Towards a Psycholinguistically Motivated Performance-Based Parsing Model

Balázs Indig
(Supervisor: Gábor Prószéky)
indig.balazs@itk.ppke.hu

***Abstract*—In this paper I present a new paradigm and framework for syntactic and semantic analysis, which is based on the following principles: (1) *Psycholiguistically* motivated means, my model hold on to the inner workings of human language processing as much as possible. (2) As a *performance-based* system the model tries to process all natural language utterances (a few sentences long coherent texts) that occur in (written) texts. My focus is not the handling of theoretically existing but in practice rather rare phenomena. Instead, I consider and try to interpret any – no matter how badly formed or agrammatic – text that appears as a natural language utterance. (3) In my model the parser *processes the text strictly left-to-right incrementally* and does not utilize or reference the parts that succeed the current position. (4) The general architecture of the parser framework is naturally *parallel* from the beginning, in contrast to the traditional approach, where the analysis is generated at the end of a pipeline of modules. Here the program processes the actual word using parallel threads (a morphological analyser thread, a corpus statistics thread, etc.). These threads analyse each word together at the same time and communicate to correct each other's errors and to make a final decision in the analysis. (5) The framework's processing and representational units are not individual sentences, rather *utterances* consisting of one or more sentences. This enables the unified handling of intra- and intersentential anaphoric relations. (6) In accordance with the principles described so far, in order to be able to handle all the different phenomena at the same time the *representation* is not necessarily a tree, but a connected graph containing different types of edges. After describing its theoretical foundations, I present a *pilot implementation* of the framework. My pilot program illustrates the basic principles and performs some of the analysis steps conforming to the theory.**

***Keywords*-psycholinguistically-motivated; performance-based, incremental, semantic parsing**

I. Introduction

In this paper I present the foundations of a performance-based parser framework, that attempts to process real language data and tries to stay close to human language processing as possible. I process text simultaneously with the generation of the utterance (i.e., strictly left-to-right) and try to use all the information that is necessary for its interpretation, even if it is not well-formed in the traditional grammatical sense. The parser therefore is limited to *assume* some not yet available parts, based on those which are heard or read until the utterance has come to an end. This does not mean that there are no utterances which "circumvent" the most likely analysis, sometimes forcing even the human parser to backtrack, but in everyday communication humans seem to follow the maxims of Grice [1], and avoid these constructions. They typically occur in jokes or intentional misrepresentations[1]. I cannot use a traditional PoS-tagger which utilizes global information about all elements of the sentence when determining the tags of an incomplete segment. I'm using an n-gram model based on data acquired from a large corpus, which can provide probabilistic information about the possible tags of the current word based on the *preceding* words only. It also continuously yields information on how typical the current span of text is, and what kind of units are expected or required at the current position. I need larger parts of text as I attempt to uniformly handle ellipsis, anaphora/coreference and coordination which may or may not span beyond sentence boundaries. My basic unit is not the sentence but an utterance/paragraph (few sentences long coherent texts). Similarly to Discourse Representation Theory [2], in our group, we construct a unified semantic representation containing all entities referred to in the processed text but in contrast to discourse representation structures, our representations are "ontologically promiscuous" in the sense of [3] and contain reified abstract entities such as eventualities and propositions in order to handle logical complexity. I employ traditional tools/modules (morphological analyser, identification of verbal and other constructions, corpus frequencies, ontologies etc.) as so-called **resource threads** constantly running and working concurrently. This allows the different tools to communicate, complement or even override each other at every parsing step in order to correct each others errors. The architecture of the analyser therefore is parallel instead of the traditional pipeline design. To store information provided by words already processed, I have a second kind of threads, the so-called **structure threads**. These threads are initiated and closed by particular words, and they realize different "offers" and "demands". To operate this offer-demand system I need a certain description of the main phenomena of the language, namely such a grammar which enumerates all possible roles for linguistic units (e.g. a noun in nominative case or a comma). The output of my parser will contain syntactic and semantic information to identify both participants and events, building a representation of the whole utterance, and formulating statements about who does what, where and when. My offer-demand mechanism and Combinatory Categorial Grammar are both strongly lexicalist,

[1] In my group, we have begun to examine large corpora to identify these complicated constructions that are in the focus of modern grammatical theories and to collect information about the frequency of their occurrences to prove that they are quite rare in everyday life.

B. Indig, "Towards a Psycholinguistically Motivated Performance-Based Parsing Model,"
in *PhD Proceedings Annual Issues of the Doctoral School - 2014*, T. Roska, G. Prószéky, P. Szolgay, Eds.
Faculty of Information Technology and Bionics, Pázmány Péter Catholic University.
Budapest, Hungary: Pázmány University ePress, 2014, pp. 133-136.

proposing that offer-demand or functor-argument construction processes are governed by deep lexical information rather than by an extra set of syntactic rules. Hungarian being a configurational language, I must deal with arbitrarily ordered offer-demand pairs in the case of verb arguments, something that is dealt with in CCG for example by relaxing category definitions [4].

II. Background

Generative models do not provide an effective solution from the computer's perspective to analyse texts. This is partly due to the fact that these models are concerned primarily with ideal speakers-listeners and focus on sentences rather than larger text segments [5]. In addition, the notion of "efficient parsing" does not belong to competence, but to performance. The term *performance-based* means that everything that actually occurs as text needs to be processed. Constructions that could theoretically exist but in practice do not are less important. Therefore, I investigated the most important performance-based parsers [6] and those theoretical approaches which state that efficient parsing belongs to the area of competence (e.g. [7]). Several efficient dependency parsers exist, although many of them are based on theories which do not agree with our concepts: MaltParser [8], Stanford Parser [9], finite-state dependency parser [10], which are in fact aimed at the description of the relationship between different linguistic units, but are heavily tied to the separate processing of subsequent sentences. A dependency description for Hungarian was developed in the late 1980s for the DLT system [11]. Another important grammatical source for Hungarian, Szeged Treebank is also available in dependency format [12], from which a statistical dependency parser was also created [13]. I think that many phenomena of language can not be described properly with dependency relations. The MetaMorpho rule-based parser [14], the most thorough Hungarian parser to date is also at our group's disposal and I am currently using its verb frame constructions along with dependency descriptions of noun phrases. None of the mentioned parsers and resources handle the task of disambiguation properly and all have bad fault tolerance: a single out-of-vocabulary or unusual word may result in the failure of the entire analysis. I also found that all the presently known parsers do one-way, pipelined processing almost exclusively, i.e. there is no communication back and forth between the different processing levels, a tool begins to operate only after the previous tool completed its task, and takes the input over "as is" from the previous tool together with the possible errors. Therefore, in our parser we abandon the idea of the traditional pipeline architecture. One of the reasons is that errors of the lower level modules in a pipeline are carried over to higher levels without correction and get amplified later and weaken the quality of later modules. Typically, parsers try to address this problem by applying a simple frequency based filter, but this solution eliminates unusual analyses even when they would actually be the correct ones. As Prószéky points out [15], decisions taken during the human analysis of linguistic structures can override the lexicon. All the aforementioned parsers are using exclusively their preliminary knowledge in their decisions. In my parser, we intend to take an approach in which the unusual structure of the actual input does not conflict with either preliminary statistics or the often misleading output of mechanistic rules. My initial hypothesis is that in the listener's/reader's mind two systems exist: one which relies upon the learned structures and another that makes current decisions and is able to perform real-time processing even if the learned structures contradict each other in their grammaticality (e.g. contain mismatching features). Therefore, my parsing algorithm is looking for a kind of consensus between different pieces of linguistic knowledge [16]. Thus, as in human language processing, the analysing and interpreting modules work in parallel and in close cooperation in my parser [17].Current competence-based models do not assume any cooperation with non-linguistic information processing systems. I can say, however, that the performance can not be separated from other cognitive processes which have impact on language, therefore, from the very first step of analysis the parser relies on the simultaneous handling of some linguistic and non-linguistic modules (world knowledge, mood, etc.) on various levels, depending on how fine-grained the model is. In the grammar, various levels of processing (morphological analysis, identification of verbal constructions, corpus frequencies, world ontologies, etc.) work concurrently in separate resource threads in the background and can complement or even override each other at every parsing step.

III. Architecture

Basically, two language element-initiated thread types seem necessary. An *offer* thread provides information on the current element (e.g. an element is in nominative case), and a *demand* thread is looking for a required element with a specific property (e.g. a possession noun looks for its possessor, a postpositional particle needs a noun, a determiner seeks the NP head, a transitive verb needs its object etc.) To overcome the problem of rejecting non-frequent but valid constructions I propose an extralinguistic decision system, which considers relevant information from the corpus frequency thread on the one hand (how well the analysis corresponds to usual patterns), and rule-based analysis on the other hand. As I process the actual word I consider the frequency relationships between the word and the preceding words. For example, after the word *esik* (to fall) I expect *sz* (word) as subject because it is relatively frequent (15% of all possible subjects for *esik*). After this step, I expect a noun phrase with the case inflection suffix *-rl* because it is very frequent (appears in 9 cases out of 10 after *esik sz*), forming an idiom meaning something is talked about. Based on the processing efficiency of the human parser our framework is trying to avoid the combinatorial explosion, so it uses aggregated statistics as "preliminary knowledge" mined from corpora. Frequent constructions are used as whole units without real-time analysis and are inserted into the representation. This method is called *caching* in IT, but in psycholinguistics is also well-known

in the sense of human language processing, and are called "**Gestalts**". The desired output is a network of semantic relations built from the underlying text, which can answer questions, and can generate statements which are contained only implicitly in the original text. I favour the use of **non-tree-based** dependency graphs as coreferences and relative pronouns are not necessarily realized in the same sentence and marking them with an edge can ruin the tree structure producing a directed acyclic graph. To cope with **ellipsis phenomena** I allow the threads to run across sentences. I argue that analysis should not stop at the sentence level, as utterances are the natural units in human communication. The topics of subsequent sentences can be identical, and therefore in natural human communication can be and are in fact mostly omitted (ellipsis phenomena), which causes most parsers to lose track of the analysis. It is very important to identify the participants in the text and to determine their **coreference** relations (which entities in the text refer to the same entities in the real world). In other words, which participants are "new" at the point of their introduction in the text and which refer to already mentioned ones, and what kind of relationship holds among them. Essentially all nouns and nominative or accusative phonologically empty arguments inferred from verb suffixes can be considered as **event participants**. According to the principles of **neo-Davidsonian event semantics** [18] even events (verbs) may be participants, as I can refer to them as well. In order to aid the resolution of such references, all the participants in all the preceding sentences are stored in our system. Further threads utilize the rich descriptions of **lexical units** and **lexico-syntactic constructions** available from the MetaMorpho parsers databases. An "offer" type thread annotates units with semantic features (such as *animate, human, abstract* etc.), available for 118,000 words and multi-word expressions. On the other hand, a "demanding" type thread, by using 35,000 of MetaMorphos open construction rules proposes connections between verbs and their possible arguments and nouns or adjectives and their arguments (e.g. *hostility towards something, interested in something* etc.)

I am also experimenting with methods to aid the prediction of verb-argument connections based on verb-noun corpus co-occurence data from the Verb Argument Browser [19] and ontological information [20]. In our group, we are working on a method to map verb arguments from the corpus to Hungarian WordNet in order to identify generalized semantic classes that correspond to the verb arguments **semantic types**.

IV. Current Work

A. Practical problems

As a first step of the approach, I have identified and implemented some formalized operation types to be used during processing. First, I had to find those elements which predict what sort of elements can come after them. For example, a determiner predicts some nominal element at the end of the construction starting with the actual determiner. There are elements which fulfil an earlier prediction, e.g. a verbal argument fulfils a slot in the argument frame of a verb that occurred earlier in the input stream. If the verb itself comes later than one of its arguments, this argument construction can fulfil automatically the right slot in the verbal frame. There are other sorts of operations: conjunction structures, for example, can be identified only when a conjunctive element arises. It can be an 'and', an 'or' or a comma: they introduce the next element of the conjunctive construction. When the system identifies an element like this, it should modify the representation of the previous element, making that element the first one in a conjunctive structure. I treat a conjunction as one unit, without deciding whether it has a head or not. A working pilot implementation of our approach attempts to deal with frequent, fundamental phenomena that are not necessarily easy to handle in other frameworks, such as linking a separated verbal prefix and the verb stem, linking parts of possessive constructions, identifying enumerations/coordination as complex units, identifying the actual role of a comma (whether it triggers e.g. another clause, an enumeration, a parenthetical/interjection or an apposition), or identifying the scope of a negation. Currently in our group, we are working on the description and formalisation of further such important linguistic phenomena in Hungarian, taking into account the constraint of left-to-right incremental processing, concerning for example exocentric constructions where the phrase does not have a head, which is a challenge for dependency parsers.

B. Implementation

My prototype implementation proceeds left-to-right taking the elements (currently words) of the input text one by one. It processes the subsequent word taking all the information provided by the resource threads into account, then either (a) *stop*s or (b) *start*s some threads or (c) *leave*s some threads unchanged or (d) *creates an edge* in the representation graph. One kind of information I use are morphological patterns to identify the features of words that will be crucial in the syntax: suffixation information, parts-of-speech and word lemmas. The aforementioned operations are currently triggered by these morphological patterns, which the actual word matches. I allow multiple morphological patterns to match a single entry where necessary, as each word suffix can contain multiple valuable information (e.g. possession and case), then all of the corresponding structural threads must start.

Each element, that has – or introduces some later element that will have – reference is noted in the list of participants. In this list, each mention of a participant is linked with coreferring mentions. This includes the special cases of pronominal verb arguments that may or may not be present in Hungarian, since the inflection on the verb carries enough information to identify their number and person. For example, the 3rd person singular form of a finite verb introduces a participant that may or may not have surface realisation in the sentence, so at the time of processing the verb the parser introduces a new participant marked as "phonologically empty". If a surface realisation of this participant actually appears in the input at a later point, the parser links it to the formerly introduced participant.

Currently the parser builds a tree over the sentence units using the recognised dependency relations combined with verb frame constructions. In the future, we will add long distance dependencies and coreference relations that will turn the representation into a less strict graph structure that is no longer a tree.

My prototype has been tested on summaries of **news articles** taken from a Hungarian news portal's RSS feed (www.inforadio.hu). The 2-3 sentence long paragraphs describe single political or economic events. Their syntactic complexity is close to what I would like to model, thus they serve as an adequate input to our parser. First, the input text is lemmatised as a pre-processing step (playing the role of a simple lexical lookup in other, less inflecting languages).

V. Future work

Among the aformentioned future development goals, some of the presented tools are not production ready, they still need to be developed for integrating in the parser program. While the parser's internal representation is still not stable I'm working toward stabilizing it and keep the way open for possible extensions for other languages for example English.

VI. Conclusion

In this paper, I described ongoing research which aims at a psycholinguistically motivated, performance-based, strictly left-to-right, utterance-based parallel processing parser framework. In my view, currently existing parser models are not sufficient at modeling such aspects of human language processing. I laid down the fundamental principles of our linguistic theory and presented some details of a pilot parser implementation.

Acknowledgment

I would like to acknowledge my supervisor, Gábor Prószéky and my collegues Bálint Sass and Márton Mihály for his kind help and his knowledge on this multidisciplinary linguistic field. Also the support TMOP-4.2.1.B 11/2/KMR-20110002 and TMOP-4.2.2/B 10/120100014 is kindly acknowledged.

References

[1] H. P. Grice and G. Harman, *Logic and conversation*. Encino:Dickenson, 1975.
[2] H. Kamp and U. Reyle, *From discourse to logic: Introduction to modeltheoretic semantics of natural language, formal logic and discourse representation theory*. Springer, 1993.
[3] J. R. Hobbs, "Ontological promiscuity," in *Proceedings of the 23rd annual meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 1985, pp. 60–69.
[4] M. Steedman and J. Baldridge, "Combinatory categorial grammar," in *Non-Transformational Syntax*, R. Borsley and K. Börjars, Eds. Blackwell, 2011, pp. 181–224.
[5] N. Chomsky, *Syntactic structures*. The Hague:Mouton, 1957.
[6] H. Bunt, P. Merlo, and J. Nivre, *Trends in Parsing Technology*. Dordrecht:Springer, 2010.
[7] B. L. Pritchett, *Grammatical competence and parsing performance*. University of Chicago Press, 1992.
[8] J. Nivre, *Inductive dependency parsing*. Springer, 2006.
[9] M.-C. De Marneffe, B. MacCartney, and C. D. Manning, "Generating typed dependency parses from phrase structure parses," in *Proceedings of LREC*, vol. 6, 2006, pp. 449–454.
[10] K. Oflazer, "Dependency parsing with an extended finite-state approach," *Computational Linguistics*, vol. 29, no. 4, pp. 515–544, 2003.
[11] G. Prószéky, I. Koutny, and B. Wacha, "A dependency syntax of Hungarian," *Metataxis in Practice (Dependency Syntax for Multilingual Machine Translation)*, pp. 151–181, 1989.
[12] V. Vincze, D. Szauter, A. Almási, G. Móra, Z. Alexin, and J. Csirik, "Hungarian dependency treebank." in *LREC*, 2010, pp. 1855–1862.
[13] J. Zsibrita, V. Vincze, and R. Farkas, "magyarlanc: A toolkit for morphological and dependency parsing of hungarian," in *Proceedings of RANLP*, Hissar, Bulgaria, 2013, pp. 763–771.
[14] G. Prószéky, L. Tihanyi, and G. Ugray, "Moose: A robust high-performance parser and generator," *Proceedings of the 9th Workshop of the European Association for Machine Translation*, pp. 138–142, 2004.
[15] G. Prószéky, "Számítógépes morfológia," in *Morfológia (Strukturális magyar nyelvtan III)*, F. Kiefer and Z. Bánréti, Eds. Akadémiai Kiadó, Budapest, 2000, vol. 3, pp. 151–1064.
[16] P. Csaba, *Mondatmegértés a magyar nyelvben*. Osiris Kiadó, Budapest, 1999.
[17] C. Valéria, "Az olvasó agy," *Akadémiai Kiadó, Budapest*, 2006.
[18] P. Terence, "Events in the semantics of English: A study in subatomic semantics," 1990.
[19] B. Sass, "The Verb Argument Browser," *11th International Conference on Text, Speech and Dialog (TSD)*, pp. 187–192, 2008.
[20] M. Mihály, C. Hatvani, J. Kuti, G. Szarvas, J. Csirik, G. Prószéky, and T. Váradi, "Methods and results of the Hungarian WordNet project," in *Proceedings of the Fourth Global WordNet Conference (GWC-2008)*, T. A., C. D., V. V., C. Fellbaum, and P. Vossen, Eds. Szeged, University of Szeged, 2008, pp. 311–321.