[7].

[4] http://www.xrce.xerox.com/ (last access: 01/03/2012)
[5] https://www.l2f.inesc-id.pt/wiki/index.php/REAP.PT (last access: 01/03/2012)
[6] In use by OOBIAN: http://www.oobian.com/ (last access: 01/03/2012).
[7] https://string.l2f.inesc-id.pt/demo/ (last access: 01/03/2012).

The demo at PROPOR will use this web interface to present STRING to the Portuguese NLP community.

Acknowledgments. The authors wish to thank Caroline Hagège (XEROX RCE, Grenoble, France) for her invaluable collaboration at the onset of the construction of the Portuguese Grammar for XIP; and to Ricardo Ribeiro, Fernando Batista and Joana Pardal (L2F/INESC-ID Lisboa) for their contribution to the initial modules of the processing chain. A final word or recognition to all L2F collaborators that helped to build STRING (in alphabetic order): Andreia Maurício, Daniel Santos, David Rodrigues, Diogo Oliveira, João Loureiro, Luís Romão, Nuno Nobre, Ricardo Portela. This work was partially supported by FCT (INESC-ID multi-annual funding), through the PIDDAC Program, the FCT project REAP.PT (Proj.Ref. CMU-PT/HuMach/0053/2008).

References

1. Diniz, C., Mamede, N.: Lexman - lexical morphological analyser. Technical report, L2F / INESC ID Lisboa, Lisboa (2011)
2. Diniz, C.: Rudrico2 - um conversor baseado em regras de transformação declarativas. Master's thesis, Instituto Superior Técnico, Universidade Técnica de Lisboa (2010)
3. Diniz, C., Mamede, N., ao Dias Pereira, J.: RuDriCo2 - a faster disambiguator and segmentation modifier. In: Simpósio de Informática - INForum, Universidade do Minho, Portugal (2010) 573–584
4. Ait-Mokhtar, S., Chanod, J., Roux, C.: Robustness behyond shalowness: incremental dependency parsing. Natural Language Engineering **8**(2/3) (2002) 121–144
5. Baptista, J., Mamede, N., Gomes, F.: Auxiliary verbs and verbal chains in european portuguese. In: Computational Processing of the Portuguese Language. Number LNCS/LNAI 6001, Berlin, PROPOR 2010, Springer (2010)
6. Correia, R., Baptista, J., Mamede, N., Trancoso, I., Eskenazi, M.: Automatic generation of cloze question distractors. In: SLaTE 2010, Waseda University, Tokyo, Japan (September 22, 2010), Interspeech 2010/ISCA SIG on Speech and Language Technology in Education (2010)
7. Portela, R., Mamede, N., Baptista, J.: Multiword identification. In: INForum, III Simpósio de Informática, Coimbra, Portugal (2011) 110–119
8. Hagège, C., Baptista, J., Mamede, N.J.a. In: Reconhecimento de entidades mencionadas com o XIP: Uma colaboração entre o INESC-L2F e a Xerox. Volume Desafios na avaliação conjunta do reconhecimento de entidades mencionadas: Actas do Encontro do Segundo HAREM (Aveiro, 11 de Setembro de 2008). Linguateca (2009)
9. Oliveira, D.: Extraction and classification of named entities. Master's thesis, Instituto Superior Técnico, Universidade Técnica de Lisboa (2010) MSc Dissertation.
10. Hagège, C., Baptista, J., Mamede, N.: Portuguese temporal expressions recognition: from te characterization to an effective ter module implementation. In: STIL'2009. 7th Brazilian Symposium in Information and Human Language Technology, NILC-CMSC/USP, São Carlos, Brasil (2009)
11. Hagège, Caroline; Baptista, J., Mamede, N.: Caracterização e processamento de expressões temporais em português. Linguamática **2**(1) (2010) 63–76