Filling Constructions: Applying Construction Grammar in the Kitchen[1]

Robert Porzel, Vanja Sophie Cangalovic and John A. Bateman
Departments of Computer Science & Linguistics, University of Bremen

Keywords: Instructions, Parsing, Language Understanding

Declarative texts are the traditional subject for constructional analysis (Langacker, 1987). Imperatives, however, have received less attention outside the domain of speech act theory (Austin, 1962). By now, several national and European projects seek to understand written recipes using computational construction grammar-based analysis. Recipes, i.e. instructions on how to prepare specific dishes, constitute an interesting domain for linguistic analysis and a challenging enterprise for natural language understanding systems. In the following we will highlight some of these challenges: Usually, subjects are almost always oanmitted in the imperative instructions, and are assumed to be the recipient/reader of the command, many clauses also contain no direct object or determiners., e.g.:

"Remove from heat and add chopped cilantro"[2]

While these phenomena also occur in declarative texts, we find them with very high frequency in recipes. We have therefore extended the Basic English Grammar (van Trip, 2017) by including an DISCOURSE-ADDRESSEE-IMPLICIT-SUBJECT-VERB-CXN that can apply after the MORPH-, VP-, NONPERFECT-, NON-PROGRESSIVE-, PRESENT-TENSE- and LEX-CXN have done so. Furthermore, we employ an IMPERATIVE- CXN that calls for the existence of a clause with two subunits - a verbal phrase and a nominal phrase. This nominal phrase, in turn, is required to have no phonological form, indicating that it has been added as an implicit subject by the DISCOURSE-ADDRESSEE-IMPLICIT-SUBJECT-VERB-CXN beforehand. More traditional grammars could have problems with such empty forms, but Fluid Construction Grammar (Steels, 2017), handles this requirement without difficulties. Additional phenomena that need to be heeded stem from the operations that are performed while executing the instructions. For example, entities denoted by count nouns are transformed into masses and occur correspondingly as mass nouns thereafter, e.g.:

"Cut three tomatoes into pieces. Now mix tomato and garlic."[3]

The biggest computational challenge lies in the recovery of implicit information. It is crucial for subsequent processing steps, i.e. robots executing these instructions, to denote that some parts of the picture are missing. This brings about a twofold challenge for the design of the grammar and the parsing of texts. First we need constructions that can apply even if specific constituents are missing, e.g. subjects or direct objects, but they also need to know that normally there would have been something there so that later processing modules can seek to fill the gaps.

References

Austin, John. How to Do Things with Words: The William James Lectures delivered at Harvard University in 1955, 1962 (eds. J. O. Urmson and Marina Sbisà), Oxford: Clarendon Press.

Langacker, Ronald. 1987. Foundations of Cognitive Grammar. Stanford University Press.

van Trijp, Remi (2017). A Computational Construction Grammar for English *The AAAI 2017 Spring Symposium on Computational Construction Grammar and Natural Language Understanding Technical Report, no. SS-17-02*, pp. 266-273. Stanford: AAAI.

Steels, Luc. 2017. Basics of Fluid Construction Grammar. *Constructions and Frames* 9(2). 178–225.

[1] The project "MUHAI" leading to this application has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 951846.

[2] Kitchen Stoires, https://www.kitchenstories.com/ last accessed 01/25/2021

[3] Kitchen Stoires, https://www.kitchenstories.com/ last accessed 01/25/2021