Shallow parsing of Portuguese texts annotated under Universal Dependencies

Guilherme Martiniano de Oliveira[0000−0002−2030−3688], Paulo Berlanga Neto[0000−0002−1985−1089], Evandro Eduardo Seron Ruiz[0000−0002−7434−897X]

Department of Computing and Mathematics – FFCLRP
University of São Paulo, Ribeirão Preto, SP – Brazil
{guizera11, pauloberlanga, evandro}@usp.br

Abstract. Shallow parsing is an intermediate step to many natural language processing tasks, such as information retrieval, question answering, and information extraction. An alternative to full-sentence parsing consists of segmentation and identifying phrases in sentences. Building such a parser for the Portuguese language is challenging considering the proposed formalism for grammar annotation, the Universal Dependency (UD). This paper addresses preliminary studies to overcome these barriers by annotating noun phrases tagged in UD.

Keywords: Shallow parsing · Universal Dependencies · Neural Networks

1 Introduction

Assigning a complete syntactic structure to sentences based on grammar and a search strategy is the goal of full parsing. However, not all-natural language processing (NLP) applications require a complete syntactic analysis [9]. For many NLP tasks, such as named entity recognition [3], sentiment analysis [15] and information retrieval [6], recovering only a limited amount of syntactic information has proved to be a valuable technology for written and spoken language domains. This chunking strategy is generally known as partial parsing or shallow parsing. Shallow parsing can also serve as a baseline for full parsing [2] since it provides a foundation for other levels of analysis.

This work focuses on extracting non-overlapping noun-phrase (NP) chunks, as proposed initially by Abney [2], including nouns and proper nouns, among other classes of words that add more meaning to these two. Shallow parsers have already been developed for the constituency tree format [5]. Here we address the challenge of developing a parser to work under Brazilian-Portuguese texts annotated with the Universal Dependencies (UD) format, which is currently used in many NLP tasks.

We propose constructing a model for recognizing noun phrases in input sentences through a neural network (NN) trained model, as proposed by Søgaard and Goldberg [16]. Some NN architectures are explicitly designed for long-term dependency learning, as written texts are. More specifically, our proposed shallow parser model processes text in three stages: 1) A *learning corpus* is built from partial parsed sentences. These sentences are extracted from the constituency version of Bosque corpus (version 8); 2) Sentences from this learning corpora are augmented with UD labels from the UD_Portuguese-Bosque version 2.2. This revised UD treebank retains the additional tags for NP. Finally; 3) A neural network-based classification model is built from the learning corpus and applied to the original test subset from the UD_Portuguese-Bosque, here called *text corpus*.

Following, we briefly introduce the main related work to shallow parsing. In Section 3, we present the data and methods used. In Section 4, we report a summary of the experiments. Some considerations about the experiment's results are detailed in Section 5. Finally, in Section 6, we present some concluding remarks.

2 Related work

The idea of text chunking was proposed in the seminal work of Steven Abney [2], where he shows the correspondence of prosodic patterns to segments of constituency grammar trees. Following this intuition, Ramshaw and Marcus [14] developed the first known method for chunking sentences similarly to traditional grammar, creating templates and rules that described chunk formation. This method is known as Transformation-Based Learning (TBL).

Alonso et al. [3], Brants [6] and a team led by Hammerton [9], among others, have also developed and applied shallow parsing to sentences annotated in the constituency tree format.

For the Portuguese language, we highlight the work of Barreto and his colleagues [4] with the TagShare project that embraces linguistic resources and tools for the shallow processing of Portuguese. These resources also include a 1M token corpus that has been accurately hand-annotated. Noun phrase chunking for English, Portuguese, and Hindi was proposed by Milidiú, Santos, and Duarte [12]. They applied Entropy Guided Transformation Learning (ETL), a machine learning strategy that combines decision trees and the classical TBL method. For the Portuguese, their proposed methodology achieved a precision of 92.62%, recall of 93.05%, and an F-measure of 92.84%.

Machine learned-based system was also used for a shallow parsing similar task called clause identification (CI). The Milidiú team extended their previous experiments to work likewise with CI [8]. They stated that CI is a phrase-chunk-like (PCL) task. PCL consists of splitting a sentence into clauses. A clause is defined as a word sequence containing a subject and a predicate. Clause identification is a special kind of shallow parsing. They proposed an Entropy Guided Transformation Learning system that achieved an F-measure of 73.9%.

Chunking received much attention, mostly when syntactic parsing was predominantly guided by constituency parsing, as it is the case for all previous works. With the UD grammar annotation surge, new methods need to be created. To our knowledge, Ophélie Lacroix [11] was the first to show that UD annotated texts can also leverage the information provided by the constituency annotation. She grouped tokens to form NP chunks and used neural networks to train and test her method. She showed that it is possible to extract NP-chunks (noun phrases) from Universal Dependencies annotated texts with accuracy similar to traditional chunks operated under constituency trees. Her NP-chunking method achieved F-measure=89.9% when applied to dependency trees.

Our project aims to deduce NP-chunks from automatically UD annotated texts using a deep neural network (NN) approach. To lead our way to a feasible NN model for NP-chunking, we based our project on the work of Søgaard and Goldberg [16]. They showed that it is possible to utilize a multi-task learning architecture (MTL) with deep bi-directional recurrent neural networks (RNNs) to make syntactic chunking more precise, achieving an F-score=94.1%. They conclude that deep neural networks are a powerful tool for syntactic analysis.

3 Methodology

3.1 Data

Using an NN-trained model, we aim to recognize and extract non-overlapping noun phrase (NP) chunks. As requested by a supervised learning approach, two corpora are needed: a) A *learning corpora*, and; b) A *test corpora*. The *learning corpora* used is composed of sentences from the Bosque corpus. Version 8.0 of the Bosque corpus[1] provides syntactic annotations of noun phrase chunks, under the 'NP' category, like other types of phrase chunks. As a constituency parsed corpus, no UD labels were provided for this version of the Bosque. UD labels were acquired from the UD_Portuguese-Bosque version 2.2[2]. This UD treebank retained the original NP tags. The *test corpora* is composed of the test subset labeled sentences from the UD_Portuguese-Bosque.

A classification engine (detailed in the next subsection, 3.2) is fed with the *test corpora* sentences. Each extracted sentence is analyzed accordingly to the knowledge acquired from the *learning corpora*. The following subsection describes the classification engine.

3.2 Method

We define the noun phrase detection task as a sequence labeling problem. Given an input sentence composed of a sequence of tokens, $w_1, \ldots, w_n$, the goal is the prediction of an output sequence $y_1, \ldots, y_n$, $y_i \in \{1, \ldots, |L|\}$, where L is a determined set of labels and y_i is the respective label for w_i.

We adopted an MTL architecture based on deep bi-directional recurrent neural networks (Bi-LSTM). The MTL can be understood as a layer-sharing method

[1] https://www.linguateca.pt/Floresta/corpus.html#download

[2] https://github.com/UniversalDependencies/UD_Portuguese-Bosque

that helps models deal with different tasks simultaneously. Therefore, such intermediary representations allow different tasks to benefit from each other, stimulating the standard practical knowledge learning process. Considering the proposed sequence labeling model, we may, for example, experiment with part-of-speech (POS) tagging and syntactic chunking predictions for the same input sentence.

Long Short-Term Memory (LSTM) [10] is a particular flavor of recurrent neural networks (RNN) widely applied in NLP tasks that enables long-term dependency learning. It may also be considered an instance that primarily aims to eliminate the vanishing gradient problem observed in the 'vanilla' RNN [7] since the latter cannot correctly handle long sequences of tokens [16].

Explained in a simple way, the LSTM architecture, consider RNNs as a blackbox abstraction. One may view LSTMs as an instance of a RNN interface. RNN may be seen as a function $R_{\Theta}(w_{1:n})$ mapping a sequence of n input vectors $w_{1:n}, w_i \in R_{\text{in}}$, to output vector $h_{1:n}, h_i \in R_{\text{out}}$. Applying $R_{\Theta}(w_{1:n})$ to all prefixes $w_{1:i}, 1 \leq i \leq n$ of $w_{1:n}$, result in n output vectors $h_{1:n}$, where $h_{1:i}$ is a summary of $w_{1:i}$.

Layers of RNN are called deep RNN. A $k-$layer RNN are a set of k RNN functions $(\text{RNN}_1, \text{RNN}_2, \ldots, \text{RNN}_k)$ feeding each other. A bidirectional RNN is composed of two RNNs, RNN_{F} and RNN_{R}, one that reads the sequence in one order, e.g., forward, and the other reading it in reverse.

We employed an architecture-based Bi-LSTM following Søgaard and Goldberg reference work [16]. They show that this architecture can explore contextual information to process long sequences. Our proposed model comprises an embedding layer that feeds two hidden layers (forward and backward), composed of 300 units. The model was trained using back-propagation and Stochastic Gradient Descent (SGD), employing batch sizes of 64 with a learning rate of 0.01. The training process lasted ten epochs. All the hyper-parameters were defined empirically. The Bi-LSTM implementation was accomplished with the `nlp-architecture`[3] Python module [1].

4 Experiment

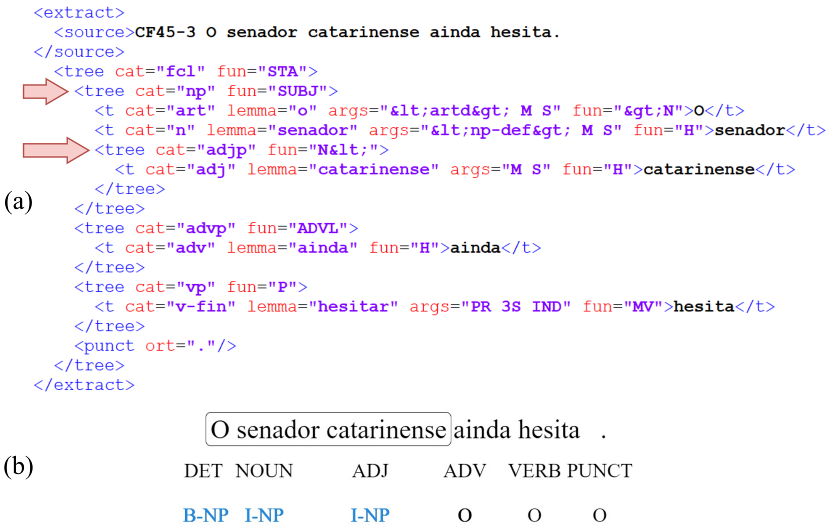
We recall that although the Bosque corpus version 8 is composed of 18,804 sentences, only part of this corpus, 9,364 sentences, were annotated under UD, assembling the UD_Portuguese-Bosque. Further, these 9,364 sentences are divided into three subsets: learning-train (8,328), dev (560) and test 476. Since not all the sentences have NP and some processing errors, such as bugs reading the XML file, only 8,585 sentences were used, corresponding to 91,6% of the 9,364. Table 1 below depicts the number of sentences used for both corpora, the *learning* (7,605) and the *test* (444) corpus.

Based on the syntactic annotations provided by the Bosque corpus (v.8), we acknowledge noun phrase chunks searching for tokens inside the noun phrase (*'NP'* category) also considering the alongside adjectives (*'adpj'* category). Figure 1(a) illustrates such annotations in the Bosque `SimTreeML` format.

[3] https://intellabs.github.io/nlp-architect/

Table 1. Number of sentences for the used corpora.

	Bosque v.8	UD_Portuguese-Bosque
subset	(constituency)	(original)
learning-train	**7,605**	8,328
dev	536	560
test	**444**	476
Total	8,585	9,364



(a)

```
<extract>
  <source>CF45-3 O senador catarinense ainda hesita.
</source>
  <tree cat="fcl" fun="STA">
    <tree cat="np" fun="SUBJ">
      <t cat="art" lemma="o" args="&lt;artd&gt; M S" fun="&gt;N">O</t>
      <t cat="n" lemma="senador" args="&lt;np-def&gt; M S" fun="H">senador</t>
      <tree cat="adjp" fun="N&lt;">
        <t cat="adj" lemma="catarinense" args="M S" fun="H">catarinense</t>
      </tree>
    </tree>
    <tree cat="advp" fun="ADVL">
      <t cat="adv" lemma="ainda" fun="H">ainda</t>
    </tree>
    <tree cat="vp" fun="P">
      <t cat="v-fin" lemma="hesitar" args="PR 3S IND" fun="MV">hesita</t>
    </tree>
    <punct ort="."/>
  </tree>
</extract>
```

(b)

O senador catarinense ainda hesita .
DET NOUN ADJ ADV VERB PUNCT
B-NP I-NP I-NP O O O

Fig. 1. Steps performed to the annotation process.

After that, we annotate each token from every sentence with the respective labels from the Universal Dependencies (UD) annotation format. In parallel, NP chunks were labeled with the IOB (Inside–Outside–Beginning) format [14]. Figure 1(b) illustrates the final annotated example.

Following the work of Lacroix [11], we aim to detect minimal, non-recursive noun phrases. For example, in the sentence *"O 7 e Meio é um ex-libris da noite algarvia."*, we consider the following constituents: *"O 7 e Meio"*, *"um ex-libris"* and *"a noite algarvia."*. Thus, we do not consider a single long noun phrase for *"um ex-libris da noite algarvia."*, but the aforesaid minimal version instead.

4.1 Evaluation

We assembled the Bosque data division in train-development-test subsets according to the work of Rademaker et al. [13]. See Table 1. Later, we trained the model with the previously mentioned method in Section 3.2. Running the test against the full reserved test set, we obtained an F-measure of 85.1%. See Table 2.

Table 2. Evaluation metrics for the Bi-LSTM network model in %.

Precision	Recall	F-measure
84.8	85.3	85.1

We may also see in Figure 2 an example of a prediction outputted by the trained model that correctly identifies the noun phrases present in the input sentence provided, based on the IOB pattern.

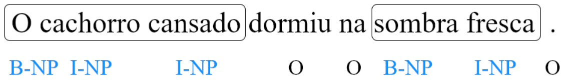


Fig. 2. Noun phrase prediction produced by the proposed model.

5 Considerations

A rudimentary qualitative analysis of the outputs reveals that the model could detect the desired minimal noun phrase chunks performing slightly better on sentences with simple syntax. Even so, many of the longest and most complex sentences were also labeled correctly. Quantitatively, an F-measure of 85.1% is not a state-of-the-art achievement. Although this work is not comparable with the work of Lacroix [11] that achieved an F-measure of 89.9%, we considered our result an encouraging preliminary one. The Bi-LSTM classifier was used with its default parameters, suggesting that an optimized gradient boosting approach like XGBoost would provide more gratifying results. The obtained F-score establishes our approach as a feasible method for Portuguese text chunking.

Although a comprehensively qualitative manual inspection of the errors shall be the subject of a prospective study, a casual manual search for minimal NP reveals some inconsistencies in the original POS tagging. Below we highlight the expression "(P)presidente da (R)república", which should not be tagged as a minimal NP. One may see a possible disagreement between human annotators in the following expressions.

1. . . . o governador do Rio e o **Presidente**[PROPN] da **República**[PROPN] chamaram o Exército.
2. . . . o **presidente**[NOUN] da **República**[NOUN] abriu uma fresta . . .
3. No caso de impedimento de o **presidente**[NOUN] da **República**[PROPN] . . .

In the previous examples the word “Presidente”, with a capital ‘P’ is tagged as *Proper Noun* while “presidente” is tagged as *Noun*. Respectively “República” appears with two distinctive tags, *Proper Noun* and *Noun*. Originally, the expression in the first sentence, “Presidente da República” was tagged as a NP, while for the second and third sentences, “presidente” and “República” were tagged individually as NP.

These last divergent examples encourage an extensive investigation, even if insufficient to justify the modest F-measure obtained. We note that the learning step might be impaired on comparable divergences due to the relatively small training dataset for the enormous variety of similar expressions.

6 Conclusion

We inferred that the method proposed has much potential for chunking detection that takes advantage of the characteristics presented in the UD pattern. We also believe that expanding learning corpora annotated under UD will foster more encouraging results. An accuracy over 95% and new methods to extract other types of chunks (prepositional, adverbial, and adjective) are some future works we are pursuing.

7 Acknowledgements

This work was carried out at the Center for Artificial Intelligence (C4AI-USP), with support by the São Paulo Research Foundation (FAPESP grant #2019/07665-4) and by the IBM Corporation.

References

1. NLP Architect, by Intel AI Laboratories (Nov 2018), https://doi.org/10.5281/zenodo.1477518
2. Abney, S.P.: Principle-Based Parsing: Computation and Psycholinguistics, chap. Parsing by Chunks, pp. 257–278. Springer Netherlands, Dordrecht (1992). https://doi.org/10.1007/978-94-011-3474-3_10
3. Alonso, M.A., Gómez-Rodríguez, C., Vilares, J.: On the Use of Parsing for Named Entity Recognition. Applied Sciences **11**(3), 1090 (2021)
4. Barreto, F., Branco, A., Ferreira, E., Mendes, A., Bacelar do Nascimento, M.F., Nunes, F., Silva, J.R.: Open resources and tools for the shallow processing of Portuguese: the TagShare project. In: Proceedings of the V International Conference on Language Resources and Evaluation – LREC2006. European Language Resources Association (2006)
5. Bick, E.: The Parsing System “Palavras”: Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework. Aarhus University Press (2000), https://books.google.com.br/books?id=ISUgDvPg7hcC
6. Brants, T.: Natural Language Processing in Information Retrieval. CLIN – Computational Linguistics in the Netherlands **111** (2003)

7. Elman, J.L.: Finding structure in time. Cognitive Science **14**(2), 179–211 (1990)
8. Fernandes, E.R., dos Santos, C.N., Milidiú, R.L.: A Machine Learning Approach to Portuguese Clause Identification. In: Pardo, T.A.S., Branco, A., Klautau, A., Vieira, R., de Lima, V.L.S. (eds.) Computational Processing of the Portuguese Language. pp. 55–64. Springer Berlin Heidelberg, Berlin, Heidelberg (2010)
9. Hammerton, J., Osborne, M., Armstrong, S., Daelemans, W.: Introduction to Special Issue on Machine Learning Approaches to Shallow Parsing. Journal of Machine Learning Research **2**(4), 551–558 (2002). https://doi.org/10.1162/153244302320884533
10. Hochreiter, S., Schmidhuber, J.: Long Short-Term Memory. Neural Comput. **9**(8), 1735–1780 (Nov 1997). https://doi.org/10.1162/neco.1997.9.8.1735, https://doi.org/10.1162/neco.1997.9.8.1735
11. Lacroix, O.: Investigating NP-chunking with Universal Dependencies for English. In: Proceedings of the Second Workshop on Universal Dependencies (UDW 2018). pp. 85–90. Association for Computational Linguistics, Brussels, Belgium (Nov 2018). https://doi.org/10.18653/v1/W18-6010, https://aclanthology.org/W18-6010
12. Milidiú, R.L., dos Santos, C., Duarte, J.C.: Phrase chunking using entropy guided transformation learning. In: Proceedings of ACL-08: HLT. pp. 647–655 (2008)
13. Rademaker, A., Chalub, F., Real, L., Freitas, C., Bick, E., de Paiva, V.: Universal Dependencies for Portuguese. In: Proceedings of the Fourth International Conference on Dependency Linguistics (Depling). pp. 197–206. Pisa, Italy (September 2017), http://aclweb.org/anthology/W17-6523
14. Ramshaw, L.A., Marcus, M.P.: Text Chunking Using Transformation-Based Learning, pp. 157–176. Springer Netherlands, Dordrecht (1999). https://doi.org/10.1007/978-94-017-2390-9_10
15. Sharma, A., Gupta, S., Motlani, R., Bansal, P., Shrivastava, M., Mamidi, R., Sharma, D.M.: Shallow parsing pipeline – Hindi-English code-mixed social media text. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 1340–1345. Association for Computational Linguistics, San Diego, California (Jun 2016). https://doi.org/10.18653/v1/N16-1159, https://aclanthology.org/N16-1159
16. Søgaard, A., Goldberg, Y.: Deep multi-task learning with low level tasks supervised at lower layers. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). pp. 231–235. Association for Computational Linguistics, Berlin, Germany (Aug 2016). https://doi.org/10.18653/v1/P16-2038, https://aclanthology.org/P16-2038