Bridging the Gap between Synthetic and Natural Questions via Sentence Decomposition for Semantic Parsing

Yilin Niu[1], **Fei Huang**[1], **Wei Liu**[2], **Jianwei Cui**[2], **Bin Wang**[2], **Minlie Huang**[1*]
[1]The CoAI Lab, DCST, Tsinghua University, Beijing, China;
[1]Institute for Artificial Intelligence, State Key Lab of Intelligent Technology and Systems, China;
[1]Beijing National Research Center for Information Science and Technology, China
[2]Xiaomi AI Lab, China
niuyl14@tsinghua.org.cn f-huang18@mails.tsinghua.edu.cn
{liuwei40,cuijianwei,wangbin11}@xiaomi.com aihuang@tsinghua.edu.cn

Abstract

Semantic parsing maps natural language questions into logical forms, which can be executed against a knowledge base for answers. In real-world applications, the performance of a parser is often limited by the lack of training data. To facilitate zero-shot learning, data synthesis has been widely studied to automatically generate paired questions and logical forms. However, data synthesis methods can hardly cover the diverse structures in natural languages, leading to a large gap in sentence structure between synthetic and natural questions. In this paper, we propose a decomposition-based method to unify the sentence structures of questions, which benefits the generalization to natural questions. Experiments demonstrate that our method significantly improves the semantic parser trained on synthetic data (+7.9% on KQA and +8.9% on ComplexWebQuestions in terms of exact match accuracy). Extensive analysis demonstrates that our method can better generalize to natural questions with novel text expressions compared with baselines. Besides semantic parsing, our idea potentially benefits other semantic understanding tasks by mitigating the distracting structure features. To illustrate this, we extend our method to the task of sentence embedding learning, and observe substantial improvements on sentence retrieval (+13.1% for Hit@1).

1 Introduction

Semantic parsing is a task of mapping natural language questions into logical forms, which serves as a backbone in knowledge base question answering (Talmor and Berant, 2018; Gu et al., 2021; Cao et al., 2022) and task-oriented dialogue (Li et al., 2021a). Traditionally, learning a powerful semantic parser relies on large-scale annotated data, which is laborious to collect. To reduce the budget of annotation, some recent efforts have been dedicated into low-resource training methods, among which data synthesis is widely used.

Data synthesis can efficiently generate large amounts of training data by means of templates and rules (Wang et al., 2015; Xu et al., 2020b), but the templated expressions of synthetic questions hinder the generalization to natural questions. The gap between synthetic and natural questions mainly derives from two kinds of textual features, phrase-level expression and sentence-level structure. The former refers to the myriad ways in which predicates, relations, and entities can be expressed (Berant and Liang, 2014), such as ''the human whose date of birth is 2000'' and ''the person born in 2000''. The latter usually manifests itself as the global rearrangement of content (see Figure 1 for an example). To alleviate the problem, paraphrasing is commonly used for increasing language diversity of synthetic data (Xu et al., 2020c; Weir et al., 2020). Although the paraphrasing methods successfully introduce diverse phrase-level expressions, they are still weak in generating sentences with diverse structures (Niu et al., 2021); this is because the generation models tend to assign significantly higher generative probability to the sentences with similar structures, making it challenging to control or diversify the structures of the generated sentences.

In contrast to the paraphrase approaches that bridge the gap by augmenting the synthetic data, we propose to unify the sentence-level structure by decomposing the questions into several

*Corresponding author.

Transactions of the Association for Computational Linguistics, vol. 11, pp. 367–383, 2023. https://doi.org/10.1162/tacl_a_00552
Action Editor: Jonathan Berant. Submission batch: 9/2022; Revision batch: 11/2022; Published 5/2023.

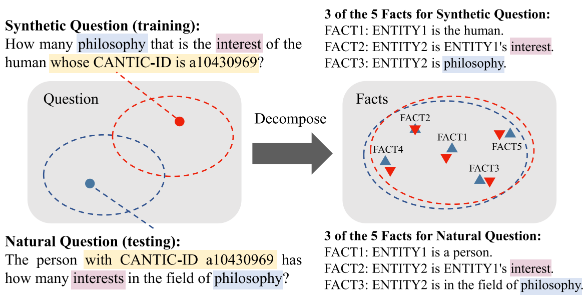

Figure 1: The motivation of our method. **Left**: synthetic and natural questions are distributed differently in terms of sentence structures. **Right**: we propose to decompose the questions into simple facts that fall into similar text spaces with lower structure gap.

simple facts (see Figure 1). Intuitively, the question decomposition maps the natural and synthetic questions into a smaller text space that only contains simple sentences, which is more controllable than generating diverse synthetic questions to cover possible sentence structures. Specifically, we decompose the questions with an in-context learning method, where the prompt is composed of several (about 10) synthetic questions as well as their decomposition results. Then we conduct semantic parsing with the decomposed facts as inputs, instead of the original questions. The decomposing-then-parsing process takes place during both the training with synthetic data and the inference for natural questions. In this way, synthetic and natural data share the similar distributions of input structures, leading to better generalization to natural questions.

We conduct a comprehensive evaluation of our method on two semantic parsing datasets, KQA (Cao et al., 2022) and ComplexWebQuestion (CWQ) (Talmor and Berant, 2018). Our method achieves a 76.7% and 51.0% exact match accuracy on KQA and CWQ, respectively—a 7.9% and 8.9% improvement compared to the best baselines. Moreover, although only synthetic data is used for training, our method has approached the performance of the semantic parser trained with full natural data on CWQ. Extensive analysis demonstrates that the improvements mainly derive from the generalization to the natural questions with novel text expressions. Besides semantic parsing, our idea can be applied to other semantic understanding tasks to mitigate distracting structure features. When extended for the task of sentence embedding learning, our method achieves a substantial improvement on sentence retrieval (+13.1% in terms of Hit@1).

Our contributions are summarized as follows:

- We propose a decomposition-based method to bridge the gap between synthetic and natural data, so that the semantic parser trained on synthetic data can better generalize to the natural questions with novel text expressions.
- Experiments on two semantic parsing datasets of different logical systems shows that our method largely outperforms previous strong baselines trained on synthetic data.
- Our method can be extended to other semantic understanding tasks, for example, sentence embedding learning, where we show its effectiveness in retrieving sentences from the candidates with divergent structures.

2 Related Work

2.1 Data Synthesis for Semantic Parsing

Data synthesis plays an important role in semantic parsing, especially in dataset construction and model training. Many datasets are constructed by first generating synthetic data by rules and then manually rewriting the questions for diverse expressions (Wang et al., 2015; Cao et al., 2022; Gu et al., 2021). Since the generation of synthetic data is controllable, researchers can restrict the distribution of logical forms in order to examine or improve various generalization capabilities of models, such as compositional generalization and scalability (Lake and Baroni, 2018; Gu et al., 2021; Oren et al., 2021).

In many scenarios, where manually rewriting is laborious or unavailable, training on synthetic data can also alleviate the problems of cold-start and privacy (Yang et al., 2022b; Yu et al., 2021). Limited by stereotyped expressions, semantic parsers trained with only synthetic data are difficult to generalize to diverse natural questions. Wu et al. (2021a) and Xu et al. (2020c) automatically rewrite synthetic questions with paraphrasing models, which introduces various phrase-level expressions but change little in sentence structures. Instead of rule-based synthesis, Zhong et al. (2020) and Yang et al. (2021) train a neural data synthesizer with natural data from other domains, and then generate training data for the target domain in a way of domain transfer. Such methods heavily rely on multi-domain resources, and thus their applicability is limited. There are some other studies attempting to bridge the data gap through mapping synthetic and natural questions into a common embedding space. Berant and Liang (2014) and Marzoev et al. (2020) leverage sentence embeddings or other paraphrase detection technologies to measure the semantic similarity between the input question and candidate logical forms. However, existing sentence embedding methods are poor at accurately capturing the semantic meanings of complex sentences, making them unsuitable for some challenging semantic parsing tasks (e.g., KQA [Cao et al., 2022]).

2.2 Other Low-Resource Semantic Parsing Methods

In addition to data synthesis, other low-resource semantic parsing methods have also attracted attention. If there exist abundant resources of other domains, cross-domain transfer learning is an effective method to boost the semantic parser for the target domain (Givoli and Reichart, 2019; Dadashkarimi et al., 2018; Herzig and Berant, 2018). Cross-lingual transfer learning performs in a similar way for low-resource languages (Liu et al., 2021; Xia and Monti, 2021; Sherborne and Lapata, 2022). When numerous unlabeled data (i.e., natural questions without logical forms) is available, self-training can carry out weakly supervised learning by generating pseudo-labels (Wang et al., 2020; Rongali et al., 2022). However, even the unlabeled questions might be not readily available in many scenarios (Yang et al., 2021). Compared with transfer learning and self-training, prompt-based methods are not restricted by external resources. These methods take advantage of the remarkable capabilities of few-shot learning (Schucher et al., 2022) and in-context learning (Shin and Van Durme, 2022; Pasupat et al., 2021; Gupta et al., 2022) of large-scale pre-trained language models. Grammar-constrained decoding is sometimes employed to guarantee grammatical logical forms (Cao et al., 2020; Shin et al., 2021). Additionally, meta-learning (Li et al., 2021b; Wang et al., 2021; Sun et al., 2020) and pre-training (Xu et al., 2020a; Jiang et al., 2021) have been investigated to provide a better start point of training.

2.3 Sentence Decomposition

Sentence decomposition is a fundamental technology for understanding complex sentences (Gao et al., 2021b). Previous work has experimentally showed that decomposing complex questions into sub-questions, and then answering the sub-questions one by one to get the final answer, can boost multi-hop question answering (Perez et al., 2020; Fu et al., 2021; Deng et al., 2022). These studies focus on decomposing the question into answerable single-hop sub-questions, whereas this paper requires the sub-sentences with the unified and simplest sentence structures, which is a finer-grained decomposition. Semantic parsing also benefits from question decomposition. Zhao et al. (2022) and Yang et al. (2022a) simplify the task with several fixed sub-questions written by humans, which is only suitable for simple logical systems. Besides, there are also some studies automatically extracting sub-questions with neural decomposer (Saparina and Osokin, 2021; Wolfson et al., 2022), which is trained on BREAK (Wolfson et al., 2020). Our method differs from this work in two aspects: (1) These methods are highly dependent on the domains and question types pre-defined by BREAK, which limits the scope of application. In contrast, our method can be easily adapted to any task with only a few (about 10) samples. (2) Our motivation is different. Previous methods use sentence decomposition to do rule-based parsing, while we aim to bridge the data gap between synthetic and natural questions for better generalizability.

In addition to sentence decomposition, some other data transformation techniques are also widely used in semantic parsing and information retrieval. Query rewriting (Carpineto and Romano, 2012; Kuzi et al., 2016; Wu et al., 2021b) transforms search queries in order to better represent the user intent. Syntactic analysis (Poon and Domingos 2009; Xu et al., 2018) provides additional dependency and constituency features, which help understand text structures. Similarly, our method decomposes complex sentences into simple sentences, that is, atomic events, so as to help the model understand complex events.

3 Method

To improve the low-resource semantic parsers only trained on synthetic data, we propose a novel method that decomposes the input question into

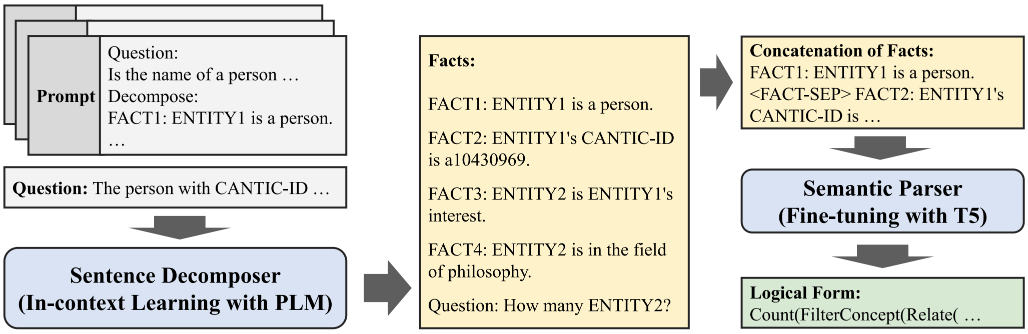

Figure 2: Overview of our method for bridging the gap between synthetic and natural questions for semantic parsing. The decomposer is implemented through in-context learning with a pre-trained language model (PLM), taking several decomposition exemplars as prompt and generating the facts semantically equivalent to the input question. The parser takes the concatenation of the facts as input and generates the corresponding logical form.

several facts before the parsing, aiming to reduce the gap between synthetic and natural data. As shown in Figure 2, our model consists of two modules, a decomposer that breaks the question into simple facts (Section 3.1) and a parser that receives the facts to do the parsing (Section 3.2). Moreover, our idea can also assist other semantic understanding tasks, for example, sentence embedding learning (Section 3.3), which illustrates the potential of our method.

3.1 Sentence Decomposer

Our sentence decomposer breaks the input question into several facts with simple syntax, unifying the sentence structure of synthetic and natural data. Specifically, we develop a prompt-based approach, where the model completes the decomposition according to several exemplars without additional fine-tuning, making use of the in-context learning ability (Brown et al., 2020; Liu et al., 2022) of pre-trained language models (PLMs).

Our prompt is composed of several (about 10) exemplars of sentence decomposition, where an example is shown in Table 1. Each exemplar in the prompt consists a question and a set of facts, which satisfies two requirements: (1) The collective meaning of the facts is consistent with the original question. (2) Each fact contains only subject, predicate, and object to maintain a simple structure. Note that the facts are expressed in natural language, not restricted by specific predicates or templates. Compared with other decomposition representations, such as AMR, the representation in the form of natural language has stronger expressive ability and is easier for pre-trained language models to generate.

sentence:
Is the name of a person educated in high school
in the mother tongue equivalent to Laura Linney?
decompose:
FACT1: ENTITY1 is a person.
FACT2: ENTITY1 is educated in high school.
FACT3: ENTITY2 is ENTITY1's name.
FACT4: ENTITY2 is in the mother tongue.
Question: Is ENTITY2 equal to Laura Linney?

sentence:
How many medals did James Stewart (who won the
13th Academy Award) receive, whose country is
the United States of America?
decompose:
FACT1: ENTITY1 is James Stewart.
FACT2: ENTITY1 won the 13th Academy Award.
FACT3: ENTITY2 is the United States of America.
FACT4: ENTITY1 received ENTITY3.
FACT5: ENTITY3 is a medal.
FACT6: ENTITY1's country is ENTITY2.
Question: How many ENTITY3?
. . .
sentence:
Which region of Italy bordering Umbria has the
lowest population?
decompose:

Table 1: An example prompt for sentence decomposition on KQA (Cao et al., 2022). The complete prompt is omitted for simplicity and can be found in `https://github.com/heyLinsir/Decomposition-for-Semantic-Parsing`. ENTITY and FACT work as placeholders for objects and events. The reference object of ENTITY is not limited to named entities (e.g., James Stewart), but can also represent concepts, nouns, etc. (e.g., medal).

Specifically, the exemplars used in the prompt are produced through human-machine collaboration, aiming to exploit the model's preferred decomposition representation. The construction process contains three steps: (1) Manually annotate two or three exemplars to form the initial prompt. (2) Utilize the model to decompose a new synthetic question with the existing prompt. (3) Manually correct the errors in the decomposition result, which is then added to the prompt as a new exemplar. Return to step 2 until the preset number of exemplars is reached.

In our experiments, we utilize OpenAI Codex (Chen et al., 2021) as the decomposition model because it performs well in the in-context learning and allows much more input tokens than OpenAI GPT3 (Brown et al., 2020). Considering that natural data is unavailable during training, only synthetic questions are used for constructing the prompt. Due to the simplicity of the sentence decomposition task, our decomposer generalizes well to natural questions. In Section 4.8, we analyze the quality of the decomposed facts, of which 89.5% preserve the exact same semantics as the original natural questions. In practice, due to the query limitation of Codex API, we employ another T5-based decomposer (Raffel et al., 2020) to imitate Codex's behavior and decompose all the synthetic training questions. The T5-decomposer is trained with the decomposition results of 5,000 synthetic questions generated by Codex. Note that the questions for evaluation are still decomposed by Codex for better generalization.

3.2 Semantic Parser

Our semantic parser predicts the logical form based on the decomposed facts instead of the original question. We concatenate the facts, in the original order they are generated during decomposition, into a single sequence with <FACT-SEP> as the separator (Figure 2), which serves as the input of the semantic parser.

To bridge the structure gap between synthetic and natural questions, two types of structure features of the concatenated sequence should be unified: (1) inner-fact structure, the sentence structure of each fact, and (2) inter-fact structure, the order of the facts. Because the decomposed facts are simple enough to have only the basic sentence structures, the gap in the inner-fact structure has been already bridged. To reduce the gap of inter-fact structure, an intuitive idea is to randomly reorder the facts before concatenation. However, no significant improvement was observed with the reordering strategy compared to using the original order. The observation implies that the pretrained Codex may tend to generate facts in a consistent order, for example, a subject first and then its properties. Therefore, we use the original order of the generated facts in the experiments for simplicity.

We employ T5-large (Raffel et al., 2020) as the backbone of our semantic parser and directly fine-tune it on the synthetic data with decomposed facts as inputs. During training, the model takes the concatenated facts as input and is optimized to maximize the conditional log-likelihood of the golden logical form. For inference, we also use the decomposed facts as the model inputs, and the logical form is generated with a beam search of size 5 conditioned on the decomposed facts.

3.3 Extension: Sentence Embedding Learning

The core idea of our method is to alleviate distracting sentence structure features in semantic understanding, which can also benefit other semantic understanding tasks even with natural training data. To illustrate this, we extend our approach to the task of learning sentence embeddings, which aims to learn representations of sentences and is helpful in many downstream tasks, such as information retrieval (Gao et al., 2021a).

Contrastive learning is an effective way to learn sentence embeddings (Gao et al., 2021a), but the spurious correlation between sentence structures and labels can mislead the model to assign high similarity scores to structurally similar but non-synonymous sentences. In order to avoid the abuse of structure features, we imitate the process described previously: (1) Decompose the original sentence x_i into a set of facts. (2) Take the original sentence x_i and the concatenation of the facts d_i as a positive example for contrastive learning:

$$\mathcal{L}(f) = -\log \frac{e^{\mathrm{sim}(f(x_i), f(d_i))/\tau}}{\sum_{j=1}^{N} e^{\mathrm{sim}(f(x_i), f(d_j))/\tau}}, \qquad (1)$$

where f is the sentence encoder, sim is the cosine similarity function, τ is the temperature, and N is the number of examples in a mini-batch. As expected, the experiments in Section 4.9 demonstrate that our method tends to assign higher similarity scores to semantically equivalent sentences rather than structurally similar sentences.

Note that sentence decomposition is only applied during training to align the representation of the original and decomposed sentences, while for evaluation, our method directly encodes the original sentence where the decomposed facts are not used. Intuitively, the decomposed facts have guided the model to learn structurally irrelevant semantic representation for the original sentence, and thus there is no need to explicitly decompose the original sentence during the test phase. Ideally, through contrastive learning, the model learns the semantic consistency between the original sentence and its decomposed facts, namely, $\mathrm{sim}(f(x_i), f(d_i)) \approx 1$. Therefore, the similarity between the embeddings of decomposed facts can be approximated by that of original sentences, that is, $\mathrm{sim}(f(x_i), f(x_j)) \approx \mathrm{sim}(f(d_i), f(d_j))$.

4 Experiments

4.1 Datasets

We conduct experiments on two semantic parsing tasks, KQA (Cao et al., 2022) and ComplexWebQuestions (CWQ) (Talmor and Berant, 2018). We choose them because: (1) These datasets provide the synthetic data generated by templates, as well as the natural data annotated by humans, which makes it convenient to investigate the gap between them. (2) The questions of these datasets are complex, which requires deep semantic understanding. (3) The logical forms of KQA and CWQ are based on KoPL (Cao et al., 2022) and SPARQL, respectively, which demonstrates the generality of our method for different logical systems. The data splits and the scripts for data processing are available at `https://github.com/heyLinsir/Decomposition-for-Semantic-Parsing`.

4.2 Baselines

Although our model only uses synthetic data in training, we compare it against baselines trained on full natural data, few-shot natural data, and synthetic data.

T5-FullNatural (Raffel et al., 2020) is fine-tuned on the full natural data, forming an upper bound for low-resource methods.

For few-shot natural data, we only utilize 100 sampled human-written examples. **T5-FewNatural** is directly fine-tuned with these examples. **T5-LoRA** (Hu et al., 2022) is a parameter-efficient fine-tuning method. **Codex-InContext** is an in-context learning method to generate the logical form by seeing a prompt consist of several exemplars (Shin and Van Durme, 2022; Chen et al., 2021). We use 20 exemplars in the prompt, which are retrieved from the 100 natural examples based on the embedding similarity to the given input.

For synthetic data, we utilize two strong baselines. **T5-Synthetic** is directly trained on the synthetic data. **Semantic Searching** re-ranks the generated logical forms according to the semantic similarity with the input questions (Berant and Liang, 2014; Marzoev et al., 2020).

4.3 Experiment Settings

For each dataset, we utilize the exact match accuracy (EM) of logical forms as evaluation metric. When evaluating each method, we early stop the training according to the performance on the development set, and then report the results on the test set. Considering the low-resource settings, the development sets consist of synthetic data, while the test sets are annotated by human experts.

Unless otherwise stated, we employ T5-large as the backbone of the semantic parser for the baseline methods and our method. During inference, we use a beam search of size 5 for generating logical forms. **Codex-InContext** leverages Davinci Codex (Chen et al., 2021) as the backbone.

Following AutoQA (Xu et al., 2020c), we augment the templated questions with a paraphrasing model, which is based on T5-large and trained for 2 million steps on 5 million sentence pairs from PARABANK 2 (Hu et al., 2019). The synthetic training data contains five synonymous questions paraphrased from each templated question. After augmentation, there is 471k and 93k synthetic training data for KQA and CWQ, respectively.

As described in Section 3.1, we employ Codex and greedy decoding for sentence decomposition. We select about 10 synthetic questions to compose the prompt for in-context learning. The selected questions are diverse and complex enough to demonstrate our requirements. Due to the query limitation of Codex API, decomposing all the synthetic training questions is time-consuming. As a trade-off between effectiveness and efficiency, we only use Codex to decompose 5,000 synthetic questions, which are used to train a T5-large based decomposer to imitate the behavior of Codex. The T5-decomposer completes the decomposition of

Method	CWQ Short	CWQ Long	CWQ All	KQA Short	KQA Long	KQA All
(Upper Bound) Full Natural Data						
T5-FullNatural	59.2	37.8	54.9	92.2	79.4	86.4
100 Natural Data						
T5-FewNatural	5.6	6.5	5.8	32.6	7.3	21.2
T5-LoRA	5.9	6.0	5.9	29.0	6.0	18.6
Codex-InContext	17.8	19.9	18.2	44.8	19.3	33.3
Large-scale Synthetic Data						
T5-Synthetic	46.6	24.4	42.1	77.8	57.9	68.8
Semantic Searching	42.9	25.4	39.4	75.2	55.7	66.4
Our Method	**54.3**	**37.8**	**51.0**	**85.8**	**65.6**	**76.7**

Table 2: Evaluation results of low-resource semantic parsing. The metric is the exact match accuracy (EM) of logical forms. The test set for each dataset is divided into two subsets, according to the number of predicates in logical forms. For both CWQ and KQA, an example is allocated into the **Short** set if it contains no more than 4 predicates, otherwise into the **Long** set.

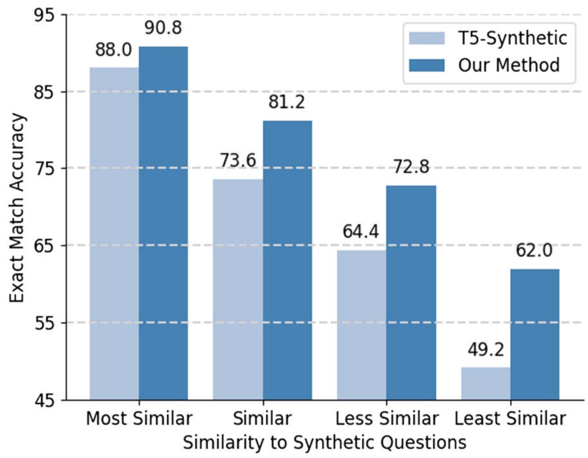

Figure 3: The exact match accuracy of logical forms on the natural questions of KQA test set, grouped by their similarity to synthetic questions. Samples with lower similarity are novel in expressions and thus harder for models.

all the synthetic training questions. Note that the questions of development sets and test sets are decomposed by Codex unless otherwise specified.

4.4 Main Results: Low-resource Semantic Parsing

Table 2 shows the results on KQA and CWQ. Among the methods using synthetic data or few-shot natural data, our method achieves the best performance. Compared to the powerful baseline, **T5-Synthetic**, our method substantially bridges the gap of the training performance on synthetic and natural data (+8.9% on CWQ and +7.9% on KQA). Additionally, the consistent improvement on both datasets demonstrates the generality of our method for different logical systems and domains.

Our method achieves significant improvements on the **Long** sets, especially on the CWQ dataset. Specifically, the performance of our method on the **CWQ-Long** set is already comparable to that of **T5-Supervised**, which is the upper bound of low-resource methods. This observation indicates that our method largely strengthens the model's ability to understand complex questions.

4.5 Generalization to Natural Questions

To understand how our method improves the parsing, we investigate the model performance on the natural questions while considering their similarity to the synthetic data. We find that our method largely contributes to the parsing accuracy of the questions with novel text expressions, verifying our motivation of bridging the gap between synthetic and natural data.

To demonstrate novelty of natural questions in text expressions, we divide the KQA test set into four subsets with the following steps: (1) For each natural question in the test set, we obtain a synonymous synthetic question provided by the original dataset. We then employ SimCSE (Gao et al., 2021a) to calculate the sentence embeddings of these two questions as well as their cosine similarity. We directly utilize their cosine similarity as the novelty of the natural questions. (2) According to the similarity, we divide all the test examples into four subsets with the same number of samples, in which the natural and synthetic questions are the **most similar**, **similar**, **less similar**, and **least similar**, respectively. In the four subsets, samples with lower similarity are novel in expressions and thus harder for models.

Figure 3 shows the semantic parsing performance on these four subsets. We find that lower similarity indicates a larger gap between the natural questions and synthetic data, leading to lower performance on both models. However, our method consistently outperforms the baseline method, especially on the subset with less similarity. It indicates that our method can effectively reduce the training and test gap for the natural questions with novel text expressions.

# Exemplars	Selection Method	Short	Long	All
5	Manual	85.4	**67.6**	**77.4**
10	Manual	85.4	66.1	76.7
15	Manual	**85.8**	65.6	76.7
15	Random (Trial 1)	84.3	63.6	75.0
15	Random (Trial 2)	**85.8**	63.4	75.7

Table 3: The exact match accuracy (EM) of our method on the KQA test set with different prompts for the decomposer. **# Exemplars** refers to the number of exemplars provided in a prompt. **Selection Method** specifies how we choose the exemplars, including random selection and manual selection. **Short** consists of the examples with no more than 4 predicates, while the examples in **Long** have longer logical forms.

4.6 Analysis on the Prompt for Sentence Decomposition

In the previous experiments, the prompt of the sentence decomposer is composed of 15 manually selected exemplars. To investigate the influence of prompt design, we vary (1) the numbers of exemplars and (2) the selection method of exemplars.

The results on KQA are provided in Table 3. In general, the change of prompts has limited effects on the overall performance of semantic parsing.

In terms of the number of exemplars, we select 5 and 10 out of the original 15 manually selected exemplars, respectively, to form two new prompts. As shown in Table 3 (5/10/15 Exemplars, Selection=Manual), these three prompts lead to comparable parsing performance.

For the selection methods, we compare our manually selected prompt against two random prompts, each of which consists of 15 randomly sampled exemplars. The results show that the two random prompts perform slightly worse on the **Long** set. The possible reason is that our manual selection strategy deliberately chooses diverse and complex questions to cover complicated situations, whereas random selection may contain more simple questions and fail to cover these situations.

4.7 Analysis on the Choice of Decomposition Representation

In this section, we show that, in our method, fact-based decomposition representation is better than traditional representations, such as abstract meaning representation (AMR) and question decomposition meaning representation (QDMR) (Wolfson et al., 2020).

Method	CWQ	KQA
T5-Synthetic	42.1	68.8
Our Method (AMR)	46.8	66.1
Our Method (QDMR)	45.7	71.7
Our Method (Fact)	**51.0**	**76.7**

Table 4: Evaluation results of the best baseline method, T5-Synthetic, and our methods with different decomposition representations. **Fact**-based representation is proposed in Section 3.1 and used in other experiments. The metric is the exact match accuracy of logical form.

Specifically, we replace fact-based representation with AMR and QDMR, and the evaluation results are shown in Table 4. For AMR, we directly utilize an open-source parsing tool[1] as the decomposer. For QDMR, we train another T5-decomposer using the training data from BREAK dataset[2] (Wolfson et al., 2020).

Overall, the variants of our method using AMR or QDMR are slightly better than the best baseline method, T5-Synthetic, but a proper decomposition representation can bring a larger performance improvement. Fact-based representation takes simple natural sentences as elementary units, which retains the powerful expressive ability of natural language. In contrast, AMR and QDMR may be limited by the scope of predefined predicates, where their models will generate unexpected outputs when the tasks contain unseen predicates.

4.8 Analysis on the Quality of Decomposed Facts

While the performance of our method serves as an extrinsic evaluation for the quality of decomposed facts, we are also interested in evaluating their quality intrinsically. In this section, we conduct manual evaluation of the decomposed facts.

We randomly select 200 synthetic questions and 200 natural questions from KQA, and then human experts examine the decomposed facts of each question: (1) whether the decomposition is

[1] `https://github.com/bjascob/amrlib`.

[2] BREAK contains the QDMRs of the natural questions from CWQ. For fair comparison, we remove these natural questions from the training set.

Question	Correct	Minor Error	Incorrect	Unknown
Synthetic	184	9	3	4
Natural	179	11	5	5

Table 5: The quality of the decomposed facts of 200 synthetic questions and 200 natural questions. **Unknown** quality occurs when the three experts cannot reach an agreement through majority vote.

Error Type: Incorrect Coreference (48.1%)

sentence:
For the higher education institution that is the education place of Dick Clark ...
decompose:
FACT1: ENTITY1 is a higher education institution.
FACT2: ENTITY2 is Dick Clark.
(✘) FACT3: ENTITY2 is the education place of ENTITY1.
(✓) FACT3: ENTITY1 is the education place of ENTITY2.

Error Type: Information Loss (38.5%)

sentence:
Which movie is shorter, Tulsa (the one whose narrative location is Oklahoma) or Jack and Jill?
decompose:
FACT1: ENTITY1 is a movie.
FACT2: ENTITY2 is Tulsa.
FACT3: ENTITY2's narrative location is Oklahoma.
FACT4: ENTITY3 is Jack and Jill.
(✘) Question: Which ENTITY1 is shorter?
(✓) Question: Which ENTITY1 is shorter, ENTITY2 or ENTITY3?

Table 6: Two main types of errors in decomposition results and their proportions.

correct, not exactly correct but with only one error (minor error), or incorrect with more than one errors; and (2) if an error exists, whether it derives from incorrect coreference, information loss, or other aspects. Each decomposition result is annotated by 3 human experts, and we aggregate their annotations using majority vote. The annotations yield moderate levels of agreement, with Fleiss Kappa $\kappa = 0.335$ (Landis and Koch, 1977). The details of manual evaluation can be found in Appendix C.

Table 5 shows the results of human evaluation. Around 90% of the synthetic and natural questions maintain exactly the original semantics after decomposition. The high quality of decomposition indicates that our proposed decomposer has good generalization for natural questions, even our prompt only contains synthetic exemplars.

We also ask the experts to annotate the error type if the decomposing is not exactly correct. The main error types are listed in Table 6, including incorrect coreference (48.1%) and information loss (38.5%). Reducing these errors may further improve the performance of our method, which we reserve for future research.

To illustrate what kinds of generalization we can expect from our method, we provide the decomposition results of five pairs of synthetic and natural questions from KQA (see Table 7). Note that each pair of questions is semantically equivalent but expressed in different ways. The decomposed facts are generated by the sentence decomposer proposed in Section 3.1. We can observe that: (1) Although the natural questions have diverse text expressions, most of them can still be decomposed properly, which demonstrates the generalization ability of the sentence decomposer. (2) For both synthetic and natural questions, most of the decomposed facts have simple and similar sentence structure. The simplification of sentence structure bridges the gap between synthetic and natural questions, thus reducing the generalization difficulty of semantic parser.

4.9 Extension: Sentence Embedding Learning

As described in Section 3.3, existing sentence embedding methods tend to assign higher matching scores to structurally similar sentences than semantically similar sentences. A possible hypothesis is that structure features distract the learning of sentence encoder. In this section, we extend our method to sentence embedding learning to show its potential on mitigating the abuse of structure features.

4.9.1 Implementation

We follow the settings and implementation of SimCSE (Gao et al., 2021a), which is trained on English Wikipedia and NLI datasets (Bowman et al., 2015; Williams et al., 2018). Specifically, we conduct unsupervised post-training on sup-simcse-roberta-large[3] in three steps: (1) Extract 17,200 premise sentences from NLI datasets. (2) Decompose each premise into several simpler facts using Codex. (3) Train the sentence encoder, initialized with sup-simcse-roberta-large, via contrastive learning (Equation 1), where each premise takes its decomposed facts as a positive example and the other facts in the batch as negative examples.

[3] `https://github.com/princeton-nlp/SimCSE`.

Synthetic Question	Natural Question
sentence: Which one has the smallest elevation (above sea level) among the mountain range whose elevation (above sea level) is less than 4000 metre **decompose:** FACT1: ENTITY1 is a mountain range. FACT2: ENTITY1's elevation (above sea level) is less than 4000 metre. Question: Which ENTITY1 has the smallest elevation (above sea level)?	**sentence:** Which mountain range whose elevation is under 4000 meters above sea level has the lowest elevation above sea level? **decompose:** FACT1: ENTITY1 is a mountain range. FACT2: ENTITY1's elevation is under 4000 meters above sea level. Question: Which ENTITY1 has the lowest elevation above sea level?
sentence: How many philosophy that is the interest of the human whose CANTIC-ID is a10430969? **decompose:** FACT1: ENTITY1 is the human. FACT2: ENTITY1's CANTIC-ID is a10430969. FACT3: ENTITY2 is ENTITY1's interest. FACT4: ENTITY2 is philosophy. Question: How many ENTITY2?	**sentence:** The person with CANTIC-ID a10430969 has how many interests in the field of philosophy? **decompose:** FACT1: ENTITY1 is a person. FACT2: ENTITY1's CANTIC-ID is a10430969. FACT3: ENTITY2 is ENTITY1's interest. FACT4: ENTITY2 is in the field of philosophy. Question: How many ENTITY2?
sentence: For the province of China that has Suzhou as a part, its ISO 3166-2 code is CN-32, what is the end time **decompose:** FACT1: ENTITY1 is a province of China. FACT2: ENTITY2 is Suzhou. FACT3: ENTITY2 is a part of ENTITY1. FACT4: ENTITY1's ISO 3166-2 code is CN-32. Question: What is the end time of FACT4?	**sentence:** ISO 3166-2 code CN-32 corresponded until what date to the Chinese province that includes Suzhou? **decompose:** FACT1: ENTITY1 is a Chinese province. FACT2: ENTITY1 includes Suzhou. FACT3: ISO 3166-2 code CN-32 corresponded to ENTITY1. Question: Until what date FACT3?
sentence: For Waterloo (the one that is headed by councillor), its population is 104986, what is the time point of it **decompose:** FACT1: ENTITY1 is Waterloo. FACT2: ENTITY1 is headed by councillor. FACT3: ENTITY1's population is 104986. Question: What is the time point of FACT3?	**sentence:** When did the Waterloo led by a councillor record a population of 104,986? **decompose:** FACT1: ENTITY1 is Waterloo. FACT2: ENTITY1 is led by a councillor. FACT3: ENTITY1 recorded a population of 104,986. Question: When did FACT3 happen?
sentence: For the democracy that is imported from Italy, its Human Development Index is 0.752, what is the time point of it **decompose:** FACT1: ENTITY1 is a democracy. FACT2: ENTITY1 is imported from Italy. FACT3: ENTITY1's Human Development Index is 0.752. Question: What is the time point of FACT3?	**sentence:** At what point in time is 0.752 the Human Development Index for the democracy that comes from Italy? **decompose:** FACT1: ENTITY1 is a democracy. FACT2: ENTITY1 comes from Italy. FACT3: 0.752 is the Human Development Index for ENTITY1. Question: At what point in time is FACT3?

Table 7: Synthetic/natural questions from KQA and their decomposed facts, which are generated by Codex.

Query: J. R. R. Tolkien is an employee of University of Leeds, what is the end time		
SimCSE	Candidate	Our Method
0.773	(✓) When did J. R. R. Tolkien stop being an employee of University of Leeds?	**0.848**
0.886	(✘) J. R. R. Tolkien is an employee of University of Oxford, what is the end time	0.781
0.816	(✘) J. R. R. Tolkien is educated at University of Oxford, what is the end time	0.717
Query: Which one has the smallest population among the region of Italy that shares border with Umbria		
SimCSE	Candidate	Our Method
0.902	(✓) Which region of Italy bordering Umbria has the lowest population?	**0.920**
0.922	(✘) Which one has the smallest population among the region of Italy that shares border with Marche	0.862
0.929	(✘) Which one has the smallest population among the region of Italy that shares border with Campania	0.871

Table 8: Two examples of the sentence retrieval task (25k candidate sentences). Each method selects the candidate with the highest cosine similarity as the retrieved result.

We apply the trained sentence encoders on two downstream tasks, semantic textual similarity (STS) and sentence retrieval, without additional fine-tuning. For simplicity, we report the average Spearman correlation on seven STS tasks: STS 2012–2016 (Agirre et al., 2012, 2013, 2014, 2015, 2016), STS Benchmark (Cer et al., 2017), and SICK-Relatedness (Marelli et al., 2014). We define the sentence retrieval task as follows: (1) Extract 12,889 examples from KQA, each of which consists of a synthetic question and a synonymous natural question. Both the query and candidate sets are composed of all the synthetic and natural questions unless otherwise stated. (2) For each question in the query set, retrieve its synonymous question from the candidate set, according to the cosine similarities between sentence embeddings. As stated in Section 3.3, sentence decomposition is only applied during training, while for evaluation, our method directly encodes the

Method	STS Spearman↑	Sentence Retrieval Hit@1↑	Hit@2↑	MR↓	MRR↑
SimCSE	**83.8**	78.1	87.4	3.3	85.0
Our Method	82.5	**91.2**	**95.8**	**1.4**	**94.5**

Table 9: Evaluation of the sentence embeddings learned by SimCSE and our method. Spearman correlation is averaged on seven STS tasks. The sentence retrieval task reports Hit@k, mean rank (MR) and mean reciprocal rank (MRR) as metrics. Hit@k refers to the proportion of the gold sentence occurring in the top-k retrieved sentences.

original sentences where the decomposed facts are not used.

4.9.2 Results

Table 9 shows the results on semantic textual similarity (STS) task and sentence retrieval task, where both the query and candidate sets are composed of all the synthetic and natural questions. Our method achieves competitive performance with SimCSE on STS, while significantly outperforming SimCSE by more than 10 points on Hit@1 of the sentence retrieval task. A potential reason is that the sentence retrieval task contains plenty of synonymous but structurally different questions, which are distracting to SimCSE. By contrast, our method is more focused on semantics rather than structure features (see Table 8 for two examples), which leads to the improved performance. However, the two sentences in a sample of STS usually have similar structures, in which situation our method is not advantageous.

4.9.3 Impact of Sentence Structure on Sentence Retrieval

To investigate the impact of structure features on sentence retrieval, we split the task into four different experiment settings according to the data sources for the query set and candidate set (see Figure 4). For example, (Natural, All) indicates that the query set consists of natural questions, while the candidate set contains all synthetic and natural questions. We can observe that: (1) Synthetic query has more distractors in the candidate set, which have different semantics but similar sentence structure. The retrieval performance degrades greatly when the experiment setting changes from (Natural, All) to (Synthetic, All). (2) The candidates from the same source as the query are more likely to mislead models.

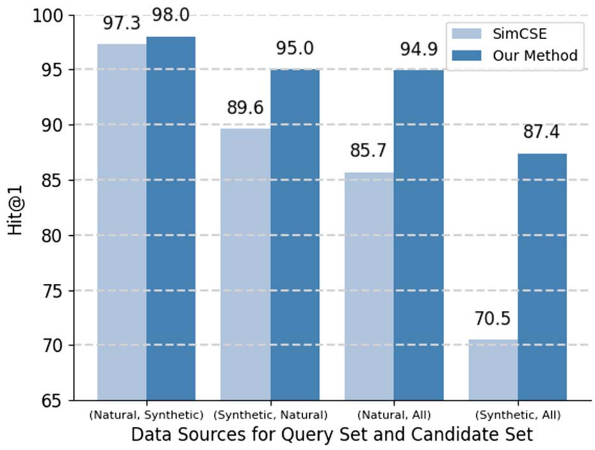

Figure 4: Hit@1 of sentence retrieval with four different experiment settings. Each setting corresponds to a pair of data sources for the query set and candidate set. For example, (Natural, All) represents that the query set consists of natural questions while the candidate set consists of all synthetic and natural questions.

For example, compared to (Natural, Synthetic), the performance of SimCSE on (Natural, All) drops significantly. (3) Our method is more robust against the distractors in all these settings.

5 Limitations

In this section, we enumerate three limitations of our method: applicability, error propagation, and inference speed. (1) Our method is designed to improve the robustness of models to distracting structure features, such as the structure gap between synthetic and natural questions. If there are no distracting structure features, such as on STS datasets, where the two sentences in a sample usually have similar structures, our method may not achieve an advantage. (2) Our method needs to decompose the input text, which may introduce errors that mislead the training and inference of models. (3) If sentence decomposition is also conducted during inference, as in the semantic parsing method proposed in this paper, it may decrease the inference speed of models.

6 Conclusion

We present a decomposition-based low-resource semantic parsing method, which can narrow the data gap and thus better generalize from synthetic data to natural data. Unlike previous methods that tackle the issue by diversifying synthetic questions, we leverage sentence decomposition to break questions into simple facts, which are more

Method	CWQ	KQA
T5-Synthetic	42.1	68.8
Our Method		
T5-decomposer	49.8	76.5
Codex-decomposer	**51.0**	**76.7**

Table 10: The exact match accuracy (EM) of our method on the test sets of CWQ and KQA with different decomposers used in the test phase. T5-Synthetic is the best baseline method.

effective in bridging the gap of sentence-level structure. Experiment results on two semantic parsing datasets show that our method achieves the best performance compared with strong baselines, and it can better generalize to the natural questions with novel text expressions. Besides semantic parsing, our method also has great potential to mitigate distracting structure features on other tasks of semantic understanding, for example, sentence embedding learning. As future work, we plan to explore more tasks that require complex sentence understanding, such as question answering, and paraphrase detection and generation.

A Analysis on the Choice of Decomposer in the Test Phase

In the previous experiments, the synthetic questions of training sets are decomposed by the T5-decomposer due to the query limitation of the Codex API, while the natural questions of test sets are decomposed by Codex for better generalization. In this section, we explore the impact of different decomposers on the performance of semantic parsing in the test phase.

Table 10 shows the semantic parsing performance based on T5- and Codex-decomposers. There is no significant performance drop when Codex is replaced by T5-decomposer in the test phase. We can speculate that, slightly weaker than Codex, the T5-decomposer also generalizes well to natural questions, even though it is only trained on synthetic data. This may be because the decomposition representation is composed of simple natural sentences, which makes it easy for models to learn and generalize. In addition, as mentioned in Section 3.1, the exemplars used for in-context learning are produced through human-Codex collaboration, so the decomposition representation may be in line with the generation bias of language models.

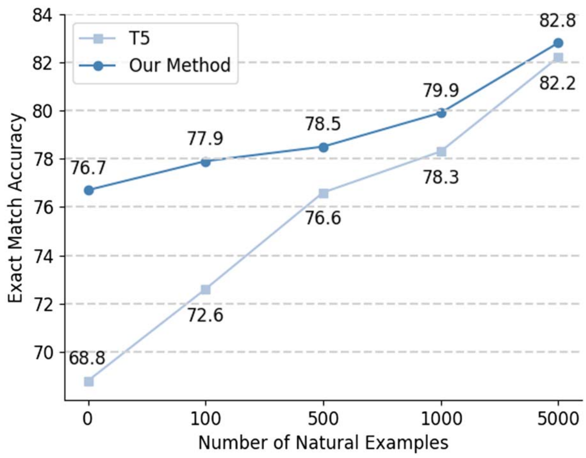

Figure 5: The performance of two methods with different number of natural examples available in training on KQA. Each method first trains a semantic parser with large-scale synthetic data (472k examples), and then fine-tunes it with the natural data.

B Combination of Synthetic Data and Few-shot Natural Data

In this section, we investigate the model performance on KQA when a few natural data are available for training together with synthetic data. The semantic parser is first trained on the large-scale synthetic data, and then further fine-tuned with the natural data. We vary the number of natural examples in a wide range from 100 to 5,000 and compare our method against T5.

As shown in Figure 5, our method consistently outperforms the T5 baseline with no more than 5,000 natural examples. It demonstrates that our pipeline (decomposing and then parsing) would not introduce serious error accumulation in the low-resource setting.

The improvement of our method compared to T5 becomes weaker with more natural data—this is because there is a smaller gap between training and test when more natural training data is available.

C Details of Manual Evaluation

For manual evaluation, we randomly select 200 synthetic questions and 200 natural questions from KQA. Six human experts, including two of the authors and four colleagues, examine the decomposed facts of each question. These six experts are all NLP researchers, who are expected to provide high-quality annotations. Each expert examines 100 synthetic questions and 100 natural questions,

Annotation Instruction

Whether the collective meaning of the decomposed facts is consistent with the original question (three levels of semantic consistency):
Level 2: the collective meaning is consistent with the original question.
Level 1: the collective meaning may be not exactly the same as the original question, or correcting a single error can achieve semantic consistency.
Level 0: the collective meaning is different from the original question, and correcting a single error cannot achieve semantic consistency.
 	
If there exist errors, annotate the error types (three possible error types) (note that an example may have more than one error type):
Error Type 1: Incorrect Coreference
    Incorrect usage of noun phrases or placeholders (i.e., ENTITY and FACT).
    Such errors can be corrected by replacing noun phrases or placeholders.
Error Type 2: Information Loss
    Missing some key information in the original question, such as description of time, place, affiliation, etc.
    Such errors can be corrected by extending a fact or adding a new fact.
Error Type 3: Others

Annotation Examples
Question: Of counties in the Texas led by a Secretary of State, which occupies the most area? Decomposed Facts: Of counties in the Texas led by a Secretary of State, which occupies the most area? FACT2: A Secretary of State is the leader of ENTITY1. Question: Which ENTITY1 occupies the most area? Whether the question is correctly decomposed(0/1/2):2 (optional) Error type (1/2/3): (Explanation: the question is correctly decomposed, so there is no need to annotate the error type.)
Question: Of counties in the Texas led by a Secretary of State, which occupies the most area? Decomposed Facts: FACT1: ENTITY1 is a county in the Texas. FACT2: A Secretary of State is the leader of ENTITY1. (✘) Question: Which FACT2 occupies the most area? (Correction: replace ‘‘FACT2’’ with ‘‘ENTITY1’’.) Whether the question is correctly decomposed(0/1/2):1 (Explanation: only one error need to be corrected.) (optional) Error type (1/2/3):1
Question: Of counties in the Texas led by a Secretary of State, which occupies the most area? Decomposed Facts: (✘) FACT1: ENTITY1 is a county in a Secretary of State. (Correction: replace ‘‘a Secretary of State’’ with ‘‘the Texas’’.) (✘) FACT2: The Texas is the leader of ENTITY1. (Correction: replace ‘‘the Texas’’ with ‘‘a Secretary of State’’.) Question: Which ENTITY1 occupies the most area? Whether the question is correctly decomposed(0/1/2):0 (Explanation: two errors need to be corrected.) (optional) Error type (1/2/3):1
Question: Which one is higher, Tom or Jerry? Decomposed Facts: FACT1: ENTITY1 is Tom. FACT2: ENTITY2 is Jerry. (✘) Question: Which ENTITY1 is higher? (Correction: add the missing information ‘‘ENTITY2’’. ‘‘Which one, ENTITY1 or ENTITY2, is higher?’’) Whether the question is correctly decomposed(0/1/2):1 (Explanation: only one error need to be corrected.) (optional) Error type (1/2/3):2

Table 11: The core part of the annotation guideline provided to the experts.

and each question is examined by three experts for majority vote.

The core part of the annotation guideline provided to the experts is shown in Table 11.

Acknowledgments

This paper was supported by the National Science Foundation for Distinguished Young Scholars (with grant no. 62125604) and the NSFC projects (Key project with grant no. 61936010). This work was also supported by the Guoqiang Institute of Tsinghua University, with grant no. 2019GQG1 and 2020GQG0005, and sponsored by Tsinghua-Toyota Joint Research Fund. We are grateful to our action editor, and the anonymous reviewers for their valuable suggestions and feedback.

References

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel M. Cer, Mona T. Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Iñigo Lopez-Gazpio, Montse Maritxalar, Rada Mihalcea, German Rigau, Larraitz Uria, and Janyce Wiebe. 2015. Semeval-2015 task 2: Semantic textual similarity, english, spanish and pilot on interpretability. In *SemEval@NAACL-HLT*, pages 252–263. https://doi.org/10.18653/v1/S15-2045

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel M. Cer, Mona T. Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2014. Semeval-2014 task 10: Multilingual semantic textual similarity. In *SemEval@COLING*, pages 81–91. https://doi.org/10.3115/v1/S14-2010

Eneko Agirre, Carmen Banea, Daniel M. Cer, Mona T. Diab, Aitor Gonzalez-Agirre, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2016. Semeval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In *SemEval@NAACL-HLT*, pages 497–511.

https://doi.org/10.18653/v1/S16-1081

Eneko Agirre, Daniel M. Cer, Mona T. Diab, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. In *SemEval@NAACL-HLT*, pages 385–393.

Eneko Agirre, Daniel M. Cer, Mona T. Diab, Aitor Gonzalez-Agirre, and Weiwei Guo. 2013. *sem 2013 shared task: Semantic textual similarity. In **SEM*, pages 32–43.

Jonathan Berant and Percy Liang. 2014. Semantic parsing via paraphrasing. In *ACL*, pages 1415–1425. https://doi.org/10.3115/v1/P14-1133

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *EMNLP*, pages 632–642. https://doi.org/10.18653/v1/D15-1075

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutsk. 2020. Language models are few-shot learners. In *NeurIPS*.

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 *ACL*, pages 6806–6817.

Shulin Cao, Jiaxin Shi, Liangming Pan, Lunyiu Nie, Yutong Xiang, Lei Hou, Juanzi Li, Bin He, and Hanwang Zhang. 2022. KQA pro: A dataset with explicit compositional programs for complex question answering over knowledge base. In *ACL*, pages 6101–6119. https://doi.org/10.18653/v1/2022.acl-long.422

Claudio Carpineto and Giovanni Romano. 2012. A survey of automatic query expansion in information retrieval. *ACM Computing Surveys*, 4(1):1–50. https://doi.org/10.1145/2071389.2071390

Daniel M. Cer, Mona T. Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. Semeval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *SemEval@ACL*, pages 1–14.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harrison Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Joshua Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021. Evaluating large language models trained on code. *CoRR*, abs/2107.03374.

Javid Dadashkarimi, Alexander R. Fabbri, Sekhar Tatikonda, and Dragomir R. Radev. 2018. Zero-shot transfer learning for semantic parsing. *CoRR*, abs/1808.09889.

Zhenyun Deng, Yonghua Zhu, Yang Chen, Michael Witbrock, and Patricia Riddle. 2022. Interpretable AMR-based question decomposition for multi-hop question answering. In *IJCAI*, pages 4093–4099. https://doi.org/10.24963/ijcai.2022/568

Ruiliu Fu, Han Wang, Xuejun Zhang, Jun Zhou, and Yonghong Yan. 2021. Decomposing complex questions makes multi-hop QA easier and more interpretable. In *EMNLP*, pages 169–180. https://doi.org/10.18653/v1/2021.findings-emnlp.17

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021a. Simcse: Simple contrastive learning of sentence embeddings. In *EMNLP*, pages 6894–6910.

Yanjun Gao, Ting-Hao Huang, and Rebecca J. Passonneau. 2021b. ABCD: A graph framework to convert complex sentences to a covering set of simple sentences. In *ACL/IJCNLP*, pages 3919–3931. https://doi.org/10.48550/arXiv.2106.12027

Ofer Givoli and Roi Reichart. 2019. Zero-shot semantic parsing for instructions. In *ACL*, pages 4454–4464. https://doi.org/10.18653/v1/P19-1438

Yu Gu, Sue Kase, Michelle Vanni, Brian M. Sadler, Percy Liang, Xifeng Yan, and Yu Su. 2021. Beyond I.I.D.: Three levels of generalization for question answering on knowledge bases. In *WWW*, pages 3477–3488. https://doi.org/10.1145/3442381.3449992

Vivek Gupta, Akshat Shrivastava, Adithya Sagar, Armen Aghajanyan, and Denis Savenkov. 2022. Retronlu: Retrieval augmented task-oriented semantic parsing. In *ConvAI@ACL*, pages 184–196. https://doi.org/10.18653/v1/2022.nlp4convai-1.15

Jonathan Herzig and Jonathan Berant. 2018. Decoupling structure and lexicon for zero-shot semantic parsing. In *EMNLP*, pages 1619–1629. https://doi.org/10.18653/v1/D18-1190

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *ICLR*.

J. Edward Hu, Abhinav Singh, Nils Holzenberger, Matt Post, and Benjamin Van Durme. 2019. Large-scale, diverse, paraphrastic bitexts via sampling and clustering. In *CoNLL*, pages 44–54.

Chengyue Jiang, Zijian Jin, and Kewei Tu. 2021. Neuralizing regular expressions for slot filling. In *EMNLP*, pages 9481–9498. https://doi.org/10.18653/v1/2021.emnlp-main.747

Saar Kuzi, Anna Shtok, and Oren Kurland. 2016. Query expansion using word embeddings. In *CIKM*, pages 1929–1932. https://doi.org/10.1145/2983323.2983876

Brenden M. Lake and Marco Baroni. 2018. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *ICML*, pages 2879–2888.

J. Richard Landis and Gary G. Koch. 1977. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174. https://doi.org/10.2307/2529310, PubMed: 843571

Haoran Li, Abhinav Arora, Shuohui Chen, Anchit Gupta, Sonal Gupta, and Yashar Mehdad. 2021a. MTOP: A comprehensive multilingual task-oriented semantic parsing benchmark. In *EACL*, pages 2950–2962.

Zhuang Li, Lizhen Qu, Shuo Huang, and Gholamreza Haffari. 2021b. Few-shot semantic parsing for new predicates. In *EACL*, pages 1281–1291.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. What makes good in-context examples for gpt-3? In *DeeLIO@ACL*, pages 100–114.

Zihan Liu, Genta Indra Winata, Peng Xu, and Pascale Fung. 2021. X2parser: Cross-lingual and cross-domain framework for task-oriented compositional semantic parsing. In *RepL4NLP@ACL-IJCNLP*, pages 112–127. https://doi.org/10.18653/v1/2021.repl4nlp-1.13

Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. 2014. A SICK cure for the evaluation of compositional distributional semantic models. In *LREC*, pages 216–223.

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. https://doi.org/10.48550/arXiv.2004.13645

Tong Niu, Semih Yavuz, Yingbo Zhou, Nitish Shirish Keskar, Huan Wang, and Caiming Xiong. 2021. Unsupervised paraphrasing with pretrained language models. In *EMNLP*, pages 5136–5150. https://doi.org/10.18653/v1/2021.emnlp-main.417

Inbar Oren, Jonathan Herzig, and Jonathan Berant. 2021. Finding needles in a haystack: Sampling structurally-diverse training sets from synthetic data for compositional generalization. In

EMNLP, pages 10793–10809. https://doi.org/10.18653/v1/2021.emnlp-main.843

Panupong Pasupat, Yuan Zhang, and Kelvin Guu. 2021. Controllable semantic parsing via retrieval augmentation. In *EMNLP*, pages 7683–7698. https://doi.org/10.18653/v1/2021.emnlp-main.607

Ethan Perez, Patrick S. H. Lewis, Wen-tau Yih, Kyunghyun Cho, and Douwe Kiela. 2020. Unsupervised question decomposition for question answering. In *EMNLP*, pages 8864–8880. https://doi.org/10.18653/v1/2020.emnlp-main.713

Hoifung Poon and Pedro M. Domingos. 2009. Unsupervised semantic parsing. In *EMNLP*, pages 1–10. https://doi.org/10.3115/1699510.1699512

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. *Journal of Machine Learning Research*, 140:1–140:67.

Subendhu Rongali, Konstantine Arkoudas, Melanie Rubino, and Wael Hamza. 2022. Training naturalized semantic parsers with very little data. In *IJCAI*, pages 4353–4359. https://doi.org/10.24963/ijcai.2022/604

Irina Saparina and Anton Osokin. 2021. Sparqling database queries from intermediate question decompositions. In *EMNLP*, pages 8984–8998. https://doi.org/10.18653/v1/2021.emnlp-main.708

Nathan Schucher, Siva Reddy, and Harm de Vries. 2022. The power of prompt tuning for low-resource semantic parsing. In *ACL*, pages 148–156. https://doi.org/10.18653/v1/2022.acl-short.17

Tom Sherborne and Mirella Lapata. 2022. Zero-shot cross-lingual semantic parsing. In *ACL*, pages 4134–4153. https://doi.org/10.18653/v1/2022.acl-long.285

Richard Shin, Christopher H. Lin, Sam Thomson, Charles Chen, Subhro Roy, Emmanouil Antonios Platanios, Adam Pauls, Dan Klein, Jason Eisner, and Benjamin Van Durme. 2021. Constrained language models yield few-shot semantic parsers. In *EMNLP*, pages 7699–7715. https://doi.org/10.18653/v1/2021.emnlp-main.608

Richard Shin and Benjamin Van Durme. 2022. Few-shot semantic parsing with language models trained on code. In *NAACL*, pages 5417–5425. https://doi.org/10.18653/v1/2022.naacl-main.396

Yibo Sun, Duyu Tang, Nan Duan, Yeyun Gong, Xiaocheng Feng, Bing Qin, and Daxin Jiang. 2020. Neural semantic parsing in low-resource settings with back-translation and meta-learning. In *AAAI*, pages 8960–8967. https://doi.org/10.1609/aaai.v34i05.6427

Alon Talmor and Jonathan Berant. 2018. The web as a knowledge-base for answering complex questions. In *NAACL-HLT*, pages 641–651. https://doi.org/10.18653/v1/N18-1059

Bailin Wang, Mirella Lapata, and Ivan Titov. 2021. Meta-learning for domain generalization in semantic parsing. In *NAACL-HLT*, pages 366–379. https://doi.org/10.18653/v1/2021.naacl-main.33

Yushi Wang, Jonathan Berant, and Percy Liang. 2015. Building a semantic parser overnight. In *ACL*, pages 1332–1342. https://doi.org/10.3115/v1/P15-1129

Yaqing Wang, Subhabrata Mukherjee, Haoda Chu, Yuancheng Tu, Ming Wu, Jing Gao, and Ahmed Hassan Awadallah. 2020. Adaptive self-training for few-shot neural sequence labeling. *CoRR*, abs/2010.03680. https://doi.org/10.1145/3447548.3467235

Nathaniel Weir, Prasetya Utama, Alex Galakatos, Andrew Crotty, Amir Ilkhechi, Shekar Ramaswamy, Rohin Bhushan, Nadja Geisler, Benjamin Hättasch, Steffen Eger, Ugur Çetintemel, and Carsten Binnig. 2020. Dbpal: A fully pluggable NL2SQL training pipeline. In *SIGMOD*, pages 2347–2361. https://doi.org/10.1145/3318464.3380589

Adina Williams, Nikita Nangia, and Samuel R. Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *NAACL-HLT*, pages 1112–1122. https://doi.org/10.18653/v1/N18-1101

Tomer Wolfson, Daniel Deutch, and Jonathan Berant. 2022. Weakly supervised text-to-sql parsing through question decomposition. In *NAACL*. https://doi.org/10.18653/v1/2022.findings-naacl.193

Tomer Wolfson, Mor Geva, Ankit Gupta, Yoav Goldberg, Matt Gardner, Daniel Deutch, and Jonathan Berant. 2020. Break it down: A question understanding benchmark. *Transactions of the Association for Computational Linguistics*, 8:183–198. https://doi.org/10.1162/tacl_a_00309

Shan Wu, Bo Chen, Chunlei Xin, Xianpei Han, Le Sun, Weipeng Zhang, Jiansong Chen, Fan Yang, and Xunliang Cai. 2021a. From paraphrasing to semantic parsing: Unsupervised semantic parsing via synchronous semantic decoding. In *ACL/IJCNLP*, pages 5110–5121. https://doi.org/10.18653/v1/2021.acl-long.397

Zeqiu Wu, Yi Luan, Hannah Rashkin, David Reitter, and Gaurav Singh Tomar. 2021b. CONQRR: Conversational query rewriting for retrieval with reinforcement learning. *CoRR*, abs/2112.08558.

Menglin Xia and Emilio Monti. 2021. Multilingual neural semantic parsing for low-resourced languages. In **SEM*, pages 185–194.

Dongqin Xu, Junhui Li, Muhua Zhu, Min Zhang, and Guodong Zhou. 2020a. Improving AMR parsing with sequence-to-sequence pre-training. In *EMNLP*, pages 2501–2511.

Kun Xu, Lingfei Wu, Zhiguo Wang, Mo Yu, Liwei Chen, and Vadim Sheinin. 2018. Exploiting rich syntactic information for semantic parsing with graph-to-sequence model. In *EMNLP*, pages 918–924. https://doi.org/10.18653/v1/D18-1110

Silei Xu, Giovanni Campagna, Jian Li, and Monica S. Lam. 2020bb. Schema2qa: High-quality and low-cost q&a agents for the structured web. In *CIKM*, pages 1685–1694.

Silei Xu, Sina J. Semnani, Giovanni Campagna, and Monica S. Lam. 2020c. Autoqa: From databases to QA semantic parsers with only synthetic training data. In *EMNLP*, pages 422–434.

Jingfeng Yang, Haoming Jiang, Qingyu Yin, Danqing Zhang, Bing Yin, and Diyi Yang. 2022a. SEQZERO: Few-shot compositional semantic parsing with sequential prompts and zero-shot models. In *NAACL*, pages 49–60. https://doi.org/10.18653/v1/2022.findings-naacl.5

Kevin Yang, Olivia Deng, Charles Chen, Richard Shin, Subhro Roy, and Benjamin Van Durme. 2022b. Addressing resource and privacy constraints in semantic parsing through data augmentation. In *ACL*, pages 3685–3695. https://doi.org/10.18653/v1/2022.findings-acl.291

Wei Yang, Peng Xu, and Yanshuai Cao. 2021. Hierarchical neural data synthesis for semantic parsing. *CoRR*, abs/2112.02212.

Tao Yu, Chien-Sheng Wu, Xi Victoria Lin, Bailin Wang, Yi Chern Tan, Xinyi Yang, Dragomir R. Radev, Richard Socher, and Caiming Xiong. 2021. Grappa: Grammar-augmented pre-training for table semantic parsing. In *ICLR*.

Wenting Zhao, Konstantine Arkoudas, Weiqi Sun, and Claire Cardie. 2022. Compositional task-oriented parsing as abstractive question answering. In *NAACL*, pages 4418–4427. https://doi.org/10.18653/v1/2022.naacl-main.328

Victor Zhong, Mike Lewis, Sida I. Wang, and Luke Zettlemoyer. 2020. Grounded adaptation for zero-shot executable semantic parsing. In *EMNLP*, pages 6869–6882. https://doi.org/10.18653/v1/2020.emnlp-main.558