Towards fast natural language parsing: FASTPARSE ERC Starting Grant*

Hacia el análisis sintáctico rápido de lenguaje natural: la ERC Starting Grant FASTPARSE

Carlos Gómez-Rodríguez
Universidade da Coruña
FASTPARSE Lab, LyS Research Group, Depto. de Computación
Facultade de Informática, Elviña, 15071 A Coruña, Spain
carlos.gomez@udc.es

Abstract: The goal of the FASTPARSE project (Fast Natural Language Parsing for Large-Scale NLP), funded by the European Research Council (ERC), is to achieve a breakthrough in the speed of natural language syntactic parsers, developing fast parsers that are suitable for web-scale processing. For this purpose, the project proposes several research lines involving computational optimization, algorithmics, statistical analysis of language and cognitive models inspired in human language processing.
Keywords: Parsing, syntax, efficiency, multilinguality, dependency parsing, constituent parsing

Resumen: El proyecto FASTPARSE (Fast Natural Language Parsing for Large-Scale NLP), financiado por el Consejo Europeo de Investigación (ERC), tiene como objetivo lograr un salto cualitativo en la velocidad de los analizadores sintácticos de lenguaje natural, desarrollando analizadores lo suficientemente rápidos para facilitar el procesado de textos a escala web. Para ello, el proyecto propone distintas líneas de investigación que combinan técnicas de optimización informática, algoritmia, análisis estadístico de propiedades del lenguaje y modelos cognitivos inspirados en el procesado humano del mismo.
Palabras clave: Análisis sintáctico, sintaxis, eficiencia, multilingüismo, análisis de dependencias, análisis de constituyentes

1 Objectives

Natural language parsing, or syntactic analysis, is the task of automatically finding the underlying structure of sentences in human languages. Parsing is a crucial process for computer applications that deal with natural language text or speech, because the syntactic analyses produced by a parser can be used to extract meaning from sentences. For example, an analysis of the simple sentence "John ate an apple" can be used to know what action has been performed (the main verb, "ate"), who performed that action (the subject, "John") and what has been eaten (the object, "an apple"). This information would not be available if we just considered the sentence as a sequence of words, without regard to its internal structure.

For this reason, natural language processing (NLP) and text mining applications that need to process written or spoken human language beyond the level of individual words rely on parsing. This includes applications such as machine translation, question answering, opinion mining, information retrieval, information extraction, and automatic summarization.

The last decade of research on parsing algorithms has notably improved their accuracy, hence the evolution of parsing from a promising, emerging technology to a practical asset that is being actively exploited in real-world applications. However, there is still an important roadblock that limits the widespread adoption of this technology and the extent of its applications: parsing algorithms have significant computation time

* This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 714150).

requirements. For example, state-of-the-art parsers based on constituency grammar exhibit speeds slower than 5 English sentences per second on standard current computers (Kummerfeld et al., 2012), which can be improved to close to 100 sentences per second by sacrificing some accuracy (e.g. Crabbé (2015)). For the other prevailing syntactic formalism, dependency grammar, state-of-the-art parsers can process around 100 sentences per second (Choi and McCallum, 2013; Rasooli and Tetreault, 2015), or up to 500-1000 for greedy models that perform significantly below state-of-the-art accuracy.

While these speeds may be good enough for interactive systems that process a few sentences or documents at a time, they are clearly prohibitive if we need to do large-scale parsing, for example of large collections of documents retrieved from the Internet. The problem is even more serious in languages other than English that present extra challenges, such as free word order, crossing dependencies or rich morphology, where the computational requirements are much higher (Bohnet, 2010; Gómez-Rodríguez, 2016b).

Now that accurate parsing has largely been achieved, it is time to shift priorities and focus on how to make parsing faster while preserving accuracy, in order to bring parsing algorithms to the web scale. The goal of this project is, therefore, to develop new models, algorithms and techniques for syntactic parsing that will significantly improve its speed. To do so, in order to cover the widest possible range of practical settings, we will develop techniques based on two different sets of requirements: on the one hand, we will significantly improve the speed of state-of-the-art parsers, without incurring any loss of accuracy. On the other hand, FASTPARSE will explore new approaches that are able to parse even much faster, under the assumption that we are willing to sacrifice some degree of accuracy in order to obtain massive speed improvements. In both cases, the research will focus on processor-independent techniques that do not require specialized hardware, and it will aim for approaches that can be applied to multiple languages.

2 Methodology

To achieve these goals, three independent research lines are proposed, whose results can be applied separately or in combination.

Speeding up parsers with case-based reasoning The so-called Zipf's law, which describes the frequency of appearance of words in language, implies that there are a few very common words that account for a significant proportion of the tokens in a text. The same basic principle has been observed to hold for other linguistic units and constructions, like lemmas (Baroni, 2009), n-grams and phrases (Ha et al., 2002). This means that a parser that processes large amounts of text is likely to find a significant proportion of short phrases and constructions that it has already seen previously.

The idea of this research line is to exploit this fact to make parsing faster by using a variant of case-based reasoning, in such a way that when a parser is given a sentence, it will check whether it contains any short phrases that have been parsed previously. In this case, the previous syntactic analyses for those parts of the sentence can be directly re-used, instead of building them again.

An advantage of this approach is that it can be applied to practically any kind of parser (constituency and dependency parsers, grammar-based or data-driven, supervised or unsupervised) by re-using the adequate type of partial analysis. A challenge for this approach is sparsity: although the repetition of phrases is frequent in human languages, it is hardly frequent enough to ensure that a given input text will contain a significant proportion of previously seen fragments. This problem will be tackled in two ways: by making the sources of re-usable partial analyses as comprehensive as possible, and by giving the system generalization capabilities so that it can re-use analysis for phrases that "almost" match an input fragment, even if they are not identical. This will require the development of linguistic rules to determine which fragments can be considered equivalent from a syntactic point of view.

Cognitively-inspired chunk-and-pass processing Human language comprehension takes place under tight resource limitations, which Christiansen and Chater (2016) call the "Now-or-Never bottleneck": we need to deal with each piece of linguistic input in an eager way, processing it as it is received, before it is replaced by new input in our working memory. To successfully operate under these conditions, the human language processing system must be highly optimized

to compress and recode linguistic input as rapidly as possible. Therefore, in spite of the differences between web-scale parsing and the everyday language comprehension that humans need, there is much to be learned from human cognition, which co-evolved with natural languages (Deacon, 1997), if we wish to find ways to process them efficiently. In fact, as observed in Gómez-Rodríguez (2016a), recent parsing research is spontaneously arriving at solutions that increasingly resemble cognitive models of human processing, even when their intention is purely application-oriented.

This research line will adapt an idea from recent cognitive models of language processing, not previously applied to NLP, to significantly reduce both the CPU and memory usage of parsers, without affecting their accuracy. This idea is that of "chunk-and-pass" processing: Christiansen and Chater (2016) explain that, to deal with the "Now-or-Never bottleneck", the brain needs to eagerly compress and recode linguistic input into successively higher representation levels, as chunks of information at one level need to be passed to the next level fast enough to avoid being overwritten by further incoming chunks. We will explore models where a very fast chunking pass generates a compressed representation of the sentence in terms of chunks rather than words, which will then be passed to the parser. This means that the length of the sequences that have to be processed by the parser is much smaller, which will considerably reduce its time and memory requirements. The proposed mechanism to perform the compressing and recoding of the input, and the interface between the chunker and the parser, is chunk embeddings, i.e. continuous vector representations of chunks.

Exploiting annotation regularities for incremental constituency parsing Grammatical formalisms for natural language parsing face a trade-off between expressivity and parsing efficiency. Formalisms that allow for an exhaustive coverage of the linguistic phenomena observed in human languages tend to have a high computational cost (Gómez-Rodríguez, 2016b). For this reason, the most widely-used constituency parsers (like the Stanford and Berkeley parsers) are based on context-free grammar (CFG). This means that these parsers cannot handle some linguistic phenomena that are not context-free, but this is compensated for with greater efficiency than parsers that use more expressive formalisms.

This research line aims to greatly improve the efficiency of constituency parsers, at the expense of losing some degree of expressivity, by imposing additional restrictions on the trees they can generate. To achieve this, we will exploit the regularities that we can find in treebanks as a result of the grammatical characteristics of each language and annotation scheme.

We will study corpora to find such regularities, and then use the obtained data to define restricted shift-reduce parsers with their transitions tailored to fast parsing. Instead of being able to generate any possible context-free tree, these parsers will be specifically designed for the specific form of trees that can be found in practice in each training corpus, allowing them to work much faster.

3 Applications

Making web-scale parsing feasible, even without a massive deployment of computing resources, has the potential of enabling the use of syntactic parsing (and, therefore, of semantic information going beyond simple keyword matching) for technologies where it is currently unfeasible, for instance:

- Monitoring applications that scan the web for new texts pertaining to a specific topic of interest, such as technology watch systems, security applications for prevention of criminal and terrorist activity, or opinion mining systems that study the evolution of public opinion on a specific issue, product or brand.
- Semantic search and question answering for web search engines.
- The creation of semantic knowledge bases at an unprecedented scale, which could be used by all kinds of knowledge-intensive applications.
- The creation of massive-scale corpora, like the recently annotated English Books corpus (created by Google, presumably using immense computational resources, and not publicly available except for a very restricted subset of the data (Goldberg and Orwant, 2013)). Public access to corpora at this scale would be greatly valuable for linguistic

and sociological studies in any language (Gulordava and Merlo, 2015; Futrell, Mahowald, and Gibson, 2015; Ferrer-i-Cancho and Gómez-Rodríguez, 2016).

4 Staff and planning

The project's scientific staff consists of 2 PhD students and 2 postdoctoral researchers, together with the PI C. Gómez-Rodríguez. The project has begun on February 1, 2017, and has a total duration of 5 years.

The three lines described above will be undertaken independently and in parallel throughout the duration of the project, and their results will be validated on existing parsers and integrated into a new software suite for multilingual dependency and constituency parsing, which will be developed within the project. More information on the project can be found at the website `http://fastparse.grupolys.org`.

References

Baroni, M. 2009. Distributions in text. In *Corpus Linguistics: An International Handbook*. Mouton de Gruyter, pages 803–821.

Bohnet, B. 2010. Top accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 89–97. Coling 2010 Organizing Committee.

Choi, J. D. and A. McCallum. 2013. Transition-based dependency parsing with selectional branching. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Vol. 1: Long Papers)*, pages 1052–1062. Association for Computational Linguistics.

Christiansen, M. H. and N. Chater. 2016. The now-or-never bottleneck: a fundamental constraint on language. *Behavioral and Brain Sciences*, 39:e62, 1.

Crabbé, B. 2015. Multilingual discriminative lexicalized phrase structure parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1847–1856. Association for Computational Linguistics.

Deacon, T. W. 1997. *The Symbolic Species: The Co-evolution of Language and the Brain*. W.W. Norton.

Ferrer-i-Cancho, R. and C. Gómez-Rodríguez. 2016. Crossings as a side effect of dependency lengths. *Complexity*, 21(S2):320–328.

Futrell, R., K. Mahowald, and E. Gibson. 2015. Large-scale evidence of dependency length minimization in 37 languages. *Proceedings of the National Academy of Sciences*, 112(33):10336–10341.

Goldberg, Y. and J. Orwant. 2013. A dataset of syntactic-ngrams over time from a very large corpus of english books. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, pages 241–247. Association for Computational Linguistics.

Gómez-Rodríguez, C. 2016a. Natural language processing and the Now-or-Never bottleneck. *Behavioral and Brain Sciences*, 39:e74, 1.

Gómez-Rodríguez, C. 2016b. Restricted non-projectivity: Coverage vs. efficiency. *Comput. Linguist.*, 42(4):809–817.

Gulordava, K. and P. Merlo. 2015. Diachronic trends in word order freedom and dependency length in dependency-annotated corpora of Latin and ancient Greek. In *Proceedings of the Third International Conference on Dependency Linguistics (Depling 2015)*, pages 121–130, Uppsala, Sweden. Uppsala University.

Ha, L. Q., E. I. Sicilia-Garcia, J. Ming, and F. J. Smith. 2002. Extension of Zipf's law to words and phrases. In *COLING 2002: The 19th International Conference on Computational Linguistics*, pages 315–320.

Kummerfeld, K. J., D. Hall, R. J. Curran, and D. Klein. 2012. Parser showdown at the wall street corral: An empirical investigation of error types in parser output. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1048–1059. Association for Computational Linguistics.

Rasooli, M. S. and J. R. Tetreault. 2015. Yara parser: A fast and accurate dependency parser. *CoRR*, abs/1503.06733.