

From Paraphrasing to Semantic Parsing: Unsupervised Semantic Parsing via Synchronous Semantic Decoding

Shan Wu^{1,3}, Bo Chen¹, Chunlei Xin^{1,3}, Xianpei Han^{1,2,*}, Le Sun^{1,2,*},
Weipeng Zhang⁴, Jiansong Chen⁴, Fan Yang⁴, Xunliang Cai⁴

¹Chinese Information Processing Laboratory ²State Key Laboratory of Computer Science
Institute of Software, Chinese Academy of Sciences, Beijing, China

³University of Chinese Academy of Sciences, Beijing, China ⁴Meituan
{wushan2018, chenbo, xianpei, sunle}@iscas.ac.cn,

xinchunlei20@mailsucas.ac.cn,

{zhangweipeng02, chenjiansong, yangfan79, caixunliang}@meituan.com

Abstract

Semantic parsing is challenging due to the structure gap and the semantic gap between utterances and logical forms. In this paper, we propose an unsupervised semantic parsing method – Synchronous Semantic Decoding (SSD), which can simultaneously resolve the semantic gap and the structure gap by jointly leveraging paraphrasing and grammar-constrained decoding. Specifically, we reformulate semantic parsing as a constrained paraphrasing problem: given an utterance, our model synchronously generates its canonical utterance¹ and meaning representation. During synchronous decoding: the utterance paraphrasing is constrained by the structure of the logical form, therefore the canonical utterance can be paraphrased controlledly; the semantic decoding is guided by the semantics of the canonical utterance, therefore its logical form can be generated unsupervisedly. Experimental results show that SSD is a promising approach and can achieve competitive unsupervised semantic parsing performance on multiple datasets.

1 Introduction

Semantic parsing aims to translate natural language utterances to their formal meaning representations, such as lambda calculus (Zettlemoyer and Collins, 2005; Wong and Mooney, 2007), FunQL (Kate et al., 2005; Lu et al., 2008), and SQL queries. Currently, most neural semantic parsers (Dong and Lapata, 2016; Chen et al., 2018b; Zhao et al., 2020; Shao et al., 2020) model semantic parsing as a sequence to sequence translation task via encoder-decoder framework.

*Corresponding Author

¹Canonical utterances are pseudo-language representations of logical forms, which have the synchronous structure of logical forms. (Berant and Liang, 2014; Xiao et al., 2016; Su and Yan, 2017; Cao et al., 2020)

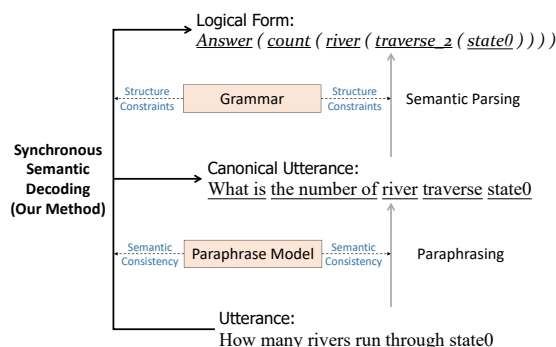


Figure 1: Different from previous staged methods (indicated by gray lines), our method generates canonical utterance and logical form synchronously. The semantic gap and the structure gap are simultaneously resolved by jointly leveraging paraphrasing and grammar-constrained decoding. Thus, our synchronous decoding employs both the semantic and the structure constraints to solve unsupervised semantic parsing.

Semantic parsing is a challenging task due to the structure gap and the semantic gap between natural language utterances and logical forms. For structure gap, because utterances are usually word sequences and logical forms are usually trees/graphs constrained by specific grammars, a semantic parser needs to learn the complex structure transformation rules between them. For semantic gap, because the flexibility of natural languages, the same meaning can be expressed using very different utterances, a semantic parser needs to be able to map various expressions to their semantic form. To address the structure gap and the semantic gap, current semantic parsers usually rely on a large amount of labeled data, often resulting in data bottleneck problem.

Previous studies have found that the structure gap and the semantic gap can be alleviated by leveraging external resources, therefore the reliance on data can be reduced. For structure gap, previous studies found that constrained decoding can effectively constrain the output structure by injecting grammars of logical forms and facts in

knowledge bases during inference. For example, the grammar-based neural semantic parsers (Xiao et al., 2016; Yin and Neubig, 2017) and the constrained decoding algorithm (Krishnamurthy et al., 2017). For semantic gap, previous studies have found that paraphrasing is an effective technique for resolving the diversity of natural expressions. Using paraphrasing, semantic parsers can handle the different expressions of the same meaning, therefore can reduce the requirement of labeled data. For example, supervised methods (Berant and Liang, 2014; Su and Yan, 2017) use the paraphrasing scores between canonical utterances and sentences to re-rank logical forms; Two-stage (Cao et al., 2020) rewrites utterances to canonical utterances which can be easily parsed. The main drawback of these studies is that they use constrained decoding and paraphrasing independently and separately, therefore they can only alleviate either semantic gap or structure gap.

In this paper, we propose an unsupervised semantic parsing method – *Synchronous Semantic Decoding (SSD)*, which can simultaneously resolve the structure gap and the semantic gap by jointly leveraging paraphrasing and grammar-constrained decoding. Specifically, we model semantic parsing as a constrained paraphrasing task: given an utterance, we synchronously decode its canonical utterance and its logical form using a general paraphrase model, where the canonical utterance and the logical form share the same underlying structure. Based on the synchronous decoding, the canonical utterance generation can be constrained by the structure of logical form, and the logical form generation can be guided by the semantics of canonical form. By modeling the interdependency between canonical utterance and logical form, and exploiting them through synchronous decoding, our method can perform effective unsupervised semantic parsing using only pretrained general paraphrasing model – no annotated data for semantic parsing is needed.

We conduct experiments on GEO and OVERNIGHT. Experimental results show that our method is promising, which can achieve competitive unsupervised semantic parsing performance, and can be further improved with external resources. The main contributions of this paper are:

- We propose an unsupervised semantic parsing method – Synchronous Semantic De-

coding, which can simultaneously resolve the semantic gap and the structure gap by jointly leveraging paraphrasing and grammar-constrained semantic decoding.

- We design two effective synchronous semantic decoding algorithms – rule-level inference and word-level inference, which can generate paraphrases under the grammar constraints and synchronously decode meaning representations.
- Our model achieves competitive unsupervised semantic parsing performance on GEO and OVERNIGHT datasets.

2 Model Overview

We now present overview of our synchronous semantic decoding algorithm, which can jointly leverage paraphrasing and grammar-constrained decoding for unsupervised semantic parsing. Given an utterance, SSD reformulates semantic parsing as a constrained paraphrasing problem, and synchronously generates its canonical utterance and logical form. For example in Fig. 2, given “How many rivers run through Texas”, SSD generates “What is the number of river traverse State0” as its canonical form and `Answer(Count(River(Traverse_2(State0))))` as its logical form. During synchronous decoding: the utterance paraphrase generation is constrained by the grammar of logical forms, therefore the canonical utterance can be generated controlledly; the logical form is generated synchronously with the canonical utterance via synchronous grammar. Logical form generation is controlled by the semantic constraints from paraphrasing and structure constraints from grammars and database schemas. Therefore the logical form can be generated unsupervisedly.

To this end, SSD needs to address two challenges. Firstly, we need to design paraphrasing-based decoding algorithms which can effectively impose grammar constraints on inference. Secondly, current paraphrasing models are trained on natural language sentences, which are different from the unnatural canonical utterances. Therefore SSD needs to resolve this style bias for effective canonical utterance generation.

Specifically, we first propose two inference algorithms for constrained paraphrasing based syn-

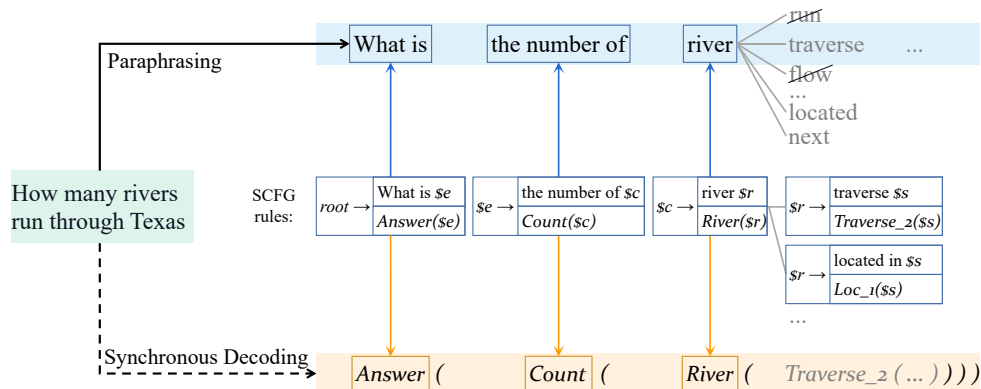


Figure 2: Overview of our approach. The sentence is paraphrased to canonical utterance and parsed to logical form synchronously. When decoding “traverse”, the paraphrase model tends to generate the words such as “run”, “flow”, “traverse” to preserve semantics. And synchronous grammar limits the next words of the canonical utterance to follow the candidate production rules. Then it is easy to discard “run” and “flow”, and select the most likely word “traverse” with its production rule from the candidates. In this paper, we propose rule-level and word-level inference methods to decode words and production rules synchronously.

chronous semantic decoding: rule-level inference and word-level inference. Then we resolve the style bias of paraphrase model via adaptive fine-tuning and utterance reranking, where adaptive fine-tuning can adjust the paraphrase model to generate canonical utterances, and utterance reranking resolves the style bias by focusing more on semantic coherence. In Sections 3-5, we provide the details of our implementation.

3 Synchronous Semantic Decoding

Given an utterance x , we turn semantic parsing into a constrained paraphrasing task. Concretely, we use synchronous context-free grammar as our synchronous grammar, which provides a one-to-one mapping from a logical form y to its canonical utterance c^y . The parsing task $\hat{y} = \arg \max_{y \in \mathcal{Y}} p_{\text{parse}}(y|x)$ is then transferred to $\hat{y} = \arg \max_{y \in \mathcal{Y}} p_{\text{paraphrase}}(c^y|x)$. Instead of directly parsing utterance into its logical form, SSD generates its canonical utterance and obtains its logical form based on the one-to-one mapping relation. In following we first introduce the grammar constraints in decoding, and then present two inference algorithms for generating paraphrases under the grammar constraints.

3.1 Grammar Constraints in Decoding

Synchronous context-free grammar(SCFG) is employed as our synchronous grammar, which is widely used to convert a meaning representation into a unique canonical utterance (Wang et al., 2015; Jia and Liang, 2016). An SCFG consists of a set of production rules: $N \rightarrow \langle \alpha, \beta \rangle$, where N is

a non-terminal, and α and β are sequence of terminal and non-terminal symbols. Each non-terminal symbol in α is aligned to the same non-terminal symbol in β , and vice versa. Therefore, an SCFG defines a set of joint derivations of aligned pairs of utterances and logical forms.

SCFGs can provide useful constraints for semantic decoding by restricting the decoding space and exploiting the semantic knowledge:

Grammar Constraints The grammars ensure the generated utterances/logical forms are grammar-legal. In this way the search space can be greatly reduced. For example, when expanding the non-terminal $\$r$ in Fig 2 we don’t need to consider the words “run” and “flow”, because they are not in the candidate grammar rules.

Semantic Constraints Like the type checking in Wang et al. (2015), the constraints of knowledge base schema can be integrated to further refine the grammar. The semantic constraints ensure the generated utterances/logical forms will be semantically valid.

3.2 Decoding

3.2.1 Rule-Level Inference

One strategy to generate paraphrase under the grammar constraint is taking the grammar rule as the decoding unit. Grammar-based decoders have been proposed to output sequences of grammar rules instead of words(Yin and Neubig, 2017). Like them, our rule-level inference method takes the grammar rule as the decoding unit. Figure 3 (a) shows an example of our rule level inference method.

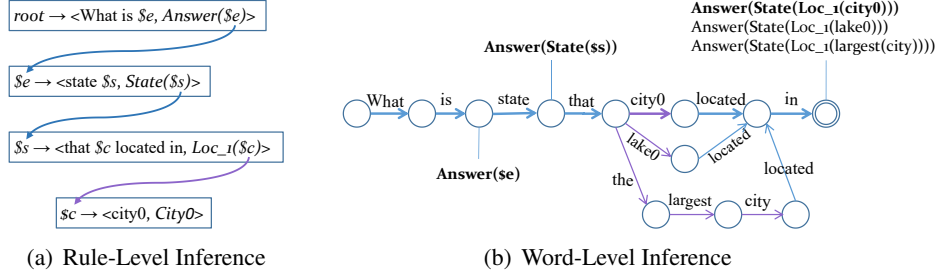


Figure 3: From the utterance “which state is city0 in”, two inference methods generate its canonical utterance “what is state that city0 located in” and its logical form $\text{Answer}(\text{State}(\text{Loc}_1(\text{City0})))$. The ways they handle non-terminal $\$c$ which is not at the end of utterance-side production rule are represented by purple lines.

Algorithm 1: Rule-level inference

Input : input utterance x , paraphrasing model $Para$, beam size B , maximum output length L , SCFG rules R , maximum search depth K ;

- 1 $beam_0 \leftarrow \{\{s\}\}$
- 2 $outputs \leftarrow \{\}$
- 3 **for** $t = 1$ **to** L **do**
- 4 **for** hypothesis c **in** $beam_{t-1}$ **do**
- 5 **for** r **in** expand rules for c **do**
- 6 **if** all non-terminals in r_β are on the right **then**
- 7 $c' \leftarrow \text{Expand}(c, r)$
- 8 $beam_t \leftarrow beam_t \cup \{c'\}$
- 9 **else**
- 10 $c' \leftarrow \text{expand}(c, r)$
- 11 $beam_t^{c'} \leftarrow \{c'\}$
- 12 **for** $k = 1$ **to** K **do**
- 13 **for** hypothesis h **in** $beam_{t+k-1}^{c'}$ **do**
- 14 $r_h \leftarrow$ expand rules for h 's first non-terminal
- 15 $beam_{t+k}^{c'} \leftarrow beam_{t+k}^{c'} \cup \text{Expand}(h, r_h)$
- 16 $beam_{t+k}^{c'} \leftarrow \text{NBest}(beam_{t+k}^{c'}, B)$
- 17 Move utterances from $beam_{t+k}^{c'}$ to $beam_{t+k}$, if non-terminals are on the right of the utterances.
- 18 $beam_t \leftarrow \text{NBest}(beam_t, B - |outputs|)$
- 19 Move full utterances from $beam_t$ to $outputs$
- 20 **if** $beam_t$ is empty **then**
- 21 **return** $outputs$
- 22 **return** $outputs$

When the non-terminal in the utterance-side production rule is at the end of the rule (e.g., $\$e \rightarrow \langle \text{state } \$s, \text{State}(\$s) \rangle$), denoting the utterance-side production rule as $r_\beta = [w_1, w_2, \dots, w_{L_r}, N]$, we can simply expand non-terminals in canonical utterances by this rule, and generate the canonical utterances from left to right with probabilities computed by:

$$P(c^{y \leq t} | x) = P(c^{y < t} | x) \prod_{i=1}^{L_r} P_{\text{paraphrase}}(w_i | x, c^{y < t}, w_{<i}) \quad (1)$$

Otherwise, we generate the next production rules to expand this rule (i.e., rule with purple line), until

there is no non-terminal on the left of words, or the generating step reaches the depth of K . We use beam search during the inference. The inference details are described in Algorithm 1.

3.2.2 Word-Level Inference

Except for rule-level inference, we also propose a word-level inference algorithm, which generates paraphrases word by word under the SCFG constraints.

Firstly, we construct a deterministic automaton using LR(1) parser (Knuth, 1965) from the CFG in utterance side. The automaton can transit from one state to another in response to an input. The inputs of the automaton are words and the states of it are utterance/logical form segments. LR(1) parser peeks ahead one lookahead input symbol, and the state transition table describes the acceptable inputs and the next states.

Then, in each decoding step we generate a word with a new state which is transited from previous state. An example is shown in Figure 3 (b). Only the acceptable words in the current state can be generated, and the end-of-sentence symbol can only be generated when reaching the final state. Beam search is also used in this inference.

4 Adaptive Fine-tuning

The above decoding algorithms only rely on a paraphrase generation model, which generates canonical utterance and logical form synchronously for semantic parsing. We can directly use general paraphrase generation models such as GPT-2 (Radford et al., 2019), T5 (Raffel et al., 2020) for SSD. However, as described in above, there exists a style bias between natural language sentences and canonical utterances, which hurts the performance of unsupervised semantic par-

ing. In this section, we describe how to alleviate this bias via adaptive fine-tuning. Given a text generation model, after pretraining it using paraphrase corpus, we fine-tune it using synthesized `<sentence, canonical utterance>` pairs.

Previous studies have shown that the pretraining on synthesized data can significantly improve the performance of semantic parsing (Xu et al., 2020a; Marzoev et al., 2020; Yu et al., 2020; Xu et al., 2020b). Specifically, we design three data synthesis algorithms:

- 1) **CUs** We sample CUs from SCFGs, and preserve executable ones. As we do not have the paired sentences, we only fine-tune the language model of the PLMs on CUs.
- 2) **Self Paras** We use the trained paraphrase model to get the natural language paraphrases of the sampled canonical utterances to form `<sentence, canonical utterance>` pairs.
- 3) **External Paras** We also use external paraphrase methods such as back translation to get the pairs.

5 Utterance Reranking

Adaptive fine-tuning resolves the style bias problem by fitting a better paraphrase model. In this section, we propose an utterance reranking algorithm to further alleviate the style bias by reranking and selecting the best canonical form.

Given the utterance x and top- N parsing results $(y_n, c_n), n = 1, 2, \dots, N$, we rerank all candidates by focusing on semantic similarities between x and c_n , so that canonical utterances can be effectively selected. Reranking for semantic parsing has been exploited in many previous studies (Berant and Liang, 2014; Yin and Neubig, 2019). These works employ reranking for canonical utterances selection. Differently, our re-ranker does not need labeled data. Formally, we measure two similarities between x and c_n and the final reranking score is calculated by:

$$\begin{aligned} score(x, c) = & \log p(c|x) + s_{rec}(x, c) \\ & + s_{asso}(x, c) \end{aligned} \quad (2)$$

Reconstruction Score The reconstruction score measures the coherence and adequacy of the canonical utterances, using the probability of reproducing the original input sentence x from c with the trained paraphrasing model: $s_{rec}(x, c) = \log p_{pr}(x|c)$

Association Score The association score measures whether x and c contain words that are likely to be paraphrases. We calculate it as:

$$\begin{aligned} s_{asso}(x, c) = & \log \prod_{i=1}^{|c|} \sum_{j=0}^{|x|} p(c_i|x_j) a(j|i) \\ & + \log \prod_{j=1}^{|x|} \sum_{i=0}^{|c|} p(x_j|c_i) a(i|j) \end{aligned} \quad (3)$$

in which, $p(c_i|x_j)$ means the paraphrase probability from x_j to c_i , and $a(j|i)$ means the alignment probability. The paraphrase probability and alignment are trained and inferred as the translation model in SMT IBM model 2.

6 Experiments

6.1 Experimental Settings

Datasets We conduct experiments on three datasets: OVERNIGHT(λ -DCS), GEO(FunQL), and GEOGRANNO, which use different meaning representations and on different domains. Our implementations are public available².

OVERNIGHT This is a multi-domain dataset, which contains natural language paraphrases paired with lambda DCS logical forms across eight domains. We use the same train/test splits as Wang et al. (2015).

GEO(FunQL) This is a semantic parsing benchmark about U.S. geography (Zelle and Mooney, 1996) using the variable-free semantic representation FunQL (Kate et al., 2005). We extend the FunQL grammar to SCFG for this dataset. We follow the standard 600/280 train/test splits.

GEOGRANNO This is another version of GEO (Herzig and Berant, 2019), in which lambda DCS logical forms paired with canonical utterances are produced from SCFG. Instead of paraphrasing sentences, crowd workers are required to select the correct canonical utterance from candidate list. We follow the split (train/valid/test 487/59/278) in original paper.

Paraphrase Model We obtain the paraphrase model by training T5 and GPT2.0 on WikiAnswer Paraphrase³, we train 10 epochs with learning rate as 1e-5. Follow Li et al. (2019), we sample 500K pairs of sentences in WikiAnswer corpus as training set and 6K as dev set. We generate adaptive fine-tuning datasets proportional to their labeled datasets, and back-translation(from English

²<https://github.com/lingowu/ssd>

³<http://knowitall.cs.washington.edu/paralex>

	Bas.	Blo.	Cal.	Hou.	Pub.	Rec.	Res.	Soc.	Avg.
Supervised									
RECOMBINATION (Jia and Liang, 2016)	85.2	58.1	78.0	71.4	76.4	79.6	76.2	81.4	75.8
CROSSDOMAIN (Su and Yan, 2017)	86.2	60.2	79.8	71.4	78.9	84.7	81.6	82.9	78.2
SEQ2ACTION (Chen et al., 2018b)	88.2	61.4	81.5	74.1	80.7	82.9	80.7	82.1	79.0
DUAL (Cao et al., 2019)	87.5	63.7	79.8	73.0	81.4	81.5	81.6	83.0	78.9
TWO-STAGE (Cao et al., 2020)	87.2	65.7	80.4	75.7	80.1	86.1	82.8	82.7	80.1
SSD (Word-Level)	86.2	64.9	81.7	72.7	82.3	81.7	81.5	82.7	79.2
SSD (Grammar-Level)	86.2	64.9	81.7	72.7	82.3	81.7	81.5	82.7	79.0
Unsupervised (with nonparallel data)									
TWO-STAGE (Cao et al., 2020)	64.7	53.4	58.3	59.3	60.3	68.1	73.2	48.4	60.7
WMDSAMPLES (Cao et al., 2020)	31.9	29.0	36.1	47.9	34.2	41.0	53.8	35.8	38.7
SSD-SAMPLES (Word-Level)	71.7	58.7	60.1	61.7	57.6	64.3	70.9	46.0	61.4
SSD-SAMPLES (Grammar-Level)	71.3	58.8	60.6	62.2	58.8	65.4	71.1	49.1	62.2
Unsupervised									
Cross-domain Zero Shot	-	28.3	53.6	52.4	55.3	60.2	61.7	-	-
GENOVERNIGHT	15.6	27.7	17.3	45.9	46.7	26.3	61.3	9.7	31.3
SYNTH-SEQ2SEQ	16.1	23.6	16.1	30.2	36.6	26.9	43.1	9.2	25.2
SYNTHPARA-SEQ2SEQ	28.4	37.3	33.9	38.1	39.1	41.7	62.7	23.3	38.1
SSD (Word-Level)	68.3	54.9	51.2	55.0	54.7	60.2	65.4	33.6	55.4
SSD (Grammar-Level)	68.8	58.1	56.5	56.1	57.8	59.3	66.9	37.1	57.6

Table 1: Overall results on OVERNIGHT.

to Chinese then translate back) is used to obtain external paraphrases data. On average, we sample 423 CUs per domain, and synthesize 847 instances per domain in Self Paras and 1252 in External Paras.

Unsupervised settings In unsupervised settings, we do not use any annotated semantic parsing data. The paraphrase generation models are fixed after the paraphrasing pre-training and the adaptive fine-tuning. The models are employed to generate canonical utterances and MRs synchronously via rule-level or word-level inference. In rule-level inference, the leftmost non-terminators are eliminated by cyclically expanded and the maximum depth K is set to 5, the beam size is set to 20. SSD uses T5 as the pre-trained language model in all the proposed components, including adaptive fine-tuning, reranking and the two decoding constraints. Ablation experiments are conducted over all components with rule-level inference.

Unsupervised settings (with external nonparallel data) Cao et al. (2020) have shown that external nonparallel data (including nonparallel natural language utterances and canonical utterances) can be used to build unsupervised semantic parsers. For fair comparison, we also conduct unsupervised experiments with external unparallel data. Specifically, we enhance the original SSD using the SAMPLES methods (Cao et al., 2020): we label

each input sentences with the most possible outputs in the nonparallel corpus and use these samples as pseudo training data – we denote this setting as SSD-SAMPLES.

Supervised settings Our SSD method can be further enhanced using annotated training instances. Specifically, given the annotated ⟨utterance, logical form⟩ instances, we first transform logical form to its canonical form, then use them to further fine-tune our paraphrase models after unsupervised pre-training.

Baselines We compare our method with the following unsupervised baselines: 1) Cross-domain Zero Shot (Herzig and Berant, 2018), which trains on other source domains and then generalizes to target domains in OVERNIGHT and 2) GENOVERNIGHT (Wang et al., 2015) in which models are trained on synthesized ⟨CU, MR⟩ pairs; 3) We also implement SEQ2SEQ baseline on the synthesized data as SYNTH-SEQ2SEQ. 4) SYNTHPARA-SEQ2SEQ is trained on the synthesized data and ⟨CU paraphrase, MR⟩ pairs, the paraphrases are obtained in the same way in Section 4.

6.2 Experimental Results

6.2.1 Overall Results

The overall results of different baselines and our method are shown in Table 1 and Table 3 (We also demonstrate several cases in Appendix). For our

	Bas.	Blo.	Cal.	Hou.	Pub.	Rec.	Res.	Soc.	OVERN. Avg.	GEO GRANNO	GEO (FunQL)
COMPLETEMODEL	68.8	58.1	56.5	56.1	57.8	59.3	66.9	37.1	57.6	58.5	63.2
Constraints											
- SEMANTIC	65.5	54.4	52.4	51.9	56.5	57.4	62.0	35.7	54.5	56.5	61.1
- GRAMMAR	63.9	52.1	47.6	50.3	45.3	52.3	56.3	29.9	49.7	50.7	53.9
Adaptive Fine-tuning											
- EXTERNAL PARAS	66.8	56.6	51.6	54.0	48.4	56.9	63.3	32.4	53.8	56.5	61.1
- PARAS (Only CUs)	64.5	55.4	50.0	51.3	47.8	55.6	62.0	31.7	52.3	54.3	59.6
- FINE-TUNING	63.9	53.6	48.4	47.6	44.9	53.2	62.5	31.4	50.7	51.8	55.4
Reranking											
- RERANKING	58.2	57.2	56.3	50.2	55.7	58.0	62.9	37.7	54.6	57.2	62.5
ORACLE (R@20)	71.2	83.2	86.9	76.9	82.7	86.9	88.1	62.8	79.8	70.9	81.4
Pretrained Language Models											
GPT-2	67.0	54.6	53.7	55.7	56.1	58.9	66.4	32.7	55.6	58.3	62.1
RAND	58.3	48.6	45.8	47.6	50.3	55.6	54.5	30.3	48.9	51.4	54.3

Table 2: Ablation results of our model with different settings on the three datasets.

	GEO GRANNO	GEO (FunQL)
Supervised		
DEPHT (Jie and Lu, 2018)	-	89.3
COPYNET (Herzig and Berant, 2019)	72.0	-
One-stage (Cao et al., 2020)	71.9	-
Two-stage (Cao et al., 2020)	71.6	-
SEQ2SEQ (Guo et al., 2020)	-	87.1
SSD (Word-Level)	72.9	88.3
SSD (Grammar-Level)	72.0	87.9
Unsupervised (with nonparallel data)		
Two-stage (Cao et al., 2020)	63.7	-
WMDSAMPLES (Cao et al., 2020)	35.3	-
SSD-SAMPLES (Word-Level)	64.0	64.3
SSD-SAMPLES (Grammar-Level)	64.4	65.0
Unsupervised		
SYNTH-SEQ2SEQ	32.7	36.1
SYNTHPARA-SEQ2SEQ	41.4	45.4
SSD (Word-Level)	57.2	62.8
SSD (Grammar-Level)	58.5	63.2

Table 3: Overall results on GEOGRANNO and GEO(FunQL).

method, we report its performances on three settings. We can see that:

1. **By synchronously decoding canonical utterances and meaning representations, SSD achieves competitive unsupervised semantic parsing performance.** In all datasets, our method outperforms other baselines in the unsupervised settings. These results demonstrate that unsupervised semantic parsers can be effectively built by simultaneously exploit semantic and structural constraints, without the need of labeled data.

2. **Our model can achieve competitive performance on different datasets with different settings.** In supervised settings, our model can achieve competitive performance with SOTA. With nonparallel data, our model can outperform Two-stage. On GEO(FunQL) our model also ob-

tains a significant improvement compared with baselines, which also verifies that our method is not limited to specific datasets (i.e., OVERNIGHT and GEOGRANNO, which are constructed with SCFG and paraphrasing.)

3. **Both rule-level inference and word-level inference can effectively generate paraphrases under the grammar constraints.** The rule-level inference can achieve better performance, we believe this is because rule-level inference is more compact than word-level inference, therefore the rule-level inference can search wider space and benefit beam search more.

6.2.2 Detailed Analysis

Effect of Decoding Constraints To analyze the effect of decoding constraints, we conduct ablation experiments with different constraint settings and the results are shown in Table 2: - SEMANTIC denotes removing the semantic constraint, -GRAMMAR denotes all constraints are removed at the same time, the decoding is unrestricted. We can see that the constrained decoding is critical for our paraphrasing-based semantic parsing, and both grammar constraints and semantic constraints contribute to the improvement.

Effect of Adaptive Fine-tuning To analyze the effect of adaptive fine-tuning, we show the results with different settings by ablating a fine-tuning corpus at a time (see Table 2). We can see that adaptive fine-tuning can significantly improve the performance. And the paraphrase generation model can be effectively fine-tuned only using CUs or Self Paras, which can be easily constructed.

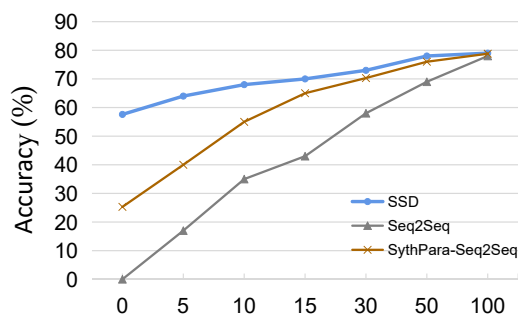


Figure 4: Semi-supervised results of different ratios of labeled data on OVERNIGHT.

Effect of Reranking To analyze the effect of reranking, we compare the settings with/without reranking and its upper bound – Oracle, which can always select the correct logical form if it is within the beam. Experimental results show that reranking can improve the semantic parsing performance. Moreover, there is still a large margin between our method and Oracle, i.e., the unsupervised semantic parsing can be significantly promoted by designing better reranking algorithms.

Effect of Adding Labeled Data To investigate the effect of adding labeled data, we test our method by varying the size of the labeled data on OVERNIGHT from 0% to 100%. In Fig. 4, we can see that our method can outperform baselines using the same labeled data. And a small amount of data can produce a good performance using our method.

Effect of Pretrained Language Models To analyze the effect of PLMs, we show the results with different PLM settings: instead of T5 we use GPT-2 or randomly initialized transformers to construct paraphrasing models. Experimental results show that powerful PLMs can improve the performance. Powered by the language generation models to do semantic parsing, our method can benefit from the rapid development of PLMs.

7 Related Work

Data Scarcity in Semantic Parsing. Witnessed the labeled data bottleneck problem, many techniques have been proposed to reduce the demand for labeled logical forms. Many weakly supervised learning are proposed (Artzi and Zettlemoyer, 2013; Berant et al., 2013; Reddy et al., 2014; Agrawal et al., 2019; Chen et al., 2020), such as denotation-base learning (Pasupat and Liang, 2016; Goldman et al., 2018), iterative searching (Dasigi et al., 2019). Semi-supervised

semantic parsing is also proposed (Chen et al., 2018a). Such as variational auto-encoding (Yin et al., 2018), dual learning framework for semantic parsing (Cao et al., 2019), dual information maximization method (Ye et al., 2019), and back-translation (Sun et al., 2019). One other strategy is to generate data for semantic parsing, e.g., Wang et al. (2015) construct a semantic parsing dataset from grammar rules and crowdsourcing paraphrase. Guo et al. (2018) produce pseudo-labeled data. Jia and Liang (2016) create new “recombinant” training examples with SCFG. The domain transfer techniques are also used to reduce the cost of data collecting for the unseen domain (Su and Yan, 2017; Herzig and Berant, 2018; Lu et al., 2019; Zhong et al., 2020). Goldwasser et al. (2011); Poon and Domingos (2009); Schmitt et al. (2020) leverage external resources or techniques for unsupervised learning.

Constrained Decoding. After neural parsers model semantic parsing as a sentence to logical form translation task (Yih et al., 2015; Krishnamurthy et al., 2017; Iyyer et al., 2017; Jie and Lu, 2018; Lindemann et al., 2020), many constrained decoding algorithms are also proposed, such as type constraint-based illegal token filtering (Krishnamurthy et al., 2017); Lisp interpreter-based method (Liang et al., 2017); type constraints for generating valid actions (Iyyer et al., 2017).

Paraphrasing in Semantic Parsing. Paraphrase models have been widely used in semantic parsing. ParaSempre (Berant and Liang, 2014) use paraphrase model to rerank candidate logical forms. Wang et al. (2015) employ SCFG grammar rules to produce MR and canonical utterance pairs, and construct OVERNIGHT dataset by paraphrasing utterances. Dong et al. (2017) use paraphrasing to expand the expressions of query sentences. Compared with these methods, we combine paraphrasing with grammar-constrained decoding, therefore SSD can further reduce the requirement of labeled data and achieve unsupervised semantic parsing.

8 Conclusions

We propose an unsupervised semantic parsing method – Synchronous Semantic Decoding, which leverages paraphrasing and grammar-constrained decoding to simultaneously resolve the semantic gap and the structure gap. Specifically, we design

two synchronous semantic decoding algorithms for paraphrasing under grammar constraints, and exploit adaptive fine-tuning and utterance reranking to alleviate the style bias in semantic parsing. Experimental results show that our approach can achieve competitive performance in unsupervised settings.

Acknowledgments

We sincerely thank the reviewers for their insightful comments and valuable suggestions. Moreover, this work is supported by the National Key Research and Development Program of China (No. 2020AAA0106400), the National Natural Science Foundation of China under Grants no. 61906182 and 62076233, and in part by the Youth Innovation Promotion Association CAS(2018141).

References

- Priyanka Agrawal, Ayushi Dalmia, Parag Jain, Abhishek Bansal, Ashish R. Mittal, and Karthik Sankaranarayanan. 2019. [Unified semantic parsing with weak supervision](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4801–4810.
- Yoav Artzi and Luke Zettlemoyer. 2013. [Weakly supervised learning of semantic parsers for mapping instructions to actions](#). *TACL*, 1:49–62.
- Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. [Semantic parsing on freebase from question-answer pairs](#). In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1533–1544.
- Jonathan Berant and Percy Liang. 2014. [Semantic parsing via paraphrasing](#). In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 1: Long Papers*, pages 1415–1425.
- Ruisheng Cao, Su Zhu, Chen Liu, Jieyu Li, and Kai Yu. 2019. [Semantic parsing with dual learning](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 51–64.
- Ruisheng Cao, Su Zhu, Chenyu Yang, Chen Liu, Rao Ma, Yanbin Zhao, Lu Chen, and Kai Yu. 2020. [Unsupervised dual paraphrasing for two-stage semantic parsing](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 6806–6817. Association for Computational Linguistics.
- Bo Chen, Bo An, Le Sun, and Xianpei Han. 2018a. [Semi-supervised lexicon learning for wide-coverage semantic parsing](#). In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 892–904. Association for Computational Linguistics.
- Bo Chen, Xianpei Han, Ben He, and Le Sun. 2020. [Learning to map frequent phrases to sub-structures of meaning representation for neural semantic parsing](#). In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7546–7553. AAAI Press.
- Bo Chen, Le Sun, and Xianpei Han. 2018b. [Sequence-to-action: End-to-end semantic graph generation for semantic parsing](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 766–777.
- Pradeep Dasigi, Matt Gardner, Shikhar Murty, Luke Zettlemoyer, and Eduard H. Hovy. 2019. [Iterative search for weakly supervised semantic parsing](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 2669–2680.
- Li Dong and Mirella Lapata. 2016. [Language to logical form with neural attention](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.
- Li Dong, Jonathan Mallinson, Siva Reddy, and Mirella Lapata. 2017. [Learning to paraphrase for question answering](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 875–886.
- Omer Goldman, Veronica Latcinnik, Ehud Nave, Amir Globerson, and Jonathan Berant. 2018. [Weakly supervised semantic parsing with abstract examples](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 1809–1819.
- Dan Goldwasser, Roi Reichart, James Clarke, and Dan Roth. 2011. [Confidence driven unsupervised semantic parsing](#). In *The 49th Annual Meeting of the Association for Computational Linguistics, ACL 2011, Portland, Oregon, USA, June 9-14, 2011*, pages 1031–1040. Association for Computational Linguistics.

- ciation for Computational Linguistics: Human Language Technologies, *Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA*, pages 1486–1495.
- Daya Guo, Yibo Sun, Duyu Tang, Nan Duan, Jian Yin, Hong Chi, James Cao, Peng Chen, and Ming Zhou. 2018. [Question generation from SQL queries improves neural semantic parsing](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 1597–1607.
- Jiaqi Guo, Qian Liu, Jian-Guang Lou, Zhenwen Li, Xueqing Liu, Tao Xie, and Ting Liu. 2020. [Benchmarking meaning representations in neural semantic parsing](#). In *EMNLP*.
- Jonathan Herzig and Jonathan Berant. 2018. [Decoupling structure and lexicon for zero-shot semantic parsing](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 1619–1629.
- Jonathan Herzig and Jonathan Berant. 2019. [Don't paraphrase, detect! rapid and effective data collection for semantic parsing](#). 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 2019, Hong Kong, China, November 3-7, 2019*, pages 3808–3818. Association for Computational Linguistics.
- Mohit Iyyer, Wen-tau Yih, and Ming-Wei Chang. 2017. [Search-based neural structured learning for sequential question answering](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1821–1831.
- Robin Jia and Percy Liang. 2016. [Data recombination for neural semantic parsing](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.
- Zhanming Jie and Wei Lu. 2018. [Dependency-based hybrid trees for semantic parsing](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2431–2441.
- Rohit J. Kate, Yuk Wah Wong, and Raymond J. Mooney. 2005. [Learning to transform natural to formal languages](#). In *Proceedings, The Twentieth National Conference on Artificial Intelligence and the Seventeenth Innovative Applications of Artificial Intelligence Conference, July 9-13, 2005, Pittsburgh, Pennsylvania, USA*, pages 1062–1068.
- Donald E. Knuth. 1965. [On the translation of languages from left to right](#). *Inf. Control.*, 8(6):607–639.
- Jayant Krishnamurthy, Pradeep Dasigi, and Matt Gardner. 2017. [Neural semantic parsing with type constraints for semi-structured tables](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 1516–1526.
- Zichao Li, Xin Jiang, Lifeng Shang, and Qun Liu. 2019. [Decomposable neural paraphrase generation](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3403–3414.
- Chen Liang, Jonathan Berant, Quoc Le, Kenneth D. Forbus, and Ni Lao. 2017. [Neural symbolic machines: Learning semantic parsers on freebase with weak supervision](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 23–33.
- Matthias Lindemann, Jonas Groschwitz, and Alexander Koller. 2020. [Fast semantic parsing with well-typedness guarantees](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 3929–3951. Association for Computational Linguistics.
- Wei Lu, Hwee Tou Ng, Wee Sun Lee, and Luke S. Zettlemoyer. 2008. [A generative model for parsing natural language to meaning representations](#). In *2008 Conference on Empirical Methods in Natural Language Processing, EMNLP 2008, Proceedings of the Conference, 25-27 October 2008, Honolulu, Hawaii, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 783–792.
- Zhichu Lu, Forough Arabshahi, Igor Labutov, and Tom M. Mitchell. 2019. [Look-up and adapt: A one-shot semantic parser](#). *CoRR*, abs/1910.12197.
- Alana Marzoev, Samuel Madden, M. Frans Kaashoek, Michael J. Cafarella, and Jacob Andreas. 2020. [Unnatural language processing: Bridging the gap between synthetic and natural language data](#). *CoRR*, abs/2004.13645.
- Panupong Pasupat and Percy Liang. 2016. [Inferring logical forms from denotations](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.
- Hoifung Poon and Pedro M. Domingos. 2009. [Unsupervised semantic parsing](#). In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, EMNLP 2009, 6-7 August 2009, Singapore, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1–10.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. [Language models are unsupervised multitask learners](#).

- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. [Exploring the limits of transfer learning with a unified text-to-text transformer](#). *J. Mach. Learn. Res.*, 21:140:1–140:67.
- Siva Reddy, Mirella Lapata, and Mark Steedman. 2014. [Large-scale semantic parsing without question-answer pairs](#). *Transactions of the Association for Computational Linguistics*, 2:377–392.
- Martin Schmitt, Sahand Sharifzadeh, Volker Tresp, and Hinrich Schütze. 2020. [An unsupervised joint system for text generation from knowledge graphs and semantic parsing](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 7117–7130. Association for Computational Linguistics.
- Bo Shao, Yeyun Gong, Weizhen Qi, Nan Duan, and Xiaola Lin. 2020. [Multi-level alignment pretraining for multi-lingual semantic parsing](#). In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 3246–3256. International Committee on Computational Linguistics.
- Yu Su and Xifeng Yan. 2017. [Cross-domain semantic parsing via paraphrasing](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 1235–1246.
- Yibo Sun, Duyu Tang, Nan Duan, Yeyun Gong, Xiaocheng Feng, Bing Qin, and Daxin Jiang. 2019. [Neural semantic parsing in low-resource settings with back-translation and meta-learning](#). *CoRR*, abs/1909.05438.
- Yushi Wang, Jonathan Berant, and Percy Liang. 2015. [Building a semantic parser overnight](#). In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 1332–1342.
- Yuk Wah Wong and Raymond J. Mooney. 2007. [Learning synchronous grammars for semantic parsing with lambda calculus](#). In *ACL 2007, Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, June 23-30, 2007, Prague, Czech Republic*.
- Chunyang Xiao, Marc Dymetman, and Claire Gardent. 2016. [Sequence-based structured prediction for semantic parsing](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.
- Dongqin Xu, Junhui Li, Muhua Zhu, Min Zhang, and Guodong Zhou. 2020a. [Improving AMR parsing with sequence-to-sequence pre-training](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 2501–2511. Association for Computational Linguistics.
- Silei Xu, Sina J. Semnani, Giovanni Campagna, and Monica S. Lam. 2020b. [Autoqa: From databases to QA semantic parsers with only synthetic training data](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 422–434. Association for Computational Linguistics.
- Hai Ye, Wenjie Li, and Lu Wang. 2019. [Jointly learning semantic parser and natural language generator via dual information maximization](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 2090–2101.
- Wen-tau Yih, Ming-Wei Chang, Xiaodong He, and Jianfeng Gao. 2015. [Semantic parsing via staged query graph generation: Question answering with knowledge base](#). In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 1321–1331.
- Pengcheng Yin and Graham Neubig. 2017. [A syntactic neural model for general-purpose code generation](#). In *ACL 2017*, pages 440–450. ACL 2017.
- Pengcheng Yin and Graham Neubig. 2019. [Reranking for neural semantic parsing](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4553–4559.
- Pengcheng Yin, Chunting Zhou, Junxian He, and Graham Neubig. 2018. [Structvae: Tree-structured latent variable models for semi-supervised semantic parsing](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 754–765.
- Tao Yu, Chien-Sheng Wu, Xi Victoria Lin, Bailin Wang, Yi Chern Tan, Xinyi Yang, Dragomir R. Radev, Richard Socher, and Caiming Xiong. 2020. [Grappa: Grammar-augmented pre-training for table semantic parsing](#). *CoRR*, abs/2009.13845.
- John M. Zelle and Raymond J. Mooney. 1996. [Learning to parse database queries using inductive logic programming](#). In *Proceedings of the Thirteenth National Conference on Artificial Intelligence and*

Eighth Innovative Applications of Artificial Intelligence Conference, AAAI 96, IAAI 96, Portland, Oregon, USA, August 4-8, 1996, Volume 2., pages 1050–1055.

Luke S. Zettlemoyer and Michael Collins. 2005. [Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars](#). In *UAI '05, Proceedings of the 21st Conference in Uncertainty in Artificial Intelligence, Edinburgh, Scotland, July 26-29, 2005*, pages 658–666.

Yuanyuan Zhao, Weiwei Sun, Junjie Cao, and Xiaojun Wan. 2020. [Semantic parsing for english as a second language](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 6783–6794. Association for Computational Linguistics.

Victor Zhong, Mike Lewis, Sida I. Wang, and Luke Zettlemoyer. 2020. [Grounded adaptation for zero-shot executable semantic parsing](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6869–6882. Association for Computational Linguistics.

dataset, we can still efficiently generate the utterances containing it, which shows our constrained-paraphrasing based semantic parser has the generalization ability on unseen words. We found that the parser maintains high recall, covering the correct canonical utterances in our n-best list of predictions. As case 2 shows the designed utterance reranking score can select the best canonical utterances by focusing on coherence and adequacy. With adaptive fine-tuning (case 3), our model can generate the utterances focusing more on semantics to alleviate the style bias.

A Appendix

A.1 Case Study

In Table 4, we present the cases generated from SSD. Cases show that SSD can output semantics-similar and grammar-legal utterances. In case 1, “take-out” does not appear in paraphrase

Cases	$x \rightarrow c$ $\log P(c x)$	$x \leftarrow c$ $s_{rec}(x, c)$	$x \leftrightarrow c$ $s_{asso}(x, c)$	Overall Reranking Score
Input: restaurants that accept credit cards and offer takeout				
Outputs: <u>restaurant that takes credit cards and that has take-out</u>	-54.2	-3.1	-20.3	-77.6
<u>restaurant that has take-out and that takes credit cards</u>	-72.0	-8.8	-20.3	-101.1
restaurant that takes credit cards	-77.2	-22.4	-31.9	-131.5
restaurant that takes credit cards and that takes credit cards	-84.2	-26.7	-28.1	-139.0
Input: meetings held in the same place as the weekly standup meeting				
Outputs: meeting whose date is date of weekly standup	-62.2	-40.2	-67.1	-169.5
<u>meeting whose location is location of weekly standup</u>	-62.7	-22.1	-62.4	-147.2
<u>meeting whose location is location that is location of weekly standup</u>	-65.0	-21.1	-63.5	149.6
meeting whose date is at most date of weekly standup	-67.2	-35.8	-73.3	-176.3
Input: meetings held in the same place as the weekly standup meeting				
Outputs: <u>meeting whose location is location of weekly standup</u>	-2.7	-22.1	-62.4	-86.2
<u>meeting whose location is location that is location of weekly standup</u>	-6.0	-21.1	-63.5	-90.6
location that is location of weekly standup	-18.0	-32.6	-62.8	-113.4
meeting whose date is date of weekly standup	-31.2	-40.2	-67.1	-138.5

Table 4: Output cases from SSD on OVERNIGHT. The outputs are sorted by the generation score $\log P(c|x)$. The underlined canonical utterances are correct. Adaptive fine-tuning is employed in the third case, and not employed in the first two cases. Accuracies on OVERNIGHT. In GENOVERNIGHT Wang et al. (2015), all the canonical utterances are also generated without manual annotation. The previous methods with superscript * means they use different unsupervised settings.