New Semantic Parsing Task with a Large Text-to-SQL Corpus

Tao Yu
Advisor: Dragomir Radev

LILY Lab

Yale University

4/27/2018

Agenda

1. Task Definition
2. Dataset
3. Methods
4. Tasks

Previous Task Definition

WikiSQL v.s. Others

Most of current semantic parsing/text-to-SQL research:

1. Complex logic forms/SQL:
   1. Logic form labels are complex/SQL covers most of classes such as GROUP BY/JOIN/Nested.
2. But simple/problematic task definition:
   1. Only one single dataset (Geo, ATIS etc.) for both train and test.
   2. The number of logic form/SQL labels is small (500). Each logic form/SQL has about 4 paraphrases of natural language problems. Thus, the same logic form/SQL appear in both train and test.

Problems: Basic seq2seq generation model (Iyer et al., 2017) can achieve descent accuracy on very complex SQL queries. Template based approaches can get even higher results (Cathy et al., 2018).

Previous Task Definition

WikiSQL v.s. Others

WikiSQL (Zhong et al., 2017)

1. Good problem definition:
   1. Databases in the test set do not appear in the train/dev set, which requires model to generalize to new databases. Over 20,000 databases.
   2. Models have to generalize to new databases.

2. But suffers from simplification of SQL and database schema.
   1. All databases have only one table. No need to predict table/JOIN.
   2. SQL only contains SELECT and WHERE. No GROUP BY/Nested etc.

Our Task Definition

The Most Realistic Seq2SQL Task

Goal: learn and test systems conducting complex semantic parsing/text-to-SQL task AND generalizing to new datasets/databases.

1. We would like to know how good the semantic parsing/text-to-SQL model performs not only on **unseen programs** but also on **unseen datasets/databases**.
2. Big contribution: we introduce a new large semantic parsing/text-to-SQL human-labeled complex question-SQL corpus!

Datasets

Current and Our Datasets

Table: seq2SQL datasets

Dataset	Q #	SQL #	DB #	Table #	SQL coverage
Geo	877	247	1	6	almost all
Scholar	817	193	1	7	almost all
ATIS	5280	947	1	32	only sel./wh./join
Advising	3898	208	1	10	almost all
WikiSQL	80,654	77,840	26,521	1	only select/where
Ours(est.)	20,000	8,000	200	avg 5	almost all/DB

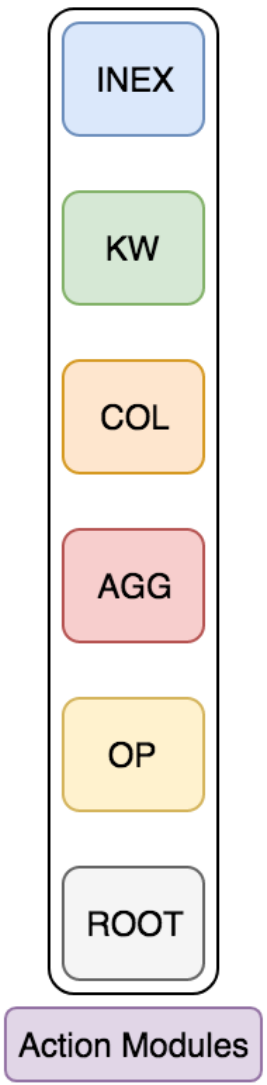
We have to collect, label, and review about 150 databases. SQL labels of each database should cover almost all SQL operators.

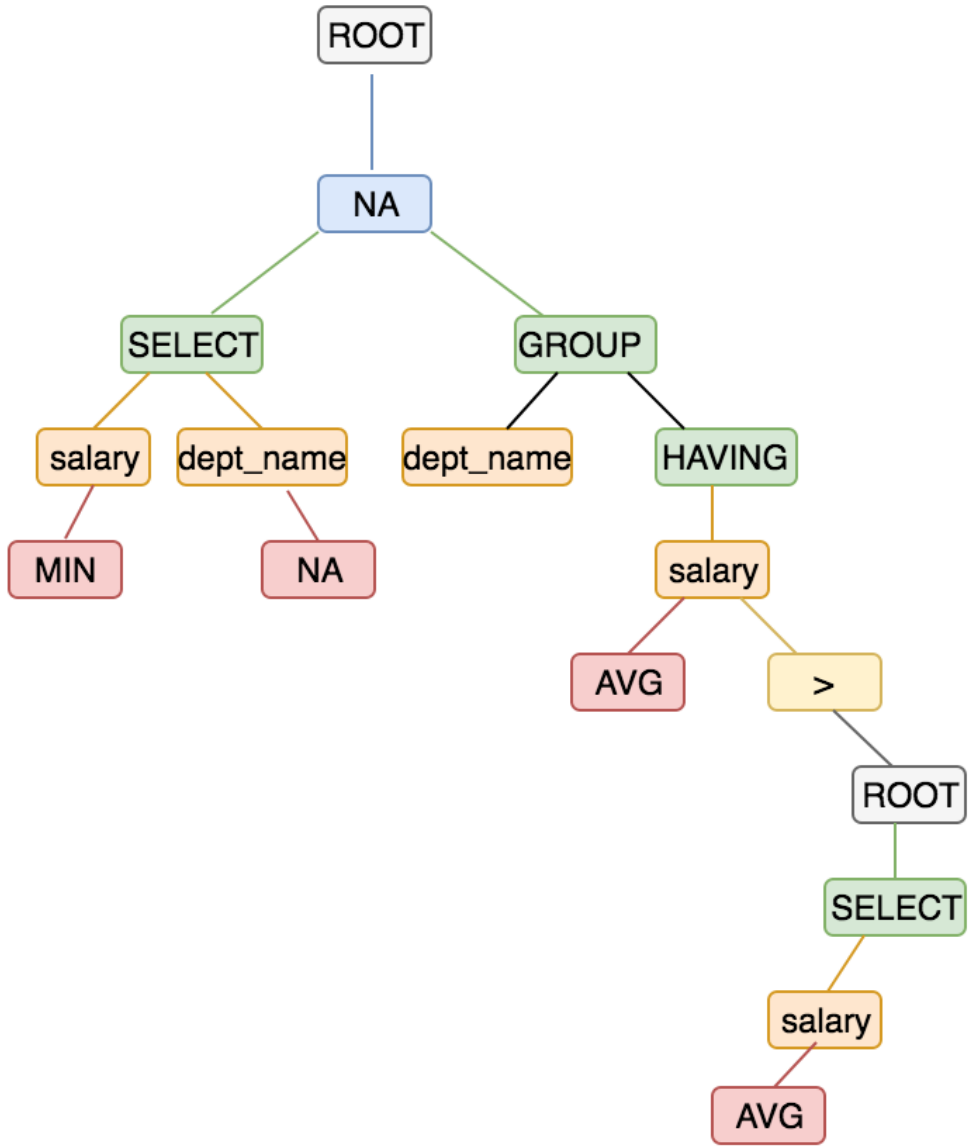
Methods

Other teammates are going to present:

1. D & L seq2seq/tree
2. (Iyer et al., 2017)
3. seq2seq + attention/copying
4. seq2SQL
5. SQLNet
6. seq2seq+set
7. Syntactic Neural Model
8. nli2SQL

Tree-based Method Using Stack





Tree-based Method Using Stack

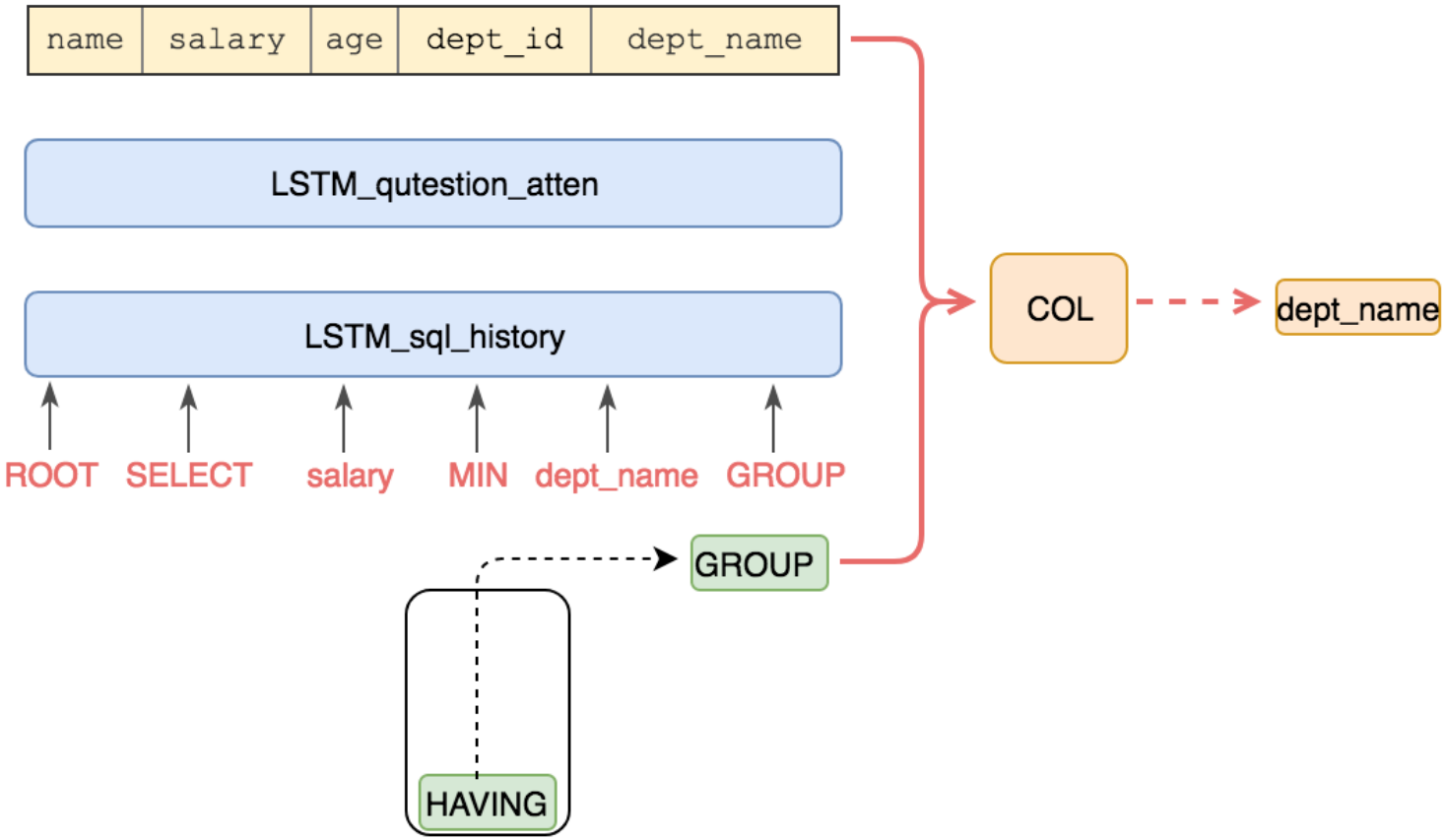


Figure: Action Prediction

Different Evaluation Matrices

First group examples by SQL Hardness criteria, then evaluate on different evaluation matrices:

1. Execution accuracy (most papers used this but not very convincing)
2. Exact matching without values (too strict)
3. Partial matching without values (BLUE score)
4. Detailed scores on different SQL components: accuracy on SELECT, WHERE, GROUP BY, ORDER BY etc.

SQL Hardness Criteria

1. SQL components 1: WHERE, GROUP BY, ORDER BY, LIMIT, JOIN, OR, LIKE
2. SQL components 2: EXCEPT, UNION, INTERSECT, NESTED
3. Counts: # of agg > 2, # of select columns > 2, # of where condition > 2, # of group by clauses > 1, # of group by clauses > 1

SQL Hardness Criteria

1. Easy: if SQL key words ONLY have no or exact ONE from [SQL components 1] and SQL do not satisfy any conditions in [Others] above.
2. Medium: SQL satisfies less than two rules in [Others] and do not have more than one word from [SQL components 1]. OR, SQL has exact TWO words from SQL components 1 and less than 1 ruls in [Others]
3. Hard: SQL satisfies more than two rules in [Others] and no more than 2 key words in [SQL components 1] but no any word in [SQL components 2] ...
4. Extra Hard: ...

Detailed Analysis

Compare Results

1. under different settings
   1. train and test have the same queries and databases
   2. train and test have different queries but same databases
   3. train and test have different queries and databases
2. Compare different models
3. Compare performances on DBs with different table/column # and genres etc.
4. Compare with human performances with different levels on SQL knowledge.

Thanks for everyone working on seq2SQL and SQA projects:

1. Rui, Kai, Michi, Dongxu, Zifan, Qingning, Zilin, and Irene
2. James, Shanelle, Michi, Rui

Thanks for Listening!