ANNUAL
REVIEWS

*Annual Review of Linguistics*

# Compositionality in Computational Linguistics

Lucia Donatelli and Alexander Koller

Department of Language Science and Technology, Saarland Informatics Campus, Saarland University, Saarbrücken, Germany; email: donatelli@lst.uni-saarland.de, koller@lst.uni-saarland.de

Annu. Rev. Linguist. 2023. 9:463–81

First published as a Review in Advance on November 2, 2022

The *Annual Review of Linguistics* is online at linguistics.annualreviews.org

https://doi.org/10.1146/annurev-linguistics-030521-044439

## Keywords

compositionality, computational linguistics, neural networks, neurosymbolic models, semantic parsing

## Abstract

Neural models greatly outperform grammar-based models across many tasks in modern computational linguistics. This raises the question of whether linguistic principles, such as the Principle of Compositionality, still have value as modeling tools. We review the recent literature and find that while an overly strict interpretation of compositionality makes it hard to achieve broad coverage in semantic parsing tasks, compositionality is still necessary for a model to learn the correct linguistic generalizations from limited data. Reconciling both of these qualities requires the careful exploration of a novel design space; we also review some recent results that may help in this exploration.

## 1. INTRODUCTION

The Principle of Compositionality—the modeling assumption that the meaning of a natural-language expression is determined by the meanings of its constituents and the ways in which they are combined into structure (Partee 1984)—is a mainstay of classical semantics. Humans can understand and produce a potentially infinite number of novel, well-formed linguistic expressions by dynamically recombining known elements (Chomsky 1957, Fodor & Pylyshyn 1988, Fodor & Lepore 2002), and compositionality helps explain this ability. Computational linguists with an interest in semantics have long embraced compositionality as a modeling principle because it facilitates the design of handwritten or learned grammars that can assign formal meaning representations to natural-language expressions and, therefore, aids in the development of computational models of human language competence.

Yet, computational linguistics has recently come to be dominated by neural modeling approaches, which tackle language processing tasks with the use of neural networks designed to emulate human cognition rather than handcrafted symbolic grammars. Modern neural methods of language have evolved dramatically from their beginnings in connectionist linguistics (Pater 2019), obliterating the previous state of the art on tasks as diverse as machine translation (Sutskever et al. 2014, Bahdanau et al. 2015), question answering (Khashabi et al. 2020), and semantic parsing (e.g., Bevilacqua et al. 2021). These tasks require the computational system to model the meaning of unseen complex sentences, and one cannot deny the ability of neural models to capture at least some aspects of meaning (but see Bender & Koller 2020 for a critical discussion). And yet, end-to-end neural networks do this without relying on the Principle of Compositionality.

As computational linguists with a passion for theoretical semantics, we are faced with the inconvenient question of whether compositionality is still a useful modeling technique in research on computational semantics or whether it can be discarded in the face of the awesome learning power of modern neural methods. In this review, we discuss some recent findings to address this issue from two angles. First, we discuss the challenge of broad-coverage semantic parsing with compositional methods (Section 3). One of the most fundamental challenges of computational linguistics is that of broad coverage: Systems for natural-language understanding must be able to produce meaningful results for any possible sentence encountered in arbitrary human-produced text. Broad coverage is a key strength of neural models, but as we discuss below, it is difficult to achieve with compositional models, at least if the Principle of Compositionality is interpreted too strictly.

Second, as a counterpoint, we discuss the challenge of achieving compositional generalization (Section 4). Several recent data sets in computational linguistics have focused on the question of whether a semantic parser that is trained on a training set with structurally simple sentences can learn to correctly parse a test set with structurally more complex sentences (e.g., ones with a depth of recursion that was never seen in training). We review a number of compositional generalization data sets and show that, at least on cases where the parser must generalize to unseen grammatical structures, purely neural models struggle to achieve high accuracy. By contrast, models that have the Principle of Compositionality built in perform well on such data sets. Taken together, these two results point to a tension that semantic parsers face: We expect them to achieve high accuracy with broad coverage, and we also want them to learn the right linguistic generalizations from limited data. Even today, a semantic parser can do well on both only when it is built around a careful computational interpretation of the Principle of Compositionality.

We conclude this review with a discussion of recent computational research on compositionality that all revolves about the design space of compositional systems. We discuss what compositionality can tell us about designing meaning representations, what can be said about the expressive capacity of compositional systems, and how our view of compositionality changes in emergent and multimodal settings.

Throughout, we focus on the task of semantic parsing, which is concerned with assigning formal meaning representations to natural-language sentences. While there are interesting things to be said about compositionality below the word level (Halle et al. 1993, Marantz 1997), in fixed syntactic constructions (Kay & Michaelis 2019), and in interaction with discourse (Janssen 2001), we do not have the space in this review article to do them justice.

## 2. BACKGROUND

Before we discuss the role of compositionality in current computational linguistics, it is useful to fix some terminology. We assume a statement of the Principle of Compositionality as follows:

> The meaning of a natural-language expression is determined by the meanings of its immediate subexpressions and the way in which they were combined.

We do not commit to the truth or self-evidence of this principle (see, e.g., Szabó 2012 for a critical discussion), but we take it as a convenient modeling assumption that is helpful across linguistic semantics and computational modeling.

The above definition of compositionality implies that the structure of a natural-language expression $w$ can be represented as a tree $t$ of some kind and that a meaning representation for $w$ can be determined by bottom-up evaluation of $t$. This compositional structure $t$ is often taken to be a syntax tree. We call the intermediate results computed at the nodes of $t$ the partial meaning representations.

In a classical semantic theory, meaning is usually defined in terms of truth conditions. By contrast, much recent work in computational linguistics does not assume that meaning representations must be truth-conditional but is instead driven by the design choices made in semantically annotated corpora known as sembanks (see the sidebar titled Sembanks). These representations may be logical formulas that support a truth-conditional interpretation, as in the Discourse Representation Theory (DRT)-annotated Parallel Meaning Bank (Abzianidze et al. 2017), certain versions of the Geoquery corpus (Zelle & Mooney 1996), and the COGS corpus (Kim & Linzen 2020). But they can also be graphs representing the predicate–argument structure of a sentence, as in the AMR Bank (Banarescu et al. 2013) and the Semantic Dependency Graphbanks (Oepen et al. 2015), or they can be executable programs, for instance, SQL queries (Yu et al. 2018, Keysers et al. 2020) or Python programs (Yin & Neubig 2017). Some sembanks are designed to support reliable annotation, some are motivated by downstream applications such as machine translation, and some are simply easy conversion targets for existing resources. In this review, our focus is not on the

### SEMBANKS

Modern computational linguistics is almost exclusively data-driven: Rather than encoding linguistic knowledge by hand (e.g., in a grammar), it is learned from corpora. A sembank is a text corpus where each sentence has been annotated with a meaning representation. Because it is hard to achieve high interannotator agreement for semantic annotations, early sembanks, such as the Geoquery corpus, were tiny (fewer than 1,000 sentences). More recent sembanks, such as the AMR Bank, are much larger.

Large sembanks that contain naturally occurring text (e.g., news text) are used to train and evaluate broad-coverage semantic parsers. By contrast, synthetic sembanks such as COGS are generated automatically from a small grammar. Such corpora can be useful for studying specific issues, such as compositional generalization, more systematically.

meaning representations themselves but rather on methods for computing them compositionally; thus, we make no distinction between mapping to a truth-conditional representation or to a graph.

Furthermore, semantic theory usually assumes that the compositional structure is a syntax tree. Traditional approaches to computational linguistics have made this assumption too because it facilitates the development of large handwritten grammars in such formalisms as Lexical Functional Grammar (LFG; Dalrymple 2022) and Head-Driven Phrase Structure Grammar (HPSG; Pollard & Sag 1994). But although, as Copestake et al. (2001) report, 95% of the development time of large handwritten grammars goes into defining the semantic component, even the largest such grammars will produce a syntactic analysis on only 85% of sentences in a news corpus; the rest is (falsely) discarded as ungrammatical. One of the biggest advantages of neural methods over grammar-based ones is that they will produce results (e.g., a syntactic or semantic analysis) for 100% of possible input sentences—that is, they achieve very broad coverage—at the risk that these results may contain mistakes. But if our computational model does not use grammars, maybe it can use compositional structures that are not syntax trees, and some models we discuss below explore this option.

## 3. COMPOSITIONALITY IN BROAD-COVERAGE SEMANTIC PARSING

With this terminology in place, we may consider the role of compositionality in semantic parsing: the task of mapping a natural-language sentence to a symbolic meaning representation. Modern research on semantic parsing is focused almost exclusively on methods that are trained on sembanks; this has turned out to be a much more effective way of achieving broad coverage than writing grammars by hand.

Here, we discuss three broad classes of compositional semantic parsers: ones that are based on Combinatory Categorial Grammar (CCG; Section 3.1), ones that are based on synchronous grammars (Section 3.2), and ones where the compositional structures are terms of some algebra (Section 3.3). Among the latter class, we pay special attention to the Apply–Modify (AM) parser, which uses a specific graph algebra to parse into graph-based meaning representations (Section 3.4). We discuss our findings in Section 3.5.

### 3.1. Models Based on Combinatory Categorial Grammar

Modern semantic parsing can be considered to start with the work of Zettlemoyer & Collins (2005), who used CCG (Steedman 2000) as their starting point. CCG is a mildly context-sensitive categorial grammar formalism. It derives a syntactic analysis by combining constituents of different categories using a small number of combinatory rules, such as forward or backward application (see **Figure 1*a***). Each such combinatory rule can be naturally extended to compositionally combine the partial meaning representations of the constituents (drawn in green in **Figure 1**). Thus, training a CCG-based semantic parser amounts to learning a lexicon, which assigns potential meaning representations and syntactic categories to each word.

The seminal paper of Zettlemoyer & Collins (2005) tackled semantic parsing on the Geoquery sembank by defining patterns for lexicon entries that were instantiated through a machine learning algorithm. Later work by Kwiatkowski et al. (2010) offered a more general approach based on higher-order unification, but even though both approaches were evaluated on Geoquery, a small sembank with very schematic language, nonstandard extensions to CCG such as type-changing rules and the ability to skip input words were needed to achieve decent coverage. Artzi et al. (2015) applied CCG to broad-coverage semantic parsing on the AMR Bank and similarly needed to skip words and potentially repair meaning representations in postprocessing. Thus, CCG-based semantic parsers can achieve good accuracy and coverage, but only after ad hoc changes to CCG itself.

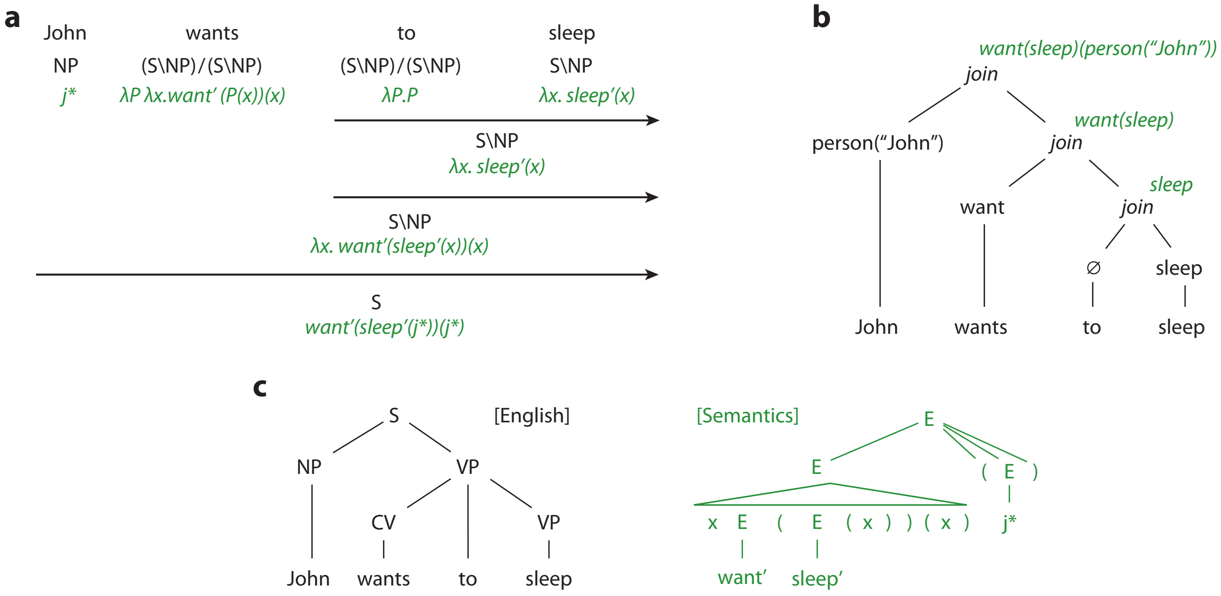

**Figure 1**

Analyses of the sentence "John wants to sleep" in the style of (*a*) Combinatory Categorial Grammar, (*b*) Herzig & Berant (2021), and (*c*) synchronous grammars. Partial meaning representations are drawn in green.

## 3.2. Models Based on Synchronous Grammars

A second class of models for compositional semantic parsing relies on synchronous grammars: grammars that derive two tree structures—one for syntactic structure, one for semantic representation—at the same time.

**Figure 1*c*** shows how a synchronous grammar [specifically, a λ-SCFG as defined by Wong & Mooney (2007)] can compositionally derive a meaning representation. It simultaneously derives a conventional syntax tree for an English sentence and the structure tree of a meaning representation, using synchronous grammar rules like the following:

| [English] | [Semantics] |
|---|---|
| $\mathrm{S} \rightarrow \mathrm{NP}_1\ \mathrm{VP}_2$ | $\mathrm{E} \rightarrow \mathrm{E}_2\ (\ \mathrm{E}_1\ )$ |
| $\mathrm{VP} \rightarrow \mathrm{CV}_1\ \text{to}\ \mathrm{VP}_2$ | $\mathrm{E} \rightarrow \lambda x\ \mathrm{E}_1\ (\ \mathrm{E}_2\ (\ x\ )\ )\ (x)$ |

Wong and Mooney show how such grammars can be learned from sembanks, based on automatically generated alignments between words in the sentence and symbols in the meaning representation. However, synchronous grammars make very strong assumptions about the structural similarity of the sentence and the meaning representation, and there are considerable mismatches between these structures even on simple sembanks like Geoquery (Wang et al. 2021). As a consequence, even the most current synchronous models achieve very low coverage on unconstrained test data (Shaw et al. 2021).

## 3.3. Algebra-Based Models

Algebra-based methods are an attempt at overcoming these limitations of synchronous grammars. They drop the assumption that the compositional structure must be a syntax tree, and they permit the use of operations for combining partial meaning representations that are more powerful than merely concatenating them as in a synchronous grammar.

An example in the style of Herzig & Berant (2021) is shown in **Figure 1*b***. The model either predicts for each word a meaning representation [e.g., *person*(*John*) for the word "John"] or predicts that this word does not contribute to the meaning of the sentence (e.g., ∅ for "to"). The model then predicts a tree that combines these meaning representations step by step. Herzig and Berant in particular define a "join" operation that combines two meaning representations into a bigger one. In this way, the tree in **Figure 1*b*** evaluates to the meaning representation *want*(*sleep*)(*person*(*John*)). Note that the particular example in **Figure 1*b*** oversimplifies matters by pushing the work of assigning an agent to the controlled verb into the interpretation of *want*; the general point is that the algebraic operations for combining partial representations can be as powerful as necessary.

In a similar vein, Liu et al. (2021) map sentences to terms over an algebra whose exact operations depend on the sembank to which the parser is applied. Their LeAR system learns to predict compositional structures for the meaning representations, but it derives a substantial increase in accuracy by compressing the information available at each node of the tree into a finite set of "syntactic" nonterminal symbols. LeAR was the first model to solve the COGS task at near-perfect accuracy, and it set a new state of the art on the Compositional Freebase Questions (CFQ) data set (Keysers et al. 2020); both data sets are discussed in Section 4.1.

## 3.4. The Apply–Modify Parser

One strength of the algebraic approach is that it easily supports semantic parsing into a variety of meaning representation formalisms. This is illustrated nicely by the AM parser (Groschwitz et al. 2018), which compositionally maps sentences to graphs. It learns to predict AM dependency trees (**Figure 2*b***), which bundle a partial meaning representation for each word (drawn below the words) with a dependency tree that combines them compositionally. The algebraic operations APP and MOD combine graphs for heads with graphs for complements and modifiers, respectively, replacing the placeholder nodes labeled in red with the bold-outlined nodes of the argument graphs.

Importantly, there is no assumption that the dependency tree represents the syntax of the sentence; its edge labels refer to algebraic operations that combine meaning representations, and the tree can be nonprojective. This allows the AM parser to achieve high accuracy on broad-coverage semantic parsing tasks, establishing new states of the art across a variety of graph-based sembanks

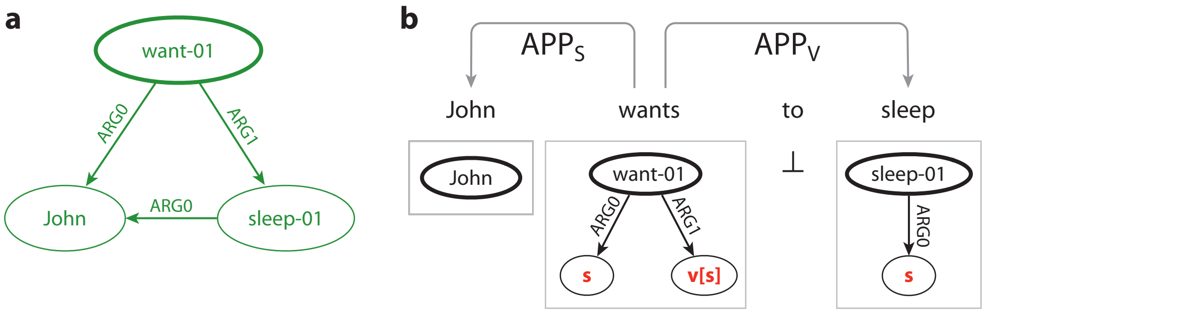

**Figure 2**

(*a*) An AMR graph with (*b*) an AM dependency tree that describes it. Abbreviations: AM, Apply–Modify; AMR, Abstract Meaning Representation.

### NEURAL MODELS OF NATURAL LANGUAGE

Since roughly 2014, research in computational linguistics has been revolutionized by the use of models based on neural networks (also known as deep learning). Neural networks are powerful machine learning models that can learn patterns of natural-language syntax and semantics from text corpora, even unannotated ones (pretraining); they excel particularly at generalizing from their training data to sentences with closely related words, thereby addressing the broad-coverage challenge. However, neural models use numeric internal representations, which are not easily human-interpretable. The widely claimed ability of large neural language models to learn meaning from text alone has been recently questioned by Bender & Koller (2020).

#### Sequence-to-Sequence Models

An important class of neural models is sequence-to-sequence (seq2seq) models, which learn to map a sequence of symbols into another sequence of symbols. Seq2seq models were originally developed for machine translation (Sutskever et al. 2014, Bahdanau et al. 2015), but they have been applied successfully to semantic parsing (see, e.g., Bevilacqua et al. 2021). They are general and powerful, but they have no notion of linguistic structure; this can sometimes make it hard for them to generalize from limited data (see Section 4).

(Lindemann et al. 2019). It is also very fast, parsing up to 10,000 tokens per second (Lindemann et al. 2020).

## 3.5. Discussion

All three approaches to semantic parsing reviewed above are compositional, in the sense that the meaning representation for a sentence is determined by bottom-up evaluation of a tree. However, they differ greatly in the exact way in which compositionality is implemented technically. As we have seen, grammar formalisms that were designed for the development of handwritten grammars are too constraining to achieve broad coverage. Algebra-based approaches can be much more flexible with respect to which partial meaning representations may be combined in each composition step and how, which allows them to handle structural mismatches between the sentence and its meaning representation. They typically use a neural network, rather than a grammar, to predict the compositional structures (see the sidebar titled Neural Models of Natural Language).

One general challenge that all compositional semantic parsers face is that a sembank contains only sentences and their meaning representations, but the parser needs to assign compositional structures to every training sentence so that it can induce a grammar or train a neural model to predict them. These compositional structures are not annotated and must be constructed heuristically or learned. Some recent research has addressed this issue (Lyu & Titov 2018, Groschwitz et al. 2021), but it remains a bottleneck of compositional approaches that puts them at a disadvantage to purely neural, noncompositional models, which can be trained directly on the sembank annotations.

# 4. COMPOSITIONAL GENERALIZATION

If compositionality is so inconvenient for broad-coverage parsing, and if sequence-to-sequence (seq2seq) models perform so well on semantic tasks, why should we develop models that incorporate compositionality? Answering this question requires disentangling the performance of language models from their underlying competence. When training data are plentiful and linguistically rich, a model can achieve good performance by learning the statistical patterns that are present in the data. But if not many training data are available, the model needs to approximate human language competence to learn the right linguistic generalizations to extend to more complex data. One would expect that a model that captures compositionality should have advantages in such a situation.

A substantial amount of recent work has investigated compositional generalization to clarify this issue. Compositional generalization is the ability to determine the meaning of unseen sentences using compositional principles. The data are typically split into a training set with relatively simple sentences and a test set whose sentences are systematically more complex than the ones seen in training. From a machine learning perspective, the test data are out of distribution: They are sampled from a different probability distribution over natural-language expressions than the training data. From a linguistic point of view, evaluation on a compositional generalization data set measures the degree to which a model has learned the right generalizations about language in training.

Below, we first introduce some of the most prominent data sets for compositional generalization (Section 4.1). We then discuss recent results on the apparent inability of seq2seq models to learn nontrivial compositional generalizations (Section 4.2). We conclude by discussing methods for achieving compositional generalization (Section 4.3).

## 4.1. Data Sets

We discuss three compositional generalization data sets that have received much attention in recent research: SCAN, CFQ, and COGS. All three are synthetic data sets automatically generated from a handwritten grammar to exercise specific abilities of semantic parsers. Further noteworthy synthetic data sets for testing the compositional generalization abilities of parsers include NACS (Bastings et al. 2018), an extension of SCAN designed to capture additional generalization abilities; and SyGYNS (Yanaka et al. 2021), which focuses on parsing sentences with novel combinations of logical expressions such as quantifiers and negation. Common nonsynthetic data sets used for compositional generalization tests include compositional splits of Geoquery (Zelle & Mooney 1996), which contains natural-language questions about US geography together with formal database queries, and SMCalFlow (Yin et al. 2021), which is composed of dataflow graphs based on dialogs about events, weather, places, and people. Visual Question Answering (VQA) data sets can also be approached from a parsing perspective; we discuss them in Section 5.3.

**4.1.1. SCAN.** The SCAN data set (Lake & Baroni 2018) consists of a set of simple compositional navigation commands paired with corresponding action sequences. For example, the command "jump" is translated to the action sequence JUMP, "turn right" to RTURN, and "jump left" to LTURN JUMP. The SCAN test data can be selected in multiple ways: (*a*) a random subset of the data ("simple"), (*b*) commands with action sequences longer than those seen during training ("length"), and (*c*) commands that compose a primitive in novel ways that was only seen in isolation during training ("primitive").

These different data splits probe different competence levels: While the simple experiment was easily solved in the original paper (Lake & Baroni 2018) by simple seq2seq models (99.8% accuracy), the length experiment requires systematic generalization to longer action sequences than those seen in training (highest seq2seq accuracy of 13.8%). For the primitive experiments, models perform differently on different sets. For commands involving "turn left," many of the original seq2seq models generalize very well to novel composed commands, with the best model achieving 90% accuracy. In contrast, models fail to generalize to composed commands with novel "jump" commands, with the best-performing model overall reaching 0.08%. The authors note that "left" (represented as LTURN) is seen many times in training as part of other action sequences, while "jump" (JUMP) is seen only in isolation, and models cannot generalize from such minimal context to new, composed sequences.

As a result of the simple grammar used to synthetically generate this data set, the SCAN language, though large, is still limited and finite. The SCAN grammar in particular lacks recursion, and the nature of the commands allows it to be solved with a fairly straightforward interpretation function that does not permit ambiguity. It is therefore unclear to which extent success on SCAN can be transferred to natural language in general. The SCAN data set nevertheless remains one of the most popular benchmarks for models testing compositional generalization. Current models report different accuracies depending on which data splits they use, but several models have now achieved close to perfect accuracy (Liu et al. 2021, Qiu et al. 2021).

**4.1.2. CFQ.** The CFQ data set (Keysers et al. 2020) consists of natural-language questions and answers, where questions are paired with corresponding SPARQL queries against the Freebase knowledge base. For instance, CFQ might pair the sentence "Who directed *Elysium*?" with the logical form $\mathsf{Person} \sqcap \exists \mathsf{RolePair}(\mathsf{Director}, \mathsf{Directee}).\mathsf{Elysium}$. The sentences in CFQ have a much more complex recursive structure than those in SCAN, but they are still generated by a handwritten grammar and are not as linguistically rich as naturally occurring text.

An important component of the CFQ data set is distribution-based compositionality assessment (DBCA), a method of splitting a corpus into a "simple" training set and a "complex" training set. DBCA requires that training and test sets have similar distributions of atomic pieces of meaning representations, but the distribution of their complex combinations differs across training and testing. This is formalized in a measure called maximum compound divergence (MCD). Thus, DBCA is a method of creating hard data splits for compositional generalization for arbitrary sembanks.

The original CFQ paper (Keysers et al. 2020) shows that the mean semantic parsing accuracy of simple seq2seq models is below 20% on DBCA splits of both SCAN and CFQ, even when trained on a large training set (roughly 96,000 instances). Compound divergence and mean accuracy of models are also negatively correlated, showing the difficulty of novel structures in particular. Many models have been developed for CFQ (Csordás et al. 2021, Herzig & Berant 2021, Jambór & Bahdanau 2021, Liu et al. 2021, Yin et al. 2021). In contrast to SCAN, CFQ has not been "solved," though the algebraic, compositional LeAR model of Liu et al. (2021) achieves quite high accuracy between 89% and 92% across all MCD splits.

**4.1.3. COGS.** COGS (Kim & Linzen 2020) is a synthetic data set that maps English sentences to logic-based, neo-Davidsonian meaning representations. It distinguishes 21 generalization types, each of which requires generalizing from training instances to test instances in a particular systematic and linguistically informed way. Lexical generalization involves recombining known grammatical structures with words that were not observed in those particular structures in training. An example is the generalization type "subject to object (common)," in which a common noun ("hedgehog") is seen only as a subject in training yet is used only as an object in the generalization test set. Note that the syntactic structure at generalization time (e.g., that of a transitive sentence) was already observed in training.

By contrast, the three structural generalization types involve generalizing to linguistic structures that were not seen in training. Below are some examples:

Table 1 Accuracy of compositional and seq2seq models on COGS

| | | Structural | | | Lexical | |
|---|---|---|---|---|---|---|
| | | Object to subject PP | CP recursion | PP recursion | Mean of 18 other types | Overall |
| **Compositional** | AM parser (Weißenhorn et al. 2022) | 78 | 100 | 99 | 99 | 98 |
| | LeAR (Liu et al. 2021) | 93 | 100 | 99 | 99 | 99 |
| **seq2seq** | Kim & Linzen 2020 | 0 | 0 | 0 | 42 | 35 |
| | Akyürek & Andreas 2021 | 0 | 0 | 1 | 96 | 82 |
| | Zheng & Lapata 2021 | 0 | 12 | 39 | 99 | 89 |
| | Conklin et al. 2021 | 0 | 0 | 0 | 94 | 75 |
| | Csordás et al. 2021 | 0 | 0 | 0 | 88 | 75 |
| | BART (Yao & Koller 2022) | 0 | 0 | 10 | 96 | 83 |

Abbreviations: AM, Apply–Modify; seq2seq, sequence-to-sequence. Table adapted with permission from Weißenhorn et al. (2022).

(1) PP recursion:
a. *Training:* Ava saw a ball **in a bowl on the table**.
b. *Generalization:* Ava saw a ball **in a bowl on the table on the floor**.

(2) CP recursion:
a. *Training:* Emma said **that** Noah knew **that** the cat danced.
b. *Generalization:* Emma said **that** Noah knew **that** Lucas saw **that** the cat danced.

(3) Object PP to subject PP:
a. *Training:* Noah ate **the cake on the plate**.
b. *Generalization:* **The cake on the table** burned.

In particular, the training data contain PP modification of nouns, but only up to nesting depth 2; the generalization set contains PP modification of nesting depths 3–12. Furthermore, at training time, PPs only modify object NPs, whereas at test time, they can modify subject NPs.

## 4.2. Sequence-to-Sequence Models Struggle with Structural Generalization

Out of these three compositional generalization data sets, COGS is unique in that it allows us to investigate the specific linguistic phenomena that are hard for different semantic parsers. A meta-analysis by Yao & Koller (2022) compares all published compositional and seq2seq models for COGS (see **Table 1**). They find that while seq2seq models can achieve an overall accuracy above 80% on the out-of-distribution COGS test set, this is an average of close to 100% on the 18 lexical generalization types and close to zero on the three structural generalization types 1–3. Thus, seq2seq models seem to systematically struggle with assigning correct meaning representations to unseen linguistic structures—exactly the scenario for which the Principle of Compositionality was introduced in linguistics.

By contrast, Yao & Koller (2022) find that compositional semantic parsers—specifically, the LeAR and AM parser models discussed in Section 3—solve COGS at near-perfect accuracy. This dichotomy persists when COGS is viewed as a corpus for syntactic parsing by training a model to predict syntax trees rather than meaning representations. BART (Lewis et al. 2020)—the pretrained seq2seq model that underlies the very accurate Abstract Meaning Representation (AMR) parser of Bevilacqua et al. (2021)—achieves very low accuracy on structural generalization, whereas the structure-aware Neural Berkeley Parser (Kitaev & Klein 2018) solves it nearly perfectly.

## 4.3. Approaches to Generalizing Compositionally

So what is needed for a model to achieve compositional generalization, especially on unseen linguistic structures? Three classes of methods seem to be effective: directly using a compositional semantic parser, augmenting the training data for a seq2seq model with compositionality-based models, and changing the learning procedure.

### 4.3.1. Compositional semantic parsing.

One effective method for achieving compositional generalization is to simply use a compositional semantic parser. On COGS, **Table 1** paints a clear picture: Structural generalization is very hard for seq2seq models, but both LeAR and the AM parser solve the test set with near-perfect accuracy (Weißenhorn et al. 2022). Both parsers prove more broadly adept: The AM parser is unique in that it works well both on COGS and on broad-coverage semantic parsing, as discussed in Section 3. LeAR, as noted above, sets a new state of the art on CFQ. Finally, Herzig & Berant's (2021) compositional parser SpanBasedSP achieves 100% accuracy on the simple and two primitive splits of SCAN. The parser also achieves near-perfect accuracy (98%) on the parsing version of the grounded data set CLOSURE (Bahdanau et al. 2019) (cf. Section 5.3), an extension of CLEVR (Johnson et al. 2017) where each image is described by a scene that holds the attributes and positional relations of all objects; the generalization task tests understanding of referring expressions that match object properties in novel contexts.

As indicated in Section 3, care must be taken when compositional methods are applied to broad-coverage semantic parsing. However, compositional models are biased toward learning the right linguistic generalizations from limited training data. This helps them in low-resource scenarios like the compositional generalization data sets.

### 4.3.2. Data augmentation.

Compositional generalization can also be improved through data augmentation methods, which automatically recombine training instances to create new artificial training data. Data augmentation is used to expose seq2seq models that otherwise struggle on compositional generalization tasks to more data that illustrate compositionality. Andreas (2020) introduces Good-Enough Compositional Data Augmentation (GECA) for this purpose. Though GECA captures only a small number of compositional principles and makes several incorrect predictions about real language data, it is quite effective across a range of tasks: improving semantic parsing, solving representative SCAN experiments, and improving language modeling in low-resource settings with six different languages.

Qiu et al. (2021) improve over this with their CSL model, which recursively recombines training examples using a quasi-synchronous grammar (cf. Section 3.2). CSL in isolation solves COGS nearly perfectly, but it fails to achieve broad coverage on naturally occurring text, such as in the SMCalFlow sembank (Andreas et al. 2020). But when CSL is used to generate additional training data for fine-tuning the T5 seq2seq model (Raffel et al. 2020), T5 achieves excellent performance on COGS, SCAN, and SMCalFlow. This is an instance of a seq2seq model performing strongly on structural generalization in COGS, but anecdotal reports by the authors suggest that T5 only learns to correctly parse, for instance, PP recursion up to the depth to which the training data were augmented. Thus, it is the compositional data augmentation model that learns to generalize correctly, not the seq2seq model.

### 4.3.3. Meta-learning.

Meta-learning capitalizes on the intuition that if a machine learning model fits itself too closely to statistical regularities of one training set, it will perform poorly on a different task; thus, it can be encouraged to learn "correct" generalizations by training on multiple different tasks at once.

Conklin et al. (2021) construct iterative meta-training tasks for compositional generalization that divide the training data into smaller chunks to assist the model in learning specific aspects of compositionality; tasks are grouped with similarity metrics to induce systematicity (e.g., object to subject NP), productivity (recursion), and primitive application (transferring a verb in isolation to its transitive context). Each meta-train, meta-test task pair is designed to simulate the divergence between training and testing and control the nature of the regularization applied for the model to learn targeted patterns. The authors find that their methods improve seq2seq performance on MCD splits of SCAN tasks focusing on systematic generalization by 10–40 points; on COGS, their methods improve the overall accuracy of a seq2seq model by 6–8 points. McCoy et al. (2020) also use meta-learning to facilitate the acquisition of distinct languages by a neural network by adjusting model loss after each meta-training task; they find this methodology to work for both abstract biases (e.g., a bias for languages with a consistent constraint ranking) and concrete biases (e.g., a bias for treating certain phonemes as vowels).

Though promising, meta-learning has been noted to be very sensitive to both the family of meta-training tasks and test data selected to be a robust method for injecting compositional bias (Mitchell et al. 2021); additionally, the biases imparted during meta-training are not always transparent. Conklin et al. (2021) note this specifically for COGS, where different meta-training setups show large variance in the bias their models acquire; McCoy et al. (2020) find their methodology falls short in generalizing to examples of longer length.

### 4.4. Discussion

We expect two qualities of a semantic parser that turn out to be complementary: the ability to parse at broad coverage, and the ability to generalize correctly from limited observations. While the former measures performance, the latter reflects a parser's ability to capture human language competence. We have seen that there is a remarkable tension between these two qualities: Semantic parsers that are too narrowly compositional struggle with broad coverage, but seq2seq models have a hard time learning the correct structural generalizations. As Shaw et al. (2021) suggest, the space of semantic parsers that have both qualities is underexplored: Arguably, it is currently occupied only by the AM parser and the CSL-T5 setup.

One takeaway from the findings in this section is that compositional generalization requires semantic parsers that embrace compositionality. This can happen either in the parsing algorithm itself (e.g., AM parser, LeAR) or in the data augmentation method (e.g., CSL). Meta-learning is an exciting research direction that aims at learning compositional generalization as a by-product of a training setup that rewards accuracy on multiple diverse tasks. Whether this is enough to broadly bias semantic parsers toward compositionality is an interesting question for the future.

## 5. THE DESIGN SPACE OF COMPOSITIONAL MODELS

We conclude this review article by briefly discussing some recent findings on compositionality that go beyond semantic parsing. The overarching question is this: If we assume that meaning representations are constructed compositionally as in Section 4, what does that mean for the design of these meaning representations and of methods for semantic construction used in computational work? We first discuss how compositionality can be used to distinguish fundamental and superficial design differences between different meaning representation schemes (Section 5.1). We then discuss the design choices for compositional semantic parsers in terms of their expressive capacity (Section 5.2). Finally, we discuss what we can learn about compositionality from experiments on language evolution and multimodal semantics (Section 5.3).

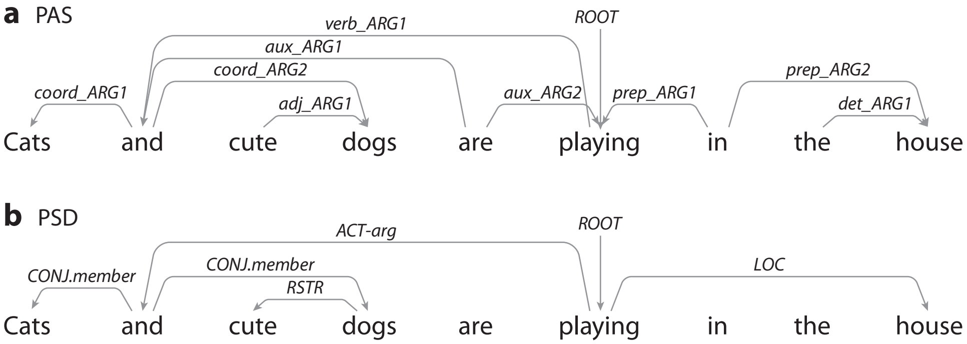

Figure 3

Meaning representations from the (*a*) PAS and (*b*) PSD sembanks. Abbreviations: PAS, Predicate–Argument Structures; PSD, Prague Semantic Dependencies. Figure adapted with permission from Donatelli et al. (2020).

### 5.1. Designing Meaning Representations for Sembanks

Whenever a sembank is developed, its annotators must make design decisions on the meaning representations that are used in the annotation. This is illustrated in **Figure 3** for the Enju Predicate–Argument Structures (PAS) and Prague Semantic Dependencies (PSD) sembanks (Oepen et al. 2015). Their meaning representations differ in many ways—some blatant (PAS includes all words in the meaning representation; PSD skips function words and copulas), others more subtle (edges point from modifiers to heads in PAS, but from heads to modifiers in PSD). This raises the question of which design decisions in a meaning representation scheme are fundamental (in that they change the information captured by the meaning representation) and which ones are superficial and more arbitrary.

Donatelli et al. (2020) offer a perspective on the "depth" of design choices that embraces compositionality as a key feature of a meaning representation. They argue that a difference between two meaning representation schemes is "shallow" if one representation $\varphi$ can be mapped to the other representation $\psi$ reversibly—that is, if the result can be deterministically mapped back into the original representation without loss of information. Importantly, such a mapping can be established systematically if both $\varphi$ and $\psi$ have the same compositional structure; in that case, the difference between $\varphi$ and $\psi$ lies exclusively in the atomic, lexical meaning representations assigned to the individual words.

Donatelli et al. (2020) model compositional structures as AM dependency trees (**Figure 2*b***) and compare the three Semantic Dependency Graph sembanks (Oepen et al. 2015); these cover the same English sentences and include the PAS and PSD sembanks illustrated in **Figure 3**. At first glance, the AM dependency trees vary quite widely across sembanks. Yet, many of these variations are quite systematic, and Donatelli et al. show that a small number of handwritten rules can resolve most of these variations. In the end, the AM dependency trees of all three graphbanks share roughly 75% of their edges, indicating that most differences between the sembanks are lexical and not structural. Design decisions with a deeper impact can then be seen more clearly. For instance, copulas, which PSD ignores and PAS treats as an auxiliary in **Figure 3*a***, show drastically different strategies for all sembanks when the copula is adjectival.

### 5.2. Expressive Capacity of Compositional Models

Compositionality can also help inform design decisions about the architecture of a semantic parser. As discussed in Section 3, we find empirically that parsers that interpret compositionality too

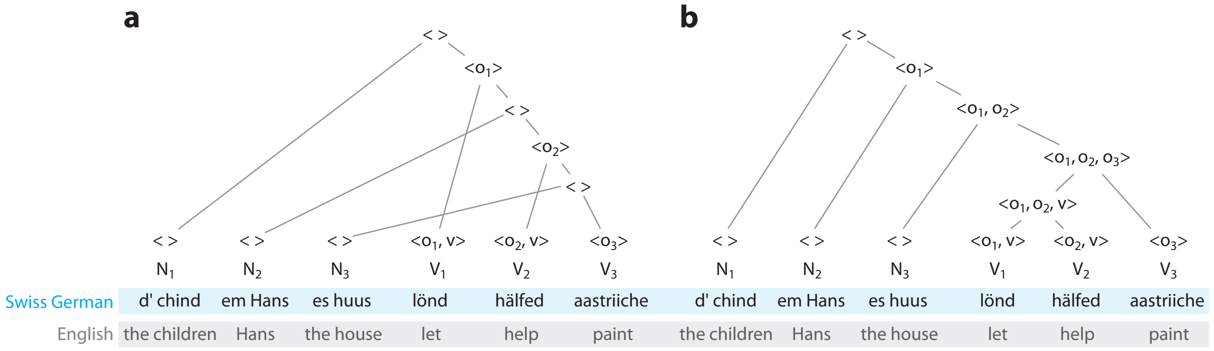
a
< >
<o1>
< >
<o2>
< >
< > < > < > <o1, v> <o2, v> <o3>
N1 N2 N3 V1 V2 V3
b
< >
<o1>
<o1, o2>
<o1, o2, o3>
<o1, o2, v>
< > < > < > <o1, v> <o2, v> <o3>
N1 N2 N3 V1 V2 V3
Swiss German
d' chind em Hans es huus lönd hälfed aastriiche
English
the children Hans the house let help paint

**Figure 4**

(*a*) A nonprojective and (*b*) a projective compositional structure for the Swiss German clause '(that we) let the children help Hans paint the house.' Figure adapted with permission from Venant & Koller (2019).

narrowly have low coverage, especially because one frequently wants to combine partial meaning representations that are far apart in the sentence, or to combine them with more powerful algebraic operations.

Venant & Koller (2019) complement these empirical findings with a theoretical perspective on the expressive capacity of a compositional semantic parser with respect to the relations between sentences and meaning representations they can describe. They find that either a compositional parser must be nonprojective (i.e., able to compositionally combine partial meaning representations that are far apart in the sentence) or its partial meaning representations must have unbounded memory capacity to remember unfilled argument positions; otherwise, the class of sentence–meaning relations it can represent is reduced. This trade-off is illustrated in the two analyses of a Swiss German subordinate clause in **Figure 4**: The nonprojective compositional structure in **Figure 4*a*** can combine words that are far apart in the sentence, and thus only needs to remember a bounded number of "object" arguments at each node, whereas the projective structure in **Figure 4*b*** must simultaneously remember the unfilled argument positions of all the verbs.

Venant & Koller's (2019) results explain why so many grammar-based methods for semantic parsing are designed as they are. Montague Grammar and CCG both require projective compositional structures and therefore must use lambda terms (with an unbounded number of variables) to achieve full expressive capacity, whereas systems with bounded memory capacity, such as HPSG (Copestake et al. 2001) or the AM parser, must allow some degree of nonprojectivity. In an era in which grammars are replaced by neural models and the design space of compositional semantic parsers in computational linguistics has opened up, these types of results can be useful milestones for navigating this design space.

## 5.3. Compositionality Beyond Semantic Parsing

Compositionality has emerged as a unifying construct across distinct tasks beyond semantic parsing, demonstrating how compositionality is both inherent in language and a useful principle to structure information beyond language. Work on emergent language in simulations between multiple neural agents explores the conditions under which language evolves in communities of agents; the emergent features can shed light on human language evolution and pinpoint what makes language flexible and interactive. Similar to compositional generalization tasks, the simplest way to probe for compositionality in such settings is to test for novel composite meanings: Can agents refer to *blue squares* upon first encounter if they have seen other blue and square things during training (Choi et al. 2018)? Recent work has found that emergent languages naturally develop the ability to refer to novel composite concepts (Lazaridou et al. 2019, Chaabouni et al. 2020). Yet, there is debate as to whether compositionality helps generalization in emergent languages or may in fact hinder it (Andreas 2019). Given these intriguing but potentially confounding empirical observations, the characterization of which architectural biases and environmental pressures favor the emergence of compositionality (or other linguistic properties) is still an open research question.

A substantial amount of recent work in computational linguistics has also investigated the use of language in multimodal settings, for instance, in the context of images or videos. Some work on language and vision has posited that semantic compositionality is a general process irrespective of the underlying modality; for example, visual compositionality consists of attribute–object relations (Nguyen et al. 2014). This research capitalizes on the fact that computational linguistics and computer vision have converged to a common way of capturing and representing the linguistic and visual information of atomic concepts through neural models. Examples of this work include compositional generalization for image captioning, which measures how well a model composes unseen combinations of concepts when describing familiar as opposed to unseen images (Holtzman et al. 2019, Nikolaus et al. 2019). VQA tasks also require models to give a (typically short) answer to a question about the content of an image (Antol et al. 2015, Johnson et al. 2017); this has been extended to compositional generalization, in which correct answers require the ability to interpret known ways of referring to objects in arbitrary contexts. The grounded data set CLOSURE (Bahdanau et al. 2019) exemplifies this: Each image is described by a scene that holds the attributes and positional relations of all objects. Generalization requires understanding of referring expressions that match object properties in novel contexts. Recent results suggest that latent compositional representations that map language to vision improve systematic generalization in this task (Bogin et al. 2021).

## 6. CONCLUSION

The Principle of Compositionality has enjoyed renewed attention in computational linguistics in the past few years. Purely neural seq2seq models have revolutionized the field and outperform grammar-based methods on many tasks, including semantic parsing. An overly narrow interpretation of compositionality severely limits the ability of a semantic parser to achieve broad coverage in semantic parsing. However, research on compositional generalization shows that models that embrace compositionality have strong advantages over purely neural models, especially when it comes to generalizing from limited training data to sentences with unseen structure. The design space of models that do well on both remains underexplored, but the Principle of Compositionality itself can offer some guidance on exploring it.

In terms of architecture choices, systems like the AM parser predict internal symbolic structures (AM dependency trees) using neural methods. Such neurosymbolic models are of interest across many of areas of artificial intelligence as they attempt to come to terms with the strengths and limitations of neural models; in his keynote lecture at the 34th Annual Meeting of the Association for the Advancement of Artificial Intelligence (AAAI-2020), Henry Kautz invokes "violent agreement on the need to bring together neural and symbolic traditions" (Kautz 2020). From this perspective, reconciling the power of the Principle of Compositionality with the power of neural models is part of a larger enterprise toward the future of artificial intelligence and cognitive science.

## DISCLOSURE STATEMENT

The authors are not aware of any affiliations, memberships, funding, or financial holdings that might be perceived as affecting the objectivity of this review.

## ACKNOWLEDGMENTS

This work was partially funded by the German Research Foundation (DFG) project KO 2916/2-2.

## LITERATURE CITED

Abzianidze L, Bjerva J, Evang K, Haagsma H, van Noord R, et al. 2017. The Parallel Meaning Bank: towards a multilingual corpus of translations annotated with compositional meaning representations. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pp. 242–47. Stroudsburg, PA: Assoc. Comput. Linguist.

Akyürek E, Andreas J. 2021. Lexicon learning for few shot sequence modeling. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 4934–46. Stroudsburg, PA: Assoc. Comput. Linguist.

Andreas J. 2019. *Measuring compositionality in representation learning*. Paper presented at the 7th International Conference on Learning Representations (ICLR 2019), New Orleans, LA, May 6–9

Andreas J. 2020. Good-enough compositional data augmentation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7556–66. Stroudsburg, PA: Assoc. Comput. Linguist.

Andreas J, Bufe J, Burkett D, Chen C, Clausman J, et al. 2020. Task-oriented dialogue as dataflow synthesis. *Trans. Assoc. Comput. Linguist.* 8:556–71

Antol S, Agrawal A, Lu J, Mitchell M, Batra D, et al. 2015. VQA: Visual Question Answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2425–33. Washington, DC: IEEE

Artzi Y, Lee K, Zettlemoyer L. 2015. Broad-coverage CCG semantic parsing with AMR. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1699–710. Stroudsburg, PA: Assoc. Comput. Linguist.

Bahdanau D, Cho K, Bengio Y. 2015. *Neural machine translation by jointly learning to align and translate*. Paper presented at the 3rd International Conference on Learning Representations (ICLR 2015), San Diego, CA, May 7–9

Bahdanau D, de Vries H, O'Donnell TJ, Murty S, Beaudoin P, et al. 2019. CLOSURE: assessing systematic generalization of CLEVR models. arXiv:1912.05783 [cs.AI]

Banarescu L, Bonial C, Cai S, Georgescu M, Griffitt K, et al. 2013. Abstract Meaning Representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pp. 178–86. Stroudsburg, PA: Assoc. Comput. Linguist.

Bastings J, Baroni M, Weston J, Cho K, Kiela D. 2018. Jump to better conclusions: SCAN both left and right. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 47–55. Stroudsburg, PA: Assoc. Comput. Linguist.

Bender EM, Koller A. 2020. Climbing towards NLU: on meaning, form, and understanding in the age of data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 5185–98. Stroudsburg, PA: Assoc. Comput. Linguist.

Bevilacqua M, Blloshmi R, Navigli R. 2021. One SPRING to rule them both: symmetric AMR semantic parsing and generation without a complex pipeline. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI-21)*, Vol. 35, pp. 12564–73. Palo Alto, CA: AAAI Press

Bogin B, Subramanian S, Gardner M, Berant J. 2021. Latent compositional representations improve systematic generalization in grounded question answering. *Trans. Assoc. Comput. Linguist.* 9:195–210

Chaabouni R, Kharitonov E, Bouchacourt D, Dupoux E, Baroni M. 2020. Compositionality and generalization in emergent languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4427–42. Stroudsburg, PA: Assoc. Comput. Linguist.

Choi E, Lazaridou A, de Freitas N. 2018. *Compositional obverter communication learning from raw visual input*. Paper presented at the 6th International Conference on Learning Representations (ICLR 2018), Vancouver, Can., Apr. 30–May 3

Chomsky N. 1957. *Syntactic Structures*. The Hague, Neth.: Mouton

Conklin H, Wang B, Smith K, Titov I. 2021. Meta-learning to compositionally generalize. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint*

*Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 3322–35. Stroudsburg, PA: Assoc. Comput. Linguist.

Copestake A, Lascarides A, Flickinger D. 2001. An algebra for semantic construction in constraint-based grammars. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*, pp. 140–47. Stroudsburg, PA: Assoc. Comput. Linguist.

Csordás R, Irie K, Schmidhuber J. 2021. The devil is in the detail: Simple tricks improve systematic generalization of transformers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 619–34. Stroudsburg, PA: Assoc. Comput. Linguist.

Dalrymple M, ed. 2022. *The Handbook of Lexical Functional Grammar: Empirically Oriented Theoretical Morphology and Syntax*. Berlin: Lang. Sci. Press

Donatelli L, Groschwitz J, Lindemann M, Koller A, Weißenhorn P. 2020. Normalizing compositional structures across graphbanks. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 2991–3006. n.p.: Int. Comm. Comput. Linguist.

Fodor JA, Lepore E. 2002. *The Compositionality Papers*. Oxford, UK: Oxford Univ. Press

Fodor JA, Pylyshyn ZW. 1988. Connectionism and cognitive architecture: a critical analysis. *Cognition* 28(1):3–71

Groschwitz J, Fowlie M, Koller A. 2021. Learning compositional structures for semantic graph parsing. In *Proceedings of the 5th Workshop on Structured Prediction for NLP (SPNLP 2021)*, pp. 22–36. Stroudsburg, PA: Assoc. Comput. Linguist.

Groschwitz J, Lindemann M, Fowlie M, Johnson M, Koller A. 2018. AMR dependency parsing with a typed semantic algebra. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1831–41. Stroudsburg, PA: Assoc. Comput. Linguist.

Halle M, Marantz A, Hale K, Keyser SJ. 1993. Distributed morphology and the pieces of inflection. In *The View from Building 20*, ed. K Hale, SJ Keyser, pp. 111–76. Cambridge, MA: MIT Press

Herzig J, Berant J. 2021. Span-based semantic parsing for compositional generalization. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 908–21. Stroudsburg, PA: Assoc. Comput. Linguist.

Holtzman A, Buys J, Du L, Forbes M, Choi Y. 2019. The curious case of neural text degeneration. arXiv:1904.09751 [cs.CL]

Jambór D, Bahdanau D. 2021. LAGr: labeling aligned graphs for improving systematic generalization in semantic parsing. arXiv:2110.07572 [cs.CL]

Janssen T. 2001. Frege, contextuality and compositionality. *J. Logic Lang. Inform.* 10(1):115–36

Johnson J, Hariharan B, van der Maaten L, Fei-Fei L, Zitnick CL, Girshick R. 2017. CLEVR: a diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1988–97. Washington, DC: IEEE

Kautz H. 2020. *The third AI summer*. Presented at the 34th Annual Meeting of the Association for the Advancement of Artificial Intelligence (AAAI-2020), New York, Feb. 10

Kay P, Michaelis LA. 2019. Constructional meaning and compositionality. In *Semantics–Interfaces*, ed. C Maienborn, K Heusinger, P Portner, pp. 293–324. Berlin: De Gruyter Mouton

Keysers D, Schärli N, Scales N, Buisman H, Furrer D, et al. 2020. *Measuring compositional generalization: a comprehensive method on realistic data*. Paper presented at the 8th International Conference on Learning Representations (ICLR 2020), virtual, Apr. 26–May 1

Khashabi D, Min S, Khot T, Sabharwal A, Tafjord O, et al. 2020. UNIFIEDQA: crossing format boundaries with a single QA system. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 1896–1907. Stroudsburg, PA: Assoc. Comput. Linguist.

Kim N, Linzen T. 2020. COGS: a compositional generalization challenge based on semantic interpretation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pp. 9087–105. Stroudsburg, PA: Assoc. Comput. Linguist.

Kitaev N, Klein D. 2018. Constituency parsing with a self-attentive encoder. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2676–86. Stroudsburg, PA: Assoc. Comput. Linguist.

Kwiatkowski T, Zettlemoyer L, Goldwater S, Steedman M. 2010. Inducing probabilistic CCG grammars from logical form with higher-order unification. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1223–33. Stroudsburg, PA: Assoc. Comput. Linguist.

Lake B, Baroni M. 2018. Generalization without systematicity: on the compositional skills of sequence-to-sequence recurrent networks. In *Proceedings of Machine Learning Research*, Vol. 80: *International Conference on Machine Learning*, ed. J Dy, A Krause, pp. 2873–82. n.p.: PMLR

Lazaridou A, Hermann K, Tuyls K, Clark S. 2019. *Emergence of linguistic communication from referential games with symbolic and pixel input*. Paper presented at the 6th International Conference on Learning Representations (ICLR 2018), Vancouver, Can.

Lewis M, Liu Y, Goyal N, Ghazvininejad M, Mohamed A, et al. 2020. BART: denoising sequence-to-sequence pretraining for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7871–80. Stroudsburg, PA: Assoc. Comput. Linguist.

Lindemann M, Groschwitz J, Koller A. 2019. Compositional semantic parsing across graphbanks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4576–85. Stroudsburg, PA: Assoc. Comput. Linguist.

Lindemann M, Groschwitz J, Koller A. 2020. Fast semantic parsing with well-typedness guarantees. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 3929–51. Stroudsburg, PA: Assoc. Comput. Linguist.

Liu C, An S, Lin Z, Liu Q, Chen B, et al. 2021. Learning algebraic recombination for compositional generalization. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 1129–44. Stroudsburg, PA: Assoc. Comput. Linguist.

Lyu C, Titov I. 2018. AMR parsing as graph prediction with latent alignment. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pp. 397–407. Stroudsburg, PA: Assoc. Comput. Linguist.

Marantz A. 1997. No escape from syntax: Don't try morphological analysis in the privacy of your own lexicon. *Univ. Pa. Work. Pap. Linguist.* 4(2):14

McCoy RT, Grant E, Smolensky P, Griffiths TL, Linzen T. 2020. Universal linguistic inductive biases via meta-learning. arXiv:2006.16324 [cs.CL]

Mitchell E, Finn C, Manning C. 2021. Challenges of acquiring compositional inductive biases via meta-learning. In *Proceedings of Machine Learning Research*, Vol. 140: *AAAI Workshop on Meta-Learning and MetaDL Challenge*, pp. 138–48. n.p.: PMLR

Nguyen DT, Lazaridou A, Bernardi R. 2014. Coloring objects: adjective-noun visual semantic compositionality. In *Proceedings of the Third Workshop on Vision and Language*, pp. 112–14. Dublin, Irel./Stroudsburg, PA: Dublin City Univ./Assoc. Comput. Linguist.

Nikolaus M, Abdou M, Lamm M, Aralikatte R, Elliott D. 2019. Compositional generalization in image captioning. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pp. 87–98. Stroudsburg, PA: Assoc. Comput. Linguist.

Oepen S, Kuhlmann M, Miyao Y, Zeman D, Cinková S, et al. 2015. SemEval 2015 task 18: broad-coverage semantic dependency parsing. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pp. 915–26. Stroudsburg, PA: Assoc. Comput. Linguist.

Partee BH. 1984. Compositionality. In *Varieties of Formal Semantics: Proceedings of the 4th Amsterdam Colloquium, September 1982*, Vol. 3, ed. F Landman, F Veltman, pp. 281–311. Dordrecht, Neth.: Foris

Pater J. 2019. Generative linguistics and neural networks at 60: foundation, friction, and fusion. *Language* 95(1):e41–74

Pollard C, Sag I. 1994. *Head-Driven Phrase Structure Grammar*. Chicago: Univ. Chicago Press

Qiu L, Shaw P, Pasupat P, Nowak PK, Linzen T, et al. 2021. Improving compositional generalization with latent structure and data augmentation. arXiv:2112.07610 [cs.CL]

Raffel C, Shazeer N, Roberts A, Lee K, Narang S, et al. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* 21(140):1–67

Shaw P, Chang MW, Pasupat P, Toutanova K. 2021. Compositional generalization and natural language variation: Can a semantic parsing approach handle both? In *Proceedings of the 59th Annual Meeting of the*

*Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 922–38. Stroudsburg, PA: Assoc. Comput. Linguist.

Steedman M. 2000. *The Syntactic Process*. Cambridge, MA: MIT Press

Sutskever I, Vinyals O, Le QV. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems (NIPS'14)*, Vol. 2, ed. Z Ghahramani, M Welling, C Cortes, ND Lawrence, KQ Weinberger, pp. 3104–12. Cambridge, MA: MIT Press

Szabó ZG. 2012. The case for compositionality. In *The Oxford Handbook of Compositionality*, ed. W Hinzen, E Machery, M Werning, pp. 64–80. Oxford, UK: Oxford Univ. Press

Venant A, Koller A. 2019. Semantic expressive capacity with bounded memory. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 65–79. Stroudsburg, PA: Assoc. Comput. Linguist.

Wang B, Lapata M, Titov I. 2021. *Structured reordering for modeling latent alignments in sequence transduction*. Paper presented at the 35th Conference on Neural Information Processing Systems (NeurIPS 2021), virtual, Dec. 6–14

Weißenhorn P, Donatelli L, Koller A. 2022. Compositional generalization with a broad-coverage semantic parser. In *Proceedings of the 11th Joint Conference on Lexical and Computational Semantics*, pp. 44–54. Stroudsburg, PA: Assoc. Comput. Linguist.

Wong YW, Mooney RJ. 2007. Learning synchronous grammars for semantic parsing with lambda calculus. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 960–67. Stroudsburg, PA: Assoc. Comput. Linguist.

Yanaka H, Mineshima K, Inui K. 2021. SyGNS: a systematic generalization testbed based on natural language semantics. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 103–19. Stroudsburg, PA: Assoc. Comput. Linguist.

Yao Y, Koller A. 2022. Structural generalization is hard for sequence-to-sequence models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 5048–62. Stroudsburg, PA: Assoc. Comput. Linguist.

Yin P, Fang H, Neubig G, Pauls A, Platanios EA, et al. 2021. Compositional generalization for neural semantic parsing via span-level supervised attention. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2810–23. Stroudsburg, PA: Assoc. Comput. Linguist.

Yin P, Neubig G. 2017. A syntactic neural model for general-purpose code generation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 440–50. Stroudsburg, PA: Assoc. Comput. Linguist.

Yu T, Zhang R, Yang K, Yasunaga M, Wang D, et al. 2018. Spider: a large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-SQL task. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 3911–21. Stroudsburg, PA: Assoc. Comput. Linguist.

Zelle JM, Mooney RJ. 1996. Learning to parse database queries using inductive logic programming. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI'96)*, Vol. 2, pp. 1050–55. Palo Alto, CA: AAAI Press

Zettlemoyer LS, Collins M. 2005. Learning to map sentences to logical form: structured classification with probabilistic categorial grammars. In *Proceedings of the 21st Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 658–66. Arlington, VA: AUAI Press

Zheng H, Lapata M. 2021. Disentangled sequence to sequence learning for compositional generalization. arXiv:2110.04655 [cs.CL]