Information Processing and Management xxx (xxxx) xxx–xxx

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

Dependency parsing with finite state transducers and compression rules

Pablo Gamallo*,[a], Marcos Garcia[b]

[a] Centro Singular de Investigación en, Tecnoloxías da Información (CiTIUS), University of Santiago de Compostela, Galiza, Spain
[b] Universidade da Corunha, Grupo LyS, Departamento de Letras, Faculdade de Filologia, Corunha, Galiza, Spain

ABSTRACT

This article proposes a syntactic parsing strategy based on a dependency grammar containing formal rules and a *compression* technique that reduces the complexity of those rules. Compression parsing is mainly driven by the 'single-head' constraint of Dependency Grammar, and can be seen as an alternative method to the well-known *constructive* strategy. The compression algorithm simplifies the input sentence by progressively removing from it the *dependent* tokens as soon as binary syntactic dependencies are recognized. This strategy is thus similar to that used in deterministic dependency parsing. A compression parser was implemented and released under General Public License, as well as a cross-lingual grammar with Universal Dependencies, containing only broad-coverage rules applied to Romance languages. The system is an almost delexicalized parser which does not need training data to analyze Romance languages. The rule-based cross-lingual parser was submitted to *CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*. The performance of our system was compared to the other supervised systems participating in the competition, paying special attention to the parsing of different treebanks of the same language. We also trained a supervised delexicalized parser for Romance languages in order to compare it to our rule-based system. The results show that the performance of our cross-lingual method does not change across related languages and across different treebanks, while most supervised methods turn out to be very dependent on the text domain used to train the system.

1. Introduction

Syntactic analysis is a crucial module for many tasks relying on natural language processing and information extraction techniques such as summarization (Abdi, Idris, Alguliev, & Aliguliyev, 2015), information retrieval (Vilares, Alonso, & Vilares, 2014), topic detection (Lee, Lee, & Jang, 2007), named entity recognition and entity linking (Derczynskia et al., 2015), text mining (Tseng, Lin, & Lin, 2007), text classification (Sokolova & Lapalme, 2009; Uysal & Gunal, 2014), or sentiment analysis (Severyn, Moschitti, Uryupina, Plank, & Filippova, 2015; Vilares, Alonso, & Gómez-Rodríguez, 2015). Besides, syntactic information is required to improve semantic applications. More precisely, knowledge on syntactic parse trees turns out to be useful to yield accurate semantic word models and embeddings based on the distributional context of words (Saif, He, Fernandez, & Alani, 2015), as well as to extract semantic relations (Zhang, Zhou, & Aw, 2008) and for semantic role labeling (Zhou, Li, Fan, & Zhu, 2011).

In this article, we propose a new (rule-based) finite-state parsing strategy based on dependencies, which minimizes the complexity of rules by using a technique we call *compression*. Compression parsing is driven by the single-head constraint of Dependency

* Corresponding author.
E-mail addresses: pablo.gamallo@usc.es (P. Gamallo), marcos.garcia.gonzalez@udc.gal (M. Garcia).

https://doi.org/10.1016/j.ipm.2018.05.003
Received 24 February 2016; Received in revised form 23 April 2018; Accepted 13 May 2018

Grammar. It simplifies the input string by progressively removing the *dependent* tokens as binary syntactic dependencies are recognized. At the end of the compression process, if all the dependencies in the sentence are recognized, the input string should contain just one token representing the main head (i.e., the *root*) of the sentence. This strategy was inspired by the *Right* and *Left* Reduce transitions used in deterministic dependency parsing (Nivre, 2003; Nivre, Hall, & Nilsson, 2004).

Deterministic dependency parsing (called 'transition based') relies on supervised techniques requiring fully analyzed training corpora (syntactic treebanks). Given that supervised techniques tend to have a loss of precision when applied to texts of domains and genres different from those used for training (Rimell, Clark, & Steedman, 2009), they need too much manual effort to create, adapt, or modify the training corpus to the target domain. It is generally accepted that supervised classifiers require some type of domain adaptation when both the training and test data sets belong to different domains. In particular, the accuracy of statistical parsers degrades when they are applied to different genres and domains (Gildea, 2001; Rimell et al., 2009). By contrast, we propose a dependency parsing strategy based on elementary linguistic information which may be applied on different domains with similar accuracy and whose performance is close to the state-of-the-art (Section 6.4).

A system based on the compression strategy was implemented in Perl and released under General Public License: *DepPattern*. In addition, we defined a high level grammar language to define dependency-based rules and developed a grammar compiler in Ruby to generate compression parsers in several languages (Gamallo & González, 2011). DepPattern has been used for several web-based IE applications, namely Open Information Extraction from Wikipedia (Gamallo, Garcia, & Fernández-Lanza, 2012), extraction of semantic relations with distant supervision (Garcia & Gamallo, 2011), and extraction of bilingual terminologies from comparable corpora (Gamallo & Pichel, 2008). It has also been integrated into commercial tools, e.g. Linguakit.[1] and Avalingua[2]

Some experiments were performed to compare our rule-based system with supervised approaches. For this purpose, we implemented a specific parser with DepPattern, called *MetaRomance*, which is based on a very basic cross-lingual grammar for Romance languages. MetaRomance was compared to the systems that participated at *CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies* (Zeman et al., 2017). The reported results showed that our system's performance remains stable across related languages and different treebanks for the same language, while most supervised methods are very dependent on the specific properties of the treebank used for training (Garcia & Gamallo, 2017).

The use of grammars in recent dependency parsers is almost non-existent since they are considered too cost to build and maintain. However, this drawback can be minimized by incorporating into the parsing system *light-weight* grammars. More precisely, our system makes use of small grammars containing broad-coverage syntactic information that can be applied to several related languages and different content domains. The cost of manually creating grammar rules is also reduced by providing a suitable rule notation for linguists.

The remainder of this article is organized as follows. Sections 2 and 3 introduce different approaches on both dependency-based and finite-state parsing, including deterministic dependency parsing and constructive parsing. Then, Section 4 is focused on the description of our compression strategy. Next, Section 5 provides a general view of the implemented system: DepPattern and MetaRomance. Section 6 reports the diverse experiments performed with MetaRomance using the treebanks provided by CoNLL 2017. Finally, some conclusions are drawn in Section 7.

2. Dependency-based syntactic parsing

Following Nivre (2006), there are two traditions in dependency parsing: grammar-driven and data-driven parsing. Within each tradition, it is also possible to distinguish between two different approaches: non-deterministic and deterministic parsing. In the latest years, most work on dependency parsing has been developed within the approach of data-driven deterministic parsing, which is also known as *transition-based parsing*, in opposition to non-deterministic strategies such as *graph-based dependency parsing*. In addition, in the latest years there has been an important growth in cross-lingual parsing research, which is the main application field of our parsing system.

2.1. Graph-based dependency parsing

A graph-based dependency parser, also known as discriminative parser, starts with all valid dependencies between the nodes/words of a sentence. This ambiguous structure is represented as a completely connected graph whose edges are weighted according to a statistical model. Then, in the disambiguation (or discriminative) process, the parser tries to find a tree covering all nodes (words) in the graph that maximizes the sum of the weighted edges (Carreras, 2007; Martins, Smith, Xing, Aguiar, & Figueiredo, 2010; McDonald & Pereira, 2006). Therefore, as in non-deterministic parsing, this technique generates first all analyses and, then, selects the most probable one according to the statistical model.

2.2. Transition-based dependency parsing

This strategy consists in inducing statistical models in combination with a deterministic strategy based on shift-reduce parsing (Gómez-Rodríguez & Fernández-González, 2012; Nivre, 2004; Yamada & Matsumoto, 2003). Nivre et al. (2004) uses the arc-eager

[1] https://linguakit.com/

[2] http://cilenis.com/en/avalingua/