CAIT: A Syntactic Parsing Toolkit for Child–Adult InTeractions

Francesca Padovani[1], Xiulin Yang[2], Bastian Bunzeck[3], Jaap Jumelet[1], Yevgen Matusevych[1], Nathan Schneider[2], Arianna Bisazza[1]

[1]Center for Language and Cognition (CLCG), University of Groningen,
[2]Georgetown University,
[3]Computational Linguistics, Department of Linguistics, Bielefeld University
Correspondence: f.padovani@rug.nl

Abstract

CHILDES is a paramount resource for language acquisition studies—yet computational tools for analyzing its syntactic structure remain limited. Leveraging the recent release of the UD-English-CHILDES treebank with gold-standard Universal Dependencies (UD) annotations, we train a state-of-the-art dependency parser specifically tailored to CHILDES. The parser more accurately captures syntactic patterns in child–adult interactions, outperforming widely used off-the-shelf English parsers, including SpaCy and Stanza. Alongside the parser, we also release a Part-of-Speech tagger and an utterance-level construction tagger, which together form the open-source Syntactic Parsing Toolkit for Child–Adult InTeractions (CAIT). Through a detailed error analysis and a case study tracking the distribution of syntactic constructions across developmental time in CHILDES, we demonstrate the practical utility of the toolkit for large-scale, reproducible research on language acquisition.[1]

1 Introduction

The kind of language that children produce and hear in their environment differs from regular everyday language (e.g., Saxton, 2009). Child Speech (CS) starts out with semantically motivated isolated words (e.g., *there*), holistic phrases (e.g., *all-gone*), and multiword utterances revolving around 'pivot words' (Braine and Bowerman, 1976) with an open slot (e.g., *"More _"*) (cf. Lieven et al., 1997, 2003; Tomasello, 2000b, 2003). Child-Directed Speech (CDS) is similarly characterized by short, syntactically simple utterances, high lexical and structural repetition, and cross-turn redundancy (Lester et al., 2022; Cameron-Faulkner et al., 2003). Both registers challenge standard linguistic analysis through non-canonical constructions, disfluencies, self-repair, and fragmentary utterances (Liu and Prud'hommeaux, 2023). As usage-based theories propose that linguistic competence emerges from repeated exposure to structured input patterns (Cameron-Faulkner et al., 2003; Lubetich and Sagae, 2014; You et al., 2021), CS and CDS are central empirical domains: the former reflects the developing linguistic system, the latter reveals the distributional patterns from which it emerges.

Many language acquisition studies rely on transcribed corpora such as CHILDES[2] (e.g., MacWhinney, 2000; Legate and Yang, 2002; Lidz et al., 2003). More recently, advances in neural language models have renewed interest in using computational methods to test acquisition hypotheses under controlled, naturalistic input conditions (e.g., Warstadt and Bowman, 2022; Portelance and Jasbi, 2024; Bunzeck et al., 2025). Despite its empirical and theoretical importance, CHILDES is difficult to analyze at scale. Existing computational tools struggle with its domain-specific idiosyncrasies, forcing researchers either to manually inspect data or rely on general-purpose parsers trained on written, adult-directed text—tools not designed for spoken, developmental language whose errors require costly post-hoc correction.

Most existing dependency parsers are trained on adult-directed written text (e.g., news, books, encyclopedic articles) and consequently perform poorly when applied to spoken, interactional, and developmental input such as CHILDES. While a small number of studies have explored dependency parsing for adult conversational speech (e.g., Bechet et al., 2014; Davidson et al., 2019), no clear consensus exists on a parser that can annotate CHILDES data reliably.

The recent publication of UD-English-

[1]Models and data can be found at https://github.com/fpadovani/CAIT-Toolkit/

[2]Throughout the paper, we use the term CHILDES as a shorthand for the English child–adult interaction domain represented in the CHILDES repository and used in our experiments. When relevant, we differentiate between Child Speech (CS) and Child-Directed Speech (CDS).

CHILDES (Yang et al., 2025) provides the first officially validated Universal Dependencies (UD) v2 treebank (Zeman, 2021) for English child–adult interactions. It consolidates several earlier annotation efforts (Liu and Prud'hommeaux, 2021; Liu and Prud'hommeaux, 2023; Szubert et al., 2025) into a unified resource that conforms to current UD guidelines. Building on this resource, our work presents four main contributions:

- We **train and evaluate several dependency parsers** on UD-English-CHILDES (Yang et al., 2025) under different training setups and assess their performance on a CHILDES test set.
- We release **CAIT** (pronounced [kaɪt]), an open-source syntactic parsing toolkit for Child–Adult InTeractions, whose central component is our best performing **dependency parser** based on `SuPar` (Zhang et al., 2020). The suite also includes a **POS tagger** trained using `Stanza` (Qi et al., 2020), and an utterance-level **construction tagger** leveraging the parser annotations.
- We perform an **in-depth error analysis**, comparing CAIT annotations with those of the off-the-shelf English `Stanza` parser, highlighting systematic differences in dependency predictions and the structural biases introduced by general-purpose parsers.
- We demonstrate the practical utility of CAIT through a **case study**, in which the construction tagger is used to track the distribution of syntactic structures across development.

Overall, our results show that an in-domain CHILDES model can enhance parsing accuracy on CS and CDS, offering a valuable tool for both computational modeling that requires strict syntactic manipulation of CHILDES and empirical studies of language development that rely on large-scale syntactic analysis.

2 Related Work

Parsing Spoken Language Over the past decades, several studies have focused on developing treebanks for dialogue and conversational data (Caines et al., 2017; Dobrovoljc and Martinc, 2018; Braggaar and van der Goot, 2021). A classic example is the pivotal work on the Switchboard (Godfrey et al., 1992) section of the Penn Treebank (Marcus et al., 1993), which consists of transcribed adult–adult telephone conversations. More recent efforts have moved towards UD-style treebanks (Kahane et al., 2021; Dobrovoljc, 2022), which have become the dominant syntactic annotation framework in NLP. For instance, Dobrovoljc and Nivre (2016), Øvrelid et al. (2018) and Kahane et al. (2021) introduced UD dependency annotations for adult conversational speech in Slovenian, Norwegian, and French, respectively. On the parsing side, models trained or fine-tuned on domain-specific spoken data have been shown to outperform those trained on written text alone, e.g., Bechet et al. (2014) on French and Davidson et al. (2019) on English human–machine dialogues, highlighting the importance of in-domain data for parsing spoken language.

Parsing CHILDES The CHILDES database (MacWhinney, 2000) provides dependency annotations originally developed by Sagae et al. (2004, 2005, 2007), which, however, do not conform to the UD framework.

With UD-style annotation gaining importance in Natural Language Processing (NLP) and language acquisition research, NLP toolkits such as `Stanza` (Qi et al., 2020) have been used to parse CHILDES data (Liu and MacWhinney, 2024). However, these automatic annotations remain inconsistent and unreliable. Liu and Prud'hommeaux (2023) present the first extensive UD-style dependency annotations for child–adult interactions. Their work extends earlier, smaller-scale studies that applied UD standards to this data (Liu and Prud'hommeaux, 2021). In their treebank, Liu and Prud'hommeaux (2023) address all critical aspects of spoken dialogue, such as speech repairs, restarts, and disfluent fragments, ensuring that the resource reflects the full range of phenomena characteristic of CHILDES. The study also presents the most recent and extensive evaluation of parsers on CHILDES, testing multiple out-of-domain models on child–adult interactions. These experiments reveal that general-purpose parsers struggle with constructions typical of this domain. Training on a small set of nine annotated child–adult interactions improves parsing accuracy, particularly for younger children, though the gains over out-of-domain models remain modest. Despite being a useful resource, their dataset is not fully consistent with UD v2, lacks UPOS tags, and has not undergone independent validation.

To address these limitations, Yang et al. (2025) have recently undertaken a major harmonization effort, integrating and expanding previous datasets

to produce a unified UD v2–compliant resource for CHILDES. Their work applies roughly 8,000 corrections, standardizes annotation practices across corpora, and results in the first official UD release for CHILDES speech data, which we use here to test different parsing architectures.[3]

Structural Annotation in Acquisition Studies CHILDES has been an important resource for acquisition studies (Diessel and Tomasello, 2000; Legate and Yang, 2002; Lidz et al., 2003). Investigations requiring structural information routinely rely on manual annotation, which is labor-intensive, thus costly and only applicable to small samples. Earlier attempts at annotating CHILDES data automatically either focus on specific phenomena (e.g., island effects, Pearl and Sprouse, 2013, logical representations, Szubert et al., 2025, modal verbs, van Dooren et al., 2022, and utterance-level constructions, Bunzeck and Diessel, 2025; Bunzeck et al., 2025), or on very broad categories of annotation like POS tags (MacWhinney, 2008; Albert et al., 2013) and simplistic dependency structures (Sagae et al., 2010). However, most of these approaches were created *ad hoc*, lack integration with standardized resources and suffer from a general mismatch between NLP tools and CHILDES data. While Bunzeck and Diessel (2025) automated utterance-level construction annotation and reported a tagging accuracy of ≈95%, they had to resort to a coarser construction hierarchy than previous, manually annotated studies (Cameron-Faulkner et al., 2003) and focused exclusively on CDS, because results on CS were too unreliable based on the rule-based POS tags available in CHILDES at that time (cf. MacWhinney, 2008).

3 Building a CHILDES-specific Dependency Parser and POS Tagger

3.1 Data

For training CAIT, we use the annotated CHILDES dataset released by Yang et al. (2025). It consists of English sentences from child–caregiver spoken interactions. The treebank contains two types of data: gold and silver. The gold portion consists of approximately 48k manually corrected sentences (237k words), while the silver portion is automatically annotated with `Stanza` (the combined off-the-shelf model) and covers an additional 1M sentences (6.9M words). The dataset is provided with predefined train, development, and test splits. The test split contains only gold UD annotations; silver data are used exclusively for training augmentation. The train and development splits draw from the same CHILDES corpora, while the test split comes from different corpora in order to evaluate model generalization.[4]

3.2 Parser Training

We train and evaluate dependency parsers for CHILDES using four Python libraries: `Stanza` (Qi et al., 2020), `SuPar`[5] (Zhang et al., 2020), `DiaParser`[6] (Attardi et al., 2021), and `Machamp`[7] (van der Goot et al., 2021). We select these parsers for their detailed documentation, strong reported performance in dependency parsing, and prior use in parsing CHILDES by Liu and Prud'hommeaux (2023). Across these frameworks, we explore multiple architectures and training configurations to assess the impact of input representations, contextualization, and training strategies on CHILDES parsing performance.

Stanza `Stanza` implements a BiLSTM-based deep biaffine dependency parser following Dozat and Manning (2017). We evaluate the off-the-shelf `Stanza` (combined model[8]) and, separately, train parsers on UD-English-CHILDES under three input settings: (i) the default configuration using a BiLSTM encoder with the pretrained word vectors released for the CoNLL-2017 UD shared task (Zeman et al., 2017, named `conll17.pt`); (ii) contextualized embeddings from *RoBERTa-large*; and (iii) contextualized embeddings from *RoBERTa-base* (Liu et al., 2019). Models were trained with a batch size of 5k for up to 50k steps, with early stopping based on development set performance using a patience of 1k steps.

SuPar `SuPar` implements a biaffine dependency parser supporting two encoding regimes: a BiLSTM/Transformer encoder, where token representations can be augmented with pretrained static embeddings (e.g., GloVe, Pennington et al., 2014;

[3] Concurrently to this work, a new UDPipe release has been made public at https://lindat.mff.cuni.cz/services/udpipe/, including a CHILDES_UD-based parser that performs similarly to our Stanza parser.

[4] For details, see https://github.com/UniversalDependencies/UD_English-CHILDES.

[5] https://pypi.org/project/supar/

[6] github.com/Unipisa/diaparser

[7] github.com/machamp-nlp/machamp

[8] Model trained on the combination of English Treebanks: EWT, GUM, GUMReddit, PUD, Pronouns; see https://stanfordnlp.github.io/stanza/combined_models.html.

Parser	Backbone Model	Training Data	Development		Test	
			LAS	UAS	LAS	UAS
SuPar	**RoBERTa-large**	**gold**	93.23	**96.23**	**92.51**	**94.91**
SuPar	RoBERTa-large	10k silver + gold	93.45	95.59	91.20	93.83
SuPar	RoBERTa-large	10k silver → gold	92.31	94.80	90.41	93.34
Stanza	BiLSTM	gold	92.51	94.91	90.34	93.44
Stanza	RoBERTa-large	gold	93.28	95.65	91.39	94.21
Stanza	RoBERTa-base	gold	93.06	95.44	91.40	94.16
Stanza	off-the-shelf	all EN-UD treebanks	86.57	90.63	85.19	89.18
DiaParser	RoBERTa-large	gold	**94.48**	91.58	89.71	93.03
DiaParser	RoBERTa-large	10k silver + gold	89.10	92.58	87.77	91.33
Machamp	RoBERTa-large	gold	85.13	–	79.00	–
Machamp	RoBERTa-base	gold	84.46	–	78.12	–
spaCy	en_core_web_trf	OntoNotes 5	61.25	69.37	61.45	68.11

Table 1: Mean LAS and UAS scores for each model on the development (3,860 sentences, 24,310 words) and test (9,591 sentences, 57,400 words) sets. Silver → gold: pretrain on silver, finetune on gold; silver + gold: finetune on both. Gold data refers to UD-CHILDES from Yang et al. (2025). Compared with the off-the-shelf Stanza baseline, our best SuPar model's (boldface) gains on the test set are statistically significant (paired t-test over per-sentence UAS and LAS scores, 9,591 paired observations; UAS/LAS: $p < 0.001$).

FastText, Bojanowski et al., 2017) and auxiliary features such as character-level representations and POS-tag embeddings; or a pretrained language model encoder, where token representations are obtained directly from the model without auxiliary inputs. In our experiments, we find that the latter encoding approach yields better results, and we therefore adopt *RoBERTa-large* (Liu et al., 2019) as the encoder backbone. We experiment with training for 10–50 epochs and observe that shorter training (around 10 epochs) leads to better generalization.

Since SuPar yields the best performance in our preliminary experiments, we further explore data augmentation strategies. We consider two settings: (i) pretraining the parser on 10k silver sentences followed by fine-tuning on the gold data, and (ii) augmenting the gold training set with an additional 10k silver sentences. The silver data are obtained by re-annotating the automatically annotated silver transcripts from Yang et al. (2025) using our best domain-specific CHILDES parser, which we publicly release.

Machamp and DiaParser Machamp (van der Goot et al., 2021) is a toolkit that supports multi-task learning, including dependency parsing, sequence tagging, and masked language modeling. For dependency parsing, it adopts the same deep biaffine parser of Dozat and Manning (2017) as Stanza. As for DiaParser (Attardi et al., 2021), it extends SuPar by incorporating transformer-based representations and by using attention weights from a selected transformer layer as additional features. These attention values are added to the biaffine arc-scoring function with a learned weight to provide structural hints for dependency edge prediction. However, none of the experimental settings we evaluated achieved performance comparable to SuPar. We report the two best-performing configurations for both parsers (Machamp and DiaParser) in Table 1.

spaCy Given its widespread use in dependency parsing (e.g., Arista et al., 2025; Isaak, 2023), we also include the off-the-shelf spaCy model en_core_web_trf (Honnibal et al., 2020) as an additional baseline. spaCy implements a transition-based dependency parser using a variant of the non-monotonic arc-eager system (Honnibal and Johnson, 2015), trained using an imitation learning objective. Unlike the other parsers considered here, spaCy is not a UD-based parser. It is trained on OntoNotes 5 (Weischedel et al., 2011) with ClearNLP constituent-to-dependency conversion (Arnold, 2017) and does not directly produce UD-compliant dependencies; its outputs therefore require conversion before they can be compared against the UD-English-CHILDES gold standard.

3.3 Parser Evaluation

We evaluate all models using two standard dependency parsing metrics: Unlabelled Attachment Score (UAS) and Labelled Attachment Score

(LAS). UAS is computed as the percentage of tokens for which the predicted head is correct, while LAS is computed as the percentage of tokens for which both the predicted head and the dependency relation label are correct. Because different parsing libraries employ distinct evaluation pipelines, direct comparison may introduce inconsistencies.

To ensure comparability, we use a unified framework based on the `Stanza` evaluation function, which computes LAS and UAS by comparing CoNLL-U predictions against gold-standard annotations from the UD-English-CHILDES test set. `SuPar` and `DiaParser` natively support CoNLL-U output. In contrast, `spaCy`'s output is not UD-compliant and required additional pre- and post-processing to convert its outputs into valid CoNLL-U files, including remapping dependency labels to UD and correcting rare structural issues (e.g., mid-sentence index resets causing cycles or incorrect roots).[9] For `Machamp`, LAS and UAS are reported using its built-in evaluation pipeline, as it does not reliably generate well-formed CoNLL-U trees compatible with the `Stanza` script.

Results Table 1 summarizes dependency parsing performance on the development and test sets for models trained in this study and the baselines. The best scores are obtained by a `SuPar` biaffine parser with *RoBERTa-large* embeddings fine-tuned for 10 epochs, which we release as a state-of-the-art CHILDES-specific parser within CAIT and adopt as the primary reference model in subsequent analyses. `Stanza` models with contextualized embeddings are closely competitive, whereas silver data augmentation with `SuPar` does not yield consistent improvements. `DiaParser` and `MaChAmp` lag behind across all configurations and are not pursued further. All domain-specific models substantially outperform `spaCy` and the `Stanza` off-the-shelf baseline, underlining the importance of in-domain training data for parsing CHILDES.

3.4 POS Tagger Training and Evaluation

While our best-performing dependency parser is built on `SuPar`, this library has a notable limitation: POS tags are not used as input features during training and are therefore not predicted at inference time. As a result, the parser outputs only dependency heads and relation labels for each token, without POS annotations. To address this limitation, CAIT also includes a dedicated POS tagger trained with `Stanza`. The tagger is trained on gold data from Yang et al. (2025), and can be directly loaded and applied using the standard `Stanza` pipeline, facilitating seamless integration with the CAIT parser and downstream annotation components. We evaluate the POS tagger using standard UD tagging accuracy for universal POS tags (UPOS) and language-specific POS tags (XPOS). It achieves 98.17 (UPOS) and 96.94 (XPOS) on the development set and 96.77 (UPOS) and 95.84 (XPOS) on the test set.

4 In-Depth Analysis of CAIT

In this section, we conduct a detailed analysis of the CAIT dependency parser, in comparison with the off-the-shelf English `Stanza`. The latter serves as our baseline, as it is one of the most widely adopted tool for generating English dependency parses in current computational modeling studies (e.g., Kallini et al., 2024; Yang et al., 2026). We first examine dependency relations for which the CHILDES-specific parser consistently outperforms the domain-generic baseline, highlighting cases where it mitigates systematic errors likely driven by biases in models trained on standard written corpora. We then analyze remaining error patterns, identifying phenomena for which performance gains are limited or where error rates remain relatively high. Finally, we show that CAIT can also help detect annotation inconsistencies in the gold data.

Figure 1 presents a per-label comparison of parsing accuracy between the CAIT parser and the `Stanza` off-the-shelf baseline. For each dependency label, errors are normalized by the number of gold instances, allowing for a fair comparison across labels of varying frequency.

4.1 Parsing Improvements

`conj` CHILDES frequently contains repetition and bare enumeration patterns (e.g., *six six seven*, *purple blue green*) that are rare in standard written corpora. In the gold annotations, these are represented as flat structures with the first element as head and subsequent items attached via `conj`. The off-the-shelf parser usually wrongly analyzes these strings by selecting the final element as the `root` and attaching antecedent items as modifiers (e.g., `amod`, `nummod`). The CAIT parser instead recovers the intended flat representation (see Appendix, Figure 4).

[9] Full details on the label remapping are available in `evaluation/mapping_parser_spacy.py` in the repository.

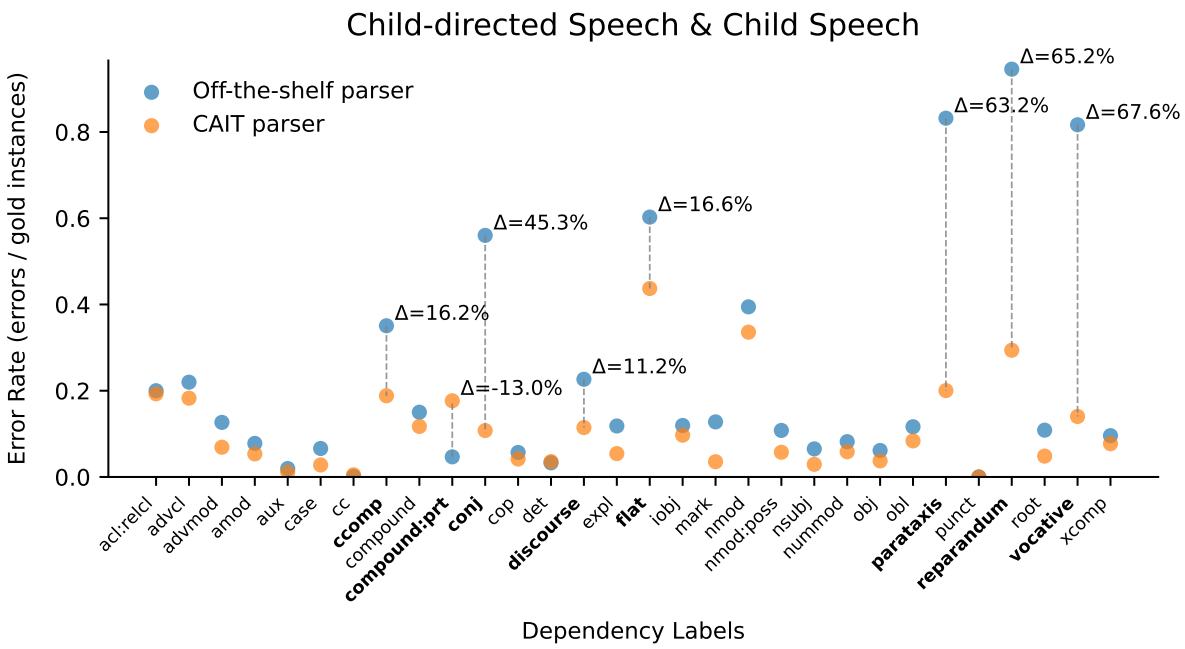


Figure 1: Per-label error rates (errors normalized by gold label count) for the CAIT parser and the `Stanza` off-the-shelf baseline on the test set. Dashed lines and bold labels mark differences greater than 10 percentage points. Only dependency labels with at least 100 gold instances are included.

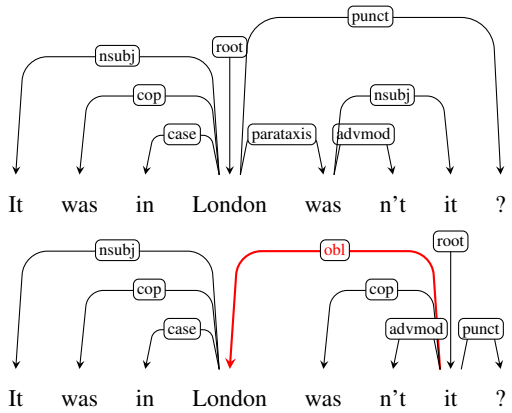


Figure 2: Gold (top) and predicted by the off-the-shelf parser (bottom) labels and relations for `parataxis`.

`vocative` In CHILDES, names and words of address like *Thomas* or *sweetheart* often appear without punctuation, unlike in typical written text. In the gold data, these are marked as `vocative` to show they are peripheral attention-getters, not part of the clause's argument structure. The off-the-shelf parser often treats them as `root` or `obj`, while the CAIT parser correctly identifies them as separate from the clause's main dependencies (see Appendix, Figure 5).

`parataxis` Parataxis covers short clauses or phrases added loosely to a sentence, without being tightly connected to its main structure. In CHILDES, speakers often add questions, repeated phrases, or side comments as they speak (e.g., *"…haven't you?"*, *"Excuse you excuse you"*). The off-the-shelf parser usually mislabels these by trying to fit them into the clause structure, treating them as complements (`ccomp`), copulas (`cop`) or obliques (`obl`), as in Figure 2. The CAIT parser handles them better by recognizing that these extra phrases are peripheral add-ons, independent of the clause's core structure.

`reparandum` The CAIT parser also improves performance on the `reparandum` label, which marks disfluent fragments such as stuttered segments (*"N n"*) and phrase restarts (*"I'm sure I'm sure..."*). The baseline parser often misclassifies these as functional categories (`det`, `cc`) or attaches them as sentence `root`, distorting the dependency structure. By correctly identifying speech repairs, the CAIT parser avoids cascading attachment errors in the rest of the tree (see Appendix, Figure 6).

4.2 Error Patterns

Due to the domain-specific features of child–adult interaction, such as fragmentary utterances, discourse markers, and repetitions, the CAIT parser occasionally overgeneralizes certain dependency relations, resulting in a series of residual error patterns (Appendix, Figure 11). Confusions involving `compound:prt` are among the most systematic errors produced by CAIT. Confusions are bidirec-

tional between compound:prt and advmod, likely reflecting the annotation harmonization of Yang et al. (2025) targeting the high density of phrasal verbs in CHILDES. While advmod marks transparent compositional modifiers (e.g., *slide down*, *going around*), compound:prt is intended for lexicalized verb–particle units whose meaning cannot be fully predicted from their parts (e.g., *hold on* 'wait', *wear out* 'make somebody tired'). CAIT also frequently over-extends compound:prt to constructions such as *put the pillows up*, likely because light verbs such as *go*, *come*, and *put* appear in both literal motion uses and idiomatic phrasal verbs. Simple adverbial modifiers are sometimes misclassified as discourse markers, as *Now* in *"Now I need the blue one"*, where CAIT treats temporal words as conversational cues. Similarly, nsubj dependents are occasionally predicted as roots or as vocatives, reflecting CAIT's difficulty in distinguishing between syntactic arguments and conversational address (e.g., *"Who's me"* or *"Mommy fix a paper"*).

4.3 Annotation Errors Identified by CAIT

CAIT can also function as a diagnostic tool for detecting annotation inconsistencies when compared with a baseline such as Stanza trained on standard UD treebanks. Because Stanza largely reflects conventional UD annotation practices, systematic divergences between the two parsers may arise either from genuine performance differences or from inconsistencies in the gold annotations. We identify two annotation inconsistencies in Yang et al. (2025).

det The CAIT parser tends to predict possessive pronouns as det rather than nmod:poss (e.g., *his coffee*, *your dolly*; see Appendix, Figure 11). A closer inspection of UD-English-CHILDES reveals that a small portion of possessive pronouns (≈3.5%) are annotated as det, deviating from standard English UD, where det is reserved for DET-tagged words. The parser learned this pattern from the data, making this an annotation-driven error.

nmod Similarly, the CAIT parser shows high error rate on nmod, frequently predicting compound for premodifying noun constructions such as *grape juice*, *space shuttle* or *tomato soup* (see Appendix, Figure 9). Per standard English UD guidelines, compound is correct for these constructions, while nmod is reserved for prepositional phrases within nominals; but in UD-English-CHILDES, ≈5% of noun-premodifying-noun instances are incorrectly annotated as nmod. The parser has correctly learned the standard compound pattern, and its elevated error rate on nmod thus reflects annotation noise.

Appendices A, B, and C present additional evaluation metrics, gold–prediction tree comparisons, and a fine-grained analysis of error directions by parser type. In the next section, we leverage the CAIT parser to improve an utterance-level construction tagger and show its effectiveness in a case study.

5 Utterance-level construction annotation

To test the usefulness of our parser for downstream tasks in language acquisition, we build an utterance-level construction tagger on top of it, similarly to Bunzeck and Diessel (2025). Utterances are "the primary psycholinguistic unit of child language acquisition" (Tomasello, 2000a). As there exists no tagger integrated with current resources and NLP tools (see Section 2), we identify this task as an ideal opportunity to test our UD-parser.

Methodology We approach construction tagging as a multiclass classification problem. For the classes, we devise an annotation scheme based on Cameron-Faulkner et al. (2003): Utterances are categorized as formulaic (FOR, such as greetings like *"Hello!"*), fragments without a predicate (FRA), wh- or yes/no- questions (QWH, QYN), copula sentences (COP, such as in Figure 2), imperatives (IMP), subject-predicate in/transitive (SPI, SPT), or complex utterances with at least two predicates (COM). For more details on the annotation scheme, see Appendix D.1. We compare the following kinds of construction taggers: i) A rule-based tagger based on the UD tags provided by CAIT (cf. Appendix D.2), ii) a rule-based tagger based on UD tags provided by off-the-shelf Stanza, iii) a rule-based tagger based only on POS tags provided by CAIT (similar to Bunzeck and Diessel, 2025), iv) a multilayer perceptron (MLP) classifier operating directly on sentence embeddings without syntactic annotation.

Construction tagger performance Table 2 reports performance on two different data sets. The dev data comes from a singular child–caregiver conversation (2,141 utterances) from the MPI-EVA-Manchester corpus (Lieven et al., 2009). The test data contains another 1,000 utterances randomly sampled from the entire CHILDES corpora to better represent potential inconsistencies in the data. All data were manually annotated ac-

Tagger	dev	test
CAIT parser	92.05	92.32
Stanza off-the-shelf	91.23	89.79
POS-only	87.54	85.74
MLP	–	70.17

Table 2: Accuracies of different tagging approaches (%).

cording to the aforementioned annotation guide. (Appendix D.5 contains another comparison for synthetic data.) As the MLP classifier is trained on the dev set, no accuracy is reported for it.

Across both data sets, the tagger based on our parser performs best. This holds for both CDS and CS individually (see Appendix D.3). While the difference on the dev data is marginal, the differences on the test data are more pronounced. As the MPI-EVA-Manchester corpus is one of the cleanest and most standardized in CHILDES, it is not overly surprising that the standard Stanza tagger works reasonably well on it, but then underperforms on the test set that contains less systematic data (non-coded repetitions, non-standard situation descriptions, etc.). In comparison, the POS-only tagger modeled after Bunzeck and Diessel (2025) yields 5–7% worse performance, while the MLP tagger based on sentence embeddings performs dramatically worse.

It seems implausible that the remaining performance gap can be closed. Even with domain-specific training, certain distinctions cannot be made from text alone, due to factors such as i) archaic syntax in songs/nursery rhymes, ii) questions and statements that differ only in prosody, iii) run-on sentences coded as one utterance and therefore indistinguishable from complex utterances, iv) elliptical utterances and singular verbs that cannot be clearly separated between fragments and imperatives, etc. (for a fine-grained analysis of specific errors, see Appendix D.4).

Case study To further show the usefulness of our tagger, we use it to replicate the general methodology of Bunzeck and Diessel (2025) with a portion of the MPI-EVA-Manchester corpus (Lieven et al., 2009). In contrast to previous studies (Cameron-Faulkner et al., 2003; Cameron-Faulkner and Hickey, 2011; Cameron-Faulkner and Noble, 2013; Noble et al., 2018; Bunzeck and Diessel, 2025), however, we expand the scope considerably by including CS as well as CDS. While other studies were focused on the input only, Bunzeck and Diessel (2025) focused exclusively on the development of construction types in CDS because CS was deemed too hard to parse correctly with POS tags alone. With our UD-based tagger, this is not the case. Therefore, we are able to compare the development of CDS with that of CS and complement this previous line of research. For the present case study, we exclude formulaics and fragments to focus on clausal constructions only (in line with Bunzeck and Diessel, 2025).

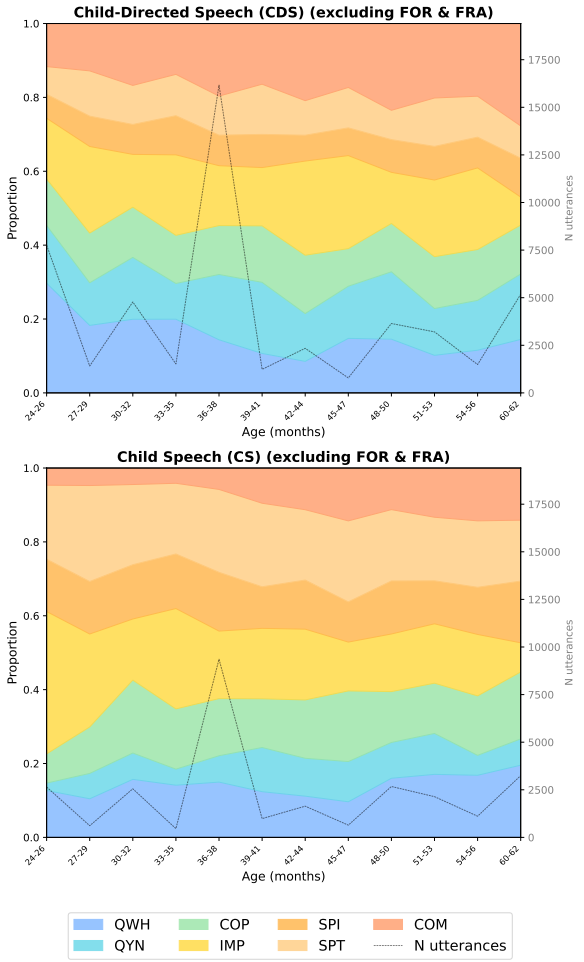


Figure 3: Development of relative construction proportions in CDS and CS, measured across 3-month bins. The dashed line shows the number of utterances annotated per bin.

Bunzeck and Diessel (2025) report two key findings: i) questions become less frequent in CDS across development, whereas ii) canonical SV(X) utterances become more frequent. Figure 3 shows the development of construction types in CDS and CS between two and five years of age. Previous findings for CDS can be generally confirmed: Questions (QWH, QYN) in the input become less frequent when comparing the earliest age group to subsequent ones, whereas SV(X) constructions become

more frequent. While standard subject–predicate utterances are fairly stable after an increase between 24 and 36 months, complex utterances see a steady increase in frequency between two and five years. Interestingly, these developments shift in CS. For the youngest age group, imperatives are most frequent. The proportion of SV(X) constructions (`SPI, SPT, COM`) increases with age, with an additional frequency shift from transitive, single-proposition sentences towards more complex utterances. Similarly, questions increase in frequency, with a clear preference for *wh*-questions. These tendencies signal a shift in children's communicative behavior: Whereas the use of imperatives already allows them to communicate effectively with one-word utterances, the more complex syntax of questions emerges later. Further, the semantically prototypical agent–patient relationships expressed in transitive clauses (`SPT`) are most frequent among early propositional utterances, whereas semantically opaque constructions (e.g., complex utterances, `COM`) become more frequent with increasing age. These results should be taken with caution as they rely on a limited subset of CHILDES. Still, this case study illustrates one type of novel analysis possible with CAIT. Many more kinds of analysis are possible, such as further investigations into dialogue-oriented aspects of language development (e.g., construction-level alignment between caregivers and children, cf. Sinclair and Fernández, 2021).

6 Conclusion & Future Work

We have shown that combining gold standard annotations (UD-CHILDES) with state-of-the-art parsers (`Stanza`, `SuPar`) enables reliable dependency parsing of informal, non-canonical registers like child speech and child-directed speech, which in turn can be leveraged for acquisition research, e.g., through the annotation of utterance-level constructions. With the release of our parser, POS and construction tagger we are making state-of-the-art NLP tools tailored to CHILDES data available to the acquisition community. Such UD resources are key for investigating structural properties, like dependency length, tree depth, and derived constructional measures, in light of recent arguments for the cognitive plausibility of dependency representations (Gibson, 2025), shedding further light on linguistic development and the input young learners receive. Extending this effort multilingually is a crucial next step: If even small UD-CHILDES treebanks become available in other languages, cross-lingual adaptation techniques could be employed to develop comparable parsing resources based on CAIT.

Acknowledgements

Francesca Padovani, Jaap Jumelet and Arianna Bisazza are supported by the project 'Polyglot Machines' (VI.Vidi.221C.009) funded by the Talent Programme of the Dutch Research Council (NWO). Nathan Schneider is supported by NSF award IIS-2144881. Bastian Bunzeck is supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – CRC-1646, project number 512393437, project A02.

References

Aviad Albert, Brian MacWhinney, Bracha Nir, and Shuly Wintner. 2013. The Hebrew CHILDES corpus: transcription and morphological analysis. *Language Resources and Evaluation*, 47(4):973–1005.

Javier Martín Arista, Ana Elvira Ojanguren López, and Sara Domínguez Barragán. 2025. Universal Dependencies annotation of Old English with spaCy and MobileBERT. Evaluation and perspectives. *Procesamiento del Lenguaje Natural*, 74:253–262.

Taylor Arnold. 2017. A tidy data model for natural language processing using cleanNLP. *The R Journal*, 9(2):248–267.

Giuseppe Attardi, Daniele Sartiano, and Maria Simi. 2021. Biaffine dependency and semantic graph parsing for enhanced Universal Dependencies. In *Proceedings of the 17th International Conference on Parsing Technologies and the IWPT 2021 Shared Task on Parsing into Enhanced Universal Dependencies (IWPT 2021)*, pages 184–188, Online. Association for Computational Linguistics.

Frederic Bechet, Alexis Nasr, and Benoit Favre. 2014. Adapting dependency parsing to spontaneous speech for open domain spoken language understanding. In *Interspeech*, pages 135–139.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Anouck Braggaar and Rob van der Goot. 2021. Challenges in annotating and parsing spoken, code-switched, Frisian-Dutch data. In *Proceedings of the Second Workshop on Domain Adaptation for NLP*, pages 50–58, Kyiv, Ukraine. Association for Computational Linguistics.

Martin D. S. Braine and Melissa Bowerman. 1976. Children's First Word Combinations. *Monographs of the Society for Research in Child Development*, 41(1).

Bastian Bunzeck and Holger Diessel. 2025. The richness of the stimulus: Constructional variation and development in child-directed speech. *First Language*, 45(2):152–176.

Bastian Bunzeck, Daniel Duran, and Sina Zarrieß. 2025. Do construction distributions shape formal language learning in German BabyLMs? In *Proceedings of the 29th Conference on Computational Natural Language Learning*, pages 169–186, Vienna, Austria. Association for Computational Linguistics.

Andrew Caines, Michael McCarthy, and Paula Buttery. 2017. Parsing transcripts of speech. In *Proceedings of the Workshop on Speech-Centric Natural Language Processing*, pages 27–36, Copenhagen, Denmark. Association for Computational Linguistics.

Thea Cameron-Faulkner and Tina Hickey. 2011. Form and function in Irish child directed speech. *Cognitive Linguistics*, 22(3):569–594.

Thea Cameron-Faulkner, Elena Lieven, and Michael Tomasello. 2003. A Construction Based Analysis of Child Directed Speech. *Cognitive Science*, 27(6):843–873.

Thea Cameron-Faulkner and Claire Noble. 2013. A comparison of book text and Child Directed Speech. *First Language*, 33(3):268–279.

Sam Davidson, Dian Yu, and Zhou Yu. 2019. Dependency parsing for spoken dialog systems. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1513–1519, Hong Kong, China. Association for Computational Linguistics.

Holger Diessel and Michael Tomasello. 2000. The Development of Relative Clauses in Spontaneous Child Speech. *Cognitive Linguistics*, 11(1-2):131–151.

Kaja Dobrovoljc. 2022. Spoken language treebanks in Universal Dependencies: an overview. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 1798–1806, Marseille, France. European Language Resources Association.

Kaja Dobrovoljc and Matej Martinc. 2018. Er ... well, it matters, right? on the role of data representations in spoken language dependency parsing. In *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 37–46, Brussels, Belgium. Association for Computational Linguistics.

Kaja Dobrovoljc and Joakim Nivre. 2016. The Universal Dependencies treebank of spoken Slovenian. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1566–1573, Portorož, Slovenia. European Language Resources Association (ELRA).

Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Edward Gibson. 2025. *Syntax: A Cognitive Approach*. The MIT Press, Cambridge, Massachusetts.

John J Godfrey, Edward C Holliman, and Jane McDaniel. 1992. Switchboard: Telephone speech corpus for research and development. In *[Proceedings] ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 517–520. IEEE.

Adele E. Goldberg. 2006. *Constructions at Work: The Nature of Generalization in Language*. Oxford Linguistics. Oxford University Press, Oxford.

Thomas Herbst and Thomas Hoffmann. 2024. *A Construction Grammar of the English Language: CASA – a Constructionist Approach to Syntactic Analysis*, volume 5 of *Cognitive Linguistics in Practice*. John Benjamins Publishing Company, Amsterdam.

Matthew Honnibal and Mark Johnson. 2015. An improved non-monotonic transition system for dependency parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1373–1378, Lisbon, Portugal. Association for Computational Linguistics.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength natural language processing in python.

Nicos Isaak. 2023. Blending dependency parsers with language models. In *ICAART (3)*, pages 813–820.

Da Ju, Hagen Blix, and Adina Williams. 2025. Domain Regeneration: How well do LLMs match syntactic properties of text domains? In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 2367–2388, Vienna, Austria. Association for Computational Linguistics.

Sylvain Kahane, Bernard Caron, Emmett Strickland, and Kim Gerdes. 2021. Annotation guidelines of UD and SUD treebanks for spoken corpora: A proposal. In *Proceedings of the 20th International Workshop on Treebanks and Linguistic Theories (TLT, SyntaxFest 2021)*, pages 35–47, Sofia, Bulgaria. Association for Computational Linguistics.

Liu Kaipeng and Wu Ling. 2026. Leveraging Lora Fine-Tuning and Knowledge Bases for Construction Identification. *Preprint*, arXiv:2601.13105.

Julie Kallini, Isabel Papadimitriou, Richard Futrell, Kyle Mahowald, and Christopher Potts. 2024. Mission: Impossible language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14691–14714, Bangkok, Thailand. Association for Computational Linguistics.

Julie Anne Legate and Charles D. Yang. 2002. Empirical re-assessment of stimulus poverty arguments. *The Linguistic Review*, 19(1-2):151–162.

Nicholas A. Lester, Steven Moran, Aylin C. Küntay, Shanley E.M. Allen, Barbara Pfeiler, and Sabine Stoll. 2022. Detecting structured repetition in child-surrounding speech: Evidence from maximally diverse languages. *Cognition*, 221:104986.

Jeffrey Lidz, Sandra Waxman, and Jennifer Freedman. 2003. What infants know about syntax but couldn't have learned: Experimental evidence for syntactic structure at 18 months. *Cognition*, 89(3):295–303.

Elena Lieven, Heike Behrens, Jennifer Speares, and Michael Tomasello. 2003. Early Syntactic Creativity: A Usage-Based Approach. *Journal of Child Language*, 30(2):333–370.

Elena Lieven, Dorothé Salomo, and Michael Tomasello. 2009. Two-Year-Old Children's Production of Multiword Utterances: A Usage-Based Analysis. *Cognitive Linguistics*, 20(3).

Elena V. M. Lieven, Julian M. Pine, and Gillian Baldwin. 1997. Lexically-Based Learning and Early Grammatical Development. *Journal of Child Language*, 24(1):187–219.

Houjun Liu and Brian MacWhinney. 2024. Morphosyntactic Analysis for CHILDES. *Language Development Research*, 4(1).

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Zoey Liu and Emily Prud'hommeaux. 2021. Dependency parsing evaluation for low-resource spontaneous speech. In *Proceedings of the Second Workshop on Domain Adaptation for NLP*, pages 156–165, Kyiv, Ukraine. Association for Computational Linguistics.

Zoey Liu and Emily Prud'hommeaux. 2023. Data-driven Parsing Evaluation for Child-Parent Interactions. *Transactions of the Association for Computational Linguistics*, 11:1734–1753.

Shannon Lubetich and Kenji Sagae. 2014. Data-driven measurement of child language development with simple syntactic templates. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2151–2160.

Brian MacWhinney. 2000. *The CHILDES Project: Tools for Analyzing Talk*, 3 edition. Lawrence Erlbaum Associates, Mahwah, NJ.

Brian MacWhinney. 2008. Enriching CHILDES for Morphosyntactic Analysis. In Heike Behrens, editor, *Corpora in Language Acquisition Research: History, Methods, Perspectives*, volume 6 of *Trends in Language Acquisition Research*, pages 165–197. John Benjamins Publishing Company, Amsterdam.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Peter Matthews. 1996. *Syntax*, reprinted edition. Cambridge Textbooks in Linguistics. Cambridge University Press, Cambridge.

Claire H. Noble, Thea Cameron-Faulkner, and Elena Lieven. 2018. Keeping it simple: The grammatical properties of shared book reading. *Journal of Child Language*, 45(3):753–766.

Elinor Ochs. 1979. Transcription as Theory. In Elinor Ochs and Bambi B. Schiefflen, editors, *Developmental Pragmatics*, pages 43–72. Academic Press, New York.

OpenAI, Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K. Arora, Yu Bai, Bowen Baker, Haiming Bao, Boaz Barak, Ally Bennett, Tyler Bertao, Nivedita Brett, Eugene Brevdo, Greg Brockman, Sebastien Bubeck, Che Chang, and 107 others. 2025. Gpt-oss-120b & gpt-oss-20b Model Card. *arXiv preprint*.

Lilja Øvrelid, Andre Kåsen, Kristin Hagen, Anders Nøklestad, Per Erik Solberg, and Janne Bondi Johannessen. 2018. The LIA treebank of spoken Norwegian dialects. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Lisa Pearl and Jon Sprouse. 2013. Computational models of acquisition for islands. *Experimental syntax and islands effects*, pages 109–131.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(85):2825–2830.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Eva Portelance and Masoud Jasbi. 2024. The Roles of Neural Networks in Language Acquisition. *Language and Linguistics Compass*, 18(6):e70001.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A python natural language processing toolkit for many human

languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3980–3990, Hong Kong, China. Association for Computational Linguistics.

Kenji Sagae, Eric Davis, Alon Lavie, Brian MacWhinney, and Shuly Wintner. 2007. High-accuracy annotation and parsing of CHILDES transcripts. In *Proceedings of the Workshop on Cognitive Aspects of Computational Language Acquisition*, pages 25–32, Prague, Czech Republic. Association for Computational Linguistics.

Kenji Sagae, Eric Davis, Alon Lavie, Brian MacWhinney, and Shuly Wintner. 2010. Morphosyntactic annotation of childes transcripts. *Journal of Child Language*, 37(3):705–729.

Kenji Sagae, Alon Lavie, and Brian MacWhinney. 2005. Automatic Measurement of Syntactic Development in Child Language. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 197–204, Ann Arbor, Michigan. Association for Computational Linguistics.

Kenji Sagae, Brian MacWhinney, and Alon Lavie. 2004. Adding Syntactic Annotations to Transcripts of Parent-Child Dialogs. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal. European Language Resources Association (ELRA).

Matthew Saxton. 2009. The Inevitability of Child Directed Speech. In Susan Foster-Cohen, editor, *Language Acquisition*, pages 62–86. Palgrave Macmillan UK, London.

Arabella Sinclair and Raquel Fernández. 2021. Construction coordination in first and second language acquisition. In *Proceedings of the 25th Workshop on the Semantics and Pragmatics of Dialogue - Full Papers*, pages 1–12, Potsdam, Germany. SEMDIAL.

Ida Szubert, Omri Abend, Nathan Schneider, Samuel Gibbon, Louis Mahon, Sharon Goldwater, and Mark Steedman. 2025. Cross-linguistically consistent semantic and syntactic annotation of child-directed speech. *Language Resources and Evaluation*, 59(2):727–776.

Michael Tomasello. 2000a. First Steps toward a Usage-Based Theory of Language Acquisition. *Cognitive Linguistics*, 11(1-2):61–82.

Michael Tomasello. 2000b. The item-based nature of children's early syntactic development. *Trends in Cognitive Sciences*, 4(4).

Michael Tomasello. 2003. *Constructing a Language: A Usage-Based Theory of Language Acquisition*. Harvard University Press.

Rob van der Goot, Ahmet Üstün, Alan Ramponi, Ibrahim Sharaf, and Barbara Plank. 2021. Massive choice, ample tasks (MaChAmp): A toolkit for multi-task learning in NLP. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 176–197, Online. Association for Computational Linguistics.

Annemarie van Dooren, Anouk Dieuleveut, Ailís Cournane, and Valentine Hacquard. 2022. Figuring out root and epistemic uses of modals: The role of the input. *Journal of Semantics*, 39(4):581–616.

Alex Warstadt and Samuel R Bowman. 2022. What artificial neural networks can tell us about human language acquisition. In *Algebraic structures in natural language*, pages 17–60. CRC Press.

Ralph Weischedel, Sameer Pradhan, Lance Ramshaw, Martha Palmer, Nianwen Xue, Mitchell Marcus, Ann Taylor, Craig Greenberg, Eduard Hovy, Robert Belvin, and Ann Houston. 2011. Ontonotes release 4.0. *LDC2011T03, Philadelphia, Penn.: Linguistic Data Consortium*, 17.

Leonie Weissweiler, Nina Böbel, Kirian Guiller, Santiago Herrera, Wesley Scivetti, Arthur Lorenzi, Nurit Melnik, Archna Bhatia, Hinrich Schütze, Lori Levin, Amir Zeldes, Joakim Nivre, William Croft, and Nathan Schneider. 2024. UCxn: Typologically informed annotation of constructions atop Universal Dependencies. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 16919–16932, Torino, Italia. ELRA and ICCL.

Xiulin Yang, Arianna Bisazza, Nathan Schneider, and Ethan Gotlieb Wilcox. 2026. A Unified Assessment of the Poverty of the Stimulus Argument for Neural Language Models. *arXiv preprint arXiv:2602.09992*.

Xiulin Yang, Zhuoxuan Ju, Lanni Bu, Zoey Liu, and Nathan Schneider. 2025. UD-English-CHILDES: A collected resource of gold and silver Universal Dependencies trees for child language interactions. In *Proceedings of the Eighth Workshop on Universal Dependencies (UDW, SyntaxFest 2025)*, pages 52–58, Ljubljana, Slovenia. Association for Computational Linguistics.

Guanghao You, Balthasar Bickel, Moritz M Daum, and Sabine Stoll. 2021. Child-directed speech is optimized for syntax-free semantic inference. *Scientific Reports*, 11(1):16527.

Daniel Zeman. 2021. Universal Dependencies: Principles and tools (tutorial). In *Proceedings of the Eighth Italian Conference on Computational Linguistics (CLiC-it 2021)*, pages 12–12, Milan, Italy. CEUR Workshop Proceedings.

Daniel Zeman, Martin Popel, Milan Straka, Jan Hajič, Joakim Nivre, Filip Ginter, Juhani Luotolahti, Sampo Pyysalo, Slav Petrov, Martin Potthast, Francis Tyers, Elena Badmaeva, Memduh Gokirmak, Anna Nedoluzhko, Silvie Cinková, Jan Hajič jr., Jaroslava Hlaváčová, Václava Kettnerová, Zdeňka Urešová, and 43 others. 2017. CoNLL 2017 shared task: Multilingual parsing from raw text to Universal Dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19, Vancouver, Canada. Association for Computational Linguistics.

Yu Zhang, Zhenghua Li, and Zhang Min. 2020. Efficient second-order TreeCRF for neural dependency parsing. In *Proceedings of ACL*, pages 3295–3305.

Limitations

Our study is accompanied by several limitations.

First, the CAIT toolkit assumes pre-tokenized input, as provided by UD-English-CHILDES. Researchers working with new, raw CHILDES transcripts would therefore first need a domain-specific tokenizer. While in this work we focus on the release of a POS tagger as a key component for morphosyntactic analysis, Stanza's (Qi et al., 2020)[10] training framework provides straightforward support for training all upstream pipeline components, including tokenization (`tokenize`), multi-word token expansion (`mwt`), and lemmatization (`lemma`), directly on the gold CoNLL-U annotations of UD-English-CHILDES. Such domain-specific components could then be combined with our parser and POS tagger to form a fully end-to-end pipeline for raw, untokenized transcripts covering the full range of UD annotation layers.

Second, for our novel, CHILDES-trained UD tagger we only have the UD-CHILDES treebank available as training. While it compiles and unifies existing resources, it is still relatively small, and more data would help in improving it further. Moreover, the official UD-CHILDES treebank (Yang et al., 2025) still contains some errors carried over from previous annotation efforts, which could inform future corrections. Some errors are likely to be not completely solvable, however, as CHILDES is not a monolithic corpus but a collection of many different corpora collected over a large time span (the English section ranges from the 1960s for the Brown corpus to the present day; the German section even contains transcripts that are over 100 years old). These corpora all adhere differently to transcription standards (e.g., pseudophonetic transcriptions like *de* instead of *the*), and are shaped by the affordances of the concrete theory employed or the goals of the study, reflecting Ochs's (1979) insight that transcription is never neutral but always a form of theoretical interpretation.

Lastly, for the construction annotation part, we concentrated on utterance-level constructions in the spirit of Tomasello (2000a); Cameron-Faulkner et al. (2003), but other construction annotations should now be possible as well. The UCxn layer (Weissweiler et al., 2024) for UD exists, and in combination with UD-CHILDES it could open up further avenues of construction identification, especially below the utterance-level. Nevertheless, our parser already outperforms off-the-shelf approaches and could be evaluated on even more existing, developmental data. For example, on manually annotated data from Diessel and Tomasello (2000), the CAIT parser finds 92.1% of relative clause constructions (537/583), whereas standard `Stanza` only finds 84.0% (490/583). Finally, it would also be interesting to compare our results to novel approaches to construction annotation, such as LoRA fine-tuning (Kaipeng and Ling, 2026).

A Additional Parser Evaluation Metrics

In Table 3 we present additional sentence-level evaluation metrics for the full set of parsers discussed in the main text. Exact Match (EM) measures the proportion of sentences for which all dependency arcs and labels are correctly predicted, while Unlabeled Exact Match (UEM) considers only the correctness of heads, disregarding labels. We report overall EM and UEM scores for dev and test set, without subdividing by child speech and CDS, as we observed no substantial differences between these subsets.

Table 4, instead, reports the LAS and UAS breakdown by speaker role on the test set for the CAIT parser and the `Stanza` off-the-shelf baseline. Both parsers perform better on CDS than on CS, which is consistent with the greater structural complexity in CDS and shorter, more fragmented utterances typical of child speech. The CAIT parser outperforms the baseline on both subsets, with the gap being more pronounced on CS.

Table 5 reports LAS and UAS scores broken down by sentence length for both CS and CDS, comparing `Stanza` off-the-shelf and CAIT parser. Sentence length is computed excluding punctuation

[10] https://stanfordnlp.github.io/stanza/training_and_evaluation.html

Parser	Backbone Model	Training Data	Development		Test Set	
			EM	UEM	EM	UEM
SuPar	**RoBERTa-large**	**gold data**	**81.17**	**89.74**	**78.22**	**87.61**
SuPar	RoBERTa-large	10k silver data + gold data	79.74	88.47	75.72	86.01
SuPar	RoBERTa-large	10k silver data → gold data	80.80	89.61	77.45	87.40
Stanza	BiLSTM	gold data	77.35	86.99	73.42	84.89
Stanza	RoBERTa-large	gold data	79.09	88.57	75.49	86.39
Stanza	RoBERTa-base	gold data	78.70	88.01	75.65	86.35
Stanza	Off-the-shelf	all EN-UD treebanks	63.78	77.77	64.07	76.90
spaCy	en_core_web_trf	OntoNotes 5	28.29	46.91	31.44	50.18
DiaParser	RoBERTa-large	gold data	74.66	85.85	71.24	83.65
DiaParser	RoBERTa-large	10k silver data + gold data	68.26	81.22	67.79	80.59
Machamp	RoBERTa-large	gold data	-	-	-	-
Machamp	RoBERTa-base	gold data	-	-	-	-

Table 3: Overall EM and UEM scores for the different parsers on the development and test sets. The EM and UEM evaluation script requires grammatical tree parses and thus cannot evaluate the generated parses by Machamp.

Parser	Backbone Model	Training Data	CS		CDS	
			LAS	UAS	LAS	UAS
SuPar	**RoBERTa-large**	**gold**	**91.30**	**93.96**	**93.48**	**95.63**
Stanza	Off-the-shelf	all EN-UD treebanks	83.33	87.81	86.73	90.31

Table 4: CS vs. CDS breakdown on the test set for the best-performing trained parser (SuPar RoBERTa-large, gold) and the Stanza off-the-shelf baseline.

tokens, eventually yielding four bins: sentences of up to 3 tokens, between 4 and 6, between 7 and 10, and more than 10 tokens. The sentence counts per bin reflect well-known distributional properties of the two portions of CHILDES: CS is dominated by very short utterances, with the $\leq$3 and 4–6 bins together accounting for the large majority of sentences, while CDS exhibits a slightly higher proportion of sentences falling in the longer bins compared to CS. The CAIT parser consistently outperforms Stanza off-the-shelf across all bins and both speakers, with the largest gains observed on the shortest sentences in CDS, where CAIT reaches a LAS of 94.08 against Stanza's 85.09. For CS, CAIT achieves its highest LAS on the 4–6 bin (92.83), while performance drops for longer sentences ($>$10), likely reflecting the scarcity of long utterances in child speech during training and the increased structural complexity they entail. Both parsers perform better on CDS than on CS across all length bins, consistent with the findings reported in Table 4.

B Gold and predicted trees to visualize errors

In this section we present graphical comparisons of selected dependency trees in their gold-standard annotation and in the erroneous structures predicted by the parsers. The error category to which each example belongs is indicated in the caption and corresponds to the paragraphs discussed in the main body of the paper. The tree visualizations are intended to facilitate interpretation of the error patterns and to illustrate the structural challenges involved in accurately parsing conversational, developmental language such as CHILDES.

C Error patterns for CAIT and off-the-shelf Stanza

Figures 10, 11, and 12 present the confusion matrices used to analyze and compare the error patterns of the two parsers: the off-the-shelf Stanza model and the CAIT parser.

Figures 10 and 11 display the normalized error distributions for the off-the-shelf Stanza parser and the CAIT parser, respectively. In both matrices, darker shades indicate higher misclassification rates, highlighting the most frequent confusion

Sen. Length	CS					CDS				
	# of Sents	Stanza off-the-shelf		CAIT parser		# of Sents	Stanza off-the-shelf		CAIT parser	
		LAS	UAS	LAS	UAS		LAS	UAS	LAS	UAS
≤3	2546	81.03	86.51	90.11	93.61	1396	85.09	87.79	94.08	95.75
4–6	1709	84.71	88.73	92.83	95.16	1740	88.63	92.10	94.21	96.43
7–10	574	84.37	88.15	92.28	94.60	987	86.70	90.68	93.10	95.51
>10	163	83.68	87.93	89.16	91.57	476	85.20	89.00	92.02	94.05

Table 5: LAS and UAS by sentence length (punctuation excluded) for Child Speech (CS) and Child Directed Speech (CDS), comparing the `Stanza` off-the-shelf baseline and `SuPar RoBERTa-large`. # of Sents indicates the number of sentences in each bin.

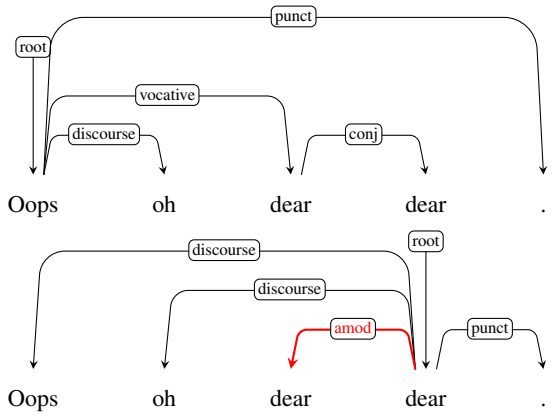


Figure 4: Gold (top) and predicted by the off-the-shelf parser (bottom) dependency labels and relations (`conj`).

patterns between gold and predicted dependency labels.

Figure 12 shows the difference in error rates between the two parsers (CAIT minus off-the-shelf `Stanza`). Positive values (red cells) indicate cases where CAIT produces more errors than the off-the-shelf model, while negative values (blue cells) indicate cases where the off-the-shelf model performs worse.

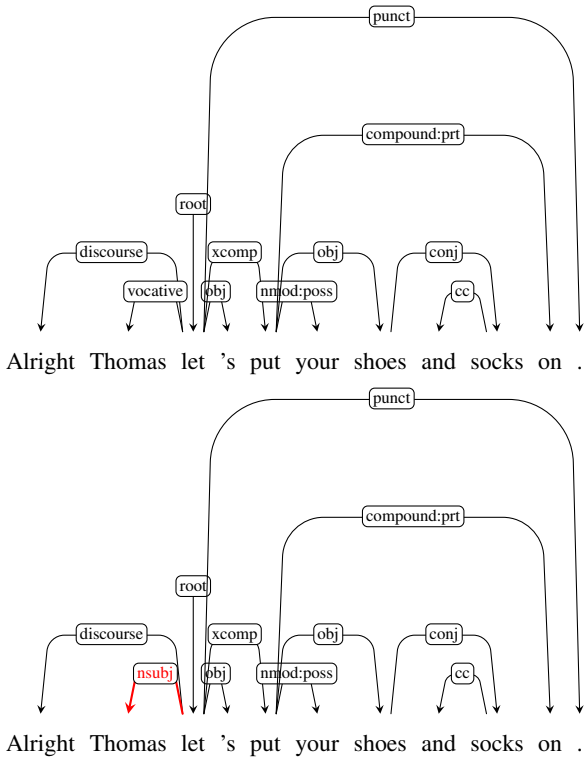


Figure 5: Gold (top) and predicted by the off-the-shelf parser (bottom) dependency labels and relations. `vocative`.

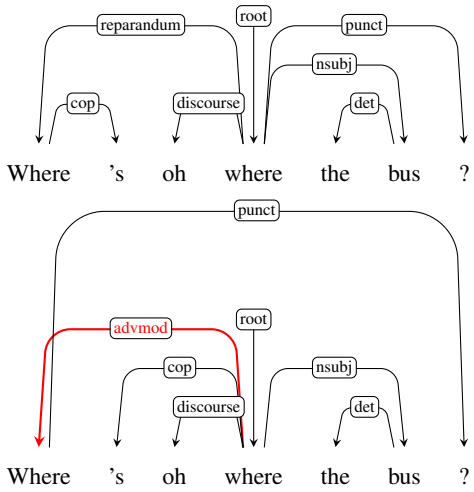


Figure 6: Gold (top) and predicted by the off-the-shelf parser (bottom) dependency labels and relations. `reparandum`.

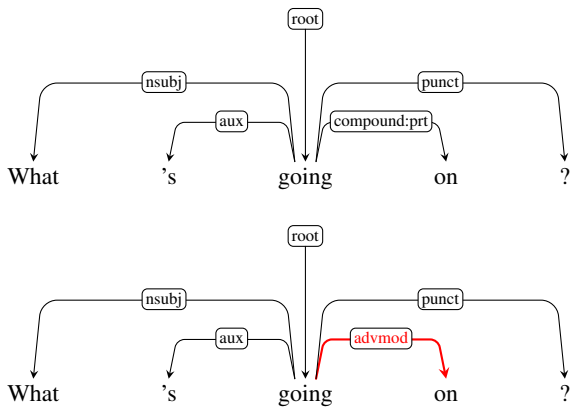


Figure 7: Gold (top) and predicted by CAIT parser (bottom) dependency labels and relations (`compound:prt` & `advmod`).

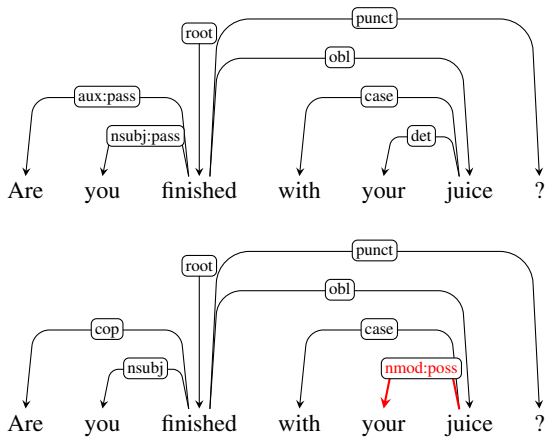


Figure 8: Gold (top) and predicted by CAIT (bottom) dependency labels and relations (`det`).

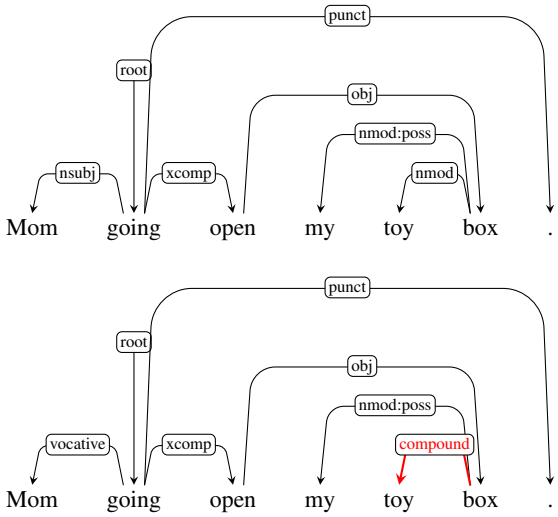


Figure 9: Gold (top) and predicted by CAIT parser (bottom) dependency labels and relations (`nmod`).

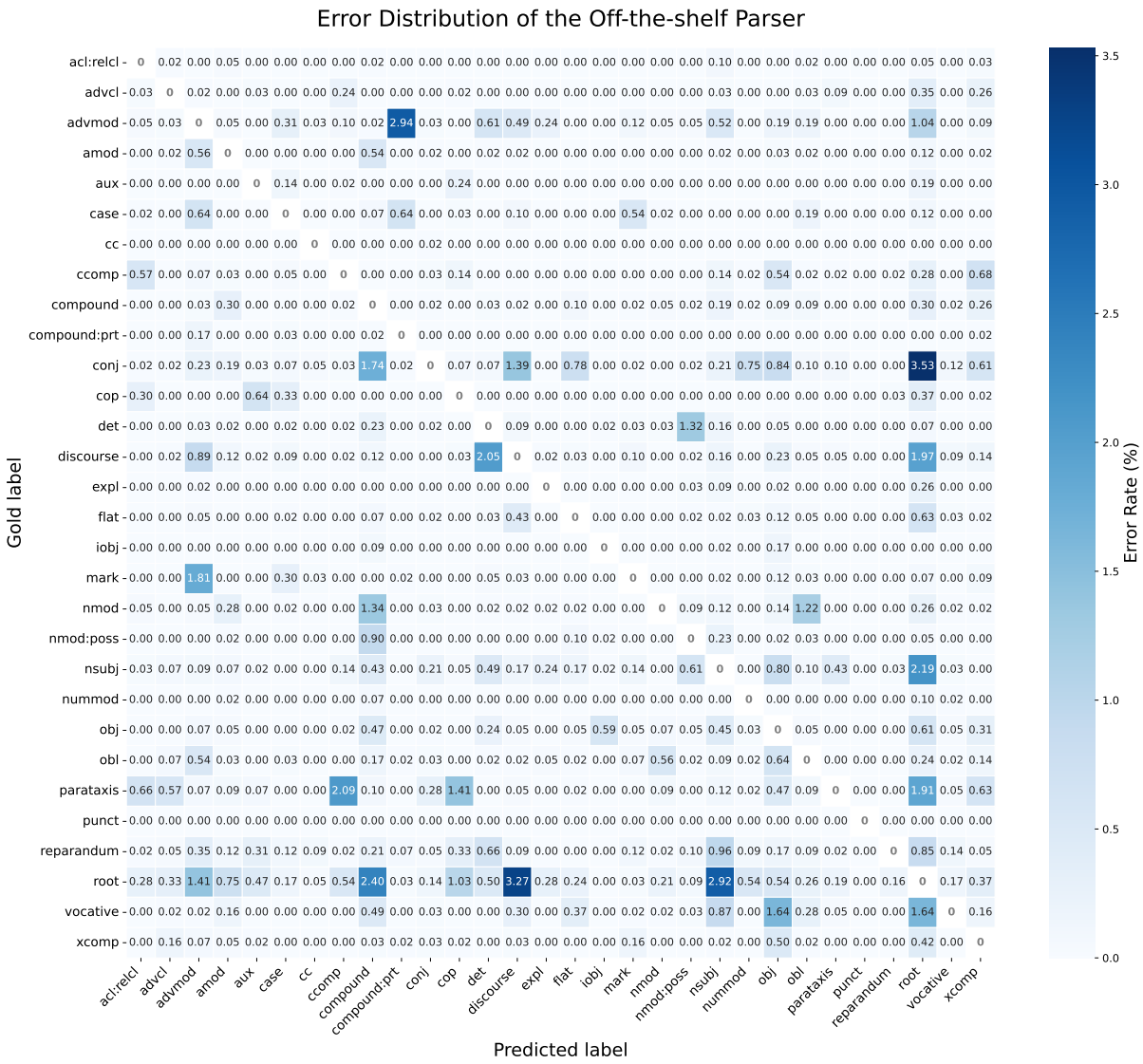


Figure 10: Confusion matrix representing the error patterns of the off-the-shelf Stanza Parser.

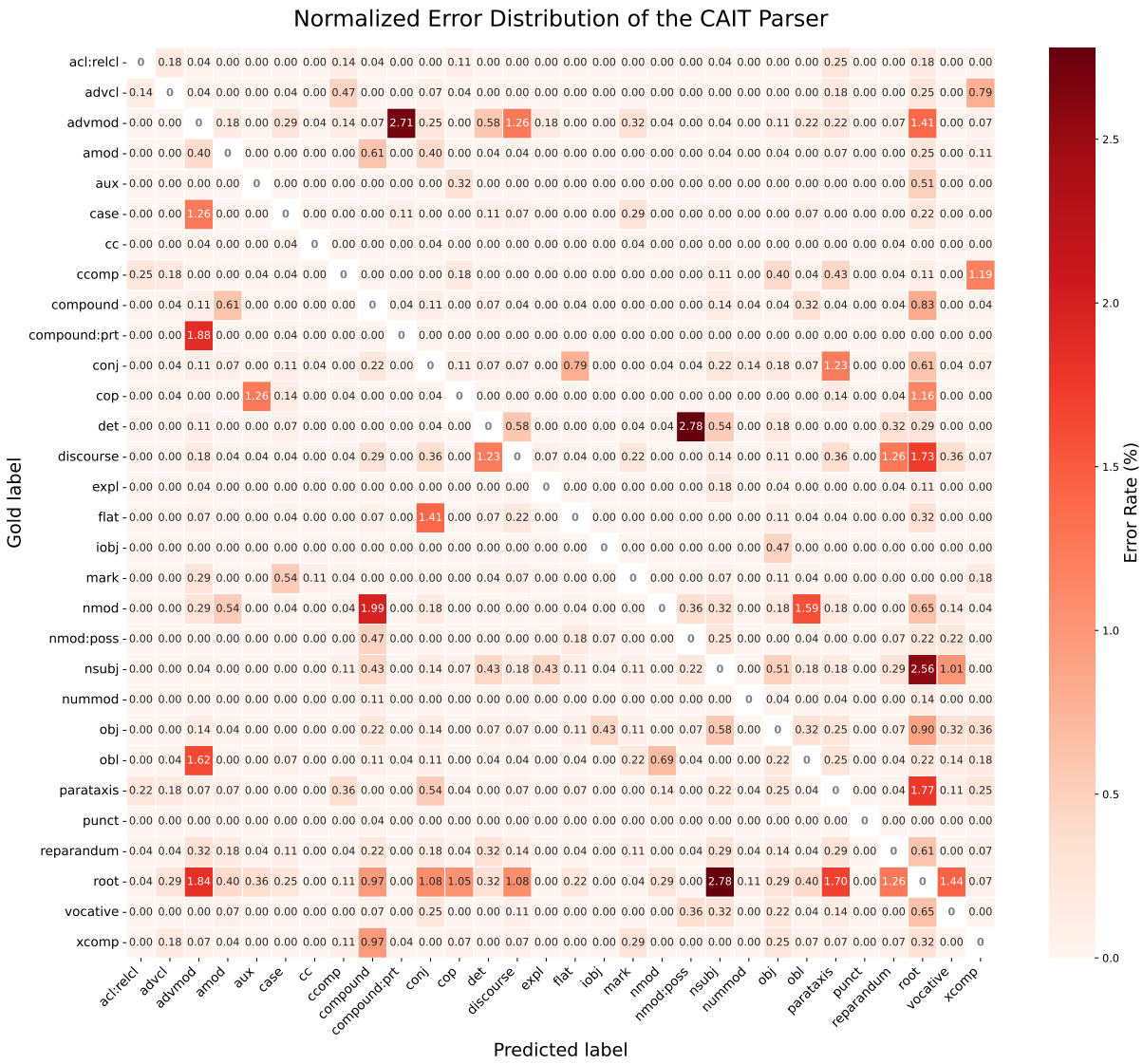


Figure 11: Confusion matrix representing the error patterns of the CAIT Parser.

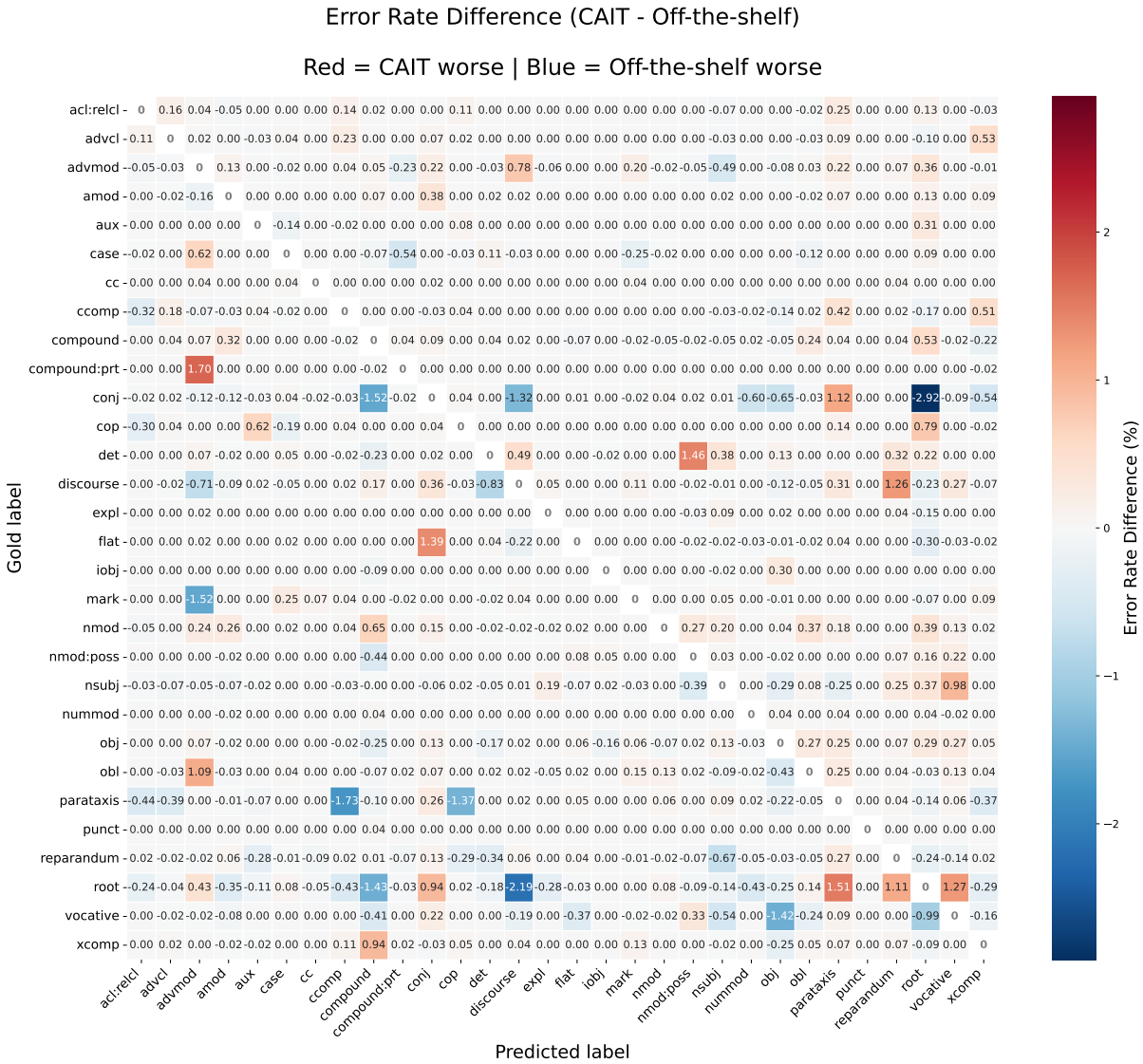


Figure 12: Confusion matrix representing the delta of the error rated between the CAIT parser and the off-the-shelf `Stanza` parser.

D Construction annotation

D.1 Annotation guidelines

We devise standard construction categories in line with comparable efforts for English (Cameron-Faulkner et al., 2003; Cameron-Faulkner and Noble, 2013; Bunzeck and Diessel, 2025), and assign one of these categories to each and every utterance. Notably, the notion of *construction* employed here ranges somewhere between the traditional grammatical notion of a construction as the "syntactic characterisation of a sentence" (Matthews, 1996, 1) to the more CxG conception of form-meaning pairings that occur with sufficient frequency (Goldberg, 2006, 5) (cf. Herbst and Hoffmann, 2024, 4–11, for a comprehensive discussion of constructionhood). In sum, our *constructions* are syntactically delineated, sufficiently frequent templates that serve a prototypical communicative function. We devise the following nine categories:

- `FOR`: Fixed formulaic expressions (social routines): *hello*, *thank you*
- `FRA`: Fragments without finite verb: *Mummy*, *a baseball*
- `QWH`: Wh-questions (fronted interrogative): *what is that?*, *where are you going?*
- `QYN`: Yes/no questions (aux inversion, to be confirmed or denied): *can you hear me?*, *is it honey?*
- `COP`: Copula constructions: *it's a kite*, *the witch was green*
- `IMP`: Imperatives: *look*, *come here*, *let's go*
- `SPI`: Subject-predicate, verb used intransitively: *she laughed*, *I'm going*
- `SPT`: Subject-predicate, verb used transitively: *I love you*, *she read the book*
- `COM`: Complex (multiple verbs/clauses, subordination or coordination pattern): *I want to go and play*, *the dog went home before it started raining*

Importantly, in line with Cameron-Faulkner et al. (2003), we ignore utterance-final tag questions and remove them programmatically before the tagging process (*that's good isn't it?* → *that's good.*).

FOR The formulaic category is only for fixed social routine expressions. The utterance must exactly match the patterns in our tagger (ignoring punctuation and case), which consist of greetings, farewells, politeness formulaics, interjections and blessings. For a complete list please consider the tagging script in the GitHub repository. Importantly, this category requires an exact match, any additional content changes the category (*oops* → `FOR`, but *oops I dropped it* → `SPT`).

FRA Fragments are utterances that lack a finite verb or are otherwise syntactically incomplete. This includes single words (*Mummy*, *ball*), response particles (*yeah*, *mhm*), bare noun phrases (*the big dog*), prepositional phrases (*in there*), adjective phrases (*very pretty*), incomplete copulas (*she's.*, exclamatives (*what a mess!*), bare participles (*running*), otherwise incomplete clauses or instances of speakers trailing off (*because I ...*).

QWH This category refers to questions introduced by a fronted interrogative pronoun (*who*, *what*, *where*, *when*, *why*, *how*, *which*, *whose*) in the main clause. The wh-word is at/near the start (before any auxiliary, ignoring additional communicatives beforehand). Although questions usually have a question mark, which is also recommended by the CHILDES guidelines, some corpora do not feature them. If it is clearly inferable from context or syntax that an utterance is meant as a question, we still do annotate it as such (and also do not strictly enforce any question mark criterion for the tagger).

QYN This category refers to questions with auxiliary inversion, typically expecting a yes/no answer. Crucially, we follow Cameron-Faulkner et al. (2003) in not counting sentences with declarative syntax but including a question mark (possibly indicating rising intonation) as yes/no-questions. Here structure takes precedence, although we want to stress that this remains debatable and should be critically reconsidered when adopting our construction scheme for further studies.

COP Copula utterances are utterances where the main predicate is a form of *be* functioning as a copula (linking verb). This includes i) subject + be + predicate (*it's a kite*), ii) existentials (*there is a dog* and iii) passives/resultatives with *be* (*it's broken*, *the store is closed*). Excluded are progressives like *she is running* where a form of *be* is an auxiliary and not a copula, and two combined copula clauses, which are considered complex utterances (*it is not hot because it was rainy yesterday*).

IMP Imperatives are commands or requests, typically verb-initial. This includes bare verb commands like *look*, typical subjectless phrases like *come here* and negative imperatives such as *don't touch it*. As mentioned, imperatives typically occur without an overt subject. Exceptions are i) hortatives or *let's*-constructions (e.g., *let's go*), and ii) emphatic *you* imperatives (like *you put that down*),

which are frequent in child/child-directed language and also subsumed under imperatives due to their communicative function.

SPI, SPT and COM Structurally and functionally these categories are fairly similar, and Bunzeck and Diessel (2025) even subsume them under one common SV(X) category. In line with Cameron-Faulkner et al. (2003), however, we distinguish them for the sake of our analysis. All three categories require a finite verb, and their finer subdistinction depends on i) transitivity (presence of a direct object) and ii) clause complexity (single vs. multiple predicates).

SPI refers to declaratives with intransitive verbs (no direct object) like *she laughed*, including progressives (*she is running*) and elliptical auxiliaries (e.g., *we did*).

SPT refers to declaratives with transitive verbs (including direct object). This includes prototypical examples like *she read the book* or *I love you*. Furthermore, we include sentences that include a control verb with an infinitive complement (*she needs to eat*) and utterances with objects in embedded infinitives (like *I want to read that book*), as these sentences semantically still transmit a singular proposition and are structurally similar to, e.g., future constructions (cf. *I will read the book* vs. *I am going to read the book* vs. *I want to read the book*).

COM then refers to utterances with multiple independent verbs or complex clausal structure. Generally, this refers to declaratives, but in the CHILDES data also combinations of other constructions are observable (like syntactic amalgams between questions and declaratives or imperatives and declaratives). We do not analyze these structures further, but subsume them under the complex category. Prototypically, this includes coordination (*I want to go to the store and I will buy some milk*) or subordination patterns (*I was sad because he hit me*). A general rule-of-thumb is that if two or more independent verbal predicates can be identified, then the utterance is a complex one.

X Finally, we used X as an exclusion category for a very minor share of utterances. We use it only as a last resort and annotate utterances as X if, and only if they i) are completely unintelligible, ii) contain only *xxx*, similar transcription markers, or only metalinguistic information, or iii) adhere to non-standard annotation that have only been used for highly specific, individual corpora.

Decision procedure for annotation In general, the decision hierarchy then looks as follows:

1. Is it an exact formulaic match? → FOR
2. Does it lack a finite verb? → FRA
3. Is it a question? Fronted wh-word in main clause? → QWH
4. Is it a question? Aux inversion? → QYN
5. Is the main predicate a copula (*be* + ADJ/NP/-passive)? → COP
6. Is it a command/request (verb-initial, *let's*, *you* + verb)? → IMP
7. Does it have multiple verbs/clauses (coordination, subordination, etc.)? → COM
8. Does it have a direct object (includes control verb + infinitive)? → SPT
9. Otherwise → SPI

D.2 Decision procedure in tagger

For our UD-based tagger, we alter the previously mentioned decision procedure to a clearly delineated, progressively less strict matching algorithm:

1. FOR → String match against formulaic patterns
2. FRA → Incomplete copula (*she's*) or exclamative (*what a day*)
3. QYN → Auxiliary inversion + question mark
4. QWH → Fronted wh-word in main clause
5. COM → Complex clausal relations (ccomp, advcl, acl, parataxis) or conjoined verbs
6. COP → Copula relation, existentials, *be* + participle
7. FRA → No verb root or standalone participle
8. IMP → Imperative mood, *you* + verb, bare verb
9. SPI → Auxiliary root with subject (elliptical)
10. SPT → Direct object or control verb + xcomp
11. SPI → Default for remaining verbs
12. FRA → Final fallback

D.3 Results by speaker type (CS vs. CDS)

To further show the robustness of our tagger, we report accuracy scores divided by child-directed and child speech.

Table 6 shows the accuracies for the dev set (991 CDS utterances, 1096 CD utterances). While general accuracy tendencies hold, it can also be said that all taggers perform slightly better on CS than CDS.

Table 7 shows the accuracies for the test set (515 CDS utterances, 474 CS utterances).

Interestingly, here the picture is slightly reversed. The taggers perform better on CDS than on CS. This suggests some remaining variation in the underlying data. As CHILDES corpora are highly

Tagger	CDS	CS	CDS-CS Δ
CAIT	91.26%	93.46%	−2.2%
Standard Stanza	88.74%	90.93%	−2.2%
POS-only	85.44%	86.08%	−0.6%
MLP	68.74%	71.73%	−3.0%

Table 6: Accuracy on dev data, separated by CDS vs. CS.

Tagger	CDS	CS	CDS-CS Δ
CAIT	93.04%	91.15%	+1.9%
Standard Stanza	92.03%	90.51%	+1.5%
POS-only	88.50%	86.68%	+1.8%

Table 7: Tagger accuracy on test data, separated by CDS vs. CS.

diverse and annotations vary widely between different datasets, this is not overly surprising. Despite this variation, the delta between CDS and CS accuracies is approximately 2% for all data sets, which we deem robust enough for our purposes.

D.4 Detailed performance analysis

To get a more detailed overview of the advantages and shortcomings of the four individual construction taggers that we evaluated, we provide category-wise accuracies in Table 8 and tagger-wise confusion matrices in Figure 13. We also report qualitative impressions that we gained from a comparison of misclassifications.

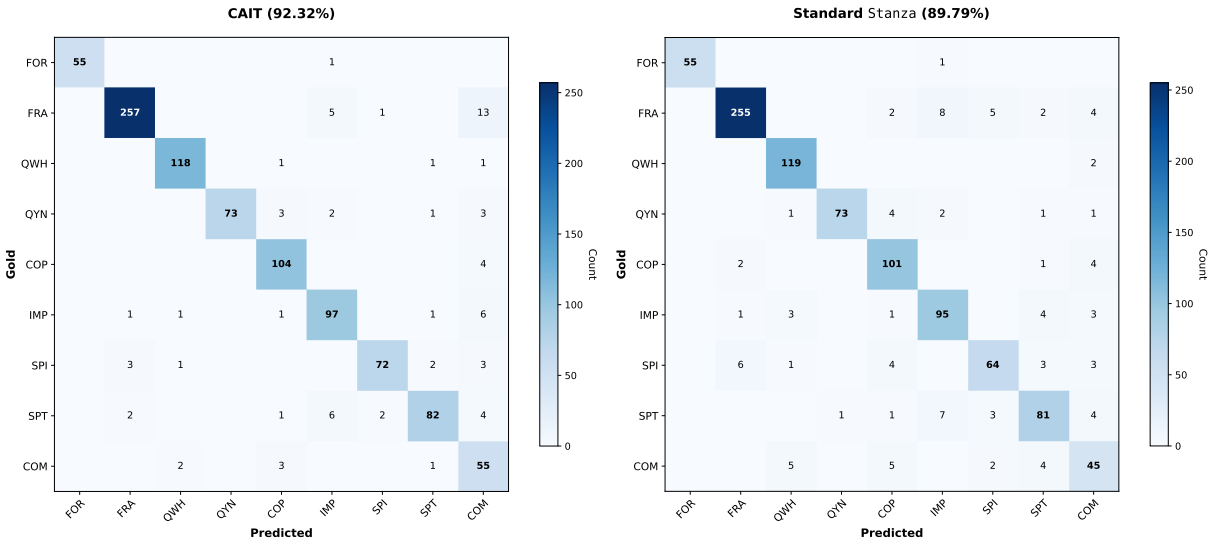
CAIT Across test and dev data, the CAIT-based tagger reaches the highest accuracy scores. The greatest improvements come from the SV(X) categories: For SPI, we observe an improvement of almost 10%. The standard parser struggles with intransitive verb detection in child speech, often misclassifying SPI as COP, FRA, or SPT. CS contains many short, contextual intransitives (*I'm going*, *she's sleeping*) that the domain-trained parser is probably better suited to. Also for complex utterances, we notice a tremendous improvement (over 16%). Here, the standard parser frequently fails to identify relative clauses (acl:relcl) and complement clauses (ccomp) in CHILDES data. As the dialogues often contain complex structures wrapped in simple vocabulary (*that's what she said*), the mismatch between parser training data and target data between the standard Stanza parser and our CHILDES dialogues could be just too strong for Stanza to correctly recognize such structures in such simplistic and not very "wordy" contexts.

In general, as Figure 13a shows, there are not many confusion clusters in the CAIT-tagged data, misclassifications are spread rather evenly across the data. The only exception are fragments misclassified as complex utterances, which form a smallish cluster. However, this might be caused by the fact that some fragments are still pretty long because they contain (unrelated) words that are strung along by the interlocutors. As dependency parsers (by design) want to assign some kind of relation, it could be that these telegraphic utterances still fulfill the criterion for complex utterances in our tagging scheme.

Stanza off-the-shelf Although it slightly underperforms the CAIT-based tagger, the off-the-shelf Stanza tagger still reaches impressive performance across the majority of categories (except the SV(X) categories). A closer look at the confusion matrix (Figure 13b) yields that especially fragments are misclassified, also in more different ways than with the CAIT-based parser. Again, this is probably caused by the register mismatch in training data. The standard Stanza parser is trained on written language and long sentences, whereas spoken language features shorter and more fragmented, hesitation- and false-start-heavy data. These specific features of spoken language are then wrongly parsed and provide incorrect input to our tagging procedure.

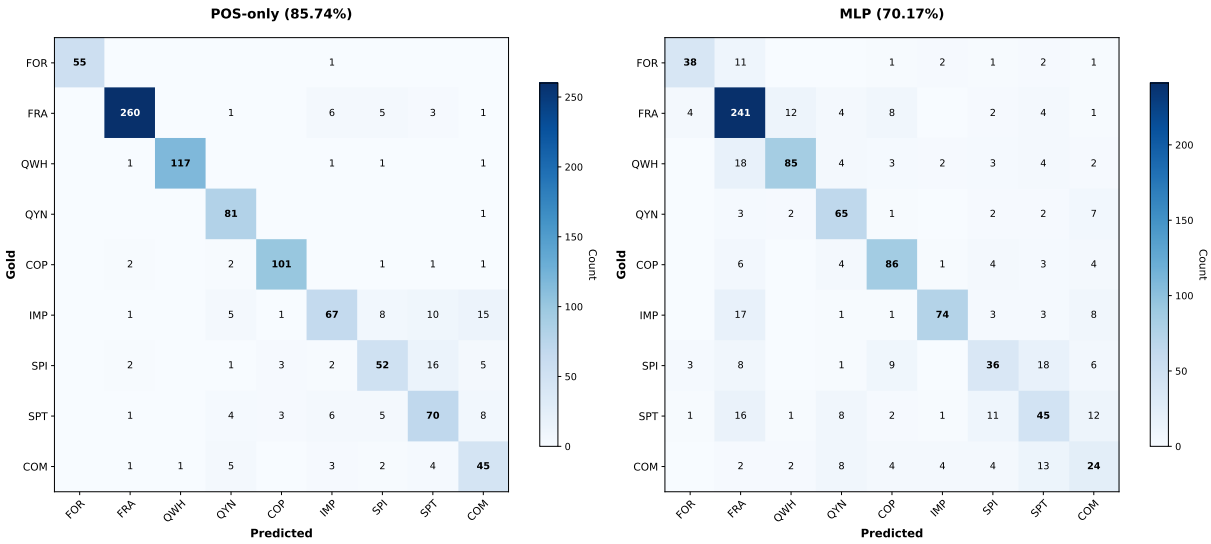
POS-based Interestingly, the POS-based tagger outperforms both UD-based taggers for two categories – fragments and yes/no-questions. On the one hand, this could be caused by the templatic nature of the POS-based tagger, which works well on the structurally rather uniform yes/no-questions. On the other hand, a closer look at the confusion matrix in Figure 13c also reveals that POS-based tagger has the best recall for both fragments and yes/no-questions, but is less precise. Both categories also contain erroneous classifications that are spread out across the gold standard categories. This differs from the UD-based taggers, which are worse in terms of recall, but provide a better precision.

In contrast to FRA and QYN, the POS-based tagger works worse for IMP and the SV(X) categories. Imperatives are drastically under-detected. Without dependency relations, imperative detection relies on verb-initial heuristics. Many imperatives are misclassified because the POS-based tagger cannot identify the imperative mood or distinguish addressee *you* from subject *you*. Also, without the different types of obj dependency labels, object de-



(a) Confusion matrix for CAIT-based tagger

(b) Confusion matrix, Stanza-based tagger



(c) Confusion matrix, POS-based tagger

(d) Confusion matrix, embedding-based tagger

Figure 13: Confusion matrices for all construction taggers on test set.

Category	CAIT	Standard Stanza	POS-only	MLP with embeddings
FOR	**98.2%**	**98.2%**	**98.2%**	67.9%
FRA	93.1%	92.4%	**94.2%**	87.3%
QWH	97.5%	**98.3%**	96.7%	70.2%
QYN	89.0%	89.0%	**98.8%**	79.3%
COP	**96.3%**	93.5%	93.5%	79.6%
IMP	**90.7%**	88.8%	62.6%	69.2%
SPI	**88.9%**	79.0%	64.2%	44.4%
SPT	**84.5%**	83.5%	72.2%	46.4%
COM	**90.2%**	73.8%	73.8%	39.3%

Table 8: Detailed construction-tagging evaluation results for all taggers.

tection relies on word order heuristics. These fail for indirect objects, PP complements, and elided objects. Finally, the low accuracy on complex utterances is also caused by a lack of dependency information (subordinate clauses simply cannot be identified without `ccomp`, `advcl`, or `acl:relcl` labels).

Embedding-based The MLP classifier uses `sentence-transformers` (Reimers and Gurevych, 2019) with the `all-mpnet-base-v2` embedding model to encode utterances into 768-dimensional vectors. On top of that, we train a multilayer-perceptron classifier with `scikit-learn` (Pedregosa et al., 2011). In comparison to the other classifiers, it underperforms dramatically across all categories, with the best accuracy of 87.3% reached in the fragment category. The confusion matrix (Figure 13d) shows that misclassifications are frequent across all categories. However, this is not overly surprising, as embeddings are most useful for representing semantic features, whereas our categories are mostly syntactically defined. The 22% between embedding and UD-based parsing demonstrates that such kinds of syntactic classification require explicit structural analysis. Again, this becomes most apparent for the SV(X) categories, where `SPI` and `SPT` are frequently confused. Similarly, complex utterances are classified into the whole spectrum of construction types (except formulaics), showing that embeddings hardly represent clause structure (for example, *I think he's happy* and *he's happy* should embed similarly, despite their important structural differences).

D.5 Synthetic data comparison

To further assess tagger generalization, we evaluated on a synthetic dataset of 800 programmatically generated utterances (100 per category, excluding FOR). We generated these utterances by prompting `gpt-oss-20b` (OpenAI et al., 2025) to produce 100 sentences that fit a short category description and contain language that is adequate for children. They were originally used for creating a very first tagging schema, but then abandoned in favor of the `dev` data set. For the sake of comparison we decided to assess tagger performance on the synthetic data after the development of all taggers was finished.

Tagger	Accuracy
CAIT	92.00%
Standard Stanza	91.87%
POS-only	**92.37%**
MLP	55.75%

Table 9: Tagger evaluation results for synthetic data.

Table 9 shows a somewhat different picture compared to the naturally occurring datasets. The POS-based tagger outperforms the UD-based taggers (by a minimal margin). This might be due to the syntactic uniformity of synthetic data, which deviates quite drastically from naturally occurring data (cf. Ju et al., 2025). As there is little variation and deviation in its patterns, the purely pattern-driven POS tagger has no problems identifying the constructions in the correct way. Interestingly, the embedding classifiers collapse completely. It is plausible that the purely distributional patterns that they learn from CHILDES data do not generalize to artificially constructed data.

D.6 Further results for case study

Figure 14 displays the mean proportion of construction types across the complete

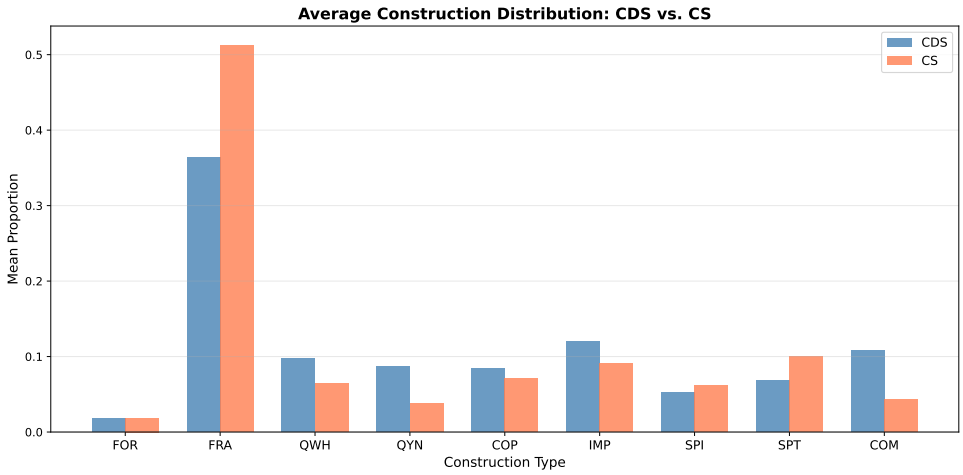


Figure 14: Mean proportion of construction types across `MPI-EVA-Manchester` corpus sample.

`MPI-EVA-Manchester` corpus sample, regardless of age. Here, the contrasts between CDS and child speech become most visible. Fragments and (in)transitive propositional utterances are more frequent in child speech than in CDS; whereas questions, copulas, imperatives and complex utterances occur more in CDS than in child speech. From a syntactic viewpoint, this contrast is somewhat expected, as the sample is taken from young children whose mental constructicon is still more item-based and less abstract/schematic. On the semantic/discourse-pragmatic level, the interaction of these categories should also be considered. Children are not linguistic mirrors, but engage in conversation with their caregivers. Questions in CDS beget answer in child speech, e.g., in the form of one-word fragments or simple propositions.

Figure 15 shows the development of construction type frequency across time, divided by CDS vs. CS. The most interesting difference here can be found in the development of wh-question frequency. Wh-questions are the only construction type where the CDS and the child speech development curves are almost mirrored. From an interaction perspective, however, this is not overly surprising. In early acquisition, caregivers try to elicit answers from children and motivate communicative behavior. This is simple with wh-questions, which can already be answered with one-word fragments (and also actively teach novel vocabulary). When this first bootstrapping period is done, the initiative in communication shifts to children who want to know things about the world they live in and actively seek out new words themselves by requesting them.

Finally, Figure 16 shows the proportions also portrayed in Figure 3, but including syntactic fragments and formulaics. In CDS, the frequency of fragments is not dramatically impacted by child age, and formulaics stay highly infrequent across development. In contrast, CS features almost 80% fragments in the first age group, which then steadily decreases towards the 40–50% range. Formulaics and social routines become frequent around 4–5 years of age.

Figure 15: Development of individual construction-type frequencies across dataset, separated by speaker (caregiver vs. child).

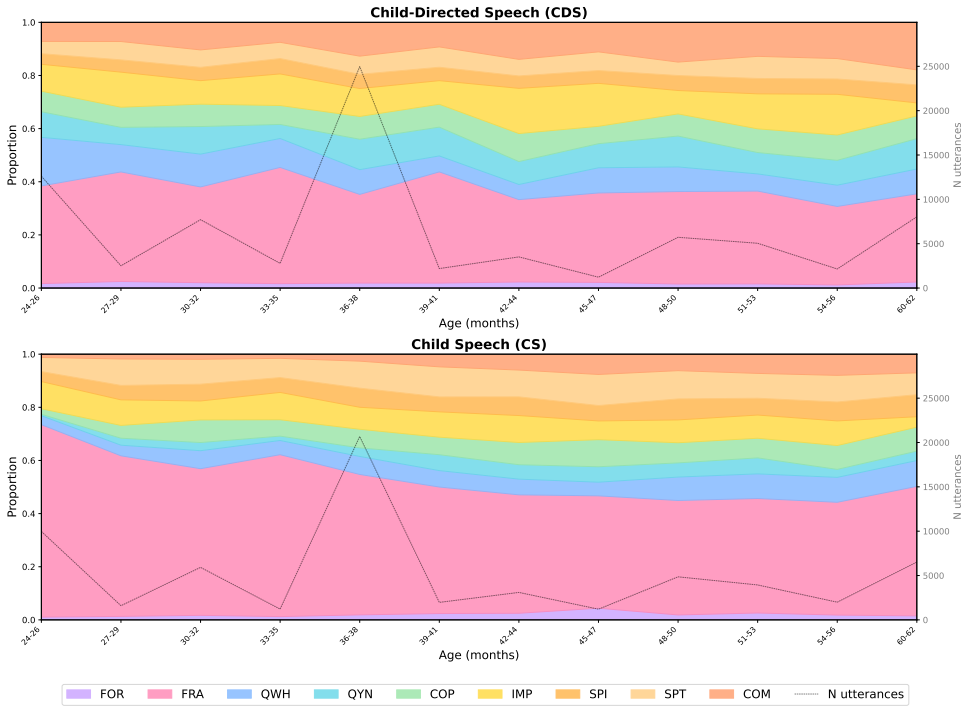


Figure 16: Development of construction type frequencies in child-directed and child speech (including FOR and FRA).