Modelling Child Learning and Parsing of Long-range Syntactic Dependencies

Louis Mahon[a], Mark Johnson[b], Mark Steedman[a]

[a]*School of Informatics, University of Edinburgh, United Kingdom*
[b]*School of Computing, Macquarie University, Australia*

Abstract

This work develops a probabilistic child language acquisition (CLA) model to learn a range of linguistic phenonmena, most notably long-range syntactic dependencies of the sort found in object wh-questions, among other constructions. The model is trained on a corpus of real child-directed speech, where each input string is paired with a logical form as a meaning representation. It then learns both word meanings and language-specific syntax simultaneously. After training, the model can deduce the correct parse tree and word meanings for a given string-meaning pair, and can infer the meaning if given only the string.

1. Introduction

This paper develops a computational model of CLA, seeking to understand the process of language acquisition by programming a computer to emulate the learning undergone by the child. We focus specifically on the learning of syntax and semantics, that is, learning the meaning of individual words, and learning the language-specific syntax by which they combine to produce a single meaning for a whole utterance. Unlike other computational approaches to language learning, such as the dominant paradigm of large language models, which make minimal prior assumptions about language structure and attempt to learn everything from the data, we distinguish between what there is reason to believe is innate vs learned. In line with semantic bootstrapping theory (Pinker, 1979), we assume the child possesses a rich system of semantic categories prior to language acquisition, such as actions, individuals and attributes, which are combined in sentence meanings or logical forms, and, together with the sentence itself, provide the input to "bootstrapping" the lexical word meanings and syntactic categories of the

language in question, via a syntactic and semantic derivation induced by the application of universal rules of functional application and composition.

The theoretical backbone of our model is provided by Combinatory Categorial Grammar (CCG, Steedman, 2000). combinatory categorial grammar (CCG) is a strongly lexicalized theory of grammar, in which all details that are language-specific, such as the linear order of clausal constituents and their mapping to logical form or meaning, is specified in the lexical entry for categories such as verbs. It is the language-specific lexicon that has to be learned by the child and the computer. The universal CCG rules of syntactic combination or merger are assumed to be innate, and exhibit tight-coupling of syntax and semantics, with a one-to-one correspondence between the semantic operations of the logical combinators, and the syntactic operations of combining grammatical constituents. This facilitates learning both in a single unified, computational model.

Learning takes place when strings from real child-directed speech taken from the CHILDES corpus (MacWhinney, 1998) are paired with a corresponding logical form (lf) representing their meanings[1] , via semantic annotation such as that provided by Szubert et al. (2024). Training uses an incremental expectation-maximization-style algorithm (Neal and Hinton, 1999): the model considers each possible valid analysis, weighted by its current estimated probability, and then increases the estimated probability on all constituents in that analysis in proportion to this weight. (So the high probability of previously learned words contributes to the probability assigned to newly encountered ones.)

The work falls between Universal Grammar (UG)-based approaches that assume expressive theories of grammar drawn from theoretical linguistics (Chomsky, 1965, 1981, 1995), and seek to identify constraints, such as Freezing Principles and Subset Conditions, that will make such grammars learnable from paired meaning representations and strings (Wexler and Culicover, 1980; Berwick, 1985; Gibson and Wexler, 1994; Fodor, 1998; Yang, 2002), and Usage-based theories based on memorization of exemplars of child directed utterance (CDU), with or without subsequent generalization (Tomasello, 2003; Bybee, 2006; Frank et al., 2007; Bannard and Matthews, 2008; Ambridge, 2020). In comparison with other UG-based approaches, CCG is syntactically of low, near-context-free, expressive power (Vijay-Shanker and Weir, 1994), and is semantically surface-compositional, requiring no constraints other than the universal rules of grammatical composition of ordered adjacent elements and a universal inventory of lexical types. Similarly

[1]This refers to interpretable logical form, rather than Chomskian 'big L' Logical Form.

to Usage-based approaches, our learner allows lexicalization of what in terms of the adult language are multi-word items, including entire CDUs, but also embodies a mechanism for automatically decomposing such items on distributional grounds.

Chater and Christiansen (2018) characterize language acquisition as the learning of a perceptuo-motor skill. Chater and Christiansen emphasise that much information relevant to language learning is forgotten quickly, necessitating that learning occurs rapidly and in real-time (in this sense, it is the direct opposite of the radical exemplar theory of Ambridge (2020), and in line with our own incremental approach). Another point they emphasize is the social context in which the child hears the utterance. We account for the first point by training on each example only once, one at a time, in the order they appear to the child. Pragmatic context is not currently represented in our input to the learner, except insofar as it is implicit in our use of adjacent CDU meanings as distractors from the intended meaning.

In the area of natural language processing (NLP), on the other hand, much work is currently focussed on large language models (LLMs), which require too much training data to be plausible models of how humans acquire language. Typically the amount is at least several orders of magnitude more tokens than a human sees in their entire life. Attempts have been made to better learn from a number of tokens more similar to that required by humans (Warstadt et al., 2023a), but this is more of an engineering challenge to improve sample efficiency to a level consistent with human exposure to language data, rather than an explicit attempt to model the learning process of a child. Such models still generally employ multi-epoch training, batched parameter updates, and arbitrary text tokenization, which are not plausible features of child language acquisition. Additionally, the datasets and order of presentation are not constrained to be realistic, whereas in our work, we use real child-directed speech (CDS), as suggested, for example, by Dupoux (2018), and present the utterances once only, in exactly the order that they appear to the child.

Some prior works have aimed to more realistically model child acquisition of syntax and semantics (Siskind, 1992, 1996a; Mahon et al., 2024; Abend et al., 2017; Kwiatkowski et al., 2012; Siskind, 1993, 1996b). Ours extends these in two main respects. Firstly, it widens the set of syntactic constructions that can be handled to the following: intransitives, transitives, ditransitives, modals, progressives, negations, subject inversion questions and, most importantly, long-range dependencies of the sort found in object wh-questions such as "what$_i$ do you want$_i$?", potentially including extraction from embedding ("what$_i$ do you think

you want$_i$?"). While the model of Abend et al. (2017) was limited in its capability for long-range dependencies, in order to handle them with cross-linguistically adequate expressive power, we have implemented aspects of CCG that are trans-context-free. (Technically, CCG is equivalent to a level 2 multiple context-free grammar (2-MCFG, Seki et al. (1991).) Secondly, our model is able to produce the entirely correct analysis for sentence-meaning pairs at train time, and even infer the meaning at test time if given only the string. While Abend et al. (2017) showed some limited ability to infer meanings from the string alone, our model can do so with a much higher accuracy. Relatedly, Abend et al. (2017) did not present evidence of fully correct parse trees for unseen strings without corresponding lfs, and it is not clear which of those meanings that were correctly inferred were the result of memorization as single-word strings. As discussed in Section 5.2, we observe that this is common behaviour for our model early in training. In contrast, we show that our model is able to produce fully correct parse trees for unseen strings, and therefore can infer corresponding meanings beyond memorization. In Section 4.9, we present several qualitative examples of such inferred trees across a variety of utterance types.

The novel contributions of this work are as follows:

- modelling the learner‘s ability to parse and interpret novel child-directed utterance;
- learning a wider variety of syntactic constructions, including object wh-questions, which contain potentially unbounded long-range dependencies (LRDs);
- higher accuracy and robustness across the various measures of learning that we test.

2. Theoretical Underpinnings

Our model deals with syntactic and semantic learning only. It assumes the child either has already learned to segment the speech stream and detect potential word boundaries, as evidenced in even young prelinguistic infants (Mattys et al., 1999), or is jointly learning phonotactics and morphology with syntax, as in the model of Goldberg and Elhadad (2013). At that point, the child must learn to combine atomic units (words) to produce a meaning representation that depends on (a) the meaning of the constituent words and (b) the manner in which they combine. Initially, for such a child, both are unknown.

In our framework, this problem manifests in the following way. When a child hears an string "Bambi is home", we assume that, from a combination of perceptual context and background and innate knowledge, it can approximately identify the meaning of the entire utterance as some object in some state: $\mathit{home}(\mathit{bambi})$. The task is then to figure out which words correspond to which parts of the meaning representation, and the language-specific principles by which they combine. As well as the correct interpretation, where English subjects precede VPs, others are also possible, e.g. "Bambi" means $\mathit{home}(\cdot)$, "is home" means *bambi* and subjects follow VPs.

2.1. Semantic and Syntactic Bootstrapping

Semantic bootstrapping (Pinker, 1979; Grimshaw, 1981; Brown, 1973; Bowerman, 1973; Schlesinger, 1971) is a theory arising from the observation that children understand semantic categories, such as action, object or property, prior to learning language, and that these categories help the child learn syntactic categories. For example, Gropen et al. (1991) showed that when children are exposed to a ditransitive verb that means making something move in a certain way, they expect the moving thing to be the direct object syntactically, whereas for verbs that mean making something change its state as a result of something else moving, they expect the moving thing to be the *indirect* object. This shows that knowing the meaning of the words in a sentence can help guide the child to understand the syntax of that sentence.

Syntactic bootstrapping (Gleitman, 1990), on the other hand, emphasises that prior *syntactic* knowledge guides children's learning of word meanings (semantic knowledge). For example, the results of Fisher et al. (1994) suggest that, when children are presented with a situation that is ambiguous semantically between two options as to which is the agent, they are able to resolve the ambiguity from their syntactic knowledge as to which noun phrase is the subject and which the object. Specifically, Fisher et al. presented children with scenes in which a ball was being transferred from an elephant to a rabbit, paired with a sentence containing the nonce word 'biffing', and were then asked which familiar word was closest in meaning to 'biffing'. If the paired sentence was "the elephant is biffing the rabbit", they selected 'give' as closest, but if it was "the rabbit is biffing the elephant", they selected 'receive' as closest, i.e., they made whatever interpretation of "biffing" allowed the agent to fall in subject position.

Abend et al. (2017) and Mahon et al. (2024) have shown for simple transitive sentences that syntactic bootstrapping can be seen simply as a later stage of semantic bootstrapping, at which the syntactic category of all words but one in

an unseen string have been learned, so that syntax and semantics can be acquired from a single model.

In this paper, we extend this method to cover a much wider set of syntactic constructions, including LRDs of the sort found in wh-questions, of the kinds investigated in CHILDES and other acqusition datasets by Klima and Bellugi (1966) and Stromswold (1995), among others discussed below.

2.2. Combinatory Categorial Grammar.

We choose CCG (Steedman, 2000) as a theory of grammar suitable for learning of this sort, because of its tight coupling of syntactic derivation and semantic composition.

All information that is specific to a given language, such as English, is specified in CCG in the lexicon, by a syntactic category, such as NP for the proper noun "Harry", or S\NP for the intransitive verb "walks". The latter category identifies the verb as a *function* applying to constituents of type NP (such as "Harry") to yield a sentence (such as "Harry walks"). The backward or left-leaning slash \ in the category S\NP further specifies the subject NP argument as having to occur to the left of the verb in this language.

In the present categorial notation, the convention is that argument-types (such as NP here) always appear to the right of the slash, so that the syntactic category of the English transitive verb "sees" is written S\NP/NP, where the forward or right-leaning slash / means that the object NP is found to the right of the verb in this language. All function categories are binary or "curried", and slashes "associate to the left", so this category is equivalent to (S\NP)/NP, specifying the object as the first argument to combine.

Each syntactic category is paired with a logical form (lf) representing its meaning, which in the case of verbs is also a function, specified as a λ-term. For example, the full lexical entries for the above categories are the following:

(1) "Harry" := NP : *harry*
"walks" := S\NP : $\lambda y.walks\,y$
"sees" := S\NP/NP : $\lambda x.\lambda y.sees\,x\,y$.

In the case of an SVO language like English, the order of combination of syntactic subject and object arguments of a trasitive verb like "sees" in 1 happens to be aligned with the (object-first) order of combination of the corresponding semantic arguments x and y at lf. The latter is assumed to be universal for transitive predicates across all languages. However, other languages are free to align syntactic and semantic combination differently, as is the case for VSO languages like Scots

Gaelic, where the subject is the first syntactic argument. Section 4 shows that our learner allows for this possibility, and considers all possible alignments.

Such categories and their lf meaning representations combine synchronously via a number of combinatory rules, of which the two most simple are the following rules that respectively apply rightward- and leftward- looking functions like verbs to their arguments such as noun-phrases:

(2) *The application rules:*

a. Forward application

$$X/Y : f \quad Y : a \quad \Rightarrow \quad X : f\,a \qquad (>)$$

b. Backward application

$$Y : a \quad X\backslash Y : f \quad \Rightarrow \quad X : f\,a \qquad (<)$$

These rules allow CCG derivations like the one shown in Figure 1 for the simple child-directed transitive sentence "You lost a shoe" from the Adam corpus.

The derivation in Figure 1 uses the application rules only. However, the CCG lexicon also includes "type-raised" categories for NPs, which have the effect of exchanging the roles of verbs and NPs as functions and arguments. Moreover, as well as the rules 2 of function application, CCG also includes rules of function composition, of which the following is the only instance used in the present paper:

(3) *The composition rules (***B***)*

a. Forward composition

$$X/Y : f \quad Y/Z : g \quad \Rightarrow \quad X/Z : \lambda z.f(g\,z) \qquad (>\mathbf{B})$$

Type-raised categories and composition rules allow some extra derivations, as shown in Figure 2. This derivational ambiguity is harmless, since, as the figure shows, they yield the same logical form for canonical sentences. However, they are crucial to the derivation of long-range dependencies, such as those involved in *wh*-questions like the one shown in Figure 3, which are not otherwise derivable.

Type-raised categories were not included in previous work related to ours (Abend et al., 2017; Mahon et al., 2024). In section 4 we will show that their inclusion, together with that of rules of function composition, allows our model to learn LRDs of this kind.

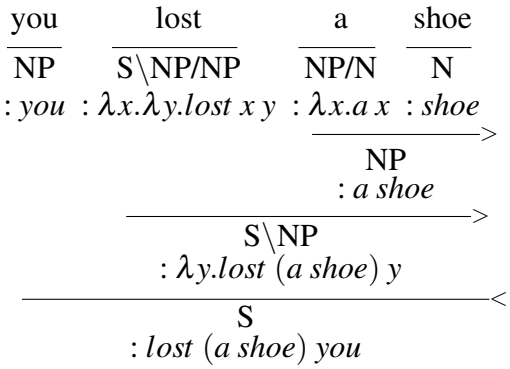

Figure 1: Example of a CCG derivation for a simple transitive sentence from the Adam (English) corpus.

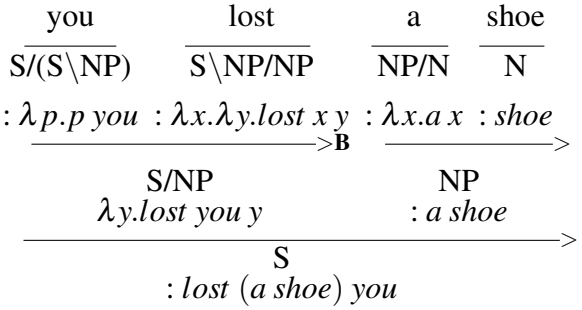

Figure 2: Example of the alternative, type-raise and compose, CCG derivation for the sentence in Figure 1 from the Adam (English) corpus.

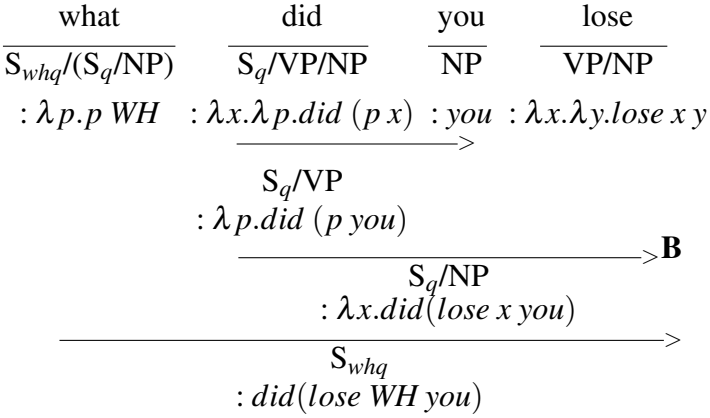

Figure 3: Example of a CCG derivation of the object-wh question corresponding to Figure 1.

It will be noticed that the logical forms exemplified in Figures 1 through 3 are, as a consequence of the process of semi-automatic annotation of the CHILDES dataset (Szubert et al., 2024), somewhat English-specific in comparison to anything we might imagine to be the form of the universal language of mind to which child language learners are assumed to have access. This means that if our learner were faced with the corresponding French strings paired with the same logical form, it would begin by learning a lot of multiword items, such as "Qu'est-ce que" with the meaning of "what", $\lambda p.p\ WH$, and "range" with the meaning of "put away", $\lambda x \lambda y.put\ away\ x\ y$. However, the learner would still learn from such data, and in many cases generalize to a more standard lexicon.

2.3. Long-range Dependencies

LRD constructions, as we use the term here, are those in which a word depends semantically on a word or set of words that are arbitrarily far away in the sentence, as in "what did you lose?". In some grammars (though not in CCG), these are treated as filler-gap dependencies, related by a discontinuous operator, such as movement. Figure 3 shows the CCG derivation of an object wh-question from the Adam corpus. Note the use of the type-raised category on the wh-word 'what', and the composition operator on the second-to-last line.[2] The ability to correctly handle LRDs is essential for accurately modelling real-world language, where such constructions are common. It is also of theoretical importance because the mechansim used in CCG to establish such dependencies properly includes that of context-free grammar in the Chomsky hierarchy (Vijay-Shanker and Weir, 1994), a property which is known to be necessary to capture natural language in general. (Chomsky, 1957; Shieber, 1985). In our corpora of child-directed speech, object wh-questions appear with high frequency, accounting for 21.6% of all utterances, and including some of the most common strings such as "what do you want?", "what are you doing?", and "what's that?".

3. Method

3.1. Probabilistic Model

The probabilistic model is broadly the same as that described in Mahon et al. (2024). In our framework for syntactic and semantic learning, each example con-

[2] In wh-questions, such as Figure 3, the wh-word is a second-order type. In the full theory, all NPs are type-raised to second-order to capture further coordination/relativization phenomena (see Steedman (2000)), though this is not a part of our model.

sists of the string of words in the string w, the meaning representation m, and the parse tree T. The parse tree is always unobserved so it is treated as a latent variable. We fit an approximation to the joint data distribution $P(w,m,T)$ via several univariate conditional distributions.

Typically, CCG parsing is discussed in terms of combining constituents via combinatory rules to derive a root. For example, the last step of the derivation in Figure 1 uses the Backward Application Rule: $Y, X\backslash Y \rightarrow X$. Our learning model, when interpreting a sentence-meaning pair, runs these combinators in reverse, that is, it proceeds by successively splitting a root into smaller chunks until they can be aligned with word spans. We will thus often speak of CCG 'splits', by which we mean the CCG combinators run in reverse. The net effect is that our model considers all possible ways to split up the sentence and the meaning representation so that the semantic constituency corresponds to the string elements.

For the parse tree, the fit distribution has the form $p_t(y|s)$, where y is either a pair (s_1, s_2) of CCG syntactic categories that combine to form s, or else a symbol *leaf*, indicating that the category should not be split in this parse tree. There is also a distribution $p_r(s)$, that predicts a root category. The distribution relating word w and meaning representation m has the form $p_w(w|m)$, and similarly for the distribution relating syntactic category s to meaning representation m. Following Abend et al. (2017), we first predict a shell lf, consisting of semantic types for all non-variables, and then predict the lf from the shell lf. Thus, the shell lf, e, for meaning m and category s, allows $p(m|s)$ to be decomposed as $p_l(m|e)$ and $p_h(e|s)$. The shell lf replaces all non-variable terms with a placeholder marked for the function of the placeholder: verb (for which we write 'vconst'), entity, determiner etc. The function is inferred from the CHILDES part-of-speech tag given in the method of Szubert et al. (2024). For example the lf $\lambda y.lost\,(a\,shoe)\,y$ from Figure 1 has shell logical form $\lambda y.\text{vconst}\,y\,(\text{quant noun})$ This allows the model to share representation power for the structure of the logical form across different examples that may have different values for the constants. See Mahon et al. (2024) for details.

Each of these distributions is modelled as a Dirichlet process, to which Bayesian updates are applied at each training example. Taking p_w as an example, the form of the posterior is then

$$p_w(w|m) = \frac{n(w,m) + \alpha H(w|m)}{n(m) + \alpha}, \tag{1}$$

where $n(w,m)$ is the number of times w and m have been observed together in the past, $n(m)$ is the number of times m has been observed in the past, and $H(x)$ is a

pre-defined base distribution (see Appendix C). An analogous definition holds for p_l, p_h and p_t. The alpha parameter is set to 1 for all distributions, corresponding to a uniform Dirichlet prior across simplices. During training, we set $\alpha = 10$ in p_t to encourage exploration of different syntactic structures, and $\alpha = 0.25$ in p_w to produce more confident predicted word meanings, which we find helps stabilize syntax learning.

The probability assigned to a full analysis, consisting of a parse tree and a meaning for each leaf node, is the product of the probabilities of all of constituent nodes given their parents. This is a stronger independence assumption than is made in head-dependency models (Collins, 1997). Our model would fail to resolve attachment ambiguity such as that between high attachment of "with"-adjuncts in "I saw a squirrel with a telescope", and low attachment, as in "I saw a squirrel with an acorn". We expect that our model would handle such ambiguities with the future addition of a "supertagger" (Srinivas and Joshi, 1994; Lewis and Steedman, 2014; Collins, 1997). This would be a neural model, e.g. a small encoder-only transformer, which predicts a small set of possible CCG categories for each word in the current string context. This model is related to two-factor theories of processing advanced in the psycholinguistic literature by Ferreira (2007) and Kahneman (2011), among others. Based on surrounding context, including words like "saw", "squirrel", and "telescope", such a model would learn to predict the category VP\VP/NP for "with", in contrast to contexts including "saw", "squirrel" and "acorn", which predict N\N/NP, thus resolving the ambiguity.

3.2. Training

The parameter updates described in Section 3.1 require tracking the number of times two different elements co-occur. For example, in p_w, the probability of predicting the logical form $\lambda x.\lambda y.lost\,x\,y$ to be realized as the word 'lost' depends on the number of times that logical form and word were observed together during training. Because we do not observe parse trees directly, we instead employ an expectation-maximization (EM) algorithm, as follows. When, at time t, the model observes a single training example (w,m), consisting of a string w and a corresponding logical form m, it uses its current parameter values $\theta^{(t)}$ to estimate a distribution over all possible parses that connect the two. The set of parameters, $\theta^{(t)}$ consists of the occurrence counts in the Dirichlet processes, e.g. the number of times a leaf meaning such as $\lambda x.\lambda y.lost\ x\ y$ occurs with a word such as 'lost'. The set of parameters can grow throughout training as new occurrences are observed. The probability assigned to a parse tree T and the training example (w,m)

is the following product

$$p(w,m,T|\theta^{(t)}) = p_r(r)\prod_{s'} p_t(s_1,s_2|s')\prod_{s} p_t(\text{leaf}|s)p_h(e_s|s)p_l(m_s|e_s)p_w(w_s|m_s)\,, \tag{2}$$

where r is the root category, s' ranges over all non-leaf nodes in T, s_1 and s_2 are the children of s', s ranges over all leaf nodes in T, and e_s, m_s and w_s are, respectively, the shell logical form, the logical form, and the word aligned to s in T. Probabilities for the leaf-level lfs m_s are determined by the root lf m, together with the shell lfs of each node, e_s, and the model parameters θ, and the same is true for w_s and w. As (w,m) is something we observe, we are interested in the conditional probability of a given parse tree

$$p(T|w,m,\theta^{(t)}) = \frac{p(w,m,T|\theta^{(t)})}{\sum_{T'\in\mathscr{T}} p(w,m,T'|\theta^{(t)})}\,, \tag{3}$$

where $\mathscr{T}$ is the set of all allowable parses of (w,y). For each parse tree, the co-occurrences that it gives rise to are recorded in proportion to the parse tree's probability. Combining the standard EM update rule with the Bayesian update for the Dirichlet process, then, for each parameter in $\theta_i \in \theta$ that tracks the co-occurrence of two elements a and b, the update rule is given by

$$\theta_i^{(t+1)} = \theta_i^{(t)} + \mathbb{E}_{T\sim p(T|w,y,\theta^{(t)})}[\delta_T(a,b)]\,,$$

where $\delta_T(a,b)$ is an indicator function that is 1 if a and b co-occur in T and 0 otherwise. The relation between the variables T, e, m, w and θ is given in Figure 4.

The set $\mathscr{T}$ of allowable parse trees is the set of all valid CCG parse trees that have the observed lf as root, the words in the observed string as leaves, and that have congruent syntactic and semantic types.[3] We require the semantic category to be congruent with the CCG category, for each node. CCG's tight coupling of syntax and semantics provides a straightforward mapping from syntactic to semantic categories. In particular, the CCG atomic categories *S* and *NP* correspond to the Montagovian t and e respectively, and the slashes in non-atomic categories correspond to functions between types.[4] For example, the CCG transitive verb

[3] We use 'semantic/syntactic type' and 'semantic/syntactic category' interchangeably.

[4] In the full theory, NP is treated as a schema, and type-raised just in time during parsing to an appropriate form as determined by the context. We pass over this detail here and simply treat NP as a category that can combine directly with others.

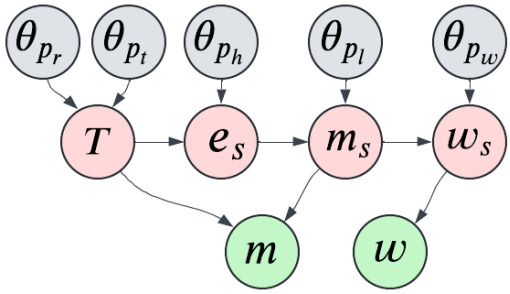

Figure 4: Graphical model for for our probabilistic model. T is the parse tree, e_s, m_s and w_s are the leaf-level shell lfs, lfs and word, m is the root-level lf, w is the utterance and θ_x is the subset of the full set θ of model parameters, consisting of the cooccurence counts in the distribution x, as described in Section 3.1. Green indicates that a variable is observed, and red indicates unobserved. These colours are for train time, at test time, m would also be red. The fact that w is observed but w_s is not reflects the fact that the model sees the full utterance, but not where the word boundaries should be, and similarly with m and m_s.

category $S\backslash NP/NP$ has the semantic type $<e,<e,t>>$. See Steedman (2000) for further details. If, when expanding a parse tree, any node violates these constraints, then that branch of search is terminated.

This constraint is based on the assumption that the child knows the semantic type of an lf (or fragment thereof on some internal parse node). For example, in the derivation of 'you lost a shoe', from Figure 1, the child knows that the constituent S\NP: $\lambda y.lost\ (a\ shoe)\ y$, in the second-to-last row, has semantic type $<e,t>$. These constraints speed up training significantly. and make training more robust by removing the noise of updates from inconsistent parse trees, i.e., those that are not in $\mathscr{T}$. We believe this is the reason for our improved robustness to noise in the lfs over Abend et al. (2017), as described in Section 4.7.

The computation of all allowable parse trees can be performed efficiently by caching the probability of each subtree.

3.3. Worked Example

Here we present a worked example for a single training example. Recall that each training example consists of a string and corresponding logical form, and the learner considers the set $\mathscr{T}$ of all compatible parses, i.e. all parses with the observed lf as root, the observed words as leaves and that obeys the constraints described in Section 3.1. We describe the training updates for a single, correct parse for the example "you can't see the music", as shown in Figure 5. The prediction of the tree proceeds from the root. We will detail the predictions made along the

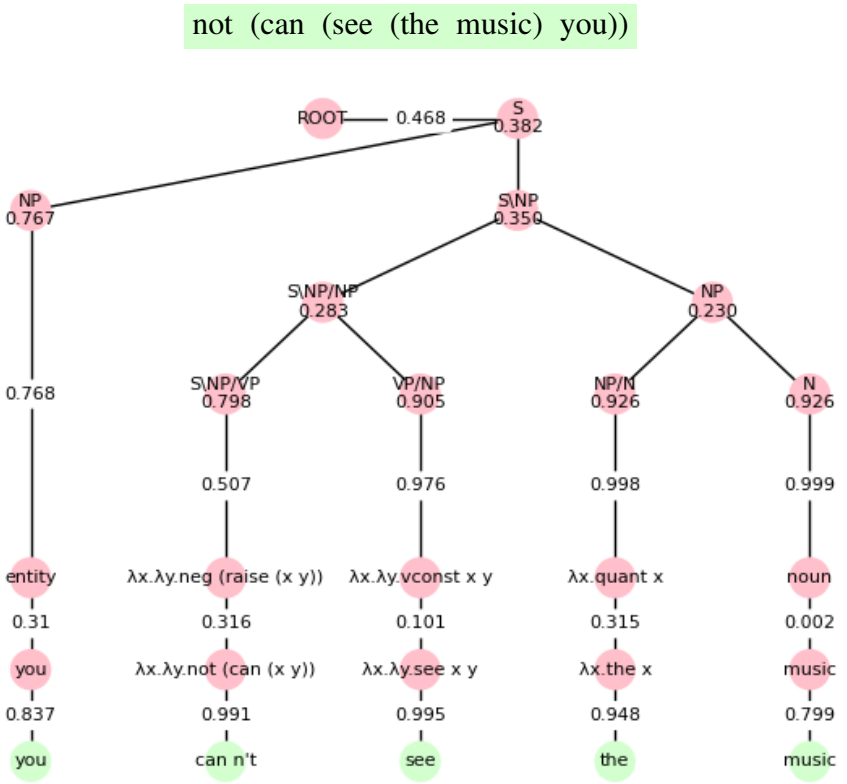

Figure 5: One of the parses considered by the learner for this example. Given information is in green, inferred information is in pink. As this is train time, the model sees both the string and the root lf. Strictly speaking, the model sees only the full utterance rather than individual words because the boundaries between words are not deterministically given. This is reflected in Figure 4.

left-most branch from the root to the leaf 'you', with all other predictions being made similarly. All predictions are made using the Dirichlet processes described in Section 3.1.

First, the learner uses the Dirichlet process, p_r, to predict a possible root category, here S with probability 0.548. Next, it uses the Dirichlet process for syntactic prediction, p_t, to predict a probability for all possible splits of this root syntactic category S. The split shown here is into NP and S\NP, with probability 0.395. Then, for the left child node NP, it again uses p_t to predict a probability for all splits into further child syntactic categories, or alternatively that this node is a leaf. Here, the prediction is that the node is a leaf, which p_t gives probability

0.765. This indicates that a substantial portion of the NP nodes it has observed in the past have been leaf nodes. The bulk of the remaining probability mass is taken up by the split into (NP/N, N).

This ends the syntactic stage of prediction, and the role of p_t. The task now is to predict what meaning and word(s) should correspond to this NP leaf node. To this end, it first uses p_l to predict a probability for all possible shell lfs, the one of which shown here is $<e>$, with probability 0.75. Again, the bulk of the remaining 0.25 probability mass is taken up by the possibility a bigram of determiner plus noun, with shell lf *quant noun*. Interpretations of this sort are discussed in Section 5.2. The fact that the input, or conditioning variable, for p_l is the syntactic leaf only, and not any other information from the tree, is the manifestation of the independence assumption discussed in Section 3.1.

Next, the learner uses p_m to predict probabilities for all likely meanings of the $<e>$, here *you* with probability 0.341. This stage of predicting lf given shell lf generally gives the smallest probability of all predictions on the tree, because there are many different meanings corresponding to a given semantic type. The relatively high value of 0.341 reflects the high frequency of *you* as a meaning in the dataset.

Finally, p_w predicts probabilities for all possible words that could correspond to the meaning *you*. The probability of 0.896 thus represents the learner's belief, at this stage in training that the word 'you' is a realisation of the meaning *you*. The remaining 0.104 is made up of a long tail of other incorrect meanings, arising from various incorrect interpretations of previous training examples. We observe that this figure continues to reduce to about 0.05 by the end of training.

The total probability for this tree and leaves given the root lf is then computed by multiplying all predicted probabilities at all locations on the tree, which gives $\sim 2.736e-20$. Given that we observe the leaves, we condition on this event by diving by the sum of the probabilities of all elements of $\mathscr{T}$. Here, that sum turns out to be $7.079e-20$, so the conditional probability for the tree in Figure 5 is

$$\frac{2.736e-20}{7.079e-7} \approx 0.386\,.$$

Thus, we assume that this tree is 'observed' weighted by this probability, and the counts for the pairings in this tree are updated in the corresponding Dirichlet processes are updated by 0.386.

This same procedure is repeated for every element of $\mathscr{T}$. In practice, for the corpora we use, this is generally $50-100$ trees.

3.4. The Learner as a Parsing Model

After training, it is possible for the learner to parse novel strings to infer their syntax trees and lf. That is, the learner observes only the string, without any corresponding lf. This differs from train time, where it observes both string and lf, and only the parse tree is unobserved. Formally, on observing w, we seek $\text{argmax}_{m,T}\, p(w,m,T|\theta_{final})$, where $p(\cdot)$ is as in (2), and θ_{final} are the model parameters after training. Computing this exactly is intractable, so we approximate using a combination of beam search and a Cocke-Young-Kasami (CYK) based chart-parsing algorithm for CCG. First, we marginalise the Dirichlet distributions p_l and p_h. This is done as follows, using p_l as an example

$$p_l(x) = \frac{\sum_{v_{sh}\in V_{sh}} p_l(x|v_{sh}) \sum_{v_m \in V_m} c_l(v_{sh}, v_m)}{\sum_{v_{sh}\in V_{sh}} \sum_{v_m \in V_m} c_l(v_{sh}, v_m)}, \tag{4}$$

where $c_l(x,y)$ is the raw count from the Dirichlet process of the occurrence of shell meaning x with meaning y. The denominator in (4) thus expresses the number of times any pair of meaning and shell meaning have been observed by the learner.

These marginal distributions then facilitate a beam search to predict a beam of highest probability lf-category pairs for each word span in the string. Recall that the learner also considers interpretations in which multiple words form a single lexical item, so this beam search is run on all contiguous spans in the utterance. Then we continue the beam search into CYK-based chart parsing to predict a CCG syntax tree. The full method for beam search of leaf nodes is specified in Algorithm 1. After this, we run CYK for CCG to predict a parse tree for the entire utterance.

Note that a typical parsing model is given the meaning and possible categories of each word, and then learns to select the correct categories from amongst these possibilities and to form the syntax tree. Our learner, on the other hand, learns the meaning and categories of the leaves from scratch, as well as learning to form the syntax tree. Due to CCG's close coupling of syntax and semantics, the parse tree, along with a meaning for each leaf, then allows us to compute the meaning for the entire utterance. This is used as an evaluation method in Section 4.4 below.

4. Results

4.1. Data

The data we use for training and testing is taken from Brown's 1973 Adam corpus, containing transcribed child-directed speech in North American English

Algorithm 1 Algorithm for parsing unseen utterances to infer the root lf.

$V_m \leftarrow$ vocabulary of all observed lfs
$V_{sh} \leftarrow$ vocabulary of all observed shell-lfs
function SEARCHLEAFSPAN(ws)
  $B \leftarrow []$
  for $v_m \in V_m$ **do**
    $p \leftarrow p_w(ws|v_m)p_m(v_m)$
    append (v_m, p) to B
  end for$B \leftarrow$ top 10 entries in B, ranked by p
  for $(v_m, p) \in B$ **do**
    for $v_t \in V_t$ **do**
      $p' \leftarrow p_m(v_m|v_t)$
      $p_{leaf} \leftarrow p_t(\text{leaf}|v_t)$
      $p \leftarrow pp'p_{leaf}\frac{p_t(v_t)}{p_m(v_m)}$
      append (v_t, v_m, p) to B
    end for
  end for
  $B \leftarrow$ top 10 entries in B, ranked by p
  return B
end function

to a child ranging in age from 2 years 3 months to 3 years 11 months. It consists of 9314 tokens and 5320 utterances, which amounts only about 2% of the data used by the child for language acquisition in the relevant period (Gilkerson et al., 2017). However, the child is simultaneously learning other skills, such as social, perceptual and motor skills, whereas our model isolates the problem of learning syntax and word-level semantics.

The utterances are extended with lfs, specifically lambda calculus expressions, as in Figures 1, 2 and 3. Each training example is then a pair of a string in English, and a corresponding lf. The lfs are produced using the method of Szubert et al. (2024), which first forms a universal dependency (UD) parse of the string and then uses the UDepLambda library `https://github.com/sivareddyg/UDepLambda`, to convert these parses into lfs. The UD parses were automatically checked for correctness using the checker at `https://github.com/UniversalDependencies/tools/`. The tokenization is taken from the CHILDES corpora.

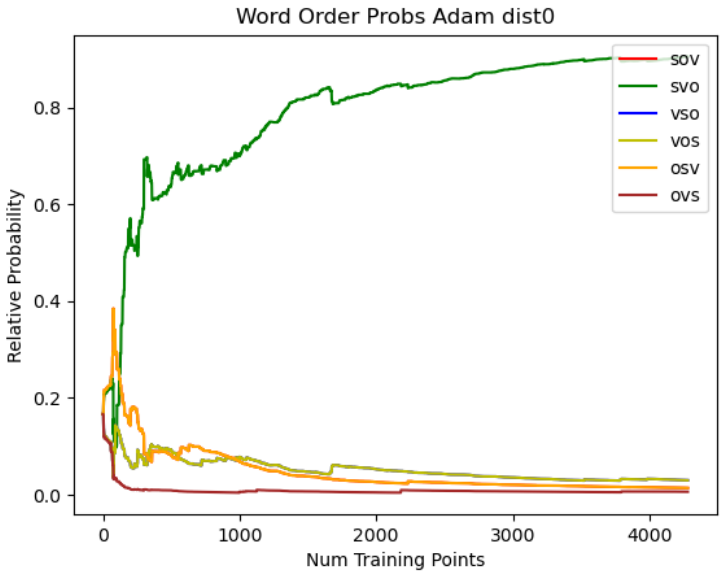

Figure 6: Relative word order probabilities, over the course of training, for each of the six possible word orders for S, V, and O as reflected by verb category. SVO order is learnt confidently within the first 500 examples, and rises to 90% by the end of training.

4.2. Word-order Learning

Prior works on similar models to ours (Abend et al., 2017; Mahon et al., 2024), evaluated the learning of word order by examining the model's internal parameters to calculate the prior probability, that is, the probability before observing any string or lf, of the transitive verb category $S\backslash NP/NP$. Specifically, this is the sum of all parse trees that would yield this category, namely the right-branching derivation consisting of forward application and backward application, as shown in Figure 1, and the left-branching derivation consisting of forward application and forward composition, as shown in Figure 2. Figure 6 shows these relative probability scores over the course of training. Clearly, the prior expectation for SVO order is learnt rapidly and confidently, which reproduces the results for similar models in Abend et al. (2017); Mahon et al. (2024).

4.3. Meaning and Category for Individual Words

We also follow Mahon et al. (2024) and evaluate the learned meanings and syntactic categories for individual words. Using Bayes' rule, we can obtain a

Figure 7: Learning curves for meanings (lfs) and syntactic categories for the 50 most common words in the corpus (see Appendix A.1). Both are learnt successfully, with word meaning higher than word category.

prediction for the meaning and category of a given word. Specifically, the inferred meaning m' for a word w is given by

$$m' \approx \underset{m}{\text{argmax}}\, p_w(w|m)p_w(m) = \underset{m}{\text{argmax}}\, p_w(w,m)\,, \tag{5}$$

where the last quantity is approximated by the observed number of times that w and m co-occur in the Dirichlet process, which is essentially the numerator in the DP For example, for the modal word 'can', the learner should predict that the meaning is the "raising to subject" verb $\lambda p.\lambda y.can\ (p\ y)$[5]. The predicted category is calculated analogously and in this case is S\NP/VP. We do this for the 50 most common words, and manually evaluate whether they are correct (our annotations are shown in Appendix A.1). Unlike Mahon et al. (2024), who report only a single figure for accuracy after training, we evaluate this throughout training, to produce a learning curve, which is shown in Figure 7.

The model learns meanings and categories for these words to a similar degree

[5]This is the 'can' of ability, rather than of possibility.

to Mahon et al. (2024): 90+% for meaning and $\sim$ 70% for category. These are learnt quickly in the first 1000 utterances. The learning of category then plateaus and fails to learn the final 30% of cases. This is due to the learner not having a systematic representation of person, tense and number, which leads to it often confuse the categories S\NP, which is an inflected verb phrase, and VP, which is an infinitival verb phrase. For example, for the word 'are' in the context of expressing identity, such as 'those are yours', meaning *equals yours those*, it predicts the category VP/NP, when it should predict S\NP. Recall from Section 3, that the model predicts the shell lf from the syntactic category, and so, after applying Bayes' rule, the experiments here predict the category from the shell lf, but currently, the shell lf is the same for inflected and infinitival verbs. This is discussed further in Section 5.

4.4. Understanding Full Utterances

Going beyond merely showing the relative, general preference for SVO order, in this work we examine the learner's ability to analyze entire utterances correctly. We do this in two different ways. In the first, called 'select acc' below, we present the model with a single string and a set of 5 lfs, only one of which is correct, and for each of these lfs, measure the estimated probability of observing the string paired with that lf. The incorrect lfs are selected from immediately before and after the utterance as it appears in the corpus, similar to the distractor setting (discussed in Section 4.7), with $n = 4$. We mark an example correct if and only if the probability is highest for the correct lf. In the second, we present only the string, and the model must infer the lf using the method described in Section 3.4, which is marked correct only if it exactly matches the true lf. In this setting, the model may encounter utterances which include words that have not been seen before in training, i.e. words that are new to the child. In Section 4.8, we show that the model can learn meanings of novel words from syntactic knowledge alone, i.e. perform syntactic bootstrapping. However, this test setting presents the model with the string alone, without the lf, so the model has no access to the meaning of the novel word and is prevented from making the correct interpretation of the utterance. One may, therefore, prefer to exclude these utterances from testing. We present results from both settings, one which includes these utterances with unseen words, which are all then scored as incorrect, and one in which they are excluded.

These three different measures of accuracy are computed, throughout the course of training, on the final 10% of utterances in the corpus, which we use as a held-out test set. All three show a steady increase, and have not yet plateaued at the

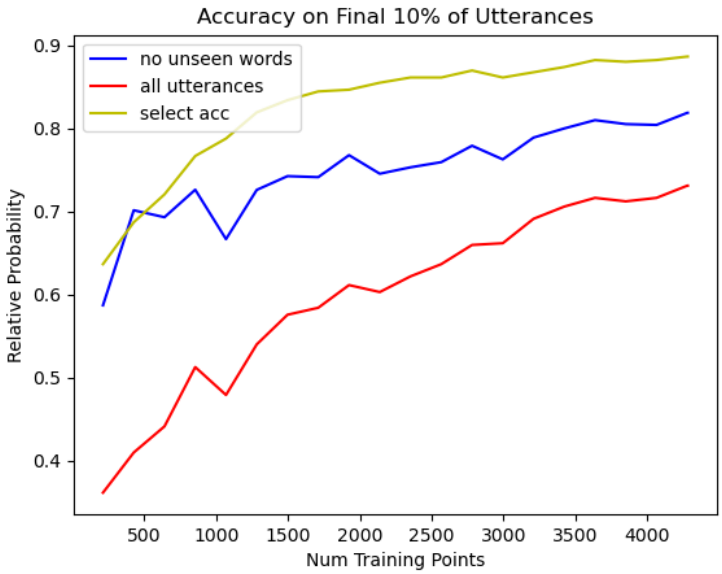

Figure 8: Ability of the learner to infer the meaning of a novel utterance: 'select acc' is the fraction of time it gives the highest probabilty to the correct lf from a set of candidate lfs; 'all utterances' and 'no unseen words' are the fraction of utterances for which the model infers the correct lf from the string alone, measured, respectively, on all utterances in the test set, and all utterances in the test set that do not have previously unseen words. The model accuracy improves steadily and reaches a high final level by all three measures.

end of training, suggesting they would continue to improve if given more training data. The red line, which shows the accuracy on all test items, including those with unseen words, is of course always lower than the blue line, where these test items have been removed. The yellow line is generally the highest, reaching 88% by the end of training, though at least one point during training, it is exceeded by the accuracy with novel words excluded (blue line).

The test set for the blue line is changing slightly over training, specifically the number of points being excluded is decreasing as the set of words seen by the model increases. Therefore we suggest the final point reached by the blue line ($\sim$ 80%) is more revealing of the ability of the model after training, but the trajectory of the red line is more revealing of the course of learning.

Note that utterances with unseen words can be, and indeed often are, correct by the first evaluation method (green line). By the end of training, there are still 53 utterances, out of 476 in the test set, with novel words. The yellow line has a

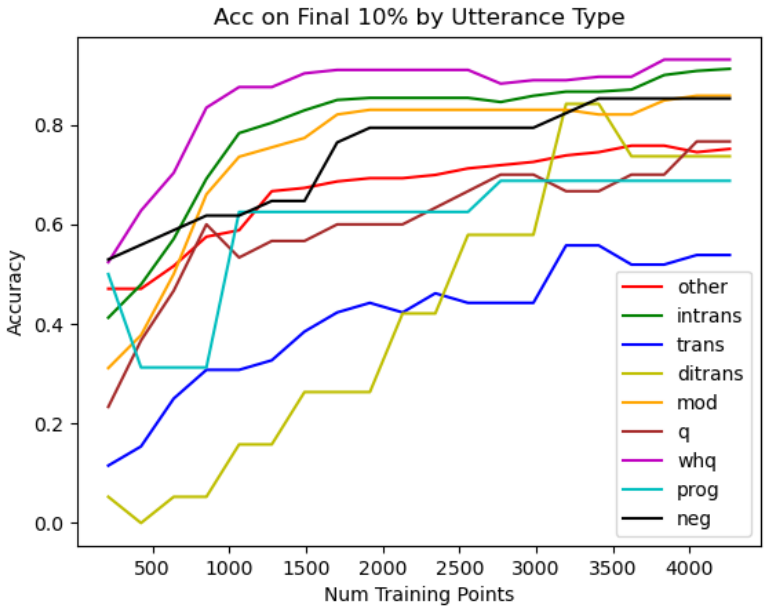

Figure 9: Breakdown of inferred meaning accuracy by different construction types, using the 'all utterances' measure. This shows the model is able to infer correct root meanings for a variety of construction types, and the high average accuracy from Figure 8 is not just the result of a few construction types.

final error rate of only 11.5%, and gets the correct answer for 27 out of these 53 utterances with novel words. This shows that the model can still make reasonable interpretations even in the presence of a novel word: if it has some rough idea of what the meaning for the entire utterance might be, it is still often able to deduce the correct analysis. This property of the model is examined further in Section 4.8.

4.5. Accuracy by Construction Type

Figure 9 shows the accuracy of the utterance meanings, separated by the construction type of the utterance. This includes all utterances, even those with novel words. Utterances that exhibit multiple listed features are counted in all the corresponding categories. For example, a negated modal like "you can't see it", is counted under 'modal' and 'neg'. The measure of accuracy used is the accuracy of the inferred root lfs when presented with an unseen string, i.e., the red line from Figure 8. This accuracy increases steadily through training for all utterance types.

This shows that the high average accuracy from Figure 8 is not restricted to just a few types of syntactic construction. Rather, our model learns to infer the correct parse with high accuracy for a variety of syntactic constructions.

The curves in Figure 9 are not inter-comparable, and in particular should not be taken as indicating order of acquisition, because the groups are quite different in size and diversity. For example, transitives ('trans', dark blue line) show the lowest final accuracy, but this reflects the fact that the set of simple transitive sentences in our data is largest and most diverse of those presented. Similarly, the fact that wh-questions ('whq', magenta line), show one of the highest accuracies is largely due to that construction type being less frequent and occupied to a greater extent by a few commonly occurring examples, such as 'what are you doing?' and 'what's that?'. There are still several examples of less common wh-questions in our test set, and we show the full predicted analyses for some of these in Section 4.9.

Note that this relationship between low variability and higher accuracy refers to variability of the test items, and is not counter to the evidence that high input variability improves language learning in children (Huttenlocher et al. (2010) and references therein), which refers to variability in the train items. The low variability in wh-questions in the dataset means the model is only tested on a small set of utterances, whereas for transitives, there is a much higher diversity of utterances it is tested on.

4.6. Accuracy on Whq Words' Categories

Because the ability to model LRDs of the sort found in object wh-questions is one of the contributions of our model, we present a further experiment focussing specifically on the accuracy for whq words. Figure 9 already showed that the accuracy in inferring root lfs for novel utterances is high for wh questions. However, as we noted in the preceding section, that result does not necessarily reflect a high accuracy in the predicted syntactic categories for wh questions, because it is possible for the model to choose an incorrect or at least non-standard syntactic analysis which nevertheless produces the correct lf. Some examples involving lexicalization of multi-world expressions are discussed further in Sections 4.9 and 5.2. Here, we show the accuracy for the syntactic category of the wh-word as it appears in fronted wh-questions in our test set, relative to ground truth categories that we annotated manually. We do not report separate scores for subject and object categories because the nature of the CHILDES data is that there are too few subject questions for the model to learn them effectively.

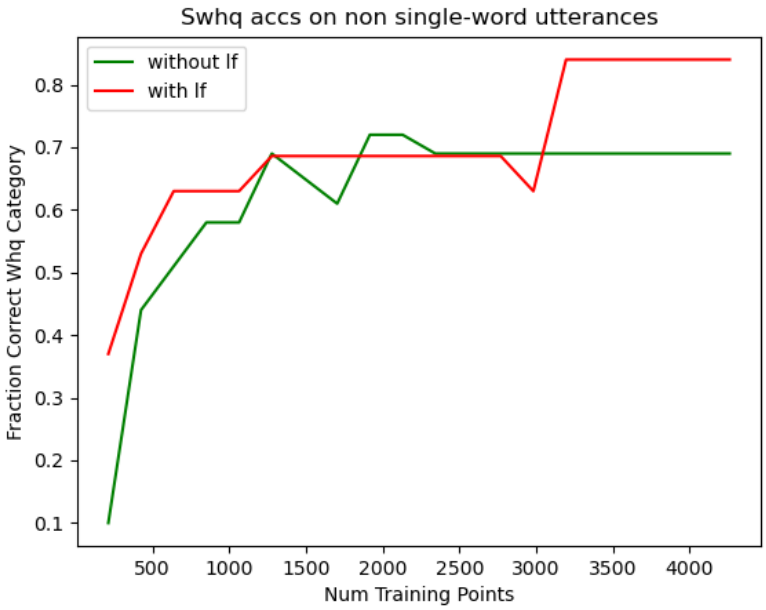

Figure 10: Accuracy of the assigned syntactic categories for whq words in our test set, relative to ground truth categories that we annotated manually. The red line corresponds to the traintime setting where the learner sees both the string and the lf, the green line to the testtime setting where it sees only the string. In both settings, the model is scored as correct whenever its favoured analysis contains a leaf with the correct whq word, meaning and category, and incorrect otherwise. The model reaches a high score by both measures, which shows that it is able to correctly assign syntactic categories to wh words at the leaf level.

The learning curves are shown in Figure 10, for the settings where the lf is seen (red line) and unseen (green line) settings. For both settings, the accuracy increases through training and reaches a high accuracy at the end of training (about 85% with the lf and 70% without). Although we do not distinguish between subject and object questions, we note that the set of wh-questions in the dataset consists almost entirely of object questions. This result is broadly consistent with observations of relatively early acquisition of object questions in children (Stromswold, 1995; De Villiers et al., 1990; Klima and Bellugi, 1966). It shows that the learner not only learns to infer the correct root lf for long-range dependencies of the sort found in object wh-questions, but also learns to model the syntax of these utterances by giving the correct syntactic category to the wh-word, and therefore also to the rest of the sentence.

4.7. Distractor Settings

Our training setting, as described in Section 3.2, presents the learner with a single lf for each utterance, i.e., it is told the single correct meaning for the corresponding string. In the case of human learning, however, it is more realistic to assume that the child apprehends several possible meanings when it hears an string, and does not know, *a priori*, which of these possible meanings the utterance expresses. To simulate this uncertainty, we repeat the experiments from Section 4.2 and 4.4 with varying numbers of 'distractor' lfs presented alongside the true lf. The learner is then free to consider any of these lfs as the meaning of the utterance. When there is a single tree that the model is very confident in, then the probability from this tree dominates anyway, and overall there is little effect from the distractor trees. However, when there is no such single confident interpretation, the distractor trees significantly reduce the probability on the trees from the correct lf, including the correct tree, and add probability, and hence parameter updates, corresponding to the incorrect trees from the distractor lfs. If the learning trajectory is not stable, the updates from these incorrect trees can derail the model.

In the real child learner, the distractor logical forms presumably originate in the child's perception and understanding of the state of the world and the conversation, which our model does not directly represent. In our experiment, we take as a proxy for such distractors, the logical forms from the utterances immediately following and preceding the given utterance. Specifically, the n distractor setting takes the $\lfloor n/2 \rfloor$ previous examples and the $\lceil n/2 \rceil$ following examples.

For example, in Adam, training examples 226-228 are as follows:

Data point 226: "you blow it"–*blow you it*

Data point 227: "you can blow"–*can* (*blow you*)

Data point 228: "you do it"–*do you it*

Thus, in the two distractor setting, when training on training example 227, we include the parse trees from all three of these lfs. In this case, one possible interpretation takes the lf from training example 226–blow(you,it) *blow it you*– and interprets "you" as meaning *you*, "can" as meaning *it*, "blow" as meaning $\lambda x.\lambda y.blow\ x\ y$, and the sentence as being in SOV order.

As shown in Figure 11, the addition of distractors slows down learning, but the shape of the trajectories remains unchanged. This robustness represents an improvement on the model of Abend et al. (2017), which exhibited some instability with respect to the number of distractors. Figure 12 shows the same stability for obtaining the correct meaning representation shown for the zero-distractor case in Figure 7. Here, the robustness to the distractor lfs is even more striking, showing

only a very marginal difference even when 8 distractor lfs are added. This suggests that, with more training data, our model would reach the same performance as for the zero-distractor setting.

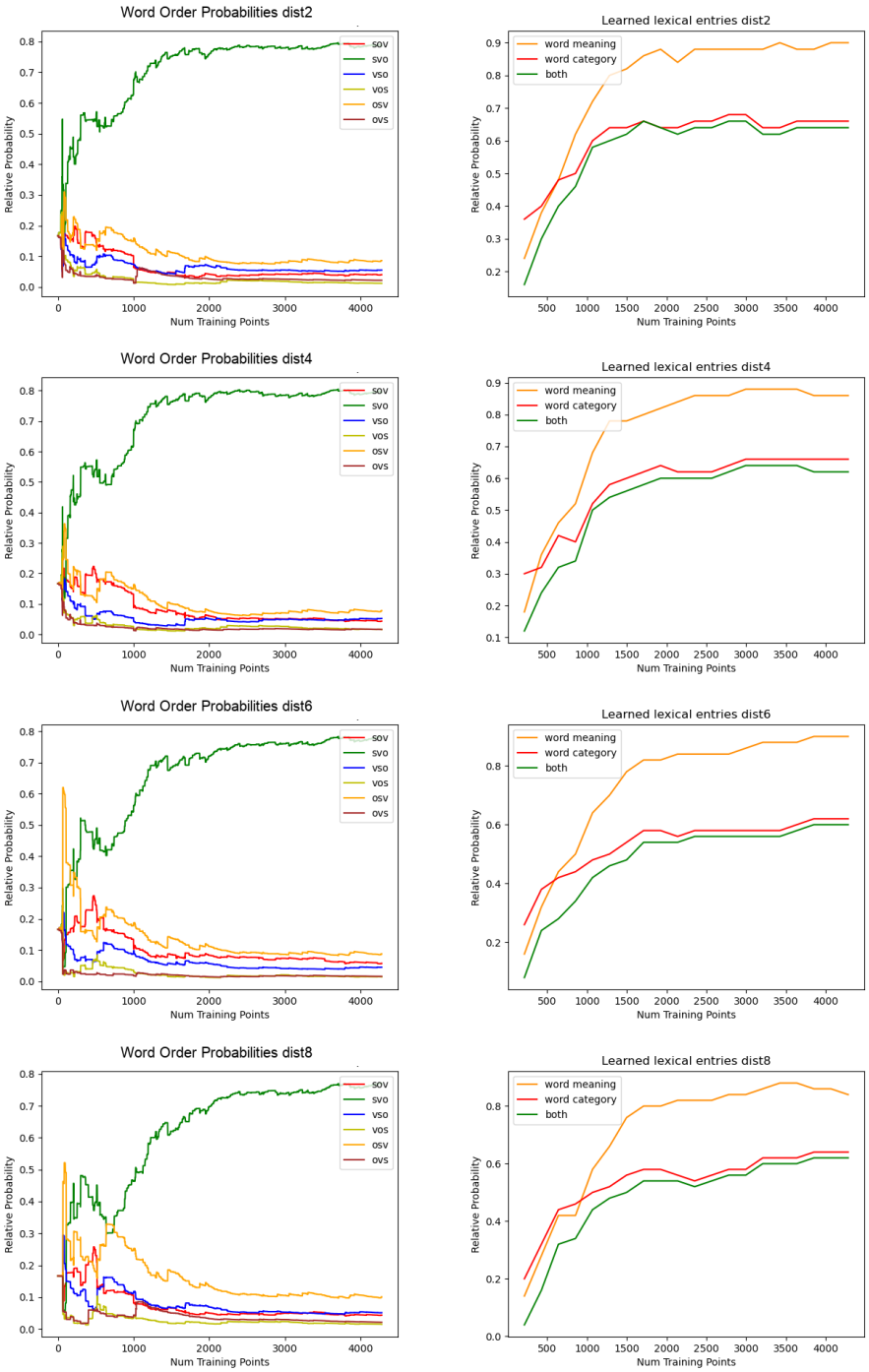

Figure 11: Repeats of experiments for word order of S, V, and O as reflected by transitive category, and word meaning/category learning, with increasing numbers of distractor lf: 2 in the top row, 4 in the second row, 6 in the third row and 8 in the fourth row. Cf. Figure 6. In this plot, and throughout the paper, a plot title ending in 'distN' indicates that there were N distractors present. This shows the learning of word order and word meanings and categories is robust to the presence of distractor meanings during training.

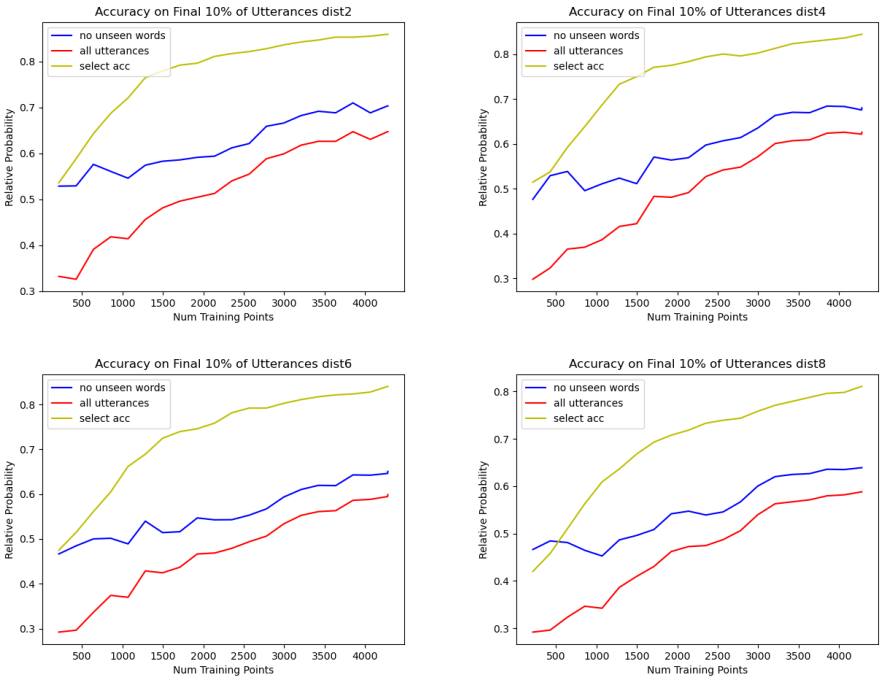

Figure 12: Repeats of experiments for predicting the correct utterance meaning at test time, with increasing numbers of distractor lfs: 2 in the top left, 4 in the top right, 6 in the bottom left and 8 in the bottom right. Cf. Figure 8. In this plot, and throughout the paper, a plot title ending in 'distN' indicates that there were N distractors present. This shows that acquiring the ability to infer the meaning of whole utterances is robust to the presence of distractor meanings during training.

4.8. One-trial Learning of Nonce Words

In this Section, we test the ability of our model for one-trial learning in a variety of syntactic contexts, that is, learning the meaning of novel words from a single exposure.

Abend et al. (2017) showed this in the case of transitive sentences. When exposed to a transitive sentence containing a nonce word 'dax', along with two possible lfs, one in which 'dax' means $\lambda x.\lambda y.dax\ x\ y$, and one in which it means $\lambda x.\lambda y.dax\ y\ x$, their model then showed a marked rise in its predicted probability that the meaning $\lambda x.\lambda y.dax\ x\ y$ is realized as the word 'dax'. The most significant advance in our model over that of Abend et al. is its ability to handle a much wider set of syntactic constructions, and we now show that this allows our model to achieve one-trial learning over this wider set.

Figure 13 shows the results of the same one-trial learning test not just for transitive sentences, but also for other more complex constructions that the child is exposed to. For each utterance type, the learner is presented two lfs that differ only in who they designate as the agent of the transitive verb, and only one, the intuitively correct one, agrees with the SVO verb category S\NP/NP. We follow Abend et al. in using two unseen names, 'Jacob' and 'Jacky' as subject and object, and in running two versions of the experiment, one with four distractor lfs and one with six. In the wh-question context, the string is "who will Jacob dax?", and one lf expresses an object wh-question, while the other expresses a subject wh-question.

Our learner, after training, is capable of one-trial learning in the context of all of these constructions. For questions, negations and modals, this measure of one-trial learning ability rises rapidly within the first 800 training examples. Such constructions contain some familiar words, namely the wh-word, the modal and the negation 'not', so in this sense they are easier than the transitives, which contain only novel words. For the progressive 'Jacob is daxing Jacky', the rise occurs later, slightly after training example 2000. The only familiar word there is the copula, which is in general a difficult word for the learner to analyse correctly because it appears frequently in a variety of different functional roles. The curve for transitives corresponds to curves presented in Abend et al., and we can see that ours rise higher and more smoothly.

This ability is the result of the model having enough language-specific syntax that, even if it encounters a new word, which must, in the first encounter, automatically get a low probability of having the correct meaning, the correct analysis has sufficiently high probability from the rest of the derivation to ensure the total

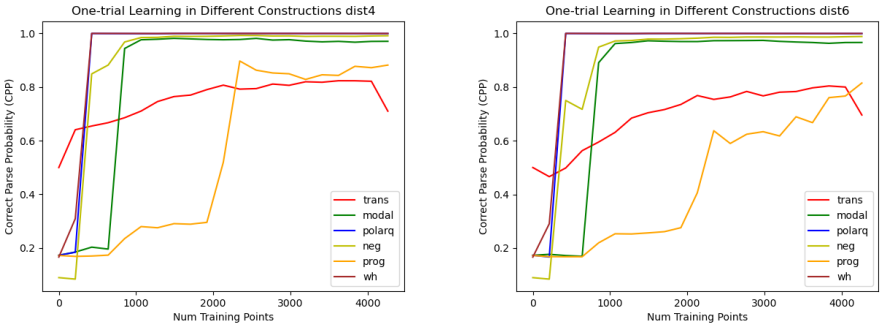

Figure 13: Evolution of the ability to learn from one-trial in the context of actor ambiguity. The y-axis shows the model's estimated probability, after the single exposure, that the word "dax" is assigned the syntactic type S \NP/NP and the logical form $\lambda\ x.\lambda\ y.dax\ x\ y$. In this plot, and throughout the paper, a plot title ending in 'distN' indicates that there were N distractors present. This shows that the model is able to acquire new word meanings from syntactic knowledge alone for a variety of construction types.

tree probability is still high. This leads to a high update weight for the novel word being aligned with its correct meaning.

The analysis in the earlier Figure 5 depicts an instance of this, of how already acquired lexical and syntactic knowledge can facilitate rapid acquisition of a new lexical item. There, the word "music" has never been observed before, and so the corresponding nodes of the tree have very low probability. However, by that stage, the learner is confident in the analysis of the rest of the sentence, so it still gives high probability to the depicted analysis. This high probability means that the co-occurrence counts for *music* and "music" get a large update, leading to a large increase in the estimated probability that the former is the meaning of the latter. The difference with the experiments in this section is that the subject and object are also novel words, so the model must rely entirely on syntactic knowledge to determine how to relate the words to the components of the lf.

The ability measured in Figure 13 is distinct from that of inferring the correct meaning for the utterance, as measured in Figure 9. For example, if a verb and argument have been observed together several times, the learner may interpret the whole verb phrase as a single lexical item (an example of this is in Figure 16b). This could give the correct meaning for the verb phrase, and hence the utterance, but, if such an analysis is made by the learner, it may not facilitate one-trial learning of a novel verb, because there it would not include a leaf node that contains just the novel verb word. For example, given the string "he is daxing", it

could analyze "is daxing" as a single lexical item, at the expense of the analysis in which "daxing" is a leaf. So, although it might acquire the meaning for this entire VP in one trial, it would not do the same for the word "daxing" itself.

Conversely, the learner may give the highest probability to an analysis that gives the *incorrect* root lf meaning, e.g. by interpreting "is daxing" as a single item meaning $\lambda x.\lambda y.see_{prog}\ x\ y$, while also having reasonably high probability on the correct analysis in which "daxing" is a root that means $\lambda x.\lambda y.dax_{prog}\ x\ y$. As "daxing" is not a leaf in the first analysis, its meaning distribution does not get any update, either correct or incorrect. Of the analyses in which it does get such an update, the correct meaning update may still dominate the probability mass, resulting in the model placing very high posterior probability on "daxing" having the correct meaning. In this case, the model would succeed at the one-trial learning test as measured in Figure 13, but fail at the test of inferring utterance meaning, as measured in Figure 9.

Figure 9 measures whether the single highest probability parse is correct, while 13 measures what fraction of the probability mass of the analyses in which the novel word is a leaf give it the correct meaning. The two measures give different, complementary views into the model's learning trajectory across construction types.

4.9. Qualitative Results

Figures 14, 15 and 16 show some examples from the final 10% of utterances, which we use as a held-out test set. These examples come from presenting the model with the string only, and having it infer the parse tree and meaning, i.e., in Figure 8, it corresponds to the red and blue lines, rather than the yellow line.

We select examples that cover a range of the important constructions that our model is able to handle. Because of our special focus on LRDs, we show three object wh-questions, one in progressive aspect.

In all of these examples, the learner infers a parse tree that will derive the correct root meaning. Some, such as the wh-question,"what does that say?", in Figure 15b, and the negated polar question in Figure 17a, exhibit the standard, correct CCG parse trees. Others, such as Figure 14a and 16a, are still textbook-correct, though non-standard in the sense that they use composition when a purely applicative derivation is available. For the other two examples, the model interprets two orthographic words as a single lexical item, e.g. "d you", in the transcription of Figure 15a, is interpreted as a single item with category S/VP and meaning $\lambda x.Q\ (do\ (x\ you))$. We observe that this often occurs for frequent bigrams. These examples still all end up with the correct root meaning for the whole utterance,

so are all counted correct by the measure of Section 4.4. The tendency to lexicalize common multi-world expressions is discussed further in Section 5. Note that, in order to correctly analyze the wh-questions, the model has to select the correct question-form of the auxiliary "do", and the correct object-wh category for the wh-word "what", rather than the subject-wh category S_{whq}/VP or the in-situ category NP.

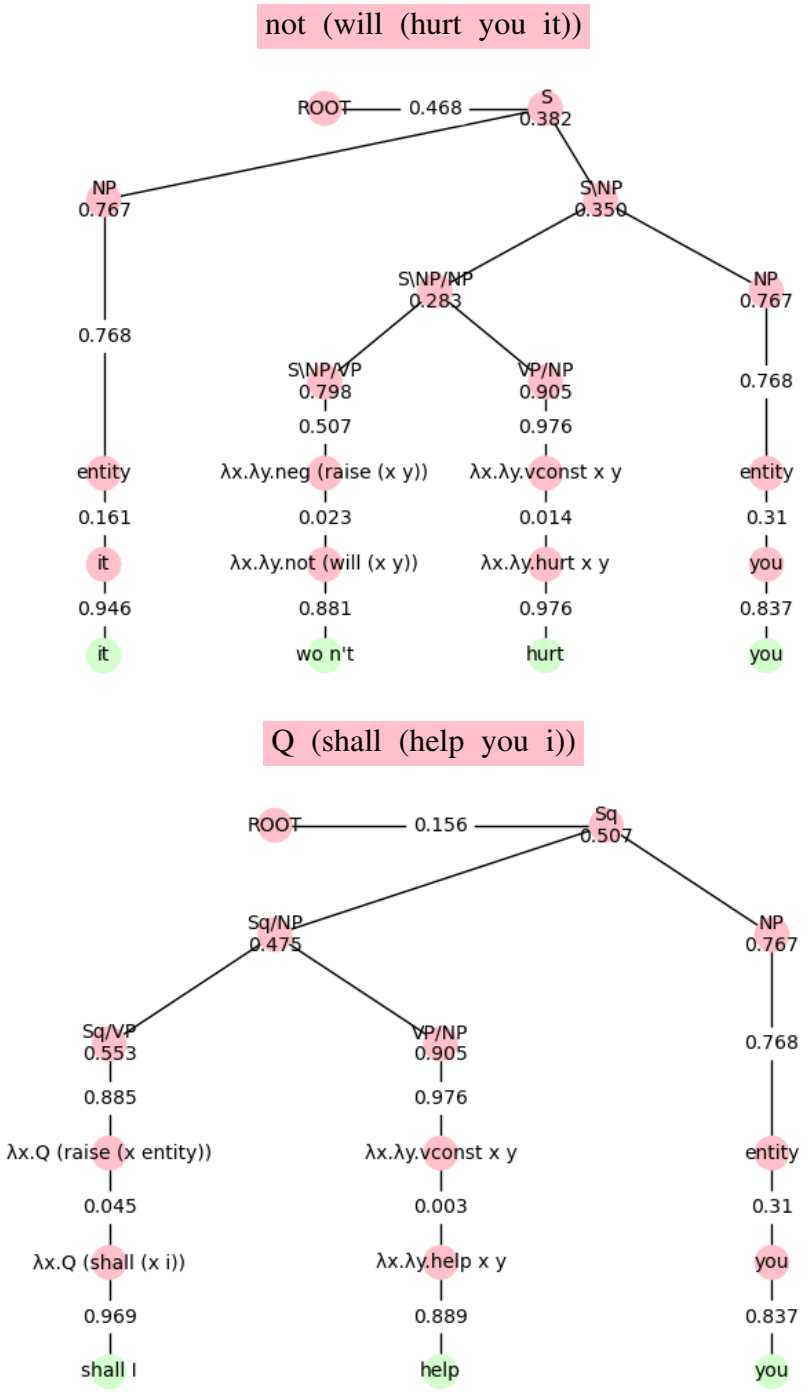

Figure 14: Examples of inferred parse trees for a negated (top) and an interrogative (bottom) modal utterance from our test set. The lf shown above the trees are those inferred by the parse of the learner. Given information is in green, inferred information is in pink. As this is test time, the model sees only the string and must infer the root lf.

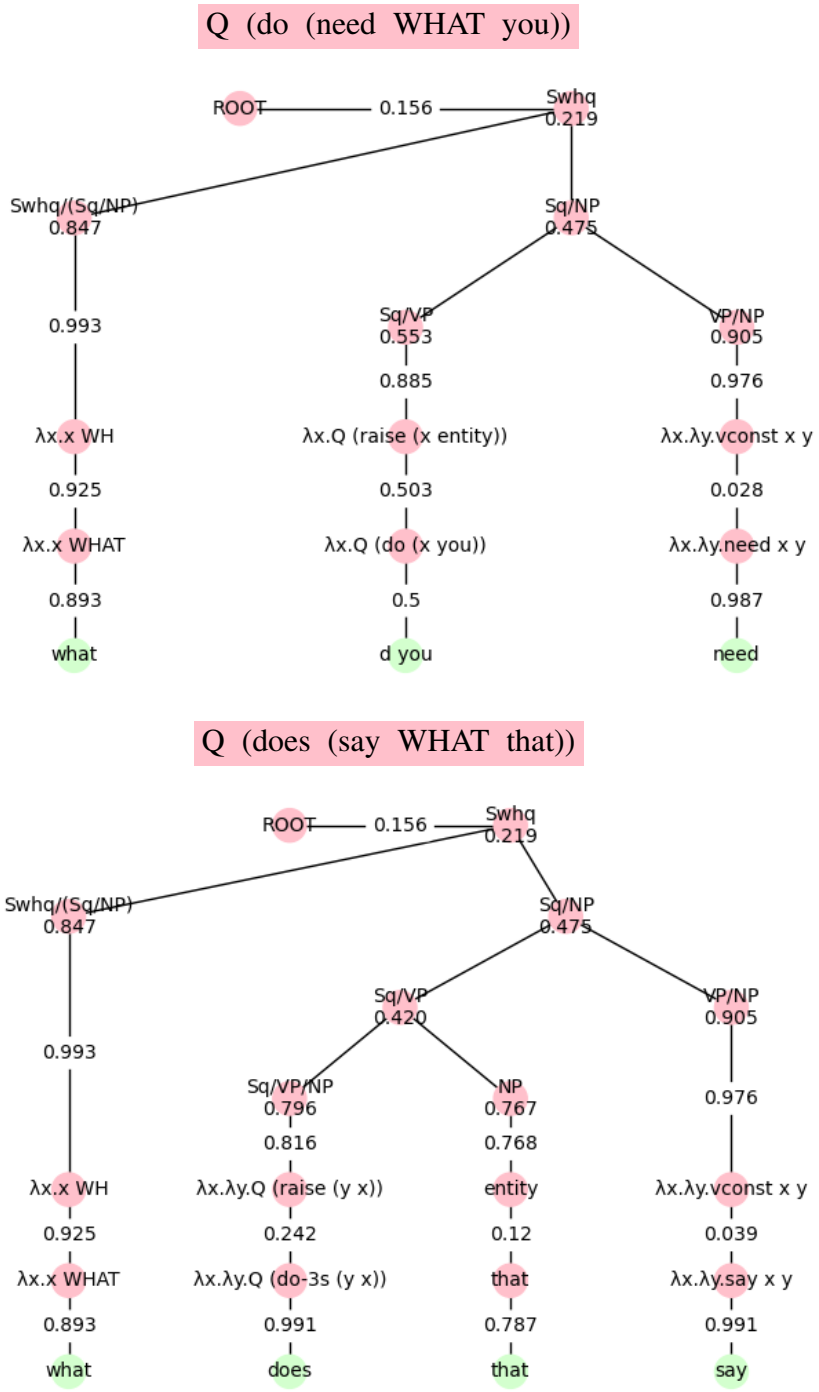

Figure 15: Examples of inferred parse trees for two object wh-questions from our test set. In the full theory, "that" would be raised as S/VP\(S/VP/NP). The lf shown above the trees are those inferred by the parse of the learner. Given information is in green, inferred information is in pink.

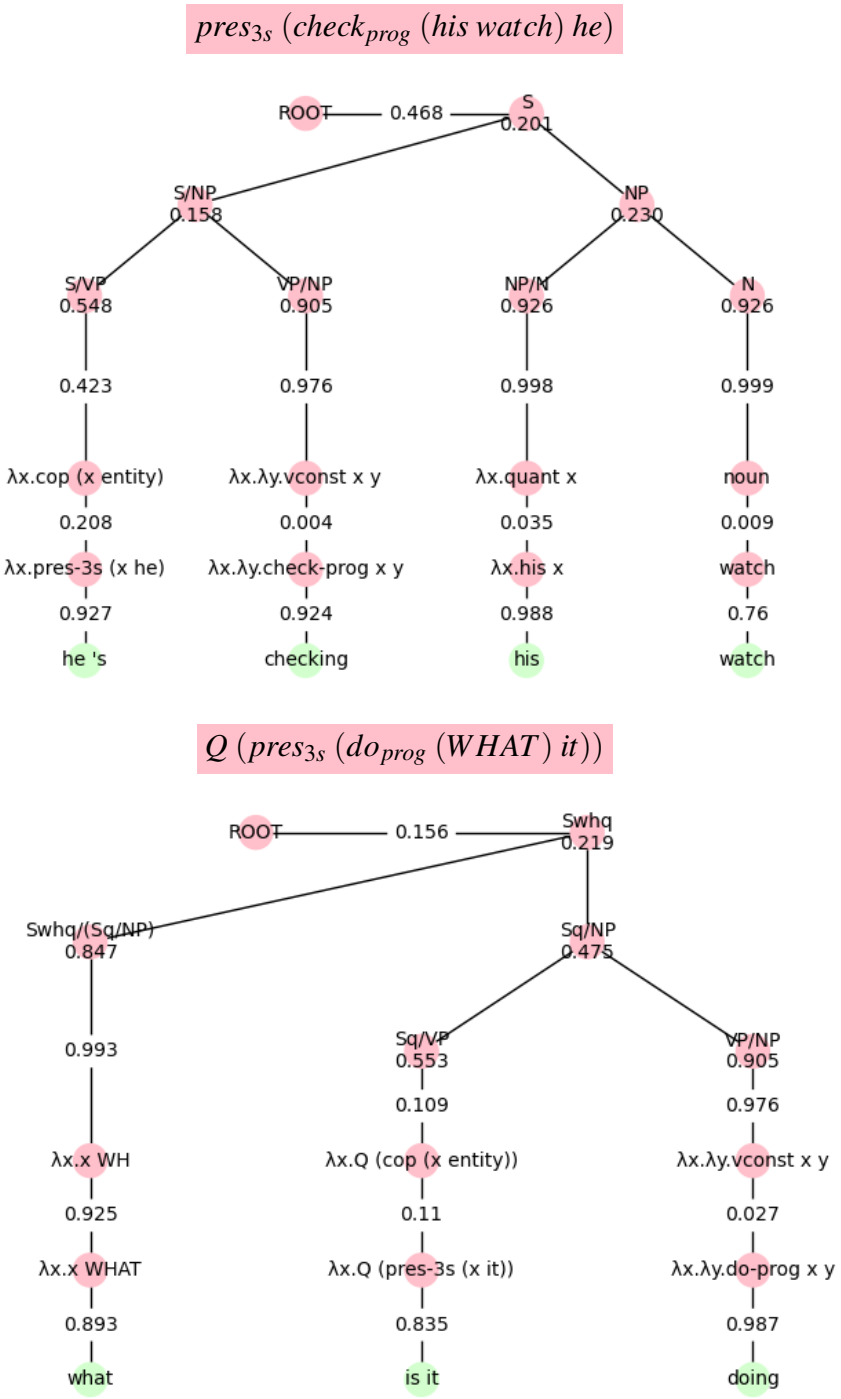

Figure 16: Example of inferred parse trees for progressives from our test set: one declarative (top) and one wh-question (bottom). The lf shown above the trees are those inferred by the parse of the learner. Given information is in green, inferred information is in pink.

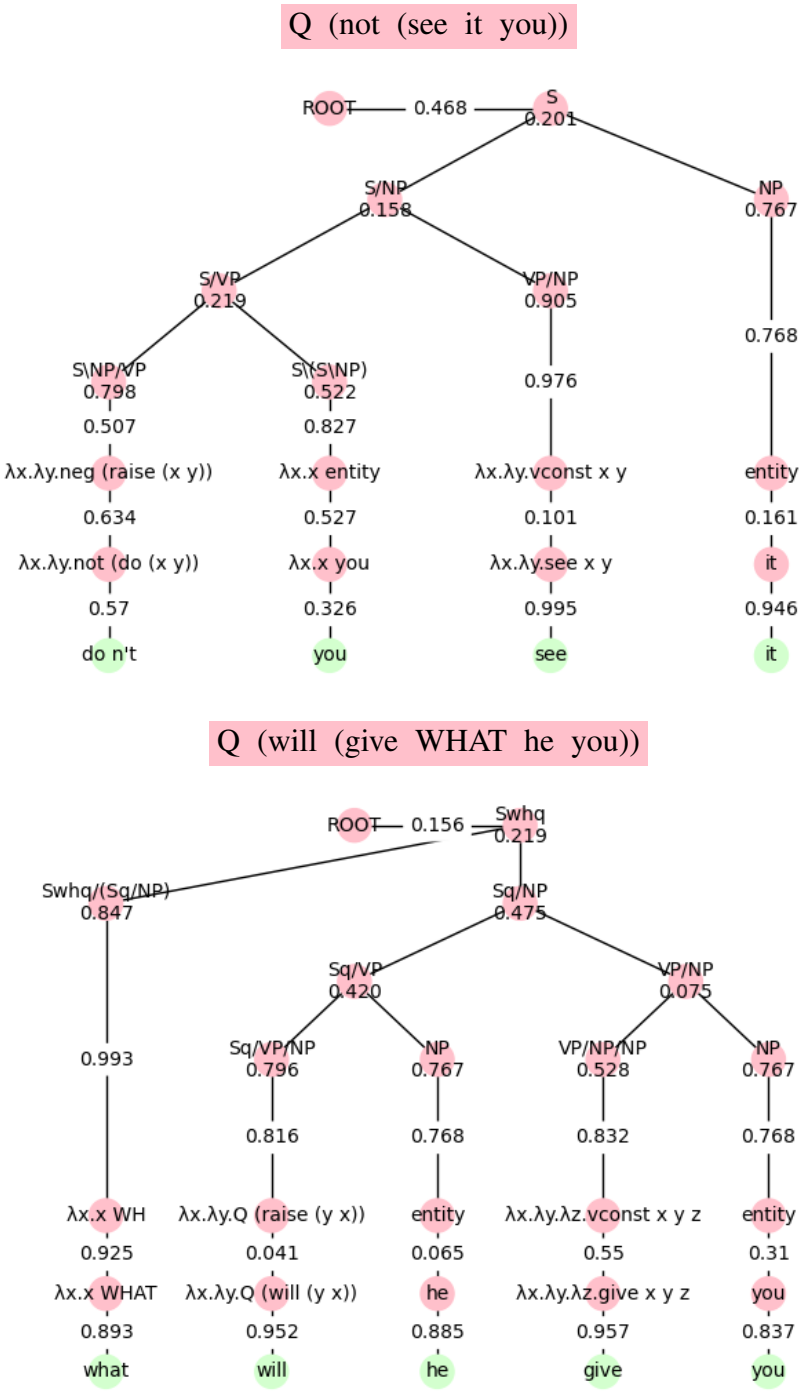

Figure 17: Example of an inferred parse tree for a polar negated question from our test set. Given information is in green, inferred information is in pink.

4.10. Summary of Empirical Improvements over Previous Models

Compared to the two most similar existing models, Abend et al. (2017) and Mahon et al. (2024), there are three main improvements offered by our model.

1. It can handle a wider variety of syntactic constructions (Sections 4.5 and 4.8), including long-range dependencies as found in object wh-questions (Section 4.6);
2. It is the first to present evidence of fully correct parses for unseen strings without corresponding lfs (Section 4.9);
3. It shows higher accuracy in inferring sentence meanings, and greater robustness to the inclusion of distractor lfs during training (Sections 4.4 and 4.7).

5. Discussion

5.1. Long-range Dependencies

It is clear from both the breakdown of accuracy by construction type, as shown in Figure 9, and from the qualitative examples in Section 4.9, that the model largely succeeds in learning the long-range dependencies in object wh-questions. In fact, in Figure 9, the learning curve for wh-questions (pink line) is, for most of training, the highest of all construction types. However, this precocity is due to the fact that there are a couple of frequent wh question utterances, such as "what are you doing?" and "what's that?", that the model learns to memorize very early, which accounts for the early jump. By 'memorize', we mean that the entire utterance is modelled as a single word, as discussed in detail in Section 5.2. The subsequent gradual rise of the pink line is then caused by the model learning the general form of wh-questions and getting more of the long tail correct. By the end of training, it can correctly analyse the large majority of novel wh-questions, producing the textbook CCG parse tree as well and meaning, even including some ditransitives, such as "what will he give you?", in Figure 17b.

5.2. Lexicalization of Common Ngrams

A common behaviour of our learner is to treat multiple orthographic words as a single lexical item, e.g. "is it" in Figure 16b. It is important to allow this interpretation, rather than tell the learner explicitly where the word boundaries are, because we assume that, while learned phonotactic constraints are able to identify possible word boundaries, they are not able to determine them exactly, and so the child, when learning syntax and semantics, must also be considering

such interpretations. In our data, the possible boundaries are those provided by the UD tokenizer, as used in Szubert et al. (2024). This amounts to potential boundaries at all spaces and around clitics.

As our model is entirely probabilistic, there is no hard line between being in or out of the lexicon: every n-gram that was observed anywhere during training has some probability of being the word for the corresponding lf, but for most n-grams, this probability is negligible, and it will never appear in the single maximum a posteriori (MAP) analysis. In this discussion, we use the term 'lexicalisation' to refer to the case where an ngram appears with significant frequency in the MAP analysis.

In the extreme case, the learner might ignore all potential boundaries and treat the entire string as a single word. Indeed, the following simple probabilistic analysis shows that, in our framework, this is the null hypothesis in that, prior to seeing any training data, it is the favoured analysis of all utterances. The probability of a parse tree is the product of the probabilities of all of the constituent nodes given their parents. Prior to seeing any training data, the probability of a category being a leaf is the same for all categories, and is strictly less than 1 (because some probability must be reserved for other possible splits of that category). Thus, the probability of the parse tree is minimized when there is just a single leaf. However, as training progresses, the model favours larger and larger parse trees and eventually, in many cases, reaches a stable interpretation comprising the standard CCG parse tree. The moment when this point is reached is the moment at which one-trial learning, as in Section 13, becomes possible.

Mostly, these cases correspond to breaking at all word boundaries, but there are some exceptions. The negation contraction "n't" is almost always analysed as a single lexical item together with the auxiliary, as in Figures 14 and 17a, even though there is a potential word boundary between them. This in fact agrees with standard linguistic assumptions (Bybee, 2002), and there is strong evidence that negated auxiliary contractions are single items in adult lexicons, as they can be inverted, while the un-contracted bigram cannot: *don't you see it* vs **do not you see it*. Many common bigrams that are lexicalised by our model agree with contractions in adult speech, e.g. "d'you" in Figure 15a and "he's" in Figure 16a[6], but there are also several that do not: "'is it" in Figure 16b and in Figure 17b. For

[6]Indeed, one could make a case either for or against including a potential boundary for words that were transcribed as clitics. The reason we do is simply that the universal dependency parser used by Szubert et al. (2024), and hence the data we use, does so.

some, such as “that’s right” and “I don’t know”, the MAP analysis remains as a one-word utterance even at the end of training. This is also consistent with adult spoken contractions transcribed as “s’right” and “I‘d‘no”.

Discussions of lexicalisation have identified several aspects to the process (Bauer, 1983), e.g. prosodic lexicalization (the effect on the phonetic realization of the segment of the utterance), morphological lexicalisation, (the characterisation of irregular inflected forms as being lexicalised), semantic lexicalization (a.k.a. idiomatization (Lipka, 1977)), and the effect of frequency on lexicalisation (Langacker, 1988; Lieven et al., 2003; Bybee, 2006; Bannard and Matthews, 2008).

Of the possible causes of lexicalisation, the only one that our model responds to is frequency. If the frequency of an ngram is large enough that the probability of it being paired with its corresponding meaning is greater than the product of the probabilities of each its constituents being paired with their corresponding meanings, then it will be lexicalised, in the sense outlined above. The closest analogue in human lexicalisation, would be to consider as lexicalised so-called conventionalised colocations, a.k.a “prefabs” (Erman and Warren, 2000)–that is, ngrams that are not idiomatic but appear unusually frequently, such as ‘ulterior motive’. Such a picture has been suggested by Bybee (2006), Erman and Warren (2000) and Bybee (1985). Note that what counts, at least in the case of our model, is not the raw occurrence frequency of the ngrams in the corpus, but rather the frequency of the ngrams in the estimated parse trees. This difference means that ngrams that cohere with the rest of the sentence into a probable parse tree count for more than those that do not. These observations may also support a lexical analysis of processes of cliticization.

The tendency of our model to lexicalise certain ngrams suggests that, from a probabilistic model of syntax and semantics alone, there is a signal to do so. However, without the other components such as phonetics, the choice for such lexicalizations may differ somewhat from those evidenced in humans.

5.3. Modeling Morphology and Phonology

The potential future extensions to include morphology were also discussed in the model of Mahon et al. (2024), who outlined two possible approaches to including morphology: either to extend the CCG parse trees down to the level of morphemes, or else replace the Dirichlet process for predicting word form given meaning with a neural model. We explored the former idea in preliminary experiments and found it not to work well. For example, we tried inserting a potential word boundary between the suffix ‘-ing’ and the root, in the idea that it

model could learn '-ing' had the category VP\(S\NP) (for transitive sentences) and meaning $\lambda p.\lambda x.prog(p\ x)$. However, it always chose to interpret the stem and the suffix together as a single lexical item. The second idea, of using a neural predictor of word form, could be more promising, as it may allow the model to learn some systematic relationship between meaning and word form without having to specify something as precise as that the orthographic suffix '-ing' always indicates progressive aspect. This neural predictor could operate on the IPA transcriptions instead of the orthographic ones , or even on the speech waveform itself, both of which are available in the CHILDES corpus we use. This would be to take a position that the syntax-semantics interface can be learnt in part by a symbolic system (namely, the one we present in the present paper), but that morphology is more suited to a connectionist model, which is consistent with the success of finite-state transducers in morphological analysis (Kay, 1987). In Section 3.1, we noted that such a mechanism is expected to be needed, in the form of the probabilistic supertagger, for the resolution of lexical ambiguity as the grammar grow towards adult size, and this would constitute a parallel "thinking fast" model component to the symbolic "thinking slow" symbolic grammar (Kahneman, 2017; Ferreira, 2007). Stanojević et al. (2023) offer neuropsychological evidence for the involvement of such a hybrid symbolic-neurocomputational mechanism in human sentence processing using a fully incremental parsing algorithm combined with a supertagger. Providing such a morphological analyser will be a necessary first step in demonstrating the universality of our syntactic learner by applying it to the similarly annotated Hagar corpus of Hebrew child-directed utterance described by Szubert et al..

6. Conclusion

This paper presented a computational model for child language acquisition of syntax and word-level semantics, trained on transcribed child-directed speech paired with manually annotated logical forms as meaning representations. Our model works with several orders of magnitude less data than even the most sample-efficient transformer-based approaches to modelling human-like learning of language (Warstadt et al., 2023b). The main advances of our model over previous similar ones lie in increased robustness and stability in learning, the extension to a wider range of constructions, and the ability to infer meanings and parse trees for unseen child-directed utterance from the held-out final sample of the corpus. We replicated the experiments of previous similar models regarding learning word order and word meanings, and showed that our model has 80% accuracy on in-

ferring the meaning of novel utterances. While prior works have demonstrated some limited ability for one-trial learning of word meanings in simple transitive sentences, our model learns these word meanings very rapidly and confidently in a wide variety of construction types. Finally, we discussed the model‘s handling of long-range dependencies, and its tendency to lexicalize common ngrams and how this might relate to usage-based lexicalization in humans. Despite the comparatively impoverished nature of our training datasets, the model's ability to acquire constructions, including those involving long-range dependencies, and its tendencies both to lexicalize frequent collocations and later to re-analyse them compositionally, appear to be broadly consistent with the course of language acquisition in real children.

7. Acknowledgements

This research was supported by ERC Advanced Fellowship GA 742137 SEMANTAX and the University of Edinburgh Huawei Laboratory.

References

Abend, O., Kwiatkowski, T., Smith, N.J., Goldwater, S., Steedman, M., 2017. Bootstrapping language acquisition. Cognition 164, 116–143.

Ambridge, B., 2020. Against stored abstractions: A radical exemplar model of language acquisition. First Language 40, 509–559.

Bannard, C., Matthews, D., 2008. Stored word sequences in language learning: The effect of familiarity on children's repetition of four-word combinations. Psychological Science 19, 241–248.

Bauer, L., 1983. English word-formation. Cambridge university press.

Berwick, R., 1985. The Acquisition of Syntactic Knowledge. MIT Press, Cambridge, MA.

Bowerman, M., 1973. Structural relationships in children's utterances: Syntactic or semantic?, in: Moore, T. (Ed.), Cognitive Development and the Acquisition of Language. Academic Press, pp. 197–213.

Brown, R., 1973. A First Language: the Early Stages. Harvard University Press, Cambridge, MA.

Bybee, J., 2002. Sequentiality as the basis of constituent structure. Typological Studies in Language 53, 109–134.

Bybee, J., 2006. From usage to grammar: The mind's response to repetition. Language , 711–733.

Bybee, J.L., 1985. Morphology: A study of the relation between meaning and form.

Chater, N., Christiansen, M.H., 2018. Language acquisition as skill learning. Current opinion in behavioral sciences 21, 205–208.

Chomsky, N., 1957. Syntactic structures. Mounton and Co.

Chomsky, N., 1965. Aspects of the Theory of Syntax. MIT Press, Cambridge, MA.

Chomsky, N., 1981. Lectures on Government and Binding. Foris, Dordrecht.

Chomsky, N., 1995. The Minimalist Program. MIT Press, Cambridge, MA.

Collins, M., 1997. Three generative lexicalized models for statistical parsing, in: Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics, ACL. pp. 16–23.

De Villiers, J., Roeper, T., Vainikka, A., 1990. The acquisition of long-distance rules. Language processing and language acquisition , 257–297.

Dupoux, E., 2018. Cognitive science in the era of artificial intelligence: A roadmap for reverse-engineering the infant language-learner. Cognition 173, 43–59.

Erman, B., Warren, B., 2000. The idiom principle and the open choice principle. Text & Talk 20, 29–62.

Ferreira, F., 2007. The "good enough" approach to language comprehension. Language and Linguistics Compass 1, 71–83.

Fisher, C., Hall, D.G., Rakowitz, S., Gleitman, L., 1994. When it is better to receive than to give: Syntactic and conceptual constraints on vocabulary growth. Lingua 92, 333–375.

Fodor, J.D., 1998. Unambiguous triggers. Linguistic Inquiry 29, 1–36.

Frank, M., Goodman, N., Tenenbaum, J., 2007. A Bayesian framework for cross-situational word learning. Advances in Neural Information Processing Systems 20, 20–29.

Gibson, E., Wexler, K., 1994. Triggers. Linguistic Inquiry 25, 355–407.

Gilkerson, J., Richards, J.A., Warren, S.F., Montgomery, J.K., Greenwood, C.R., Kimbrough Oller, D., Hansen, J.H., Paul, T.D., 2017. Mapping the early language environment using all-day recordings and automated analysis. American journal of speech-language pathology 26, 248–265.

Gleitman, L., 1990. The structural sources of verb meanings. Language acquisition 1, 3–55.

Goldberg, Y., Elhadad, M., 2013. Word segmentation, unknown-word resolution, and morphological agreement in a Hebrew parsing system. Computational Linguistics 39, 121–160.

Grimshaw, J., 1981. Form, function, and the language acquisition device. the logical problem of language acquisition, ed. by cl baker and john j. mccarthy, 165–182.

Gropen, J., Pinker, S., Hollander, M., Goldberg, R., 1991. Affectedness and direct objects: The role of lexical semantics in the acquisition of verb argument structure. Cognition 41, 153–195.

Huttenlocher, J., Waterfall, H., Vasilyeva, M., Vevea, J., Hedges, L., 2010. Sources of variability in children's language growth. Cognitive psychology 61, 343–365.

Kahneman, D., 2011. Thinking, fast and slow. Macmillan, New York.

Kahneman, D., 2017. Thinking, Fast and Slow. Farrar, Straus, and Geroux, New York.

Kay, M., 1987. Nonconcatenative finite-state morphology, in: Third Conference of the European Chapter of the Association for Computational Linguistics, pp. 2–10.

Klima, E., Bellugi, U., 1966. Syntactic regularities in the speech of children, in: Lyons, J., Wales, R. (Eds.), Psycholinguistics Papers:The Proceedings of the 1966 Edinburgh Conference. Edinburgh University Press, pp. 183–207.

Kwiatkowski, T., Sharon, G., Zettlemoyer, L., Steedman, M., 2012. A probabilistic model of syntactic and semantic acquisition from child-directed utterances and their meanings, in: EACL.

Langacker, R., 1988. A usage-based model, in: Rudzka-Ostyn, B. (Ed.), Topics in Cognitive Linguistics. John Benjamins, Amsterdam, pp. 127–161.

Lewis, M., Steedman, M., 2014. A^* CCG parsing with a supertag-factored model, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing, ACL, Doha, Qatar. pp. 990–1000.

Lieven, E., Behrens, H., Speares, J., Tomasello, M., 2003. Early syntactic creativity: A usage-based approach. Journal of Child Language 30, 333–370.

Lipka, L., 1977. Lexikalisierung, idiomatisierung und hypostasierung als probleme einer synchronischen wortbildungslehre .

MacWhinney, B., 1998. The childes system. Handbook of child language acquisition , 457–494.

Mahon, L., Abend, O., Berger, U., Demuth, K., Johnson, M., Steedman, M., 2024. A language-agnostic model of child language acquisition. arXiv preprint arXiv:2408.12254 .

Mattys, S., Jusczyk, P., Luce, P., Morgan, J., 1999. Phonotactic and prosodic effects on word segmentation in infants. Cognitive Psychology 38, 465–494.

Neal, R., Hinton, G., 1999. A view of the em algorithm that justifies incremental, sparse, and other variants, in: Jordan, M. (Ed.), Learning in Graphical Models. MIT Press, Cambridge, MA, pp. 355–368.

Pinker, S., 1979. Formal models of language learning. Cognition 7, 217–283.

Schlesinger, I., 1971. Production of utterances and language acquisition, in: Slobin, D. (Ed.), The Ontogenesis of Grammar. Academic Press, New York, pp. 63–101.

Seki, H., Matsumura, T., Fujii, M., Kasami, T., 1991. On multiple context-free grammars. Theoretical Computer Science 88, 191–229.

Shieber, S., 1985. Evidence against the context-freeness of natural language. Linguistics and Philosophy 8, 333–343.

Siskind, J., 1992. Naive Physics, Event perception, Lexical Semantics, and Language Acquisition. Ph.D. thesis. MIT.

Siskind, J., 1996a. A computational study of cross-situational techniques for learning word-to-meaning mappings. Cognition 61, 39–91.

Siskind, J.M., 1993. Naive physics, event perception, lexical semantics, and language acquisition .

Siskind, J.M., 1996b. A computational study of cross-situational techniques for learning word-to-meaning mappings. Cognition 61, 39–91.

Srinivas, B., Joshi, A., 1994. Disambiguation of super parts of speech (or supertags): Almost parsing, in: Proceedings of the International Conference on Computational Linguistics, ACL.

Stanojević, M., Brennan, J., Dunagan, D., Steedman, M., Hale, J., 2023. Modeling structure-building in the brain with CCG parsing and large language models. Cognitive Science 47, 1–39.

Steedman, M., 2000. The syntactic process. MIT press.

Stromswold, K., 1995. The acquisition of subject and object *wh*-questions. Language Acquisition 4, 5–48.

Szubert, I., Abend, O., Schneider, N., Gibbon, S., Mahon, L., Goldwater, S., Steedman, M., 2024. Cross-linguistically consistent semantic and syntactic annotation of child-directed speech. Language Resources and Evaluation , 1–50.

Tomasello, M., 2003. Constructing a Language: A Usage-based Theory of Language Acquisition. Harvard University Press.

Vijay-Shanker, K., Weir, D., 1994. The equivalence of four extensions of context-free grammar. Mathematical Systems Theory 27, 511–546.

Warstadt, A., Mueller, A., Choshen, L., Wilcox, E., Zhuang, C., Ciro, J., Mosquera, R., Paranjabe, B., Williams, A., Linzen, T., Cotterell, R. (Eds.), 2023a. Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning, Association for Computational Linguistics, Singapore. URL: https://aclanthology.org/2023.conll-babylm.0.

Warstadt, A., Mueller, A., Choshen, L., Wilcox, E., Zhuang, C., Ciro, J., Mosquera, R., Paranjabe, B., Williams, A., Linzen, T., et al., 2023b. Findings of the babylm challenge: Sample-efficient pretraining on developmentally plausible corpora, in: Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning.

Wexler, K., Culicover, P., 1980. Formal Principles of Language Acquisition. MIT Press, Cambridge, MA.

Yang, C., 2002. Knowledge and Learning in Natural Language. Oxford University Press, Oxford.

Appendix A. Mapping from CHILDES POS Tags to Montagovian Semantic Types

Table A.1 shows how we infer the Montagovian semantic type from the CHILDES POS tags that are available in our lfs. Some are defined schematically, the avoid overly long expressions. For example, the category for conjunctions (conj) and coordinations (coord), we use the variable X to stand for any other semantic category. The reason the mapping from tags to semantic types is many-to-one is that this allows learning to be shared across categories. For example, if the model learns that the general category 'det' precedes nouns, it knows that this is true for all types of determiners, whereas if we distinguish between 'det:art', 'det:poss', 'det:num' etc., then it has to learn this separately for each.

Appendix A.1. Manually Annotated Lexicon for Fifty Most Common Words

This section shows the ground-truth logical form meaning representation and CCG syntactic category for the fifty most common words in each dataset. As described in Section 4.3, these are used to evaluate the learner's ability to acquire the correct lexicon. Note, the lfs that appeared in the main paper were abbreviated for clarity. Here, we write the full lf, including the CHILDES part of speech tag. The full lexical entry is of the form $\langle \text{lf} \rangle \; || \; \langle \text{syntactic-category} \rangle$

. Where a word has two common meanings, we include two different lexical entries, separated with a comma.

```
'll:λ x.λ y.mod|~will (x y) || S\\NP/(S\\NP)
're:λ x.λ y.v|hasproperty y x || S\\NP/NP,
    λ x.λ y.v|equals y x || S\\NP/NP
's:λ x.λ y.v|equals y x || S\\NP/NP,
    λ x.λ y.v|hasproperty y x || S\\NP/NP
Adam:n:prop|adam || NP
I:pro:sub|i || NP
a:λ x.det:art|a x || NP/N
an:λ x.det:art|a x || NP/N
another:λ x.qn|another x || NP/N
are:λ x.λ y.v|equals x y || S\\NP/NP,
    λ x.λ y.v|hasproperty y x || S\\NP/NP
break:λ x.λ y.v|break y x || S\\NP/NP
can:λ x.λ y.mod|can (x y) || S\\NP/(S\\NP),
    λ x.λ y.mod|can (x y) || S/NP/(S\\NP)
d:λ x.λ y.mod|do (x y) || S\\NP/(S\\NP),
    λ x.λ y.mod|do (x y) || S/NP/(S\\NP)
did:λ x.λ y.mod|do-past (x y) || S/NP,
    λ x.λ y.mod|do-past (x y) || S/NP/(S\\NP)
do:λ x.λ y.v|do y x || S\\NP/NP,
    λ x.λ y.mod|do (x y) || S/NP/(S\\NP)
does:λ x.λ y.mod|do-3s (y x) || S\\NP/(S\\NP),
    λ x.λ y.mod|do-3s (x y) || S/NP/(S\\NP)
dropped:λ x.λ y.v|drop-past y x || S\\NP/NP
have:λ x.λ y.v|have y x || S\\NP/NP
he:pro:sub|he || NP
his:λ x.det:poss|his x || NP/N,pro:poss|his || NP
hurt:λ x.λ y.v|hurt-zero y x || S\\NP/NP
in:λ x.λ y.prep|in (y x) || S\\NP\\(S\\NP)/NP,
    λ x.prep|in x || S/S
is:λ x.λ y.v|equals x y || S\\NP/NP,
    λ x.λ y.v|hasproperty y x || S\\NP/NP
it:pro:per|it || NP
like:λ x.λ y.v|like y x || S\\NP/NP
lost:λ x.λ y.v|lose-past y x || S\\NP/NP
may:λ x.λ y.mod|may (x y) || S\\NP/(S\\NP)
missed:λ x.v|miss-past x || S\\NP,
    λ x.λ y.v|miss-past y x || S\\NP/NP
my:λ x.det:poss|my x || NP/N
name:n|name || N
need:λ x.λ y.v|need y x || S\\NP/NP
no:λ x.qn|no x || NP/N
```

```
not:λ x.λ y.not (x y) || S\\NP/(S\\NP)\(S\\NP/(S\\NP))
on:λ x.prep|on x || S\\NP\\(S\\NP)/NP
one:pro:indef|one || NP
pencil:n|pencil || N
say:λ x.λ y.v|say y x || S\\NP/NP
see:λ x.λ y.v|see y x || S\\NP/NP
shall:λ x.λ y.mod|shall (x y) || S\\NP/(S\\NP)
some:λ x.qn|some x || NP/N
that:pro:dem|that || NP,λ x.pro:det|that x || NP/N
the:λ x.det:art|the x || NP/N
they:pro:sub|they || NP
this:pro:dem|this || NP,λ x.pro:det|this x || NP/N
those:pro:dem|those || NP,λ x.pro:det|those x || NP/N
was:λ x.λ y.v|equals x y || S\\NP/NP,
    λ x.λ y.v|hasproperty y x || S\\NP/NP
we:pro:sub|we || NP
what:pro:int|WHAT || Swhq/Sq/NP,pro:int|WHAT || NP
who:pro:int|WHO || Swhq/Sq/NP,pro:int|WHO || NP
you:pro:per|you || NP
your:λ x.det:poss|your x || NP/N
```

Appendix B. Mapping from CHILDES POS Tags to Shell lf Terms

As described in Section 3.1, we use the CHILDES part of speech tags, which are included in the logical forms of Szubert et al. (2024), to choose the marking on the constant in the shell logical form. Table B.2 gives full correspondence. In the main text in Section 4.2, we indicated the marking with the first letter of the right column, e.g. 'verb' gives 'vconst'.

Appendix C. Base Distributions

As described in Section 3.1, each of the components of our model, p_r, p_t, p_e, p_l and p_w use a base distribution, which is then updated with the expected observed cooccurrence counts during training. The base distributions for each of these models are as follows:

- for p_r and p_t: $H(y) = 0.9^n$, where n is the number of atomic categories in y;
- for p_l and p_e, $H(y) = 0.25^n$, where n is the number of variables and constants in y;
- for p_w, $H(y) = 0.72^n$, where n is the number of letters in y.

These are unnormalised distributions, because they do not sum to 1, though their sum is finite. In principle, these could be normalised by fixing a vocabulary size, however we simply leave them unnormlised in our experiments. Normalisation is immaterial for p_w anyway, because all analyses for an observed utterance will have the same number of letters in the leaf node words, so normalising would just multiply the numerator and the denominator of Equation (3) by the same factor.

Table A.1: Our mapping from CHILDES part of speech tags of terms in the logical form to Montagovian semantic types.

CHILDES TAG	const marking in shell lf
adj	<<e,t>,<e,t>>
adv	not considered
adv:int	not considered
adv:tem	not considered
aux	not considered
conj	<X,<X,X>>
coord	<X,<X,X>>
cop	handled separately
det	<<e,t>,e>
det:art	<<e,t>,e>
det:dem	<<e,t>,e>
det:int	<<e,t>,e>
det:num	<<e,t>,e>
det:poss	<<e,t>,e>
mod	<<<e,t>,<e,t>>,<e,t>>
mod:aux	<<e,t>,e>
n	<e,t>
n:pt	<e,t>
n:gerund	e
n:let	e
n:prop	e
neg	<<e,<e,t>>,<e,<e,t>>>, <<e,t>,<e,t>>t,t
prep	<<e,t>, <e, t>>
pro:dem	e
pro:indef	e
pro:int	e
pro:obj	e
pro:per	e
pro:poss	<e,t>
pro:refl	e
pro:sub	e
qn	<e,t>
v	<e,<e,t>>, <e,t>

Table B.2: Our mapping from CHILDES part of speech tags of terms in the logical form to the marking on the constant in the corresponding shell logical form.

CHILDES TAG	const marking in shell lf
adj	adj
adv	adv
adv:int	adv
adv:tem	adv
aux	aux
conj	connect
coord	connect
cop	cop
det	quant
det:art	quant
det:dem	quant
det:int	quant
det:num	quant
det:poss	quant
mod	raise
mod:aux	quant
n	noun
n:pt	noun
n:gerund	entity
n:let	entity
n:prop	entity
neg	neg
prep	prep
pro:dem	entity
pro:indef	entity
pro:int	WH
pro:obj	entity
pro:per	entity
pro:poss	quant
pro:refl	entity
pro:sub	entity
qn	quant
v	verb