Out-of-the-Box Robust Parsing of Portuguese

João Silva, António Branco, Sérgio Castro, and Ruben Reis

University of Lisbon
{jsilva,antonio.branco,sergio.castro,ruben.reis}@di.fc.ul.pt

Abstract In this paper we assess to what extent the available Portuguese treebanks and available probabilistic parsers are suitable for out-of-the-box robust parsing of Portuguese. We also announce the release of the best parser coming out of this exercise, which is, to the best of our knowledge, the first robust parser widely available for Portuguese.

Key words: parsing, probabilistic, robust, out-of-the-box, Portuguese

1 Introduction

The task of robust parsing seeks to address one of the well-known data acquisition bottlenecks in the field of natural language processing.

To its input, typically sentences, a parser associates the corresponding grammatical analysis or representation. Due to the inherent incompleteness of their lexica and grammar rules, hand-coded rule-based parsers are often brittle and eventually fail to deliver any outcome to input sentences whose lexical items are missing in its lexicon or with syntactic constructions not covered by its grammar rules. Robust parsers however are specifically designed to deliver an outcome to any input, even though at the cost of performing at suboptimal accuracy level.

In the last two decades, the field of robust parsing has undergone substantial research effort and progress. The basic technology of probabilistic context free grammars was researched and expanded to a point where the state-of-the-art is consistently scoring in the window 85–90% [1, p. 440] for the most successful solutions—obtained for the most studied natural language, English, with the largest and most widely used dataset, the Penn Treebank, and for its basic task, labeled syntactic constituency analysis.

This research effort has supported the development of a number of packages that implement language-independent approaches, have public releases, and allow to train top performing parsers. Concomitantly, there have been increased efforts to construct treebanks for languages other than English, many of them widely available with public distribution.

These circumstances have permitted to release out-of-the-box, top performing robust parsers for those languages for which such datasets were developed. And this observation brings us to the central issue motivating the present paper.

Though there have been treebanks developed for Portuguese and widely available, to the best of our knowledge, no encompassing and thorough research exercise has been performed to assess whether out-of-the-box, top performing robust parsers for Portuguese can be obtained, and a fortiori no such parser, if possible, has been widely released so far. The present paper is thus guided by the objective of gathering answers to the following leading questions:

Out-of-the-box? To what extent do the software packages currently available, that permit to train a robust parser out of a treebank, are language independent and support a smooth application to Portuguese data? Given their design features, to what extent do the available treebanks permit a smooth application of these packages and the development of an out-of-the-box robust parser?

Top-performing? What range of performing scores is attainable with such parsers? To what extent, if any, does the circumstance that they are out-of-the-box parsers, working in a stand-alone mode with no pre- or post-processing auxiliary tools, affect their chance to attain state-of-the-art performance?

On a par with these objectives, another important goal of this paper is to describe the characteristics and announce the public distribution of the best performing parser resulting from the present research exercise.

In Section 2, we indicate the datasets available for this exercise and their features, and assess their suitability to support the development of out-of-the-box parsers. The software packages available and their suitability are described and assessed in Section 3. In Section 4, we report on the impact on the performance of the parsers that results from the fact that they are used as they come out of their training over the dataset, without resorting to any pre- or post-processing auxiliary tool. Finally, Section 5 is devoted to concluding remarks.

2 Datasets

To the best of our knowledge, there are two treebanks of contemporary Portuguese available, Bosque and CINTIL Treebank. In this section, we ponder their suitability for training out-of-the-box parsers.

2.1 Bosque

Bosque is a treebank of newspaper articles (Brazilian and European Portuguese) that has been automatically annotated by the PALAVRAS parser and subsequently manually revised.[1]

Syntactic heads are explicitly marked, as well as arguments and other modifiers (with indication on whether they are pre- or post-modifiers). Trees are also annotated with tags for syntactic functions at the phrase level. Named entities and closed class multi-word expressions appear as one syntactic unit as their parts are concatenated (separated by underscore) into a single token. A sample, taken from [2], is shown in Figure 1.

In [2], the authors report on the use of Bikel's package [3] to train a parser over Bosque. Most of that paper actually consists in the description of the many difficulties that the authors need to cope with when adapting the tree format of Bosque to a format suited for training the parser.

[1] Distribution at `http://www.linguateca.pt`.

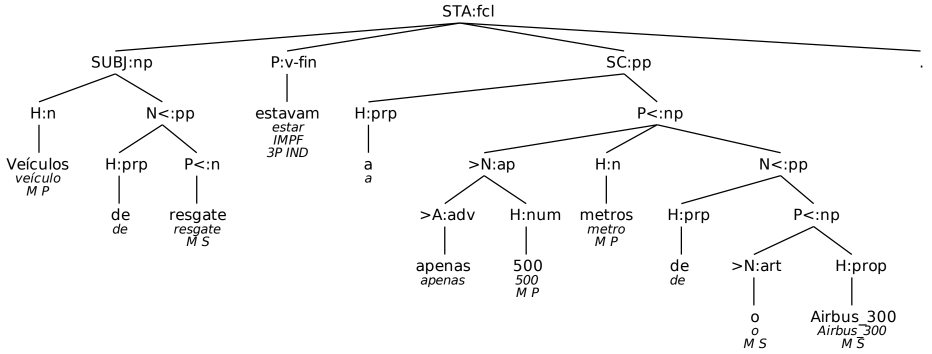

Figure 1. A sample of Bosque

A dataset of 1,877 sentences was held-out for testing and the remaining 7,497 sentences were used for development. An out-of-the-box Bikel parser achieved a PARSEVAL f-score[2] of 36.3%, which can be taken as a baseline. Another, improved parser was then obtained by refining the training parameters while running 10-fold cross-validation tests over the development dataset. This parser was extended with Portuguese-specific head-finding rules, and with inflectional and derivational features for classifying unknown Portuguese words. Additionally, the annotation of Bosque was enriched to support a better performance of the parser. This improved parser achieved a PARSEVAL f-score of 63.2%.

2.2 CINTIL Treebank

The CINTIL treebank was produced from the output of LXGram, a deep linguistic processing grammar [4] by manually selecting the correct parse for a sentence from among all the possible parses that are delivered by the grammar [5].[3]

Like Bosque, constituents are marked with syntactic functions at the phrase level. Although heads are not explicitly indicated by a tag, they can be directly picked from the X-bar syntactic structure, which this treebank adheres to. Named entities and multi-word expressions have their internal constituency explicitly represented. For instance, `Hong Kong` is represented as `(N' (N Hong) (N Kong))`. In addition, it is possible to access a variety of morphological information, like the lemma of the words. A sample is shown in Figure 2.

As there were no published results on training a parser over CINTIL, we ran an experiment to assess its suitability for the exercise reported in this paper.

In its current version, CINTIL has 1,204 sentences, mostly from newspaper articles (European Portuguese). Since the sentences are already represented in the *de facto* standard Penn Treebank format, there was no need to adapt it for training with Bikel's parser, which is the one we chose to ease comparison with the results from [2]. However, keeping in mind that our interest is in testing to what extent the parser can be used out-of-the-box, we did not adapt the parser for Portuguese.

[2] See Section 3.5 for more details on evaluation metrics.

[3] Distribution at `http://nlx.di.fc.ul.pt/lxgram/`.

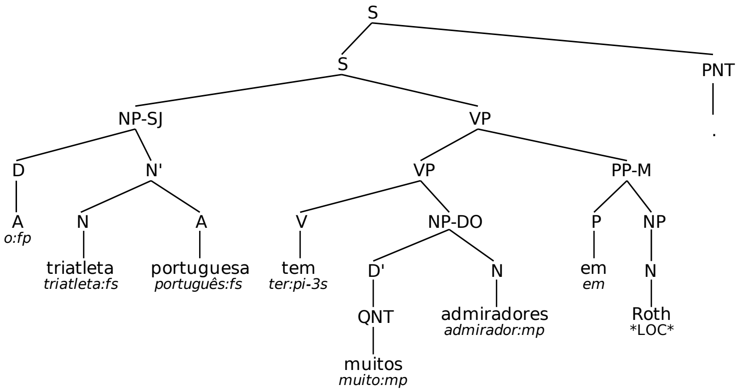

Figure 2. A sample of CINTIL Treebank

We ran 10-fold cross-evaluation, by iteratively training over a randomly selected 90% portion of the treebank and testing over the remaining 10% to get a baseline score. The average labeled Parseval f-score obtained was 76.18%.

This represents an improvement of more than 20% over the best result (63.2%) in [2] with Bosque. Significantly, when comparing the two baseline scores, the improvement over the Bosque-supported parser is more than 100%.

Despite the fact that the results in [2] were obtained with a parser fine-tuned for Portuguese, and a much larger training dataset, Bikel's parser fared much better over CINTIL than over Bosque.[4] Accordingly, we opted for using the CINTIL treebank for the remainder of this paper given this dataset not only ensures better parser performance as it adheres to the *de facto* standard format and can actually be used out-of-the-box.

3 Parsers

Having cleared the issue on which dataset, if any, would support out-of-the-box parsing, we proceeded with a detailed experimentation of several packages.

In order to gather as many as possible freely available packages that were good candidates for training out-of-the-box parsers we searched the ACL software repository[5] and collected the replies to a post (September 2009) on Corpora-List[6] querying on this type of packages.

The following five appeared as good candidates: Bikel [3], Stanford [8], Berkeley [1], BitPar [9] and Reranking/Charniak [10] parsers.

3.1 Bikel

Bikel's parser [3] is a language-independent, head-driven, statistical parsing engine. Language-independence is achieved through "language packages" which

[4] It is known that annotation errors adversely affect the performance of classifiers [6,7].

[5] `http://www.aclweb.org/`

[6] `http://gandalf.aksis.uib.no/corpora/`

encapsulate all the procedures that are specific to a language or treebank. The package comes with specific support for English, Chinese and Arabic.

For English, running in a mode emulating Model 2 of Collins' parser [11], and when evaluating over sentences up to 40 words in length from Section 00 of the Penn Treebank, it performs at 90.01% Parseval f-score [3, p. 181].

3.2 Stanford

The Stanford package [8] allows training a factored parser, which models phrase-structure and lexical dependencies separately. Phrase-structure is modeled using a probabilistic context-free grammar similar to that in [11], which includes features such as parent annotation of nodes and English-specific splits of certain features.[7] The model for lexical dependencies takes into account direction, distance and valence between a constituent and its dependents. The probability of a tree is then given by the product of the probabilities that the phrase-structure model and the lexical dependencies model assign to that tree. This software package comes with specific support for English, Chinese, Arabic and German.

For English, the parser scores 86.7% Parseval f-score over sentences up to 40 words in length from Section 23 of the Penn Treebank [8, p. 8].

3.3 Berkeley

The Berkeley parser [1] iteratively refines a base X-bar grammar by repeatedly splitting and merging non-terminal symbols. For instance, the symbol NP could be split into the symbols NP_0 (NP in subject position) and NP_1 (NP in object position). The base X-bar is obtained directly from the training dataset by a binarization procedure that introduces left-branching X-bar nodes in order to ensure that each node has two children. In each iteration, every symbol is split in two. Since this would quickly become unwieldy, a merging step checks which splits can be undone without a great loss in likelihood.

For English, this parser achieves 90.15% Parseval f-score over the sentences up to 40 words in length from Section 23 of the Penn Treebank [1, p. 440].

3.4 Other packages

BitPar [9] is a bit-vector implementation of the CYK parsing algorithm. The compact bit-vector representation greatly minimizes the memory and runtime costs during parsing, allowing BitPar to more easily represent the parse forest of all possible analysis of a sentence.

Reranking/Charniak [10] uses a maximum entropy model to select the best parse from the 50-best parses delivered by Charniak's coarse-to-fine *n*-best generative parser [12].

[7] Infinitive VPs are marked, the tag for preposition is split into subtypes, etc.

Like the three packages presented above, these two packages were originally developed for English. However, they eventually revealed to be much more language dependent. A great deal of their source code has hard-coded features for English and in particular for the English specific tag set and annotation conventions used in the Penn Treebank. Moreover, to a considerable extent, for the training phase of the parser to be induced, they resort to rule-based preprocessing of this dataset in order to extract or make available linguistic features specific of English but not explicitly present in the Penn treebank.

Their adaptation to train a parser for Portuguese, from whatever training dataset, would thus require a very large effort to change the source code. Accordingly, we realized that BitPar and Reranking/Charniak cannot be considered for *out-of-the-box* parsing of Portuguese. For this reason, they ended up not being further used in the research exercise described here.

3.5 Evaluation results

As there is nowadays a specific and productive research area on dependency parsing, our focus here will be on parsers for syntactic constituency alone. Accordingly, the evaluation results reported from this point onwards were obtained with one of the available versions of CINTIL that contains constituency information and no syntactic function tags.

We maintain the same evaluation methodology used in the quick experiment ran for assessing the datasets in Section 2. We use 10-fold cross-evaluation, iteratively training over a random 90% portion of the treebank and evaluating over the remaining 10%, and averaging the scores. The following performance metrics were computed:

Parseval is the classic metric of bracketing correctness [13]. We use a subsequent adaptation [14] that also takes into account whether the constituent label is correct. This metric provides labeled recall and labeled precision, from which the labeled f-score[8] is calculated.

Evalb is similar to PARSEVAL in that it is also a metric of bracketing correctness, providing, among other metrics, labeled f-score [15]. The most important difference, however, is that pre-terminals nodes are taken separately, allowing for a separate measure of tagging (part-of-speech) accuracy.

LeafAncestor is argued to mirror more closely our intuitive notions of parsing accuracy [16]. Instead of looking at bracketing correctness, it checks the *lineage* of terminal elements, i.e. the sequence of nodes connecting a terminal element to the root of its tree.

Three parsers were trained with each one of the packages described above that were suitable for out-of-the-box parser induction. Each one of the resulting parsers was evaluated along the metrics just outlined. The results are summarized in Table 1.

[8] F-score is defined as the harmonic mean of precision and recall: $f = \frac{2pr}{p+r}$

	f_{Parseval}	f_{Evalb}	POS acc.	LeafAnc.
Bikel	84.97%	73.08%	88.82%	90.48%
Stanford	88.07%	78.75%	92.91%	91.87%
Berkeley	89.33%	80.79%	91.62%	93.72%

Table 1. Performance scores of parsers for Portuguese

The first point worth noting is that the results obtained with the Parseval metric are within the 85–90% window that has been consistently obtained for top-performing parsers, running over English.

These are extremely encouraging scores, specially if one takes into account that they were obtained over a small treebank and with out-of-the-box parsers, with no effort applied to adapt them to Portuguese or to the training dataset.

Moreover, to our knowledge, these are the best published results for the probabilistic parsing of Portuguese.

Coming now to a comparison among parsers, Berkeley's parser has the best overall performance, which is in line with it also being one of the best parsers for English [1, p. 440]. A possible contribution for its better score may come from the fact that this is possibly the least language-dependent parser, since it does not use head-finder rules. As such, it was not as severely penalized as the other parsers for running out-of-the-box over a language that is not English.

4 Assessing enhanced performance

In this section we seek to assess, at least in part, to which extent the performance of these out-of-the-box parsers can be expected to be improved.

Improvement will likely come by pursuing two lines of action, each moving away from the out-of-the box status of the parsing task supported by the induced parsers. On the one hand, each parser can be put under systematic testing so that their parameters can be progressively fine-tuned in order to set a running configuration that support optimal performance. On the other hand, each parser can be aided by pre-processing modules so that the resulting parsing pipeline has better performance than the stand-alone parser.

In this section, we will concentrate on the second line of action as it can be implemented under a straightforward approach. It is immediate to test the *upper bound* for the improvement of performance in parsing due to the contribution of a pre-processing module. It is enough to use the correct data already present in the treebank as if they were the output of a perfect, 100% accurate pre-processing module.

Taking the values from Table 1 as a baseline, we test different ways of pre-processing the input delivered to the parser, typically in view of reducing data-sparseness, and measure how that improves the overall parsing performance. We test each of the following pre-processing procedures:

Lemmatization. Portuguese has a rather rich morphology, in particular in what concerns verbal inflection. Abstracting away from the variations caused by inflection should help mitigate the problems caused by data-sparseness.

To test this, the parser is trained and evaluated over a treebank where each noun, adjective and verb has been replaced by its lemma.

Named-entity recognition. Named-entities and multi-word expressions are a well-known problem for natural language processing [17]. Recognizing these sequences of words as an entity is very helpful for the parsing process since the parser can handle them as being a syntactic atom.

In CINTIL, certain types of named entities (NE) are marked and classified.[9] This allowed us to create a variant of the treebank, we termed NE-joined, where those named entities appear as one syntactic unit with their parts concatenated (separated by a plus sign) into a single token. For instance, `Hong Kong` is represented as the single node `(N Hong+Kong)`.

In addition, given that the named entities in CINTIL are semantically classified, we created another variant, we termed NE-sem, where each named entity expression was replaced by its semantic type. For instance, `Hong Kong` is represented as `(N *LOC*)` (for location), while `David Maia` is represented as `(N *PER*)` (for person).

POS tagging. As can be seen in Table 1, the POS tagging accuracy of each parser is below what is attainable by a state-of-the-art, dedicated POS tagger.[10] This is likely due to the small size of the training treebank we are using here.

It is worth noting that even if the parser could assign POS tags with as much accuracy as a POS tagger, such pre-processing by a dedicated, stand-alone POS tagger is often used and coupled to robust parsers as a way to reduce ambiguity and speed up the parsing process [20, p. 240].

To test a contribution by an hypothetical optimal POS tagger, we assume that the input to the parser has been previously annotated by a faultless tagger. This is quite easy to simulate since the parsers we are using already support a mode where they accept POS-tagged input. All that we need to do is to keep the pre-terminal tags from the treebank in the text that is going to be annotated.

The results obtained when pipelining the above different pre-processing modules with each parser are displayed in Table 2. Each pre-processing procedure is able to improve on the baseline scores, albeit by different amounts.

The relative ranking of the parsers (Berkeley > Stanford > Bikel) is generally preserved across the different experiments. Also, for each parser, the relative differences in the scores obtained through each evaluation metric ($\mathrm{f}_{\mathrm{Parseval}}$, $\mathrm{f}_{\mathrm{Evalb}}$, POS accuracy and LeafAncestor) are also generally preserved across the experiments.

Given this, and for the sake of simplicity, the results in Table 2 are commented taking the Berkeley parser and the $\mathrm{f}_{\mathrm{Parseval}}$ metric as a reference. This is also the parser that is being released as an outcome of the present research exercise, at `http://lxparser.di.fc.ul.pt/`.

[9] Expressions for persons, organizations, locations, events, works (e.g. movies, books, paintings, etc.) and miscellaneous.

[10] For instance, for Portuguese, an accuracy of 97% has been obtained for POS tagging also with the CINTIL tagset [18,19].

		baseline	lemmas	NE-joined	NE-sem	POS
Bikel	f_{Parseval}	84.97%	87.68%	85.16%	87.68%	92.34%
	f_{Evalb}	73.08%	74.35%	73.48%	74.71%	79.72%
	POS acc.	88.82%	92.03%	90.26%	91.13%	n.a.
	LeafAnc.	90.48%	91.79%	90.49%	91.03%	94.06%
Stanford	f_{Parseval}	88.07%	89.49%	88.69%	88.91%	93.69%
	f_{Evalb}	78.75%	80.74%	79.63%	80.41%	84.60%
	POS acc.	92.91%	94.05%	94.48%	94.63%	n.a.
	LeafAnc.	91.87%	92.81%	92.06%	92.27%	94.97%
Berkeley	f_{Parseval}	89.33%	90.21%	89.55%	90.34%	95.61%
	f_{Evalb}	80.79%	81.11%	81.60%	83.15%	87.42%
	POS acc.	91.62%	92.82%	92.74%	93.29%	n.a.
	LeafAnc.	93.72%	94.63%	94.16%	94.36%	96.55%

Table 2. Performance scores for each pre-processing procedure

NE-joined provides the least improvement (89.55%). While it does simplify the phrase-structure, it does not help in reducing the lexicon.

If the entities are replaced by their types, as in NE-sem, the improvement becomes more apparent (90.34%), gaining 1 percentage point over the baseline. When every entity is mapped into a small set of semantic types, it greatly compacts the lexicon and any new named entity, when classified with a semantic type seen in training, will not be considered an unknown word.

Using lemmas instead of inflected forms brings about a similar amount of improvement (90.21%). This improvement, however, was not caused by better handling of named entities. Therefore, assigning lemmas as a pre-processing step could be applied together with NE-sem for an even bigger increase in scores.

Providing the parsers with correct POS yields by far the largest improvement, of 6 percentage points over the baseline, bringing the score to 95.61%.

5 Conclusions

With the experiment reported in this paper we showed that it is possible to apply out-of-the-box, state-of-the-art software packages for training parsers for Portuguese. More importantly, not only is this possible, as the results are in line with those obtained for English with top-performing parsers given the best treebank available, namely the CINTIL Treebank.

Given that the results described above were obtained over a modestly sized treebank and with out-of-the-box parsers it will be possible to obtain new parsers with improved performance. The improvements achieved by experimenting with different pre-processing procedures confirm this.

The best parser for Portuguese, obtained with the Berkeley package, is released at `http://lxparser.di.fc.ul.pt/`.

As future work, our goal will be to retrain this parser over upcoming and larger versions of CINTIL Treebank and find the configuration of parameters that eventually support its optimal performance.

References

1. Petrov, S., Barrett, L., Thibaux, R., Klein, D.: Learning accurate, compact, and interpretable tree annotation. In: Proceedings of the 44th ACL. (2006) 433–440
2. Wing, B., Baldridge, J.: Adaptation of data and models for probabilistic parsing of Portuguese. In: Proceedings of the 7th Encontro para o Processamento Computacional da Língua Portuguesa Escrita e Falada (PROPOR). (2006) 140–149
3. Bikel, D.: Design of a multi-lingual, parallel-processing statistical parsing engine. In: Proceedings of the 2nd Human Language Technology Conference. (2002)
4. Branco, A., Costa, F.: A computational grammar for deep linguistic processing of Portuguese: LXGram, version A.4.1. Technical Report DI-FCUL-TR-08-17, University of Lisbon (2008)
5. Branco, A., Costa, F.: A deep linguistic processing grammar for portuguese. In: this volume. (2010)
6. Padró, L., Màrquez, L.: On the evaluation and comparison of taggers: The effect of noise in testing corpora. In: Proceedings of the 17th COLING. (1998) 997–1002
7. Dickinson, M., Meurers, D.: Detecting inconsistencies in treebanks. In: Proceedings of the 2nd Workshop on Treebanks and Linguistic Theories. (2003)
8. Klein, D., Manning, C.: Fast exact inference with a factored model for NLP. Advances in Neural Language Processing Systems **15** (2003) 3–10
9. Schmid, H.: Efficient parsing of highly ambiguous context-free grammars using bit vectors. In: Proceedings of the 20th COLING. (2004) 162–168
10. Charniak, E., Johnson, M.: Coarse-to-fine *n*-best parsing and maxent discriminative reranking. In: Proceedings of the 43rd ACL. (2005)
11. Collins, M.: Head-Driven Statistical Models for Natural Language Parsing. PhD thesis, University of Pennsylvania (1999)
12. Charniak, E.: A maximum-entropy-inspired parser. In: Proceedings of the 1st North American Chapter of the ACL. (2000) 132–139
13. Black, E., Abney, S., Flickinger, D., Gdaniec, C., Grishman, R., Harrison, P., Hindle, D., Marcus, M., Santorini, B.: A procedure for quantitatively comparing the syntactic coverage of English grammars. In: Proceedings of the Workshop on the Evaluation of Parsing Systems. (1991) 306–311
14. Magerman, D.: Statistical decision-tree models for parsing. In: Proceedings of the 33rd ACL. (1995) 276–283
15. Sekine, S., Collins, M.: Evalb website `http://nlp.cs.nyu.edu/evalb/`.
16. Sampson, G., Babarczy, A.: A test of the leaf-ancestor metric for parse accuracy. Natural Language Engineering **9**(4) (2003) 365–380
17. Sag, I., Baldwin, T., Bond, F., Copestake, A., Flickinger, D.: Multiword expressions: A pain in the neck for NLP. In: Proceedings of the 3rd Conference on Intelligent Text Processing and Computational Linguistics. (2002) 1–15
18. Branco, A., Silva, J.: Evaluating solutions for the rapid development of state-of-the-art POS taggers for Portuguese. In: Proceedings of the 4th Language Resources and Evaluation Conference (LREC). (2004) 507–510
19. Silva, J.: Shallow processing of Portuguese: From sentence chunking to nominal lemmatization. Master's thesis, University of Lisbon (2007) Published as Technical Report DI-FCUL-TR-07-16.
20. Bangalore, S., Joshi, A.: Supertagging: An approach to almost parsing. Computational Linguistics **25**(2) (1999) 237–265