# New Defined Semantic Parsing and Text-to-SQL Tasks

Tao Yu, Dragomir Radev PhD

Department of Computer Science, Yale University

LILY Lab

## Introduction

Semantic Parsing is one of most important tasks in natural language understanding (NLU). It maps natural language to meaningful executable programs such as logic forms, SQL and even Python code. It has been studied for decades. Some of current state-of-art methods adapt basic seq2seq encoder-decoder architectures and are able to achieve over 80\% exact matching accuracy on even on complex benchmarks such as ATIS and GeoQuery. These simple but efficient models seem to already solve the most of problems in this field. However, we argue that most of current systems conduct the task of semantic matching instead of semantic parsing. They fail to parse the meaning of the sentences and generalize to unseen programs and datasets. To solve this problem, we introduce a large semantic parsing and text-to-SQL corpus including about 200 databases and more than 10,000 human labeled complex SQL queries to the research community. Also, we define a more realistic and challenging semantic parsing/text-to-SQL task based on this new dataset.

## Problems in Prior Works

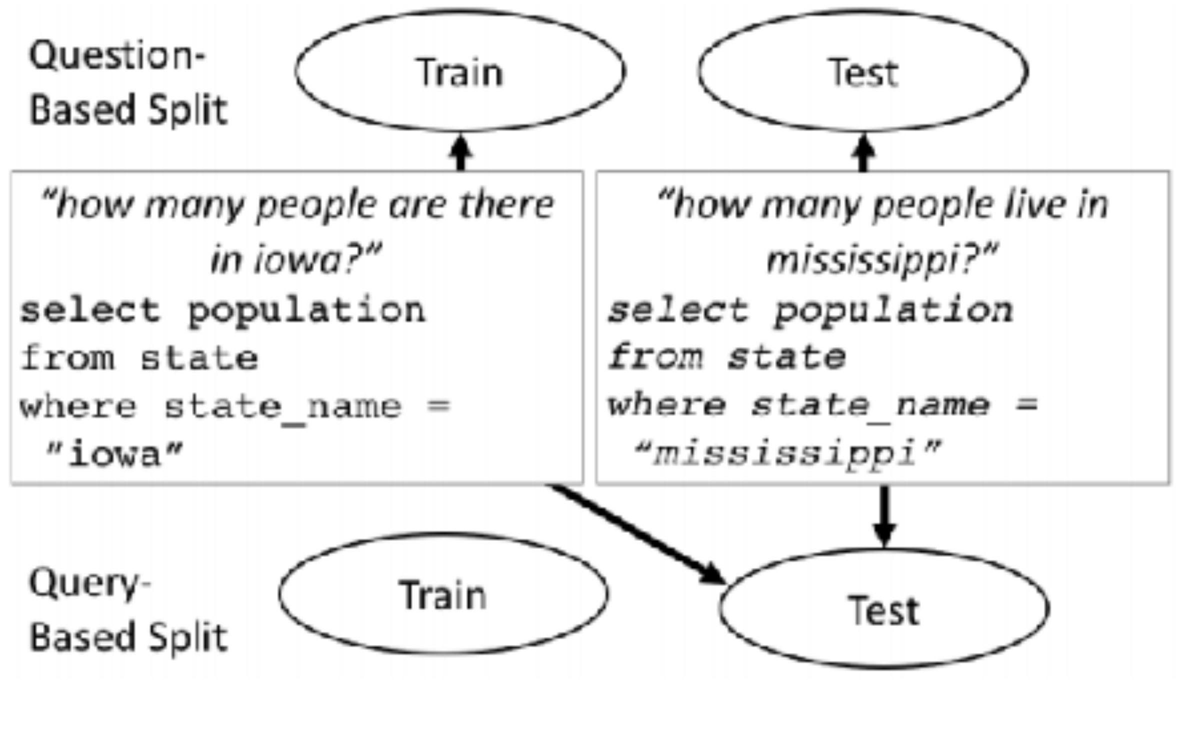
Most of previous work on text-to-SQL use either simple datasets or simple train-test splits on a single database. WikiSQL is an example of a simple SQL dataset. Its databases only contain single tables. This leads to the fact that corresponding SQL queries do not have complex structures such as JOIN and GROUP BY etc. On the other hand, (Iyer et al, 2017) applied a basic seq2seq model on the seq2SQL task with very complex SQL queries. However, their experiments are on one single database and use question-splitting datasets during training and testing. Thus, their model memorizes database-specific SQL templates and only needs to decide which template to use during testing. Unsurprisingly, it performs very bad on unseen SQL queries despite having high test accuracy under the question-splitting setting on one single database. To avoid this issue, as shown in figure 1, Cathy et al. proposed a new way of splitting datasets so that the same SQL queries do not appear in both train and test data. The template-based approach fails under this query-splitting setting.

**Table 1. Results on WikiSQL Task**

| | Dev | | | Test | | |
|---|---|---|---|---|---|---|
| | $Acc_{lf}$ | $Acc_{qm}$ | $Acc_{ex}$ | $Acc_{lf}$ | $Acc_{qm}$ | $Acc_{ex}$ |
| Content Insensitive | | | | | | |
| Dong and Lapata (2016) | 23.3% | - | 37.0% | 23.4% | - | 35.9% |
| Augmented Pointer Network (Zhong et al., 2017) | 44.1% | - | 53.8% | 42.8% | - | 52.8% |
| Seq2SQL (Zhong et al., 2017) | 49.5% | - | 60.8% | 48.3% | - | 59.4% |
| SQLNet (Xu et al., 2017) | - | 63.2% | 69.8% | - | 61.3% | 68.0% |
| TypeSQL w/o type-awareness (ours) | - | 66.5% | 72.8% | - | 64.9% | 71.7% |
| TypeSQL (ours) | - | **68.0%** | **74.5%** | - | **66.7%** | **73.5%** |
| Content Sensitive | | | | | | |
| Wang et al. (2017a) | 59.6% | - | 65.2% | 59.5% | - | 65.1% |
| TypeSQL+TC (ours) | - | **79.2%** | **85.5%** | - | **75.4%** | **82.6%** |

**Table 2. Results on Different Dataset using Query-splitting  Evaluations**

| | Advising | | ATIS | | GeoQuery | | Scholar | |
|---|---|---|---|---|---|---|---|---|
| Split | Quest. | Query | Quest. | Query | Quest. | Query | Quest. | Query |
| seq2seq | 4.4 | 0.0 | 7.6 | 0.0 | 31.5 | 3.3 | 19.3 | 0.0 |
| + Attention | 25.5 | 0.0 | 45.9 | 17.9 | 53.8 | 22.0 | 32.6 | 0.0 |
| + Copying | 73.7 | 0.3 | 50.8 | 32.0 | 68.5 | 30.8 | 59.2 | 5.4 |
| D & L seq2seq | 32.3 | 0.0 | 38.9 | 7.2 | 61.3 | 24.2 | 43.6 | 7.6 |
| D & L seq2tree | 41.5 | 0.0 | 46.3 | 23.1 | 62.4 | 27.5 | 44.0 | 6.4 |
| Iyer et al. | 40.9 | 0.5 | 44.3 | 17.9 | 62.0 | 27.5 | 28.4 | 1.0 |
| Template LSTM Baseline | 80.4 | 0.0 | 46.2 | 0.0 | 55.7 | 0.0 | 51.9 | 0.0 |
| Template LSTM Oracle | 99.0 | 0.0 | 69.4 | 0.0 | 77.8 | 0.0 | 83.5 | 0.0 |



"how many people are there in iowa?"

```
select population
from state
where state_name =
"iowa"
```

"how many people live in mississippi?"

```
select population
from state
where state_name =
"mississippi"
```

**Figure 1.** Query-based Splitting Evaluation

## New Semantic Parsing Task Definition

In the real world, however, we would like to know how good the seq2SQL model performs not only on unseen queries but also on unseen databases. In this project, we are going to introduce the most realistic seq2SQL task and explore new methods to solve this problem. First, we will label a new corpus including about 10000 SQL-question pairs for about 200 databases with multiple tables. In this task, we split the dataset in a way so that different databases are seen during training and testing. Second, we are going to discover new seq2SQL approaches that take not only questions and SQL pairs but also table schemas and database structures as inputs. How can we learn a seq2sql parser and generalize well to new databases?

## Results

In table 1, we can find the overall results on WikiSQL task where models have to generalize to new databases.  But the task suffers from simplification of SQL and database schema. Table 2 shows basic seq2seq generation models such as (Iyer et al., 2017) can achieve descent accuracy on very complex SQL queries on previous semantic parsing evaluation matrix but fail on predicting unseen queries. Current approaches memorize table-specific SQL templates on Question-splitting datasets (the same SQL queries appear in both train and test datasets). They just have to decide which templates to use and replace condition values during testing. Also, template based approaches can get higher results. These results in two tables show we still have a long way to go in order to develop a semantic parsing system that can generalizes to new dataset and parse complex questions into programs.

## Conclusion

We show that there are some problems in current semantic parsing task definition and evaluation. The results show we still have a long way to go in order to develop a semantic parsing system that can generalizes to new dataset and parse complex questions into programs. As one of our contributions to this research area, we introduce a large corpus that contains many datasets and complex hand-labelled complex queries. Moreover, we introduce a new realistic task which most of current systems fail. We plan to explore different approaches and develop a novel approach to tackle this task in certain degrees.