The Proof is in the Almond Cookies

van Trijp, Remi; Beuls, Katrien; Van Eecke, Paul

Published in:
Narrative-based Understanding of Everyday Activities

Publication date:
2024

The Proof is in the Almond Cookies

A Case Study on Narrative-Based Understanding of Recipes

Remi van Trijp[1], Katrien Beuls[2], and Paul Van Eecke[3]

[1]*Sony Computer Science Laboratories Paris, France*
[2]*Faculté d'informatique, Université de Namur, Belgium*
[3]*Artificial Intelligence Laboratory, Vrije Universiteit Brussel, Belgium*

This paper presents a case study on how to process cooking recipes (and more generally, how-to instructions) in a way that makes it possible for a robot or artificial cooking assistant to support human chefs in the kitchen. Such AI assistants would be of great benefit to society, as they can help to sustain the autonomy of aging adults or people with a physical impairment, or they may reduce the stress in a professional kitchen. We propose a novel approach to computational recipe understanding that mimics the human sense-making process, which is narrative-based. Using an English recipe for almond crescent cookies as illustration, we show how recipes can be modelled as rich narrative structures by integrating various knowledge sources such as language processing, ontologies, and mental simulation. We show how such narrative structures can be used for (a) dealing with the challenges of recipe language, such as zero anaphora, (b) optimizing a robot's planning process, (c) measuring how well an AI system understands its current tasks, and (d) allowing recipe annotations to become language-independent.

Keywords: narrative, grounded language understanding, human-centric AI, mental simulation

Please cite as:
van Trijp, R., Beuls, K. & Van Eecke, P. 2024. The Proof is in the Almond Cookies. In Steels, L. & Porzel, R. (eds). *Narrative-based Understanding of Everyday Activities: A Cookbook.* Venice: Venice International University. Pages 59–77.

1 Introduction

This paper explores what kind of grounded language processing model is needed for enabling robots or computational cooking assistants to support human chefs in the kitchen. Such human-centric AI assistants would be of great benefit for society because they could sustain the autonomy of aging adults or people with a physical impairment, or they could reduce the pressure on professional chefs who have to work in high-stress situations. We propose a novel approach to computational recipe understanding that mimics the narrative-based sense-making process of humans (Bruner, 1991), which may lead to more intuitive and meaningful human-robot interactions.

We first discuss the main challenges of recipe understanding and related work before introducing *narrative-based understanding* (section 2, also see Van Eecke et al., 2023). We then illustrate the approach through a concrete case study on an English recipe for almond crescent cookies, shown in Figure 1. Finally, we evaluate the benefits and the scalability of the approach, and provide more

Almond Crescent Cookies

PREP TIME	COOK TIME	TOTAL TIME
15 mins	20 mins	35 mins

SERVINGS
30 servings

Ingredients
- 226 grams butter, room temperature
- 116 grams sugar
- 4 grams vanilla extract
- 4 grams almond extract
- 340 grams flour
- 112 grams almond flour
- 29 grams powdered sugar

Instructions
- Beat the butter and the sugar together until light and fluffy.
- Add the vanilla and almond extracts and mix.
- Add the flour and the almond flour.
- Mix thoroughly.
- Take generous tablespoons of the dough and roll it into a small ball, about an inch in diameter, and then shape it into a crescent shape.
- Place onto a parchment paper lined baking sheet.
- Bake at 175°C for 15-20 minutes.
- Dust with powdered sugar.

Figure 1: This Figure shows an English recipe for almond crescent cookies, adapted from `https://www.simplyrecipes.com/recipes/almond_crescent_cookies/`.

information about resources made available to the community for researchers who wish to experiment with narrative-based understanding.

1.1 Challenges

Recipe understanding is a challenge for robotics because kitchens are rich and dynamically changing environments (Bollini et al., 2013). From a linguistic perspective, recipes come with their own genre-specific syntax and semantics (see a.o. Cotter, 1997; Gerhardt et al., 2013; Cani, 2022) that challenge traditional NLP solutions, of which we summarize the most important ones here:

- *How-to instructions:* Recipes use procedural language, such as imperative commands, which leads to reduced performance of off-the-shelf parsers (Tellex et al., 2020).
- *Zero anaphora:* Recipes are abundant with zero anaphora (e.g. no direct object in the phrase "mix thoroughly") because cooking takes place in an actual kitchen that provides the necessary context for filling in the blanks.
- *Dynamic Environment:* Kitchens are dynamic environments in which entities are changed into "resultant objects" – often without explicit mention of that happening. For instance, the almond cookie recipe (Figure 1) introduces the phrase "the dough" for the first time in its fifth instruction without making explicit that it is the resultant object of mixing together various ingredients such as butter, sugar and flour.
- *Complex Semantics:* Recipes require careful management of time, measurement and ordering. Instructions can be explicit (such as "340 grams flour" or "for 15-20 minutes"), but recipes also often use vague measurements ("generous tablespoons") and evaluative phrases ("until

light and fluffy") that require a tight integration of language processing and sensorimotor perception.

1.2 Related Work

Computational recipe understanding and other tasks in *Digital Gastronomy* (Zoran, 2019) have always enjoyed academic interest (see e.g. the Computer Cooking Contests; Najjar and Wilson, 2017), but especially in the past few years there has been a surge of attention for the broader field of *food computing* (Harper and Siller, 2015; Min et al., 2019). This surge is driven on the one hand by the explosion of large-scale online data such as recipes and cooking videos; and on the other hand by the breakthroughs in deep learning for handling such large data (e.g. LeCun et al., 2015).

Most research therefore focuses on aggregating and cleaning up the data; and on the creation of datasets, benchmarks, representations and classification systems for food-related information (e.g. Smith and Lin, 2012; Kicherer et al., 2018; Yagcioglu et al., 2018; Marin et al., 2019; Popovski et al., 2019; Jiang et al., 2020; Tian et al., 2021). This information is then used for various tasks such as recipe generation (e.g. Jabeen et al., 2020; Wang et al., 2022), recipe recommendation (e.g. Haussmann et al., 2019; Tian et al., 2022), question-answering systems (e.g. Manna et al., 2020; Khilji et al., 2021), and so on. Ultimately, such systems aim to provide an appropriate response to a particular input, such as proposing relevant recipes based on the user's preferences.

Even though such work is relevant to the present study, their goals only require a shallow understanding of recipes, while our objective is to parse recipes in such a way that a robot can successfully execute it (or more generally speaking, that the robot can successfully execute instructions). This objective requires adequate systems for *grounded* (Harnad, 1990) *natural language understanding* (NLU; Allen, 1994, also see Tellex et al., 2020, for a survey).

Despite a longstanding research history going back to the 1970s (e.g. Winograd, 1971; Hart and Nilsson, 1972), a recent benchmark study has shown that grounded language understanding is still a largely unsolved problem (Shridhar et al., 2020). That is not to say that no progress has been made: thanks to more sophisticated language technologies and the increasing availability of online data, the field has moved away from limited sets of natural language instructions, and has instead set its ambition on mapping open-ended instructions from the web onto everyday manipulation tasks (Tenorth et al., 2010).

In the cooking domain, several prototypes and experiments have been reported (Sugiura et al., 2010; Beetz et al., 2011; Bollini et al., 2013; Bezaleli Mizrahi et al., 2023). These studies are usually performed from the perspective of robotics, and mainly examine how existing NLP techniques can be repurposed for the generation of executable robot plans (Tenorth et al., 2010). Language processing therefore typically involves translating instructions onto syntactic parse trees from which semantics can be inferred; or more recently, applying neural network models for directly mapping sentences onto formal semantic representations (Tellex et al., 2020).

2 Narrative-Based Understanding

We propose to treat recipes as a form of narrative, taking inspiration from discourse-analysis studies in linguistics (Cotter, 1997) and recent work on the value of narratives for human-centered AI (e.g. Szilas, 2015; Blin, 2022; Steels, 2022; Van Eecke et al., 2023). Narratologists divide a narrative into three interconnected layers (Bal, 1985), illustrated in Figure 2:

1. The *fabula* (or *story*) is a collection of facts, events and actions;

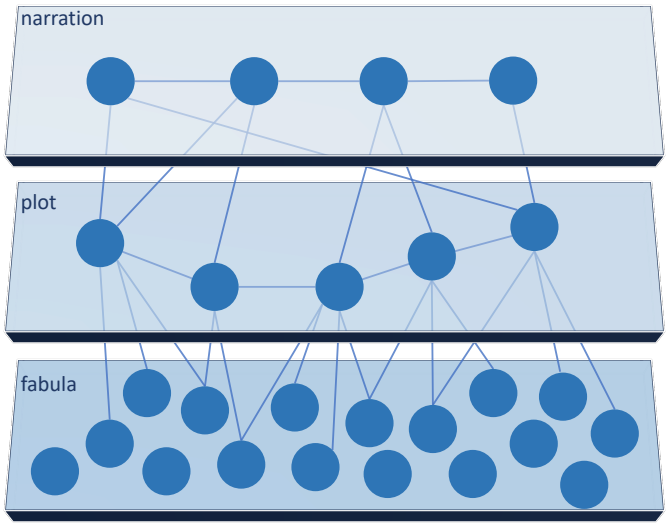


Figure 2: A narrative is a three-layered structure consisting of a fabula, plot, and narration. Narrative-based understanding involves constructing the plot using the narration and the fabula, thereby integrating language processing, memory, mental simulation, perception, and so on.

2. The *plot* (or *syuzhet*) is a structure that arranges the relevant items of the fabula in a causal network of events that lead to a conclusion (called *narrative closure*; Carroll, 2007);

3. The *narration* (or *discourse*) is how the narrative is presented.

2.1 Recipes as Narratives

Let us apply these three layers to recipes-as-narratives. There are two observable layers: the *fabula* and the *narration*. The *fabula* is so vast (i.e. most of its content is irrelevant such as ingredients that won't be used) that a cooking agent can only maintain a partial model, which it obtains through sensorimotor processing and retrieving facts from memory (e.g. which drawer contains the cutlery). While the fabula can be considered as the background against which the narrative should be situated, the *narration* concerns how the narrative is presented, which can be a written recipe, a cooking video, a dialogue, and so on. This layer is typically analyzed using (multimodal) language processing techniques.

At the heart of narrative-based understanding is the *plot*, which is invisible to the cooking agent and which therefore has to be constructed. The plot is a rich content model in which the relevant elements of the fabula are arranged in a causal network of events. By integrating the diverse and often fragmented and ambiguous input from various knowledge sources (such as language processing, vision and pattern recognition, mental simulation, action monitoring, ontologies, knowledge graphs, and so on), the plot provides a coherent and structured path towards the goal of the narrative (in our case study: delicious almond crescent cookies).

In the case of cooking, our main guideline for constructing the causal network of events is the narration. In the simplest case, the narration follows the same order as the plot, but even for recipes there exist many variations (Cotter, 1997). The recipe for almond cookies, for instance, starts with

a flash-forward by stating that there will be 30 servings. Other recipes may rely on base recipes (subplots) or include alternative ways to prepare a dish.

Important to note is that the construction of the plot is not the main goal of narrative-based understanding: it rather serves to find *narrative closure* (Carroll, 2007), which is the state in which the plot arrives at a satisfactory conclusion. In the case of recipe understanding, narrative closure is obviously achieved if the desired food is ready to be served.

2.2 Language as a Form of Action

Just like narratives involve the active construction of a plot in order to make sense of reality (Bruner, 1991), functional theories of linguistics have considered language to be a form of action ever since the influential works of Wittgenstein (1953), Austin (1962) and Searle (1969). The instructions found in recipes are textbook examples of such *speech acts*: linguistic expressions that invite the addressee to perform a (mental) action.

In robotics and grounded language understanding, those actions take the form of plans that can be simulated or executed by a robot. Traditional approaches typically involve a pipeline going from linguistic expressions to truth-conditional semantic representations (Eckardt, 2006; Tellex et al., 2020), which are then mapped onto an executable robot plan. For instance, the phrase "take the dough" can be associated with the logical form $\exists x : \{DOUGH(x) \wedge TAKEN(x)\}$ ("there exists an x that is dough and that is taken"), which (using temporal logics; Kress-Gazit et al., 2018) can be specified as becoming True once the robot executes the correct operation and takes the dough. This formal semantic specification is then used for generating an executable robot plan (e.g. Beetz et al., 2011; Bollini et al., 2013; Sugiura et al., 2010).

In our approach, we dispense with an intermediary truth-conditional representation and propose that the meaning of a sentence (or indeed, the meaning of the recipe as a whole) *is* an executable robot plan. For example, the phrase "take generous tablespoons of the dough and roll it into a small ball" in the almond crescent cookie recipe directly maps onto an actual operation in which the cooking agent uses a tablespoon as a tool of measurement for making several spheres made of dough.

2.3 Personal Dynamic Memory

Narratives are *personal* as they are based on past experiences, individual beliefs and values, and on which perspective is taken (Steels, 2020; Van Eecke et al., 2023). For instance, if at the world cup football a small nation eliminates one of the tournament's favourites, their supporters may praise their team's courage and efficient counter tactics, while the losing side might condemn them for playing a defensive "anti-football" game.

Narratives are therefore not constructed out of the blue, but are integrated into a *personal dynamic memory* (PDM; Steels, 2020). A personal dynamic memory consists of persistent knowledge (e.g. an agent's linguistic inventory, its ontology, and so on) and of past experiences and past narratives. The more cooking experience an agent has acquired throughout its lifetime, the easier it will be able to construct a recipe's narrative.

3 Almond Crescent Cookies

There is an English proverb that goes *the proof is in the pudding*, which comes from the older saying *the proof of the pudding is in the eating*. This expression used to mean quite literally that you have

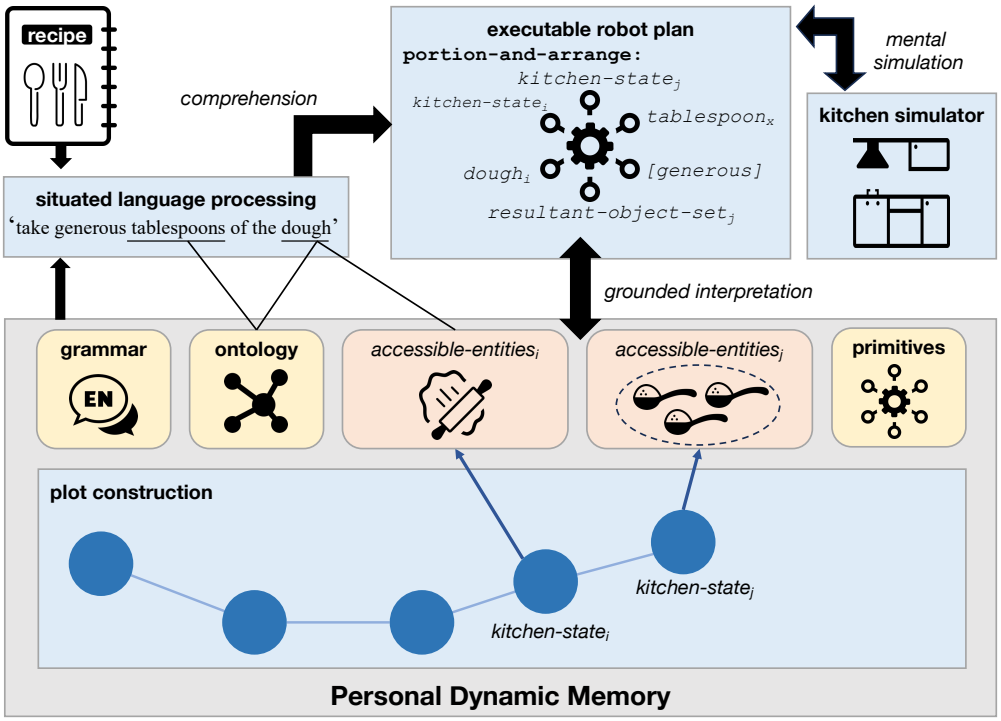


Figure 3: This figure illustrates a single cycle in the construction of the recipe's plot. On the top left: while parsing, the language processor has access to the cooking agent's grammar, ontology, and the entities that are currently under its attention (*accessible-entities*$_i$). Comprehension results in a partial executable robot plan (here the operation *portion-and-arrange*). Through interaction with a kitchen simulator and the agent's personal dynamic memory, a complete plan is generated and executed, leading to a new plot beat (*kitchen-state*$_j$), which includes new accessible entities (tablespoons of dough). The cycle can then repeat itself with the next instruction until the recipe is finished.

to try out food to know how tasty it is, and nowadays it can be used to say that you can only know the value or quality of something through direct experience or by obtaining concrete results. The same goes for evaluating the value of our narrative-based approach to recipe understanding.

While the previous section offered a conceptual, implementation-independent overview of narrative-based understanding, we will now proceed with a specific operationalization through a concrete case study on an English recipe for almond crescent cookies. This section will make heavy use of Figure 3, which illustrates one cycle of the back-and-forth between language processing, semantic interpretation, mental simulation, and personal dynamic memories. Interested readers can find more technical details at the web demonstration of our case study at `https://ehai.ai.vub.ac.be/demos/recipe-understanding`, and the recipe execution benchmark which we developed for evaluating our approach (see section 4).

3.1 Situated Language Processing

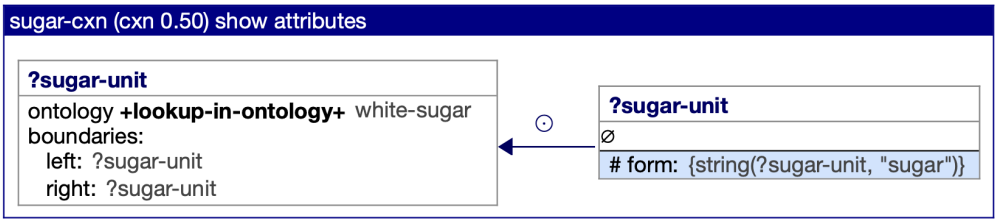
In order to handle the genre-specific challenges of recipes, we have chosen a construction grammar approach (Fillmore, 1988; Goldberg, 2003; Fried and Östman, 2004), which we implemented in the open-source formalism Fluid Construction Grammar (FCG; Steels, 2004; van Trijp et al., 2022; Beuls and Van Eecke, 2023). The motivation for this approach is threefold. First, construction grammar is a linguistic theory in which *all* linguistic information is represented as mappings between form and meaning (called "constructions"), which makes it convenient to represent both the idiosyncrasies of recipe language as well as its more abstract syntactic structures in a uniform way. Secondly, semantics can but needn't be directly coupled to syntactic structures, which makes it possible to parse sentences directly into language-independent executable robot plans. Finally, the functional scope of constructions is not limited to the sentence level, which means that constructions can represent discourse-level information as well (Fried, 2021).

The latter feature of construction grammar is of great importance for recipes. As discussed in section 1.1, recipes are abundant with zero anaphora, which are used by recipe authors as a strategy for cohesion building since these zero anaphora refer to entities that are highly salient in the current discourse context (Cani, 2022). For instance, in the almond cookies recipe, the direct object is omitted in phrases such as "mix thoroughly" and "place onto a parchment paper lined baking sheet".

Our solution is to include non-linguistic information in language processing, which is supplied by the personal dynamic memory as shown in Figure 3. The PDM is where the recipe's plot is constructed: at each node, the PDM tracks which entities are currently under the attention of the cooking agent (called *accessible entities*). Accessible entities are like characters that were introduced in prior scenes and that are still present in the current scene.

More concretely, linguistic processing makes use of a kind of blackboard that contains all of the information about the accessible entities and the input phrase or sentence. This blackboard is called *transient structure* because it changes over time as different constructions access and expand its information. Figure 4 illustrates such a transient structure at the beginning of a parsing task for the phrase "116 grams sugar". This transient structure consists of four units (which are simply lists of feature-value pairs). The unit called *root* (on top) contains unhandled information from the input sentence, such as which strings (or tokens) were observed and which strings are adjacent to each other. At this point in the recipe, the cooking agent has already fetched a medium bowl of butter, so there are two accessible entities: the current kitchen state, and the bowl of butter.

In Fluid Construction Grammar (FCG), constructions are formalized as schemas that consist of a conditional pole (right-hand side) and a contributing pole (left-hand side). A construction is allowed to add the information of its contributing pole to the transient structure if the feature-value pairs from its conditional pole can be matched against the information already present in the transient structure (Steels and De Beule, 2006). Here is an example of a lexical construction for the word "sugar":



The application of a construction leads to a new and expanded transient structure, which may in turn trigger the application of other constructions. Parsing is thus operationalized as a search

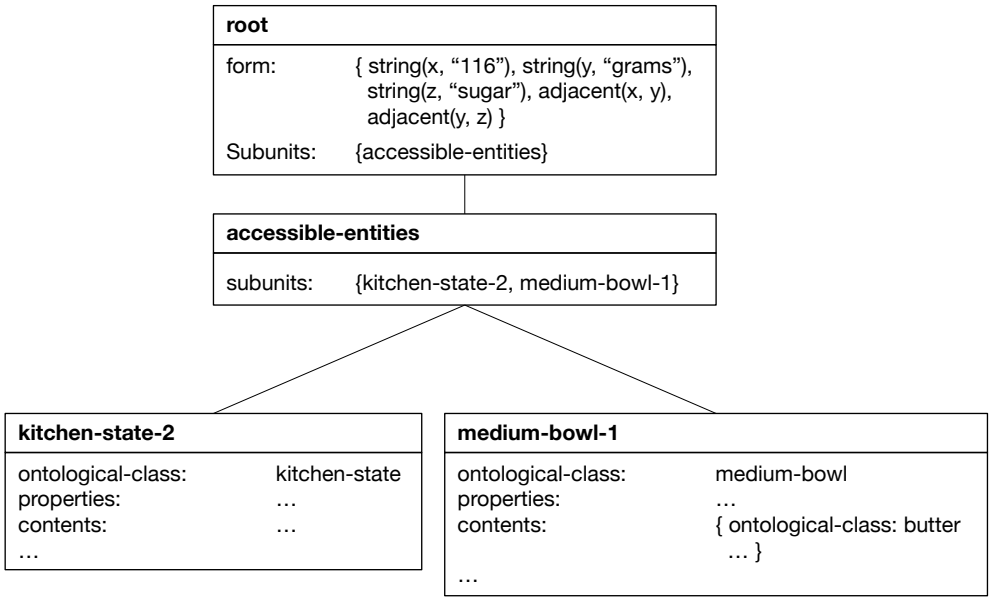


Figure 4: A *transient structure* contains information about both the input sentence and the entities that are accessible from discourse context.

problem for finding the best set of constructions for extracting the meaning of an input sentence (Van Eecke et al., 2022).

Constructions represent most information as declarative features, but they can also use procedural attachment (Steels, 1979; Bundy and Wallen, 1984). For instance, in the sugar-cxn, the value of the feature *ontology* is not directly specified: instead, a procedure `+lookup-in-ontology+` is embedded, which is able to fetch all of the features that are associated with the concept [white-sugar]. Procedural attachment is necessary for dealing with uncertainty. Some examples are:

1. Human-centric AI systems must be open-ended learners, hence the ontology may change with every novel experience.

2. Higher-level constructions may involve generalizations, e.g. over phrases with ingredients such as "two tablespoons of sugar" or "120 grams of flour". Procedural attachment allows an on-the-fly check whether sugar or flour both qualify as ingredients.

3. Constructions that handle quantities must be able to parse such units on the fly. For instance, a construction responsible for portions should recognize both the tokens "114" as well as "two", so a procedure for checking whether a token can be parsed as a number helps to make such generalizations possible.

For the present case study, we hand-coded a small grammar of 56 constructions that can be inspected in our web demo. Hand-coding is a necessary first step to identify what kind of grammatical structures are necessary for mapping recipes onto executable robot plans and for evaluating the feasibility of a constructional approach (we will address the question of learning in section 5). Our grammar includes constructions for lemmatization, lexical constructions, idiomatic and semi-schematic constructions (e.g. the "until light and fluffy"- and "place-X-onto-y"-constructions), and

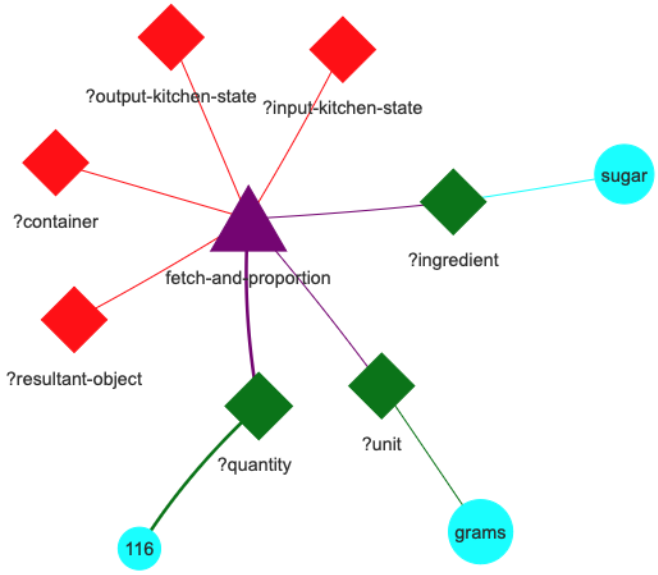


Figure 5: Parsing "116 grams sugar" leads to a partial robot plan that the cooking agent needs to complete into an executable one.

abstract constructions such as the English Resultative (Boas, 2003; Goldberg and Jackendoff, 2004) for analyzing phrases such as "beat the butter and sugar together until light and fluffy".

3.2 Meaning and Mental Simulation

The goal of language processing is not to derive the most accurate syntactic analysis of a sentence: syntactic structures are only built insofar as they help to get to the meaning as efficiently as possible. "The" meaning is a bit misleading because language is an inferential coding system (Sperber and Wilson, 1986) so not all the meaning is in the message. For instance, a phrase such as "put two eggs in a bowl" does not specify which eggs and bowl to take, or that you have to crack the eggs open and get rid of the shells.

Parsing therefore only leads to a partial robot plan that the cooking agent needs to complete and expand upon. For instance, the phrase "116 grams sugar" maps onto an operation called *fetch-and-proportion*, illustrated in Figure 5. Some of the operation's arguments are already provided by the recipe instructions such as which food product to fetch (*sugar*), shown as cyan circles. However, there are several open slots such as the resultant object, shown as red diamonds. The cooking agent thus needs to find fillers for those slots using different knowledge sources such as its personal dynamic memory and mental simulation. To enable the agent to do so, we used the open-source software tool Incremental Recruitment Language (IRL) for representing, generating and executing robot plans (Steels, 2000; Spranger et al., 2012).

More specifically, we implemented a new representation language for cooking that includes 40 predefined cooking operations (called "primitives") that encode meaning, temporality and dependencies. The IRL-system can then combine these primitives into graphs that represent complete recipe execution plans. Recurrent graphs can be automatically chunked and stored as

composite operations for more efficient plan generation in the future, and users may extend the representation language with additional cooking operations.

As shown in Figure 3, plan execution relies on mental simulation, which is a distinct human capacity that allows us to project ourselves into hypothetical realities (Waytz et al., 2015), typically in the form of narratives (Escalas, 2004). In our case study, we therefore included our own (qualitative) kitchen simulator as well as a quantitative simulator (Pomarlan, 2021) for simulating cooking operations and their effects (see section 4.2).

One of the reasons for selecting the IRL-system for plan generation and plan execution is that it allows a *data flow* approach in which the handling of data adapts to which data is available at a given moment, as opposed to explicit control flow in which all operations need to be ordered beforehand. For example, given an initial kitchen state, a food product and the unit of measurement, the operation *fetch-and-proportion* can compute the resultant object and output kitchen state. Suppose however that a human chef has already taken a cup of 116 grams of sugar, then the agent could apply the same operation in a different direction for verifying whether the resultant object corresponds to what is written in the recipe, or for backtracking which actions the human user must have taken.

Data flow is important for handling the dynamic nature of a kitchen environment (where many things can go wrong) and for adapting to the user's needs, who may use different variations of a recipe or who may have different preferences about which actions they like to perform themselves and which to delegate to their AI assistant. Moreover, data flow allows the final robot plan to be greatly optimized. For instance, the cooking agent does not need to wait until an operation such as *boil* is completely finished: the IRL-system will already provide a placeholder for the resultant object (including a timing for when it will be ready) so the cooking agent can already start planning and executing other tasks.

The result of plan generation and execution is a new node in the plot that the agent is constructing in its personal dynamic memory, as shown in Figure 3. This new node includes an update of the kitchen state and which entities are currently accessible. Indeed, the "meaning" of each recipe instruction is a small executable robot plan that is causally linked to the next one. The interlinked plans as a whole form a detailed and coherent robot execution plan for the whole recipe, which makes it possible to backtrack to earlier kitchen states whenever necessary.

4 Evaluation and Self-Assessment

In this section we describe the steps taken towards evaluation as well as first results. Moreover, we discuss how a cooking agent may reason about its own performance.

4.1 Integrative Narrative Networks

One of the challenges of narrative-based understanding is to monitor how different knowledge sources are integrated with each other, and whether the resulting plot offers a coherent and sensible content model. Crucially, the cooking agent itself should also have a way to monitor its own understanding process.

We are approaching this challenge using a new data structure called *integrative narrative networks* (Baroncini et al., 2023). The key inspiration for such networks comes from the narratological concept of "narrative questions", which are the questions that are raised in the audience's mind by a narrative (or that an author wants to be raised). For instance, when a new character is introduced in a movie,

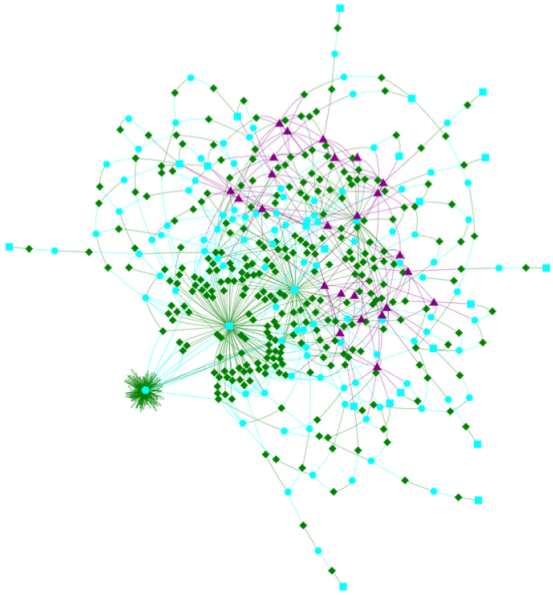


Figure 6: An Integrative Narrative Network for the almond crescent cookies recipe.

this may raise the narrative question "who is this person?" Compelling narratives use such questions to connect plot points to each other and to keep the audience engaged, until narrative closure is reached – the point where all salient questions have been satisfactorily answered (Carroll, 2007). Likewise, we can frame understanding as a process in which an agent poses narrative questions to itself, and then searches for answers for those questions until it reaches narrative closure (or a conclusion).

The network in Figure 5 illustrates this idea. This network shows that the activation of the primitive *fetch-and-proportion* raises a number of questions (depicted as diamond-shaped nodes), such as which ingredient to fetch and what the resultant object will be. Some of these questions are immediately answered by parsing "116 grams sugar" (the green nodes), while other questions are still open (red). The agent now has to search for answers for those questions. Narrative questions and answers can be introduced by various knowledge sources, and this process is continued until all salient questions have been satisfactorily answered. Figure 6 shows a complete Integrative Narrative Network, as built by the cooking agent, for the almond crescent cookies recipe.

Figure 7 shows some results of a first experiment in monitoring and measuring narrative-based understanding, described in more detail in Steels et al. (2022). The black line shows the number of narrative questions that are raised as the cooking agent goes through the recipe. The coloured segments show how many questions were answered by different knowledge sources. From bottom to top, these are: language (blue), mental simulation (orange), ontology (green) and the discourse model/PDM (red).

Our current work focuses on how such integrative narrative networks can be used for allowing agents to monitor and reason about their own understanding process, and to optimize the decisions that they make in the face of uncertainty.

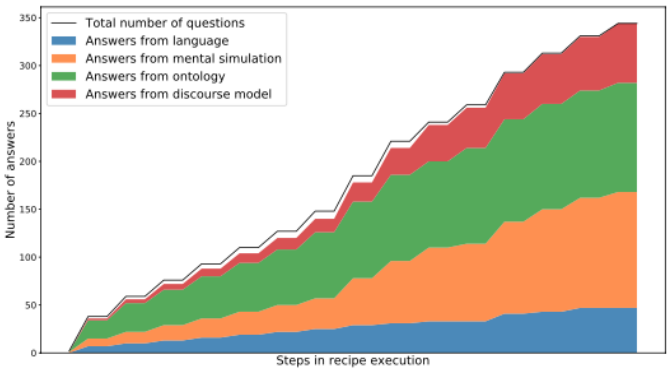


Figure 7: This Figure illustrates the number of narrative questions raised during recipe understanding and how many of them have been answered by which knowledge source.

4.2 Recipe Execution Benchmark

In order to evaluate our work as well as invite the research community to advance the field of grounded language understanding (and computational recipe understanding in particular), we have released a fully documented recipe execution benchmark (Nevens et al., 2024), which consists of the following components:

- A representation language for cooking (see section 3.2) that can express complete recipe execution plans. This representation language is independent from syntax or a particular natural language, so knowledge about syntax is not necessary for annotation.
- A test set of 30 English recipes with gold standard annotations. These recipes have been selected for the specific linguistic and extralinguistic challenges in recipe understanding.
- A qualitative kitchen simulator that is able to execute the recipe execution plans, and which returns both execution and evaluation results for further inspection.
- A suite of metrics that allow multiperspective estimates to optimize transferability to real-world utility. These consist of Smatch (Cai and Knight, 2013), goal-condition success, Dish Approximation Score, and Recipe Execution Time.

5 Conclusions and Future Work

This paper explored how narrative-based language understanding can be used for processing cooking recipes in a way that allows robots or artificial cooking agents to execute those recipes in a dynamic kitchen environment. Through a case study on an English recipe for almond crescent cookies, we have shown how the rich content models built during narrative-based understanding can be exploited for tackling the specific challenges of recipe language, such as resolving zero anaphora by keeping track of which entities are currently under the cooking agent's attention. We have thereby shown how language processing can be embedded in a system for representing, generating and executing robot plans, coupled to a kitchen simulator. Moreover, we have proposed how narratives may offer a new framework for monitoring and measuring understanding.

Even though we have illustrated our approach through a concrete implementation and accompanying web demonstration, we hope to have convinced the reader that the framework of narrative-based understanding can be operationalized in a multitude of ways. To this end we have published a recipe execution benchmark for comparing different solutions. One key component of this benchmark is a new representation language for cooking, which allows recipes to be annotated in a syntax- and natural-language-independent fashion.

Our current and future work focuses on automatically learning computational construction grammars for recipe understanding in order to scale our approach. For the reasons detailed in section 2.2, we believe that construction grammar shows great promise to deal with the specific challenges of grounded language understanding, and computational recipe understanding in particular. Important breakthroughs in the automated learning of such grammars have recently been achieved in the domain of Visual-Question Answering (Nevens et al., 2022; Doumen et al., 2024; Beuls and Van Eecke, 2024), which requires a mapping from questions to visual queries in similar ways as recipe instructions map onto executable robot plans.

By mimicking the sense-making process of humans, narrative-based language understanding can become a key component in the development of human-centric AI systems. Such systems have many potential benefits for society, as they may interact with humans in more intuitive and meaningful ways.

Acknowledgements

The research reported on in this chapter was funded by the European Union's Horizon 2020 research and innovation programme under grant agreement no. 951846 (MUHAI).

References

Allen, J. F. (1994). *Natural Language Understanding*. Benjamin/Cummings, Redwood City, 2nd edition.

Austin, J. L. (1962). *How to do things with words*. Oxford University Press, London.

Bal, M. (1985). *Narratology: Introduction to the Theory of Narrative*. Toronto Univeristy Press, Toronto.

Baroncini, S., Steels, L., and van Trijp, R. (2023). Semantic data retrieval and integration for supporting artworks interpretation through integrative narrative networks. In *SWODCH'23: International Workshop on Semantic Web and Ontology Design for Cultural Heritage*.

Beetz, M., Klank, U., Kresse, I., Maldonado, A., Mösenlechner, L., Pangercic, D., Rühr, T., and Tenorth, M. (2011). Robotic roommates making pancakes. In *2011 11th IEEE-RAS International Conference on Humanoid Robots*, pages 529–536.

Beuls, K. and Van Eecke, P. (2023). Fluid Construction Grammar: State of the art and future outlook. In Bonial, C. and Tayyar Madabushi, H., editors, *Proceedings of the First International Workshop on Construction Grammars and NLP (CxGs+NLP, GURT/SyntaxFest 2023)*, pages 41–50. Association for Computational Linguistics.

Beuls, K. and Van Eecke, P. (2024). Humans learn language from situated communicative interactions. What about machines? *Computational Linguistics*, 50(4):1277–1311.

Bezaleli Mizrahi, A., Lachnish, A. Z., and Zoran, A. R. (2023). Digital gastronomy testcase: A complete pipeline of robotic induced dish variations. *International Journal of Gastronomy and Food Science*, 31:100625.

Blin, I. (2022). Building narrative structures from knowledge graphs. In Groth, P., Rula, A., Schneider, J., Tiddi, I., Simperl, E., Alexopoulos, P., Hoekstra, R., Alam, M., Dimou, A., and Tamper, M., editors, *The Semantic Web: ESWC 2022 Satellite Events. ESWC 2022*, volume 13384 of *Lecture Notes In Computer Science*, pages 234–251, Cham. Springer.

Boas, H. C. (2003). *A Constructional Approach to Resultatives*. Stanford Monograph in Linguistics. CSLI, Stanford.

Bollini, M., Tellex, S., Thompson, T., Roy, N., and Rus, D. (2013). Interpreting and executing recipes with a cooking robot. In *Experimental Robotics: The 13th International Symposium on Experimental Robotics*, pages 481–495. Springer.

Bruner, J. (1991). The narrative construction of reality. *Critical Inquiry*, 18(1):1–21.

Bundy, A. and Wallen, L. (1984). Procedural attachment. In Bundy, A. and Wallen, L., editors, *Catalogue of Artificial Intelligence Tools*. Springer, Berlin/Heidelberg.

Cai, S. and Knight, K. (2013). Smatch: an evaluation metric for semantic feature structures. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 748–752, Sofia, Bulgaria. Association for Computational Linguistics.

Cani, P. (2022). De-constructing recipes: A constructionist comparative analysis of recipe texts. Master's thesis, Universitá Ca' Foscari Venezia, Venice.

Carroll, N. (2007). Narrative closure. *Philosophical Studies*, 135(1):1–15.

Cotter, C. (1997). Claiming a piece of the pie: How the language of recipes defines community. In Bower, A. L., editor, *Recipes for Reading: Community Cookbooks, Stories, Histories*, pages 51–72. University of Massachusetts Press, Amherst.

Doumen, J., Beuls, K., and Van Eecke, P. (2024). Modelling constructivist language acquisition through syntactico-semantic pattern finding. *Royal Society Open Science*, 11(7):231998.

Eckardt, R. (2006). Truth conditional semantics. In *Meaning Change in Grammaticalization: An Enquiry into Semantic Reanalysis*, pages 59–90. Oxford University Press.

Escalas, J. E. (2004). Imagine yourself in the product: Mental simulation, narrative transportation, and persuasion. *Journal of Advertising*, 33(2):37–48.

Fillmore, C. J. (1988). The mechanisms of "construction grammar". In *Annual Meeting of the Berkeley Linguistics Society*, volume 14, pages 35–55.

Fried, M. (2021). Discourse-referential patterns as a network of grammatical constructions. *Constructions and Frames*, 13(1):21–54. ISBN: 1876-1933 Publisher: John Benjamins Type: https://doi.org/10.1075/cf.00046.fri.

Fried, M. and Östman, J.-O. (2004). Construction grammar: A thumbnail sketch. In Östman, J.-O. and Fried, M., editors, *Construction grammar in a cross-language perspective*, pages 1–86. John Benjamins, Amsterdam, Netherlands.

Gerhardt, C., Frobenius, M., and Ley, S., editors (2013). *Culinary Linguistics: The Chef's Special*, volume 10 of *Culture and Language Use*. John Benjamins, Amsterdam.

Goldberg, A. E. (2003). Constructions: A new theoretical approach to language. *Trends in Cognitive Sciences*, 7(5):219–224.

Goldberg, A. E. and Jackendoff, R. (2004). The English Resultative as a Family of Constructions. *Language*, 80(3):532–568.

Harnad, S. (1990). The Symbol Grounding Problem. *Physica D*, 42:335–346.

Harper, C. and Siller, M. (2015). Openag: A globally distributed network of food computing. *IEEE Pervasive Computing*, 14(4):24–27.

Hart, P. and Nilsson, N. (1972). Shakey: An experiment in robot planning and learning. Movie.

Haussmann, S., Seneviratne, O., Chen, Y., Ne'eman, Y., Codella, J., Chen, C.-H., McGuinness, D. L., and Zaki, M. J. (2019). FoodKG: a semantics-driven knowledge graph for food recommendation. In Ghidini, C., Hartig, O., Maleshkova, M., Svátek, V., Cruz, I., Hogan, A., Song, J., Lefrançois, M., and Gandon, F., editors, *The Semantic Web – ISWC 2019*, pages 146–162, Cham. Springer.

Jabeen, H., Weinz, J., and Lehmann, J. (2020). Autochef: Automated generation of cooking recipes. In *2020 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–7.

Jiang, Y., Zaporojets, K., Deleu, J., Demeester, T., and Develder, C. (2020). Recipe instruction semantics corpus (RISeC): Resolving semantic structure and zero anaphora in recipes. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 821–826, Suzhou, China. Association for Computational Linguistics.

Khilji, A. F. U. R., Manna, R., Laskar, S. R., Pakray, P., Das, D., Bandyopadhyay, S., and Gelbukh, A. (2021). CookingQA: Answering Questions and Recommending Recipes Based on Ingredients. *Arabian Journal for Science and Engineering*, 46(4):3701–3712.

Kicherer, H., Dittrich, M., Grebe, L., Scheible, C., and Klinger, R. (2018). What you use, not what you do: Automatic classification and similarity detection of recipes. *Data & Knowledge Engineering*, 117:252–263.

Kress-Gazit, H., Lahijanian, M., and Raman, V. (2018). Synthesis for robots: Guarantees and feedback for robot behavior. *Annual Review of Control, Robotics, and Autonomous Systems*, 1:211–236.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.

Manna, R., Das, D., and Gelbukh, A. (2020). Information retrieval-based question answering system on foods and recipes. In Martínez-Villaseñor, L., Herrera-Alcántara, O., Ponce, H., and Castro-Espinoza, F. A., editors, *Advances in Computational Intelligence*, pages 260–270, Cham. Springer International Publishing.

Marin, J., Biswas, A., Ofli, F., Hynes, N., Salvador, A., Aytar, Y., Weber, I., and Torralba, A. (2019). Recipe1m+: A dataset for learning cross-modal embeddings for cooking recipes and food images. *IEEE Trans. Pattern Anal. Mach. Intell.*

Min, W., Jiang, S., Liu, L., Rui, Y., and Jain, R. (2019). A survey on food computing. *ACM Computing Surveys*, 52(5).

Najjar, N. A. and Wilson, D. C. (2017). Computer cooking contest 2017 – preface. In Sanchez-Ruiz, A. A. and Kofod-Petersen, A., editors, *Proceedings of ICCBR 2017 Workshops (CAW, CBRDL, PO-CBR), Doctoral Consortium, and Competitions co-located with the 25th International Conference on Case-Based Reasoning (ICCBR 2017)*, pages 224–227, Trondheim. CEUR Workshop Proceedings.

Nevens, J., De Haes, R., Ringe, R., Pomarlan, M., Porzel, R., Beuls, K., and Van Eecke, P. (2024). A benchmark for recipe understanding in artificial agents. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation*, pages 22–42.

Nevens, J., Doumen, J., Van Eecke, P., and Beuls, K. (2022). Language acquisition through intention reading and pattern finding. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 15–25, Gyeongju.

Pomarlan, M. (2021). AbeSim. `https://github.com/mpomarlan/abe_sim`.

Popovski, G., Seljak, B. K., and Eftimov, T. (2019). FoodBase corpus: a new resource of annotated food entities. *Database*, 2019:baz121.

Searle, J. R. (1969). *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, Cambridge.

Shridhar, M., Thomason, J., Gordon, D., Bisk, Y., Han, W., Mottaghi, R., Zettlemoyer, L., and Fox, D. (2020). Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10737–10746.

Smith, N. and Lin, K.-I. (2012). Automatic information extraction from the web: Case study with recipes. In *Proceedings of the 50th Annual Southeast Regional Conference*, ACM-SE '12, pages 369–370, New York, NY, USA. Association for Computing Machinery.

Sperber, D. and Wilson, D. (1986). *Relevance: Communication and cognition*. Harvard University Press, Cambridge, MA, USA.

Spranger, M., Pauw, S., Loetzsch, M., and Steels, L. (2012). Open-ended procedural semantics. In Steels, L. and Hild, M., editors, *Language Grounding in Robots*, pages 153–172. Springer, New York, NY, USA.

Steels, L. (1979). Procedural attachment. Technical Report AIM-543, MIT, Cambridge MA.

Steels, L. (2000). The emergence of grammar in communicating autonomous robotic agents. In Horn, W., editor, *Proceedings of the 14th European Conference on Artificial Intelligence*, pages 764–769, Amsterdam, Netherlands. IOS Press.

Steels, L. (2004). Constructivist development of grounded construction grammar. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 9–16.

Steels, L. (2020). Personal dynamic memories are necessary to deal with meaning and understanding in human-centric ai. In Saffiotti, A., Serafini, L., and Lukowicz, P., editors, *Proceedings of the First International Workshop on New Foundations for Human-Centered AI (NeHuAI) co-located with 24th European Conference on Artificial Intelligence (ECAI 2020)*, volume 2659, pages 11–16, Santiago de Compostella. CEUR Workshop Proceedings.

Steels, L., editor (2022). *Foundations for Meaning and Understanding in Human-Centric AI*. Venice International University, Venice.

Steels, L. and De Beule, J. (2006). Unify and merge in Fluid Construction Grammar. In *International Workshop on Emergence and Evolution of Linguistic Communication (EELC 2006)*, pages 197–223.

Steels, L., Verheyen, L., and van Trijp, R. (2022). An experiment in measuring understanding. In *Workshop on semantic techniques for narrative-based understanding: Workshop at IJCAI-ECAI 2022*, pages 36–42.

Sugiura, Y., Sakamoto, D., Withana, A., Inami, M., and Igarashi, T. (2010). Cooking with robots: Designing a household system working in open environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2427–2430, New York. Association for Computing Machinery.

Szilas, N. (2015). Towards Narrative-Based Knowledge Representation in Cognitive Systems. In Finlayson, M. A., Miller, B., Lieto, A., and Ronfard, R., editors, *6th Workshop on Computational Models of Narrative (CMN 2015)*, volume 45 of *OpenAccess Series in Informatics (OASIcs)*, pages 133–141, Dagstuhl, Germany. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

Tellex, S., Gopalan, N., Kress-Gazit, H., and Matuszek, C. (2020). Robots that use language. *Annual Review of Control, Robotics, and Autonomous Systems*, 3(1):25–55.

Tenorth, M., Nyga, D., and Beetz, M. (2010). Understanding and executing instructions for everyday manipulation tasks from the world wide web. In *2010 IEEE International Conference on Robotics and Automation*, pages 1486–1491.

Tian, Y., Zhang, C., Guo, Z., Huang, C., Metoyer, R., and Chawla, N. V. (2022). Reciperec: A heterogeneous graph learning model for recipe recommendation. In De Raedt, L., editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 3466–3472. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Tian, Y., Zhang, C., Metoyer, R., and Chawla, N. V. (2021). Recipe representation learning with networks. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, pages 1824–1833, New York, NY, USA. Association for Computing Machinery.

Van Eecke, P., Nevens, J., and Beuls, K. (2022). Neural heuristics for scaling constructional language processing. *Journal of Language Modelling*, 10(2):287–314.

Van Eecke, P., Verheyen, L., Willaert, T., and Beuls, K. (2023). The Candide model: How narratives emerge where observations meet beliefs. In Akoury, N., Clark, E., Iyyer, M., Chaturvedi, S., Brahman, F., and Chandu, K., editors, *Proceedings of the 5th Workshop on Narrative Understanding (WNU)*, pages 48–57, Toronto.

van Trijp, R., Beuls, K., and Van Eecke, P. (2022). The FCG Editor: An innovative environment for engineering computational construction grammars. *PLOS ONE*, 17(6):e0269708.

Wang, H., Lin, G., Hoi, S. C. H., and Miao, C. (2022). Learning structural representations for recipe generation and food retrieval.

Waytz, A., Hershfield, H. E., and Tamir, D. I. (2015). Mental simulation and meaning in life. *Journal of Personality and Social Psychology*, 108(2):336–355.

Winograd, T. (1971). Procedures as a representation for data in a computer program for understanding natural language. AI Technical Reports AITR-235, MIT, Cambridge MA.

Wittgenstein, L. (1953). *Philosophical investigations*. Macmillan Publishing Company, London, United Kingdom.

Yagcioglu, S., Erdem, A., Erdem, E., and Ikizler-Cinbis, N. (2018). RecipeQA: A challenge dataset for multimodal comprehension of cooking recipes. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1358–1368, Brussels, Belgium. Association for Computational Linguistics.

Zoran, A. (2019). Cooking with computers: The vision of digital gastronomy [point of view]. *Proceedings of the IEEE*, 107(8):1467–1473.