Natural Language Interfaces
Text-to-SQL

Tao Yu

11/04/2020

Natural Language Interfaces (NLI)

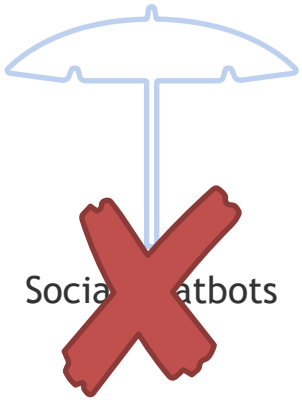
In this presentation, we will focus on interfaces that support performing a useful and concrete task, such as

Search for info in large collections of documents

Booking flights

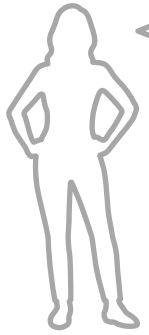
Get insights from statistical data

Instruct domestic robots



Socia atbots

NLI to Databases



User

Which European countries have some players who won the Australian Open at least 3 times?

Database

Table 1: Matches

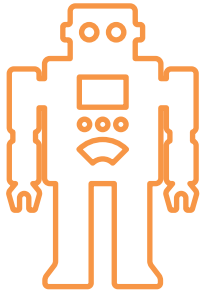
Id	Tourney	Year	Winner id	… …
1	Australian Open	2018	3	… …

Table 2: Ranking

Ranking	Points	Player id	Tours	… …
1	9,985	3	11	… …

Table 3: Players

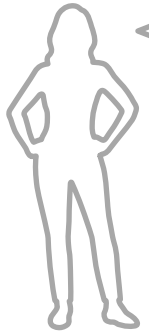
Id	Name	Nation	Continent	… …
1	Djokovic	Serbia	Europe	… …
2	Osaka	Japan	Asia	… …
3	Federer	Switzerland	Europe	… …



Semantic Parser

```
SELECT T1.nation
FROM players AS T1 JOIN matches AS T2
  ON T1.id = T2.winner_id
WHERE T2.Tourney = “Australian Open”
  AND T1.continent = “Europe”
GROUP BY T2.winner_id
HAVING COUNT(*) >= 3
```

NLI to Databases



User

Which European countries have some players who won the Australian Open at least 3 times?

Database

Table 1: Matches

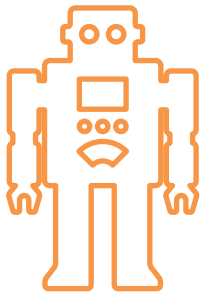
Id	Tourney	Year	Winner id	… …
1	Australian Open	2018	3	… …

Table 2: Ranking

Ranking	Points	Player id	Tours	… …
1	9,985	3	11	… …

Table 3: Players

Id	Name	Nation	Continent	… …
1	Djokovic	Serbia	Europe	… …
2	Osaka	Japan	Asia	… …
3	Federer	Switzerland	Europe	… …



Semantic Parser

```
SELECT T1.nation
FROM players AS T1 JOIN matches AS T2
  ON T1.id = T2.winner_id
WHERE T2.Tourney = “Australian Open”
  AND T1.continent = “Europe”
GROUP BY T2.winner_id
HAVING COUNT(*) >= 3
```

Execute

Table Semantic Parsing Task

Input

User

Which European countries have some players who won the Australian Open at least 3 times?

Database

Table 1: Matches

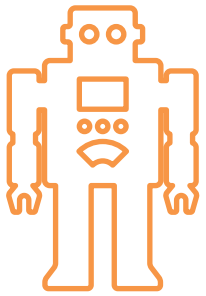
Id	Tourney	Year	Winner id	... ...
1	Australian Open	2018	3	... ...

Table 2: Ranking

Ranking	Points	Player id	Tours	... ...
1	9,985	3	11	... ...

Table 3: Players

Id	Name	Nation	Continent	... ...
1	Djokovic	Serbia	Europe	... ...
2	Osaka	Japan	Asia	... ...
3	Federer	Switzerland	Europe	... ...



Semantic Parser

```
SELECT T1.nation
FROM players AS T1 JOIN matches AS T2
  ON T1.id = T2.winner_id
WHERE T2.Tourney = “Australian Open”
  AND T1.continent = “Europe”
GROUP BY T2.winner_id
HAVING COUNT(*) >= 3
```

Key Task of NLI: Table Semantic Parsing

Input

User

Which European countries have some players who won the Australian Open at least 3 times?

Database

Table 1: Matches

Id	Tourney	Year	Winner id	... ...
1	Australian Open	2018	3	... ...

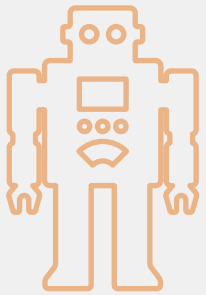
Table 2: Ranking

Ranking	Points	Player id	Tours	... ...
1	9,985	3	11	... ...

Table 3: Players

Id	Name	Nation	Continent	... ...
1	Djokovic	Serbia	Europe	... ...
2	Osaka	Japan	Asia	... ...
3	Federer	Switzerland	Europe	... ...

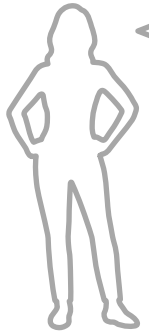
Output



Semantic Parser

```
SELECT T1.nation
FROM players AS T1 JOIN matches AS T2
  ON T1.id = T2.winner_id
WHERE T2.Tourney = "Australian Open"
  AND T1.continent = "Europe"
GROUP BY T2.winner_id
HAVING COUNT(*) >= 3
```

Key Challenges in Table Semantic Parsing



User

Which European countries have some players who won the Australian Open at least 3 times?

Database

Table 1: Matches

Id	Tourney	Year	Winner id	... ...
1	Australian Open	2018	3	... ...

Table 2: Ranking

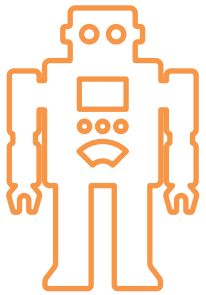
Ranking	Points	Player id	Tours	... ...
1	9,985	3	11	... ...

Table 3: Players

Id	Name	Nation	Continent	... ...
1	Djokovic	Serbia	Europe	... ...
2	Osaka	Japan	Asia	... ...
3	Federer	Switzerland	Europe	... ...

Requires jointly understanding mentions in questions and tables

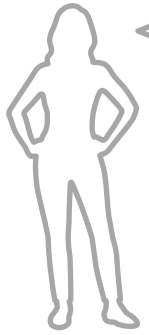
Cell values



Semantic Parser

```
SELECT T1.nation
FROM players AS T1 JOIN matches AS T2
  ON T1.id = T2.winner_id
WHERE T2.Tourney = "Australian Open"
  AND T1.continent = "Europe"
GROUP BY T2.winner_id
HAVING COUNT(*) >= 3
```

Key Challenges in Table Semantic Parsing



User

Which European countries have some players who won the Australian Open at least 3 times?

Database

Table 1: Matches

Id	Tourney	Year	Winner id	... ...
1	Australian Open	2018	3	... ...

Table 2: Ranking

Ranking	Points	Player id	Tours	... ...
1	9,985	3	11	... ...

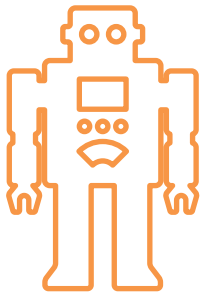
Table 3: Players

Id	Name	Nation	Continent	... ...
1	Djokovic	Serbia	Europe	... ...
2	Osaka	Japan	Asia	... ...
3	Federer	Switzerland	Europe	... ...

Requires jointly understanding mentions in questions and tables

Cell values

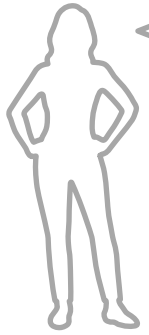
Column names



Semantic Parser

```
SELECT T1.Nation
FROM Players AS T1 JOIN Matches AS T2
  ON T1.id = T2.winner_id
WHERE T2.Tourney = "Australian Open"
  AND T1.Continent = "Europe"
GROUP BY T2.winner_id
HAVING COUNT(*) >= 3
```

Key Challenges in Table Semantic Parsing



User

Which European countries have some players who won the Australian Open at least 3 times?

Database

Table 1: Matches

Id	Tourney	Year	Winner id	... ...
1	Australian Open	2018	3	... ...

Table 2: Ranking

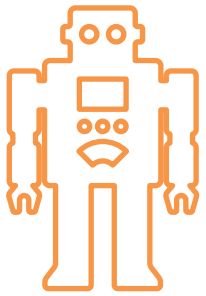
Ranking	Points	Player id	Tours	... ...
1	9,985	3	11	... ...

Table 3: Players

Id	Name	Nation	Continent	... ...
1	Djokovic	Serbia	Europe	... ...
2	Osaka	Japan	Asia	... ...
3	Federer	Switzerland	Europe	... ...

Requires jointly understanding mentions in questions and tables

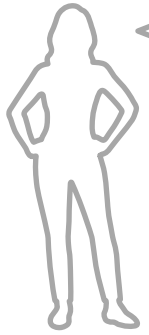
- Cell values
- Column names
- Table names



Semantic Parser

```
SELECT T1.Nation
FROM Players AS T1 JOIN Matches AS T2
  ON T1.id = T2.winner_id
WHERE T2.Tourney = "Australian Open"
  AND T1.Continent = "Europe"
GROUP BY T2.winner_id
HAVING COUNT(*) >= 3
```

Key Challenges in Table Semantic Parsing



User

Which European countries have some players who won the Australian Open at least 3 times?

Database

Table 1: Matches

Id	Tourney	Year	Winner id	... ...
1	Australian Open	2018	3	... ...

Table 2: Ranking

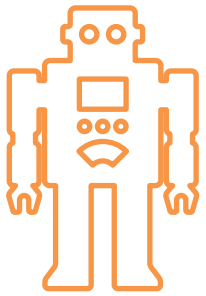
Ranking	Points	Player id	Tours	... ...
1	9,985	3	11	... ...

Table 3: Players

Id	Name	Nation	Continent	... ...
1	Djokovic	Serbia	Europe	... ...
2	Osaka	Japan	Asia	... ...
3	Federer	Switzerland	Europe	... ...

Requires jointly understanding mentions in questions and tables

- Cell values
- Column names
- Table names
- Logic snippets

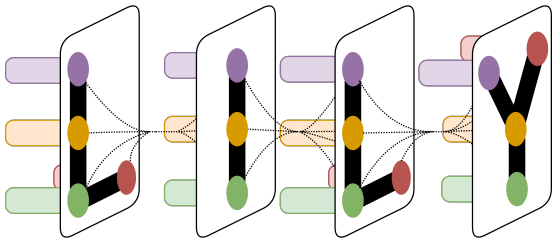


Semantic Parser

```
SELECT T1.Nation
FROM Players AS T1 JOIN Matches AS T2
  ON T1.id = T2.winner_id
WHERE T2.Tourney = "Australian Open"
  AND T1.Continent = "Europe"
GROUP BY T2.winner_id
HAVING COUNT(*) >= 3
```

GraPPa: Grammar-Augmented Pre-Training for Table Semantic Parsing

Salesforce Research

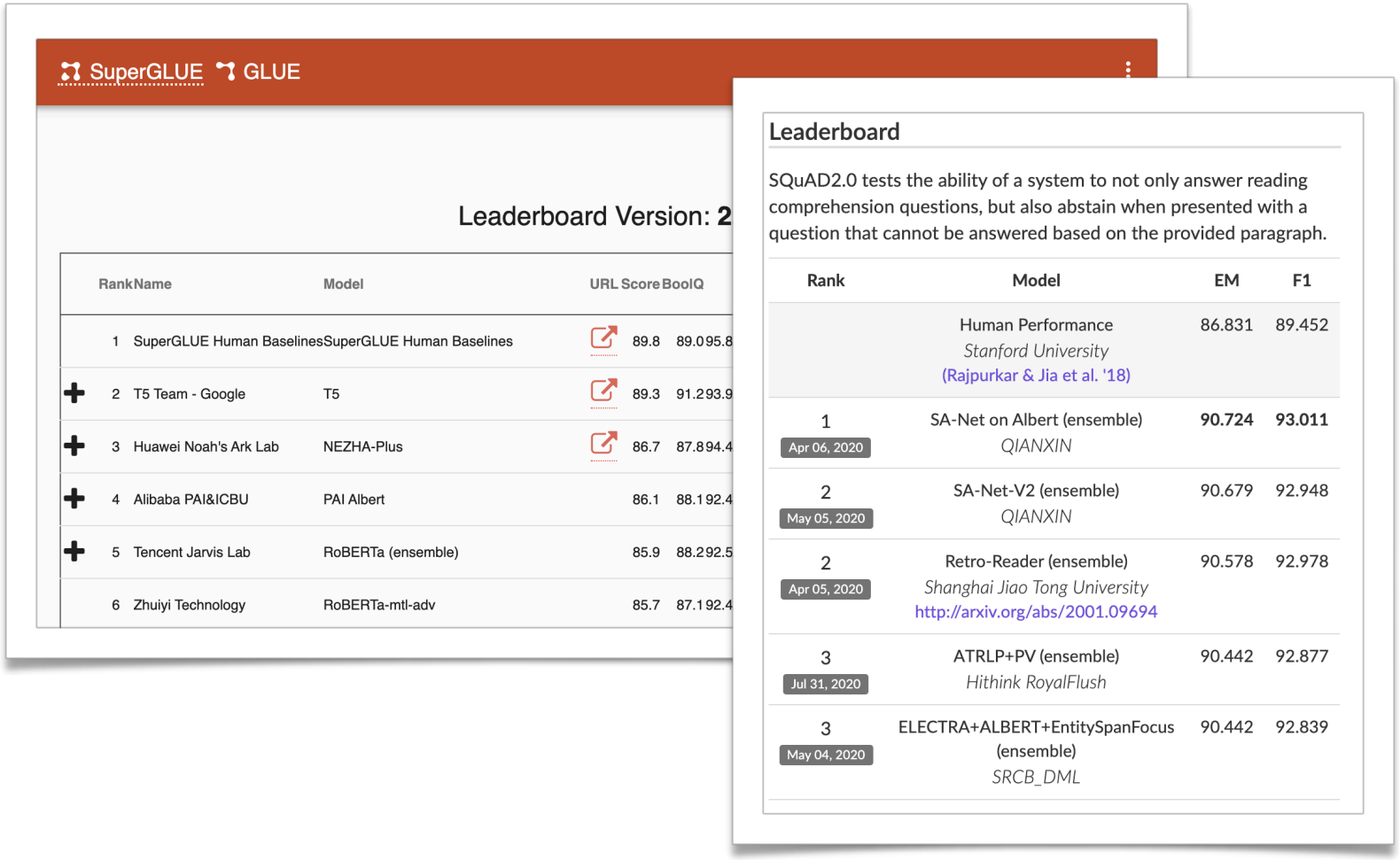


Yale NLP Group

Tao Yu, Chien-Sheng Wu, Xi Victoria Lin,
Bailin Wang, Yi Chern Tan, Xinyi Yang,
Dragomir Radev, Richard Socher, Caiming Xiong

Language Models for Text Understanding

Recent pre-trained language models (LMs) such as BERT and RoBERTa achieve tremendous success on many natural language processing tasks



SuperGLUE GLUE

Leaderboard Version: 2

Rank	Name	Model	URL	Score	BoolQ
1	SuperGLUE Human Baselines	SuperGLUE Human Baselines		89.8	89.0 95.8
2	T5 Team - Google	T5		89.3	91.2 93.9
3	Huawei Noah's Ark Lab	NEZHA-Plus		86.7	87.8 94.4
4	Alibaba PAI&ICBU	PAI Albert		86.1	88.1 92.4
5	Tencent Jarvis Lab	RoBERTa (ensemble)		85.9	88.2 92.5
6	Zhuiyi Technology	RoBERTa-mtl-adv		85.7	87.1 92.4

Leaderboard

SQuAD2.0 tests the ability of a system to not only answer reading comprehension questions, but also abstain when presented with a question that cannot be answered based on the provided paragraph.

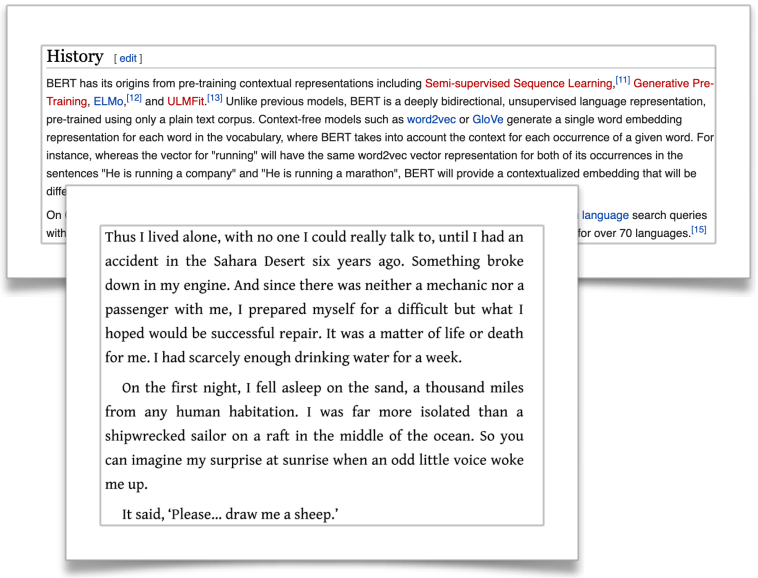
Rank	Model	EM	F1
	Human Performance *Stanford University* (Rajpurkar & Jia et al. '18)	86.831	89.452
1 Apr 06, 2020	SA-Net on Albert (ensemble) *QIANXIN*	**90.724**	**93.011**
2 May 05, 2020	SA-Net-V2 (ensemble) *QIANXIN*	90.679	92.948
2 Apr 05, 2020	Retro-Reader (ensemble) *Shanghai Jiao Tong University* http://arxiv.org/abs/2001.09694	90.578	92.978
3 Jul 31, 2020	ATRLP+PV (ensemble) *Hithink RoyalFlush*	90.442	92.877
3 May 04, 2020	ELECTRA+ALBERT+EntitySpanFocus (ensemble) *SRCB_DML*	90.442	92.839

Language Models for Text Understanding

Most of LMs are pre-trained only on text such as

Wikipedia articles

Books



?

However, in table semantic parsing, the model also has to

Encode tables

Id	Tourney	Year	Winner	... ...
1	Australia	2018	3	... ...

Ranking	Points	Player id	Tours	... ...
1	9,985	3	11	... ...

Id	Name	Nation	Continen	... ...
1	Djokovic	Serbia	Europe	... ...
2	Osaka	Japan	Asia	... ...
3	Federer	Switzerla	Europe	... ...

Jointly understand questions and tables

- Cell values
- Column names
- Table names
- Logic snippets

Figure out logic compositionality

Which European countries have some players who won the Australian Open at least 3 times?

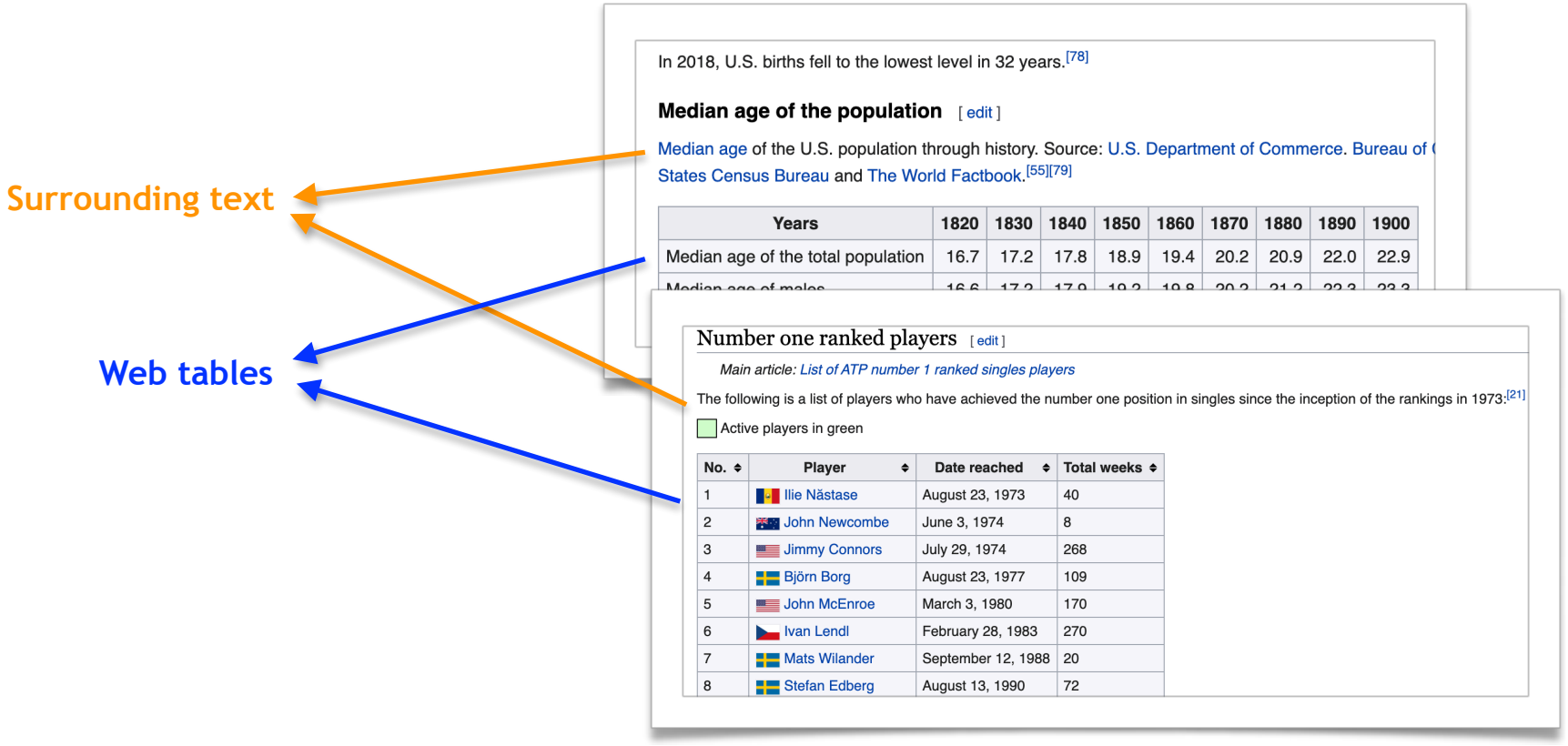
Language Models for Semantic Parsing

Yin et al. (2020) and Herzig et al. (2020) pre-train LMs on over 20M text-table examples extracted from web

Noisy

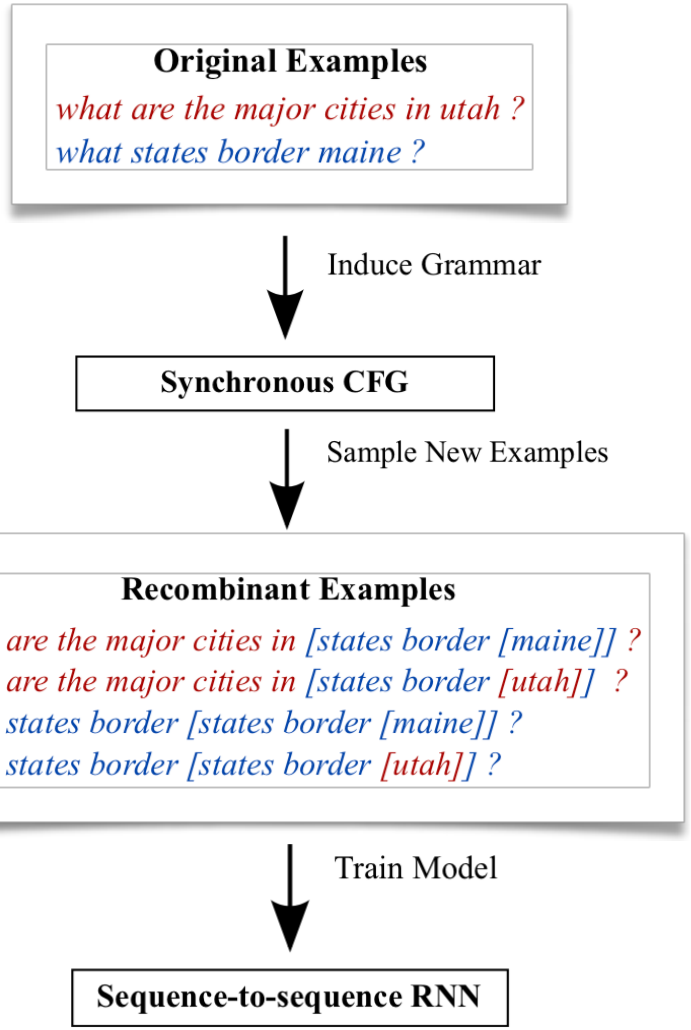
↓

ineffective

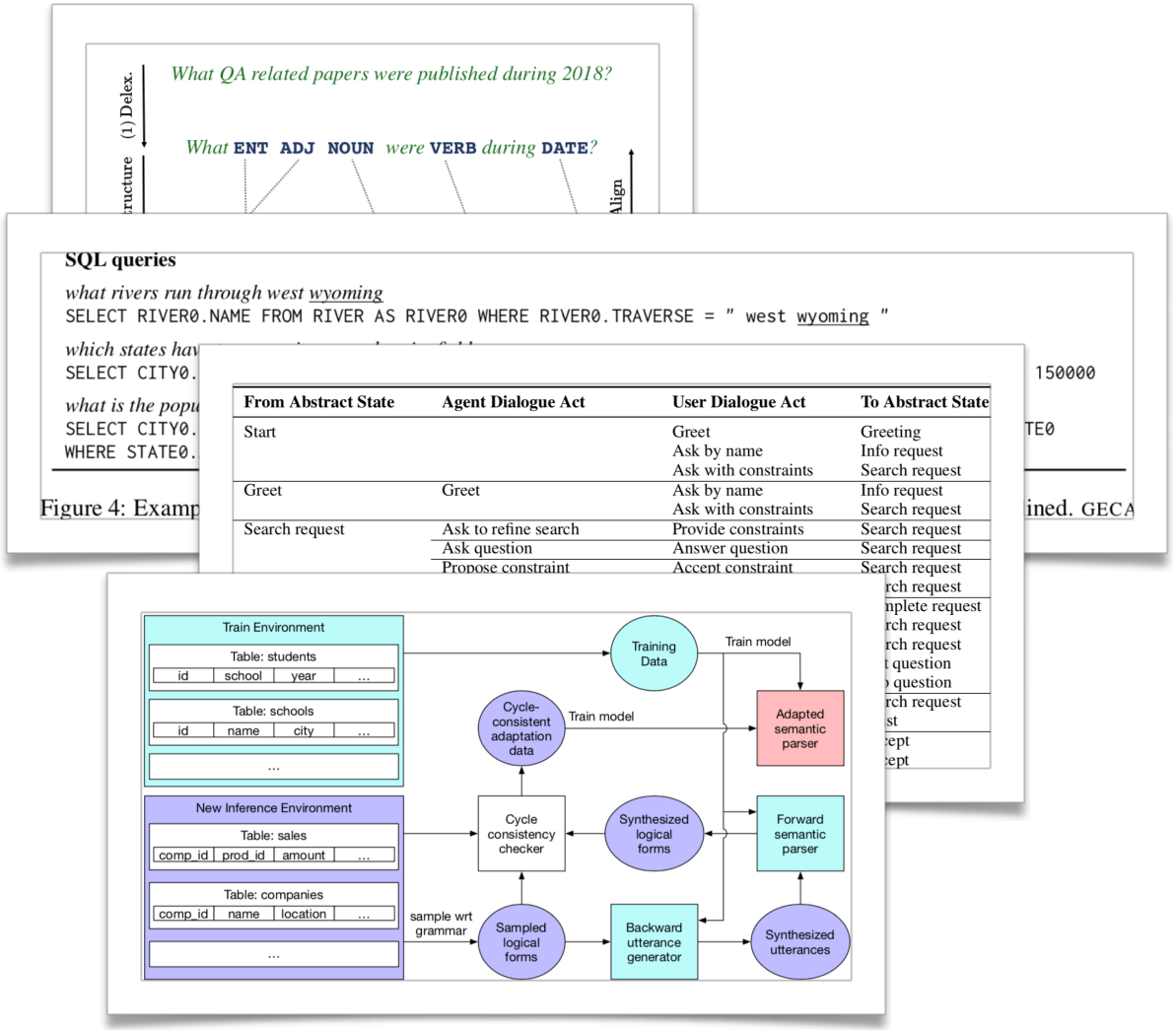


Data Augmentation in Semantic Parsing

Semantic parsing data is more compositional, data augmentation is widely applied to enlarge data size.



(Jia and Liang, 2016)

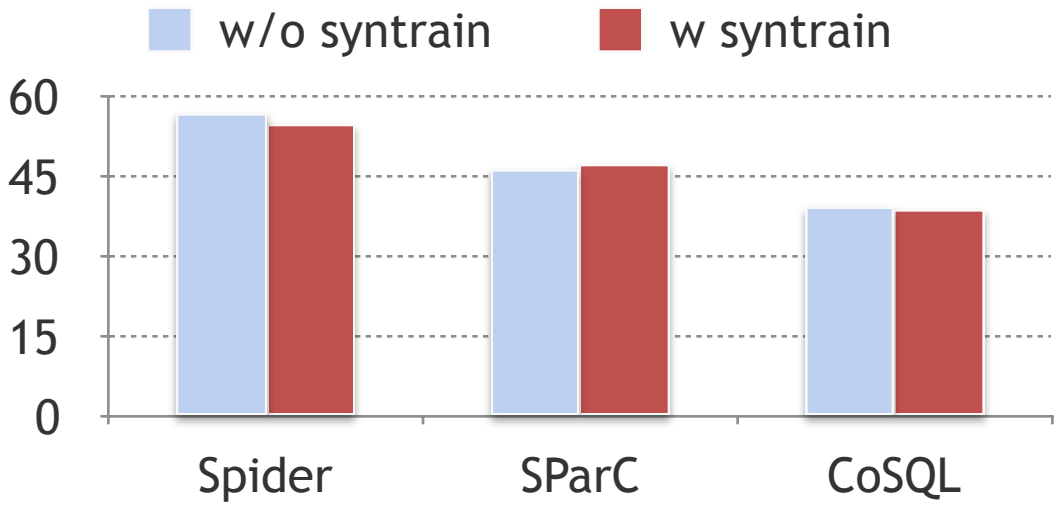


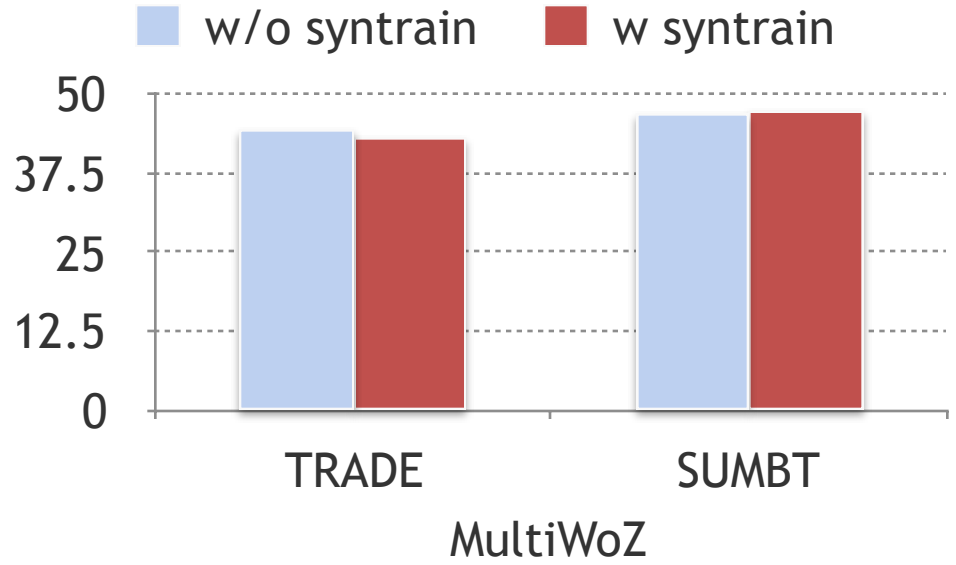
(Herzig and Berant, 2018) (Andreas, 2020) (Campagna et al., 2020) (Zhong et al., 2020)

Data Augmentation Does Not Always Help

Models do **NOT** always benefit from data augmentation especially when the original data is relatively large and cross-domain.

Note: Direct training with the augmented data tends to overfit on the synthetic data.





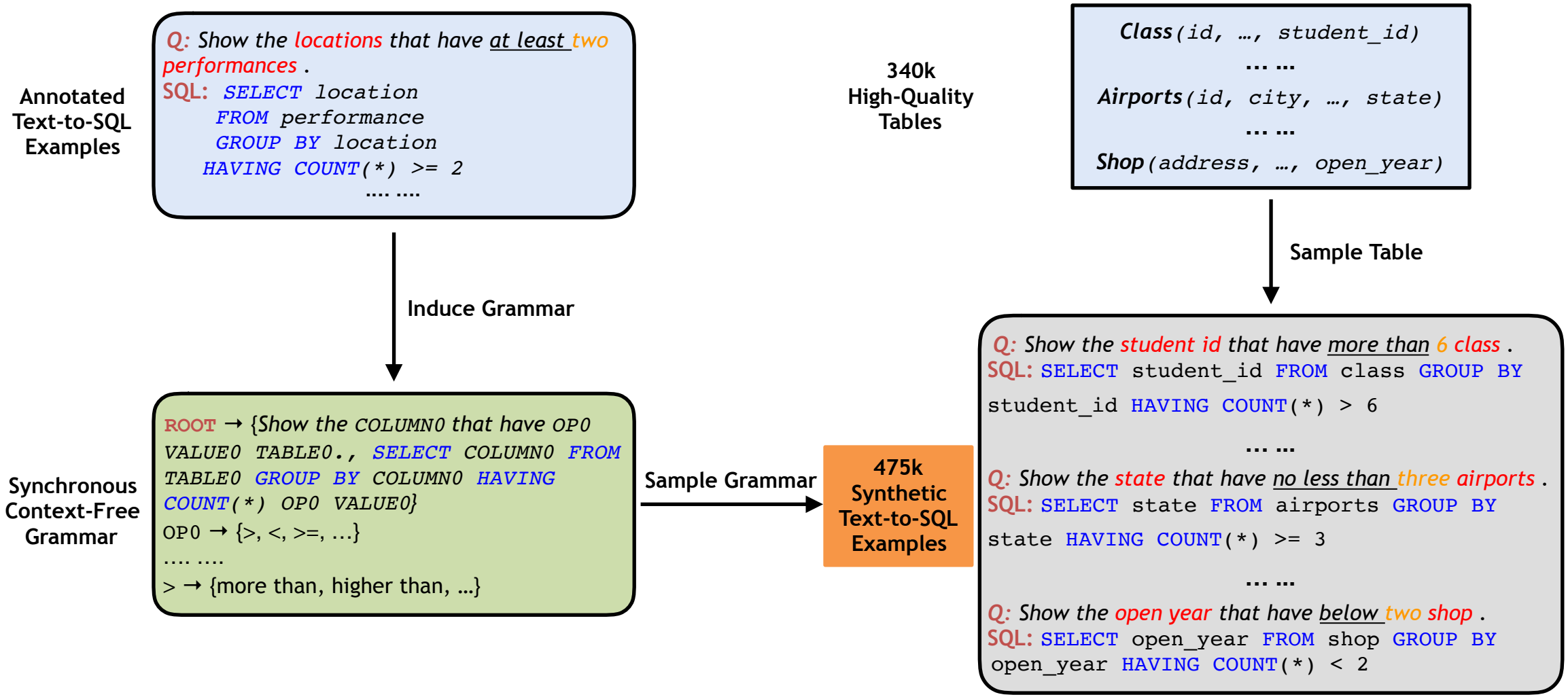
syntrain: training models on a combination of the augmented data and original training data

(Zhong et al., 2020)

(Campagna et al., 2020)

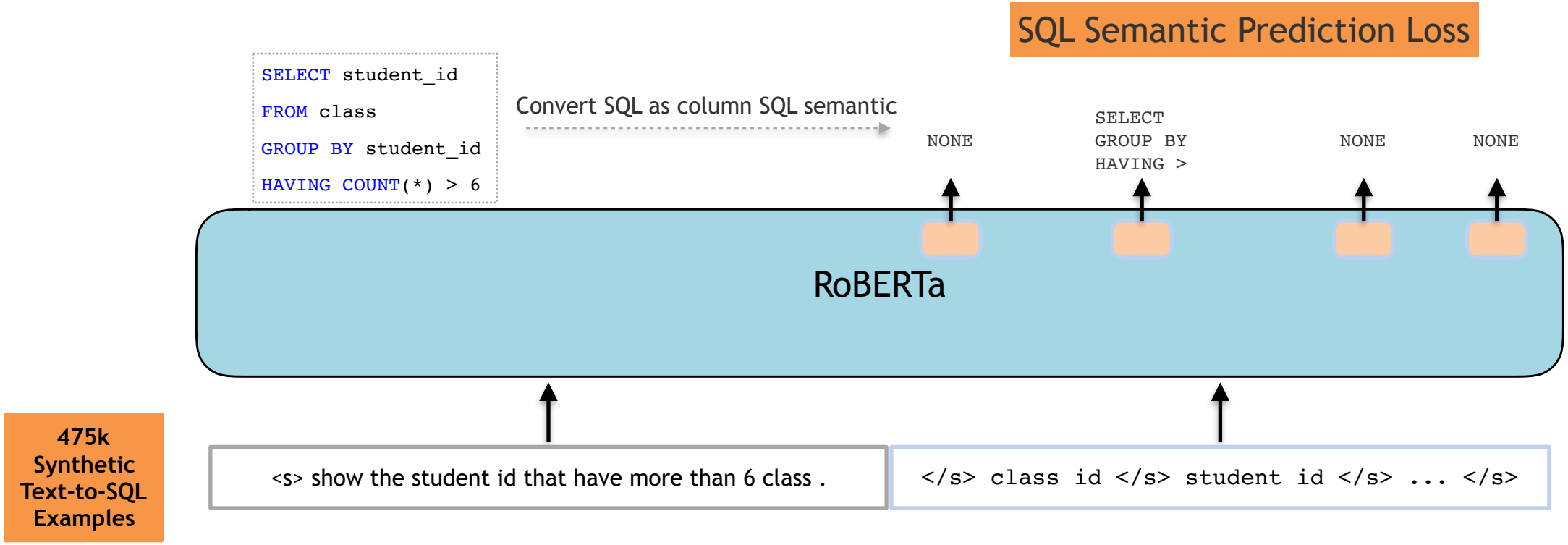
Data Augmentation for Pre-Training

Instead of directly training semantic parsers on the augmented data, we use it in pre-training to inject a compositional inductive bias in the joint representations of textual and tabular data to LMs.



Pre-Training GraPPa

Pre-train RoBERTa on the synthetic data using a novel SQL semantic prediction (SSP) objective that predicts the semantic role of a table column in the SQL for each question-SQL pair.



Pre-Training GraPPa

To avoid overfitting on the synthetic data, we also include masked language modeling (MLM) on several existing table-and-language datasets to regularize our pre-training process.

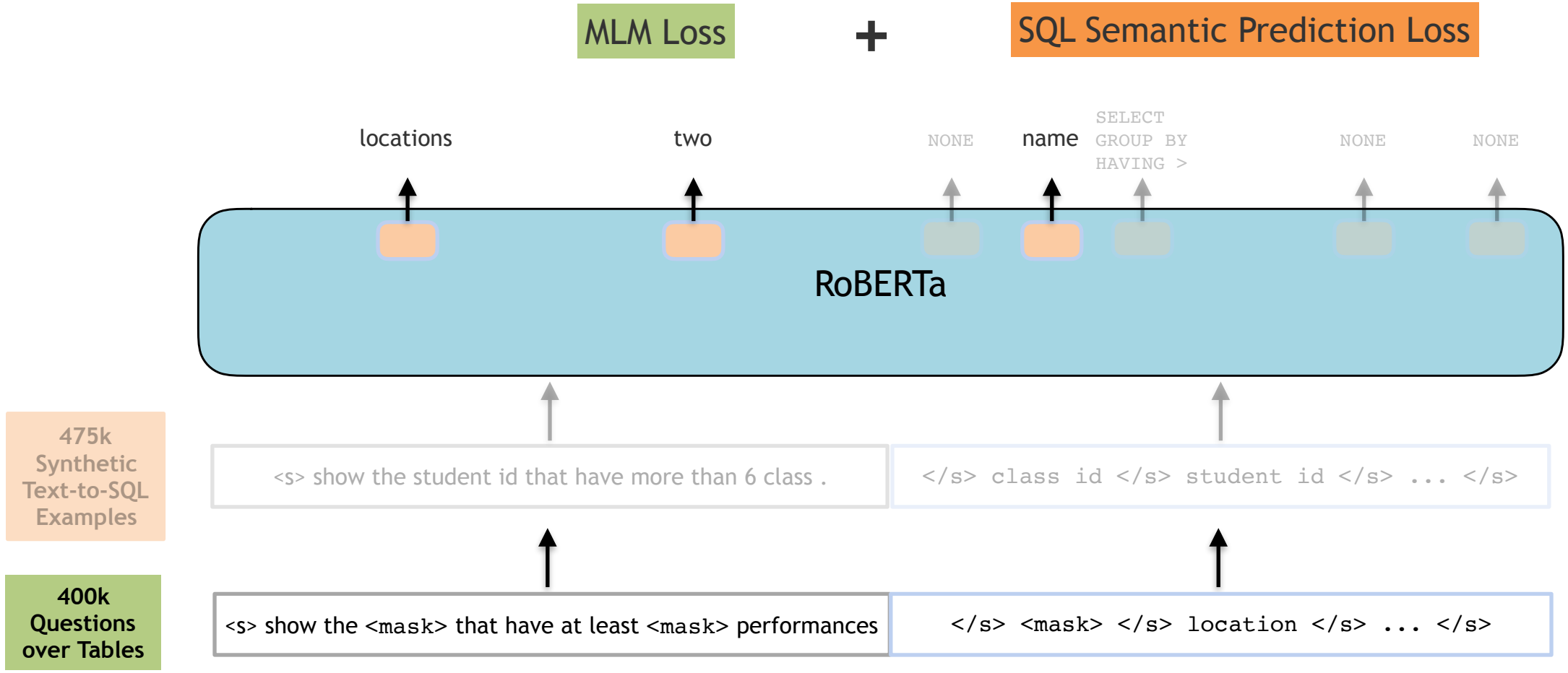


Table Semantic Parsing Benchmarks

When incorporated with strong base semantic parsers, GraPPa achieves new state-of-the-art results on all the four tasks.

Fully Supervised Semantic Parsing Tasks
Spider and **WikiSQL**

Find the first and last names of the students who are living in the dorms that have a TV Lounge as an amenity.

database with 5 tables

```
SELECT T1.FNAME, T1.LNAME
FROM STUDENT AS T1 JOIN LIVES_IN AS T2
     ON T1.STUID=T2.STUID
WHERE T2.DORMID IN
   ( SELECT T3.DORMID
     FROM HAS_AMENITY AS T3 JOIN DORM_AMENITY AS T4
          ON T3.AMENID=T4.AMENID
     WHERE T4.AMENITY_NAME= 'TV LOUNGE')
```

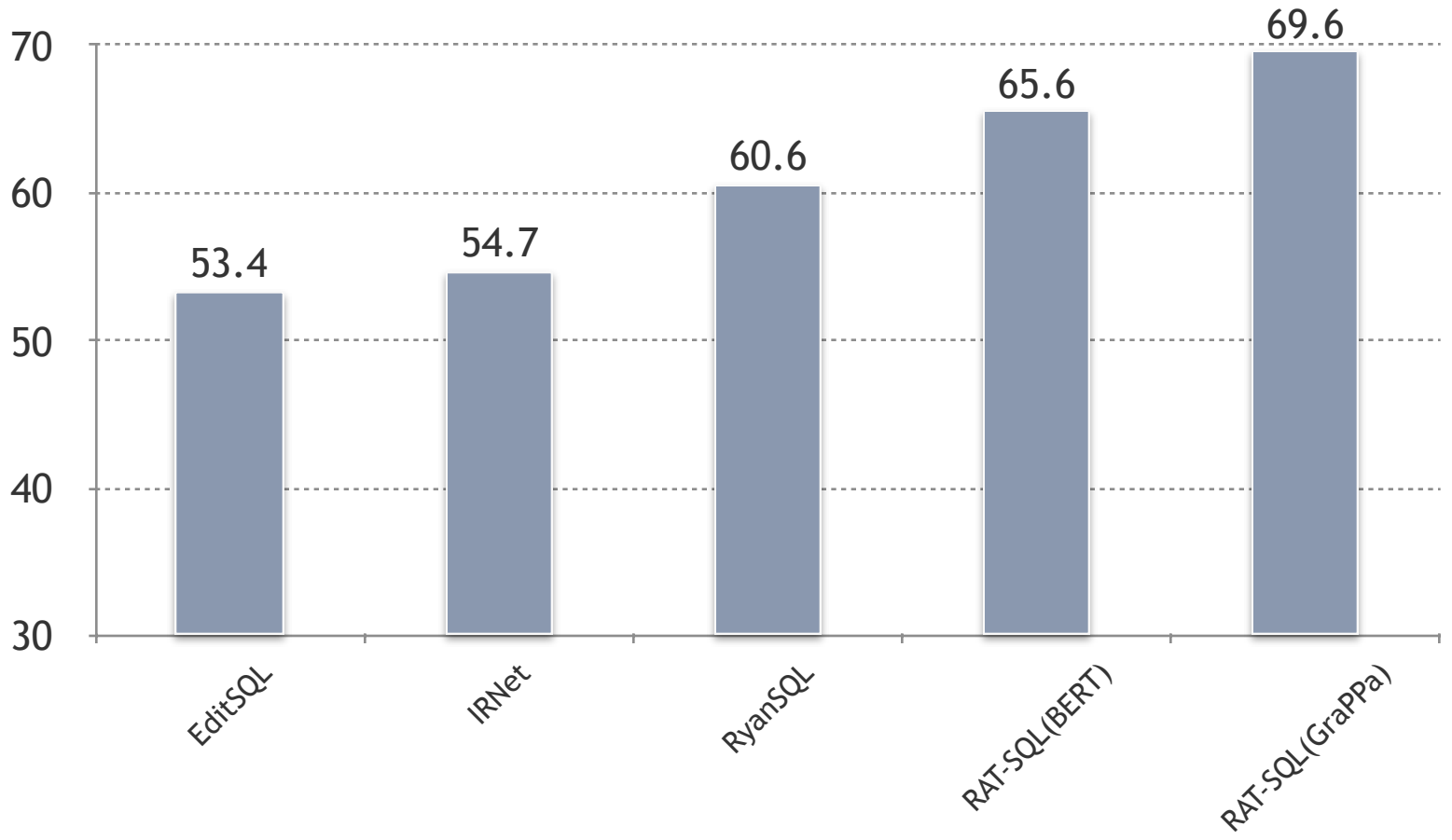
Weakly Supervised Semantic Parsing Tasks
WikiTQ and **WikiSQL**

In what city did Piotr's last 1st place finish occur?

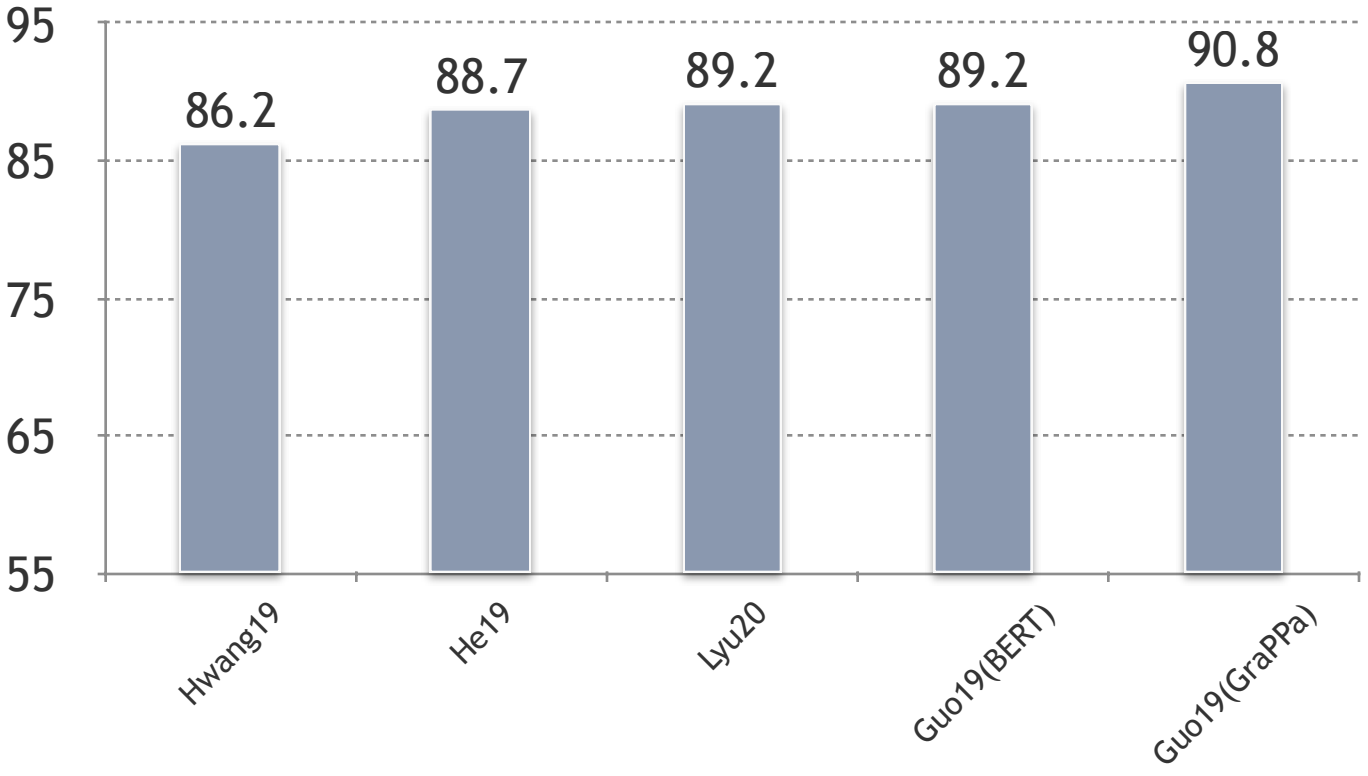
a table with 6 columns

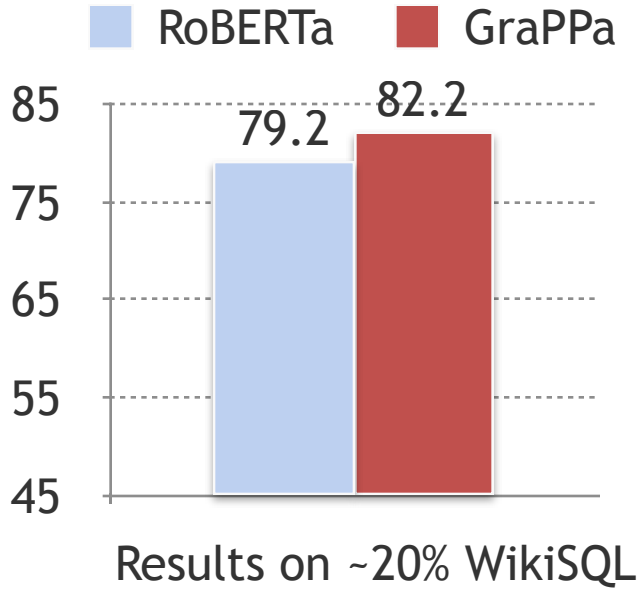
"Bangkok, Thailand"

Test Results on Spider

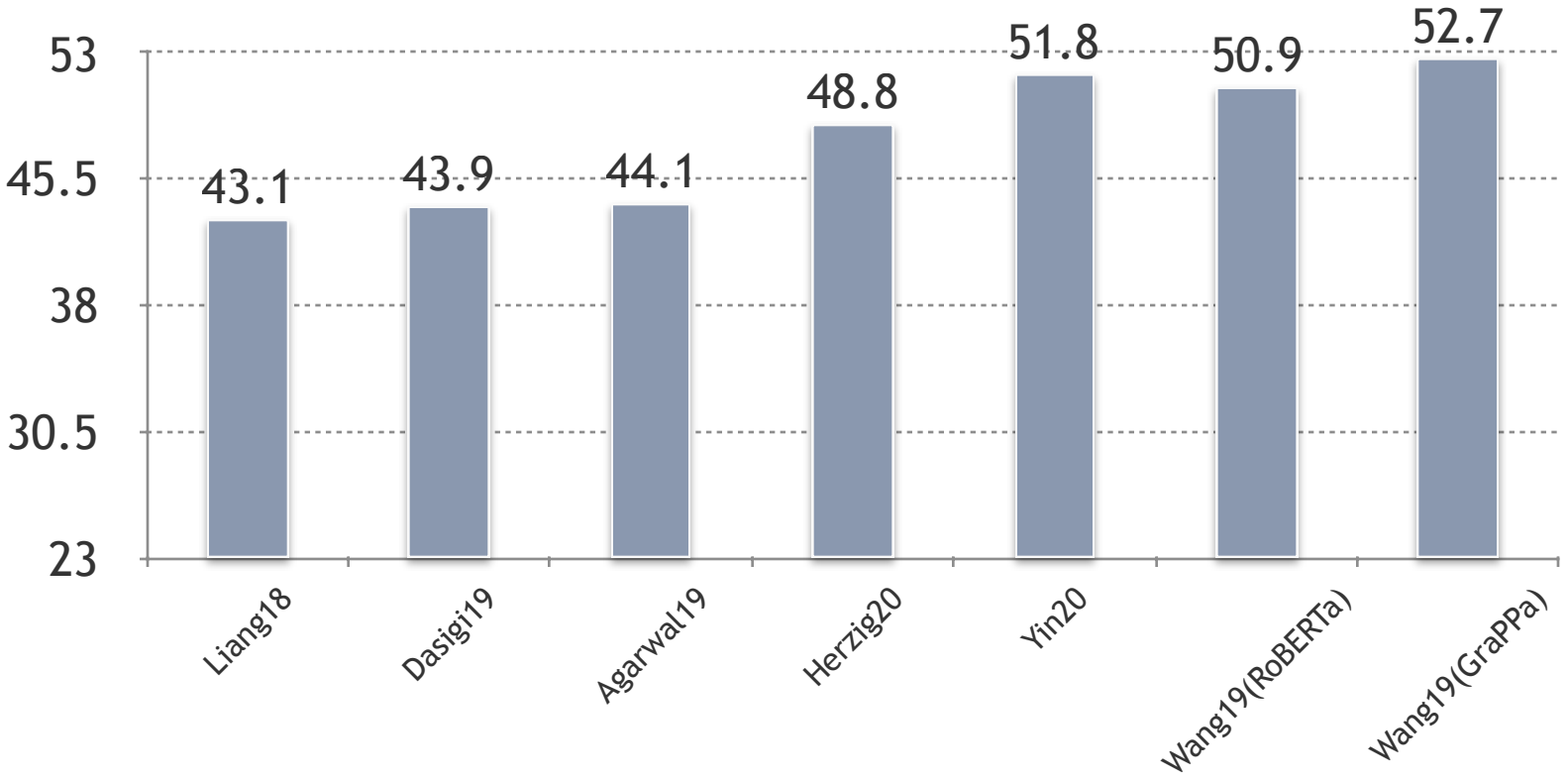


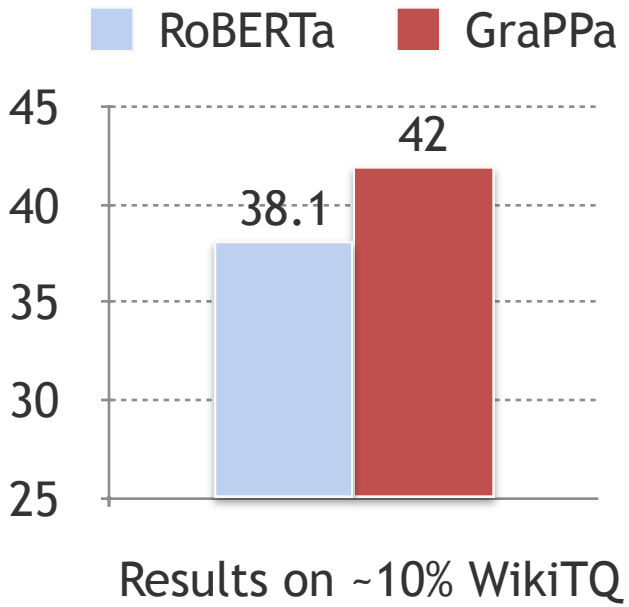
Test Results on Fully-Sup. WikiSQL





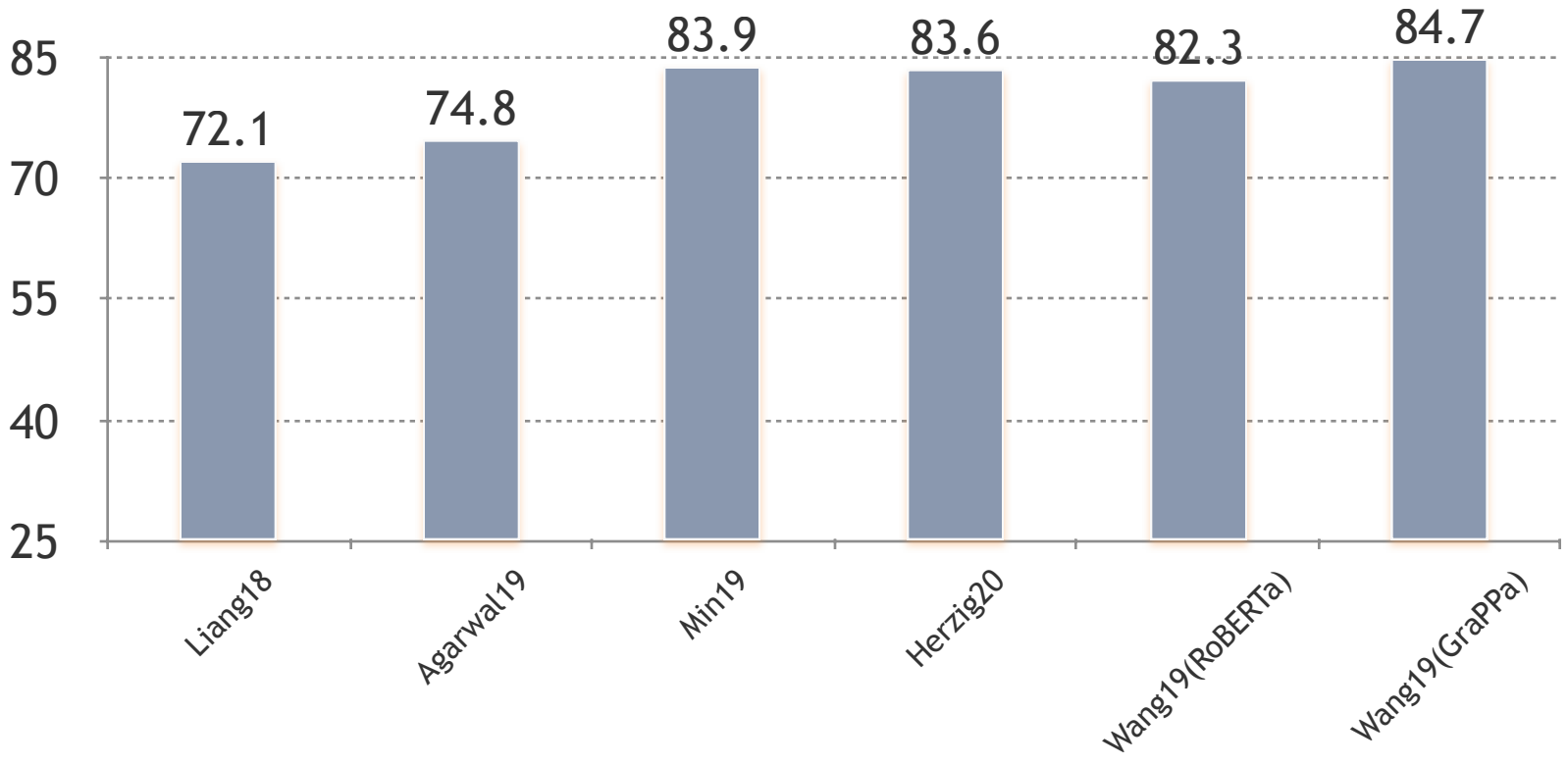
Test Results on WikiTQ





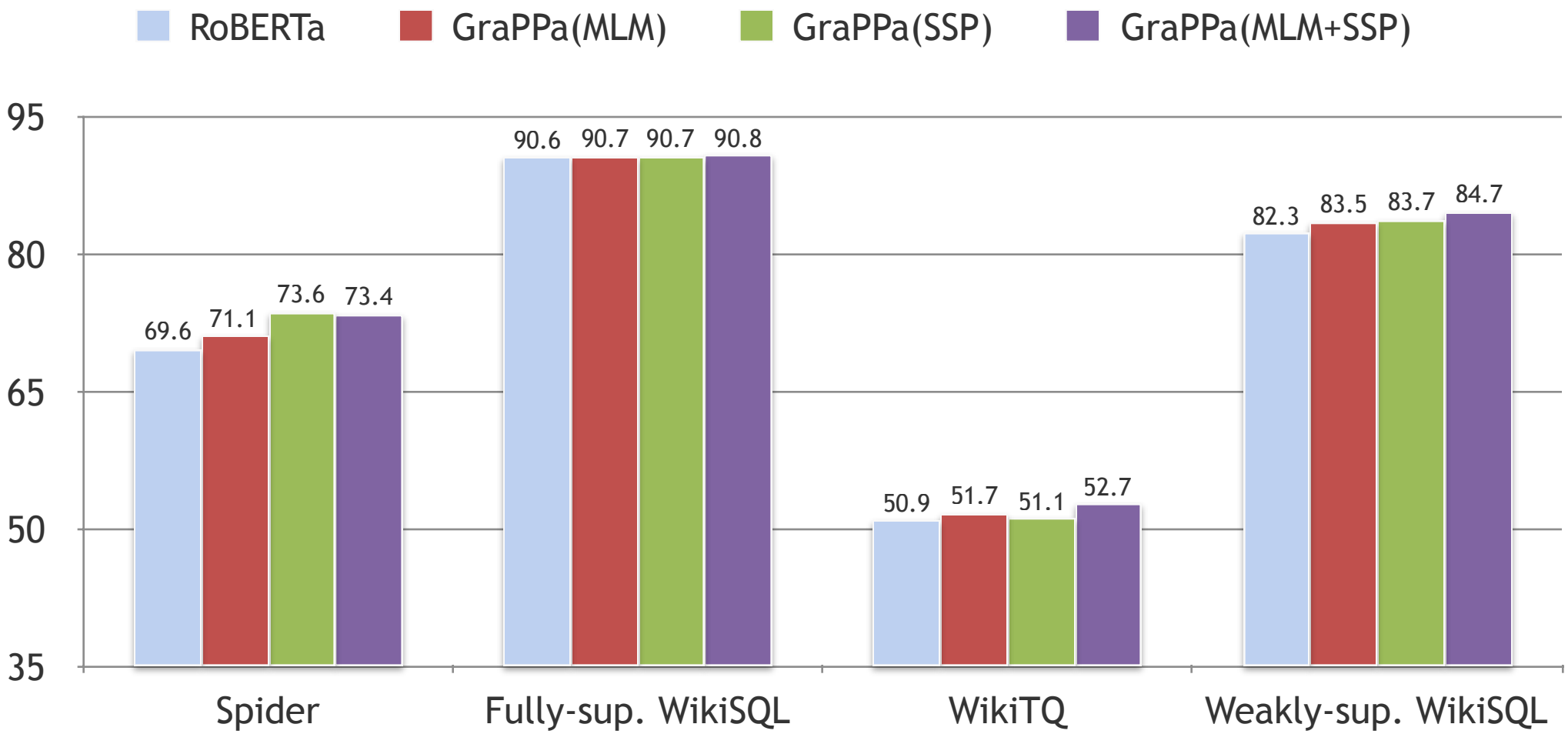
Results on ~10% WikiTQ

Test Results on Weakly-Sup. WikiSQL



Effect of Pre-Training Objectives

GraPPa trained with both MLM and SSP loss consistently outperforms the one trained with one of them.



Generalization

- We design our SCFG solely based on Spider, and then sample from it to generate synthetic examples. Despite the fact that GraPPa pre-trained on such corpus is optimized to the Spider data distribution, which is very different from WikiSQL and WikiTQ, GraPPa is still able to improve performance on the two datasets.

- In particular, for WikiTQ where the underlying distribution of programs (not necessarily in the form of SQL) are latent, GraPPa can still help a parser generalize better, indicating GraPPa can be beneficial for general table understanding even though it is pre-trained on SQL specific semantics.

- We believe that higher performance can be achieved if a SCFG is developed specifically for WikiSQL and WikiTQ.

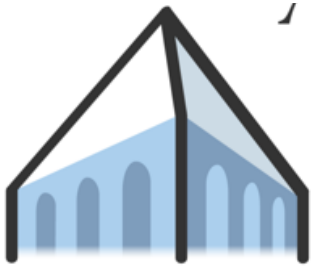
- While the pre-training method is surprisingly effective in its current form, this work relies on hand-crafted grammar which often generates unnatural questions; Further improvements are likely to be made by applying more sophisticated data augmentation techniques.

Pre-Training Time and Data

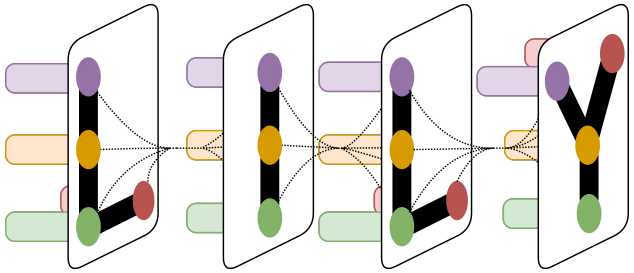
- We pre-train GraPPa for 300k steps on eight 16GB Nvidia V100 GPUs. The pre-training procedure can be done **in less than 10 hours** (Compared with 6 days on >100 V100 GPUs/3 days on 32 TPUs for TaBERT and TAPAS)!

- Our experiments on the Spider task show that longer pre-training doesn't improve and can even hurt the performance of the pre-trained model. The best result on Spider is achieved by using GraPPa pre-trained for only 5 epochs on our relatively small pre-training dataset.

Semantic Evaluation for Text-to-SQL with Distilled Test Suites



Berkeley NLP Group

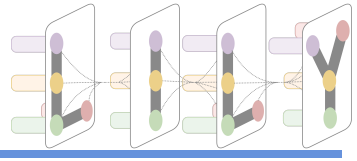


Yale NLP Group

Ruiqi Zhong, Tao Yu and Dan Klein

{ruiqi-zhong, klein}@berkeley.edu, tao.yu@yale.edu

Semantic Parsing Evaluation



Inputs

Natural Language

How old is the youngest person from department A?

"People" Database

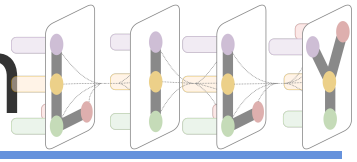
NAME	Age	Department
Alice	26	A
Bob	23	A

Output → Predicted Parse

```
SELECT MIN(Age) from People WHERE Department = 'A'
```

How do we evaluate the predicted parse?

Prior Metric 1: Exact String Match



"People" Database

NAME	Age	Department
Alice	26	A
Bob	23	A

Gold Parse

```
SELECT MIN(Age) from People WHERE Department = 'A'
```

Attempt 1: Exact String Match.
gold parse string == predicted parse string ?

Different but Equivalent

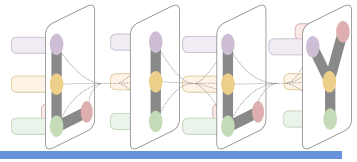
Predicted Parse

```
SELECT Age from People WHERE Department = 'A'
    ORDER BY Age ASC LIMIT 1
```

Totally reasonable prediction.
But gold parse != predicted parse

False Negative!

Prior Metric 2: Answer Match



"People" Database

NAME	Age	Department
Alice	26	A
Bob	23	A

Gold Parse

```
SELECT MIN(Age) from People WHERE Department = 'A'
```

Attempt 2: Answer Match.
gold answer == predicted answer ?

Predicted Parse

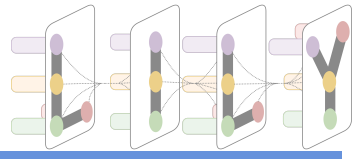
```
SELECT MIN(Age) from People
```

```
Answer: 23
```

Wrong prediction.
But gold answer == predicted answer

False Positive!

Semantic Correctness



"People" Database

NAME	Age	Department
Alice	26	A
Bob	23	A

Gold SQL

```
SELECT MIN(Age) from People WHERE Department = 'A'
```

Semantic Correctness:
gold answer == predicted answer ?
on all possible databases

Predicted SQL

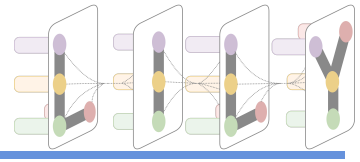
```
SELECT Age from People WHERE Department = 'A'
SELECT MIN(Age) from People
    ORDER BY Age ASC LIMIT 1
```

Wrong
Correct

NAME	Age	Department
Alice	26	A
Bob	23	B

UNDECIDABLE in general

An Online Judge Analog



Problem Description

Return the sum of all even numbers in the array

Your Solution

Predicted SQL

```
def even_sum(arr):
    return sum(a for a in arr if a % 2 == 0)
```

Test Suite

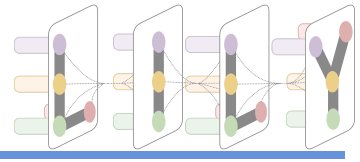
Database

```
Input: [1,2,3]
Expected Output: 2

Input: [3, 5, 7]
Expected Output: 0

Input: [2,7,8,2,1]
Expected Output: 12

[More test cases omitted]
```

Answer

Solution is considered correct if it passes the entire test suite

Our Metric: Test Suite Accuracy



Natural Language

How old is the youngest person from department A?

Predicted Parse

```
SELECT MIN(Age) from People
    WHERE Department = 'A'
```

Execute on

Test Case 1

Name	Age	Department
Alice	26	A
Bob	23	A

Predicted Answer: 23 ✓

Test Case 2

Name	Age	Department
Alice	26	A
Bob	23	B

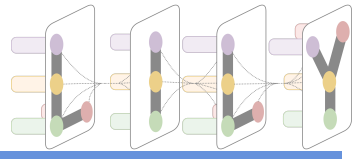
Predicted Answer: 26 ✓

→ Correct if the predicted parse passes all the test cases (test suite)

More Cases

[More Database-Answers Omitted]

Criteria for Test Suites



Criteria 1:
Fast to run

Criteria 2:
Code Coverage

↓

Search Objectives

Cannot enumerate all possible databases (smaller than a certain length) …

Effectively tests the use of every clause and constants

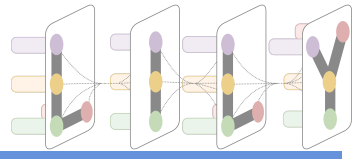
Gold `SELECT MIN(Age) from People WHERE Department = 'A'`

not covered not covered

Database

NAME	Age	Department
Alice	26	A
Bob	23	B

Define Code Coverage



Gold

```
SELECT MIN(Age) from People WHERE Department = 'A'
```

Test Suite

DB 1

NAME	Age	Department
Alice	26	A
Bob	23	A

Correct Answer 23 == Neighbor 1 Answer 23

DB 2

NAME	Age	Department
Alice	26	A
Bob	23	B

Correct Answer 26 != Neighbor 1 Answer 23

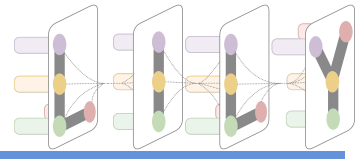
modify

Neighbor 1

```
SELECT MIN(Age) from People ~~WHERE Department = 'A'~~
```

discriminated by DB 2

Define Code Coverage



Gold

```
SELECT MIN(Age) from People WHERE Department = 'A'
```

Test Suite

DB 1

NAME	Age	Department
Alice	26	A
Bob	23	A

Correct Answer
23

DB 2

NAME	Age	Department
Alice	26	A
Bob	23	B

Correct Answer
26

modify

For each neighbor, at least one database in the test suite discriminates the neighbor.

Neighbor 1

```
SELECT MIN(Age) from People ~~WHERE Department = 'A'~~
```

DB 2

Neighbor 2

```
SELECT MIN(Age) from People WHERE Department = 'B'
```

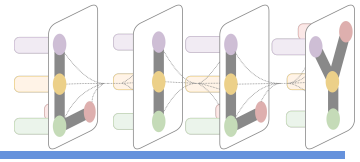
discriminated by DB 1 and DB 2

Neighbor 3

```
SELECT MAX(Age) from People WHERE Department = 'A'
```

DB 1

Define Code Coverage



Gold `SELECT MIN(Age) from People WHERE Department = 'A'`

DB 1

NAME	Age	Department
Alice	26	A
Bob	23	A

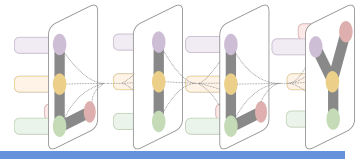
Correct Answer
23

Test Su

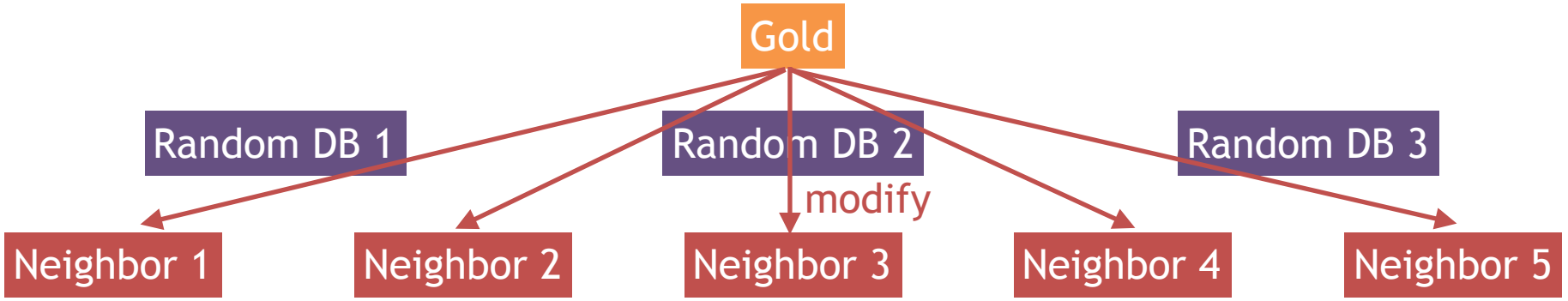
Neighbor 1 cannot be distinguished by any database.

Neighbor 1	`SELECT MIN(Age) from People ~~WHERE Department = 'A'~~`	None
Neighbor 2	`SELECT MIN(Age) from People WHERE Department = 'B'` discriminated by	DB 1
Neighbor 3	`SELECT MAX(Age) from People WHERE Department = 'A'`	DB 1

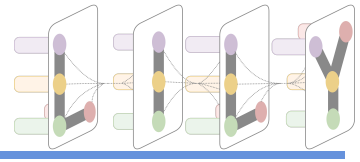
Constructing a Test Suite



Optimization Objective Find a small set of databases that can discriminate all neighbors.

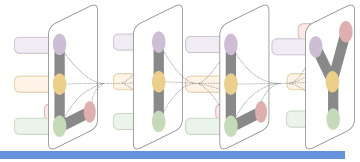


Constructing a Test Suite

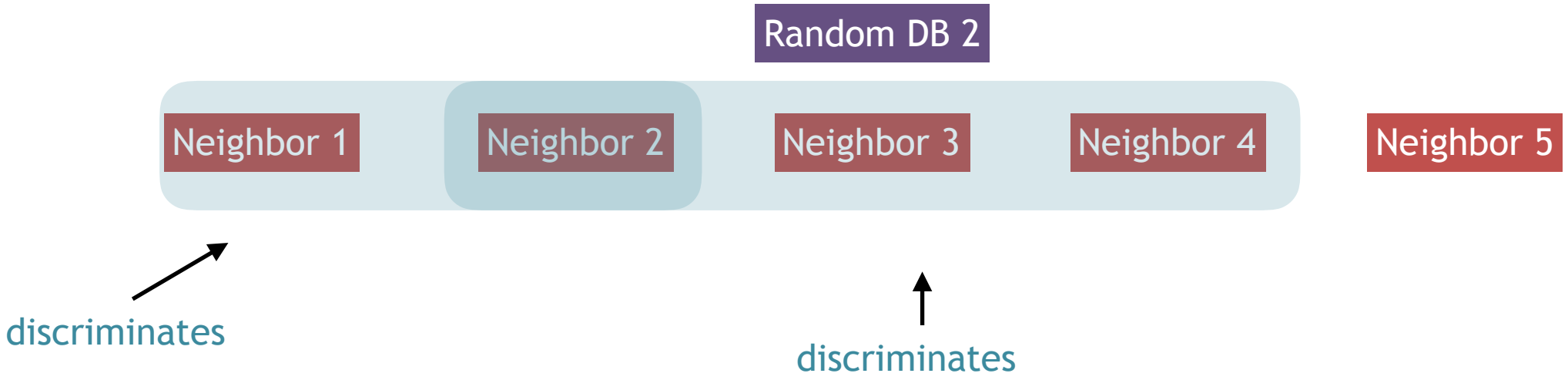


Optimization Objective Find a small set of databases that can discriminate all neighbors.

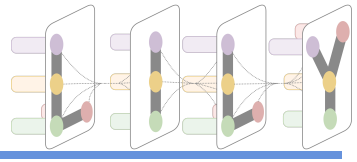
Constructing a Test Suite



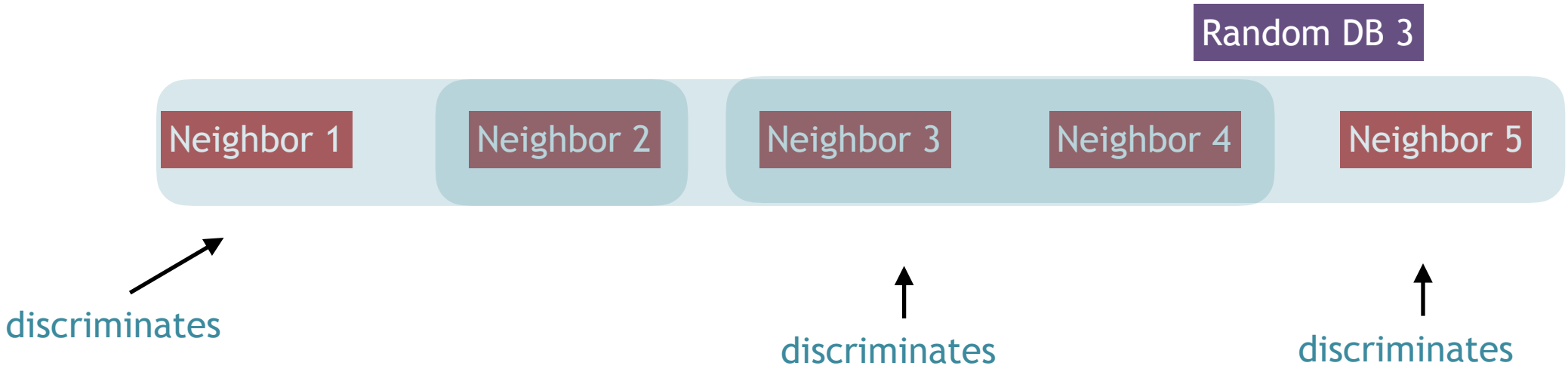
Optimization Objective Find a small set of databases that can discriminate all neighbors.



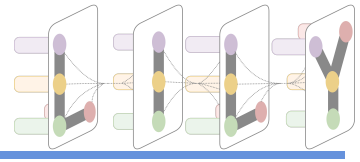
Constructing a Test Suite



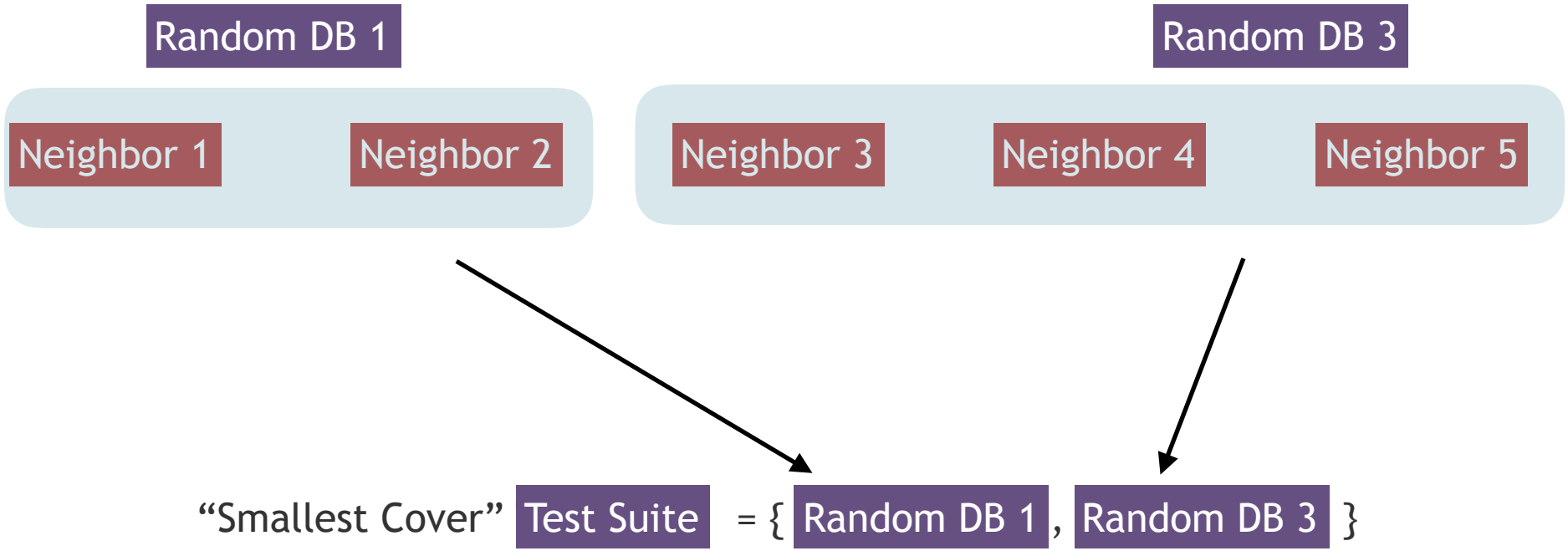
Optimization Objective Find a small set of databases that can discriminate all neighbors.



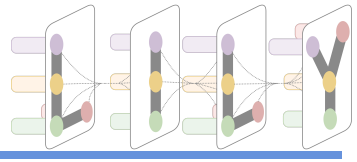
Constructing a Test Suite



Optimization Objective Find a small set of databases that can discriminate all neighbors.



Data



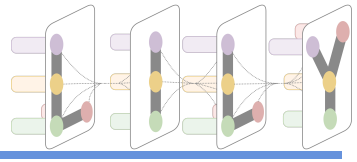
SPIDER development set: 1034 pairs of English-SQL parallel data.

Official metric “Exact Set Match”: whether the predicted SQL contains the same set of clauses as the gold.

21 Leaderboard submissions with Exact Set Match ranging from 40% to 65%.

Rank	Model	Dev	Test
1 May 02, 2020	RATSQL v3 + BERT (DB content used) *Microsoft Research* (Wang and Shin et al., ACL '20) code	69.7	**65.6**
2 Sep. 8, 2020	YCSQL + BERT (DB content used) *Anonymous*	-	65.3
3 Sep. 8, 2020	ShadowGNN (DB content used) *Anonymous*	-	64.8
4 May 31, 2020	AuxNet + BART (DB content used) *Anonymous*	70.0	61.9
4 Dec 13, 2019	RATSQL v2 + BERT (DB content used) *Microsoft Research* (Wang and Shin et al., ACL '20) code	65.8	61.9

No Errors Made



False Negative — The predicted query is in fact semantically correct, but considered wrong.

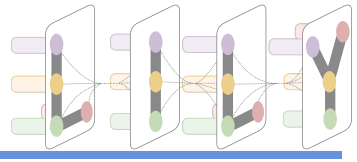
This provably never happens.

False Positive — The predicted query is in fact semantically wrong, but considered correct.

Manually examined 100 random examples where
our metric differs from exact set match
(i.e. the official metric).

Our metric judges correctly all the time.

Official Metric in Hindsight

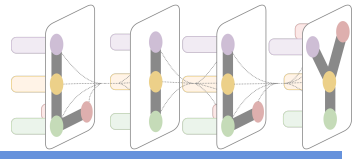


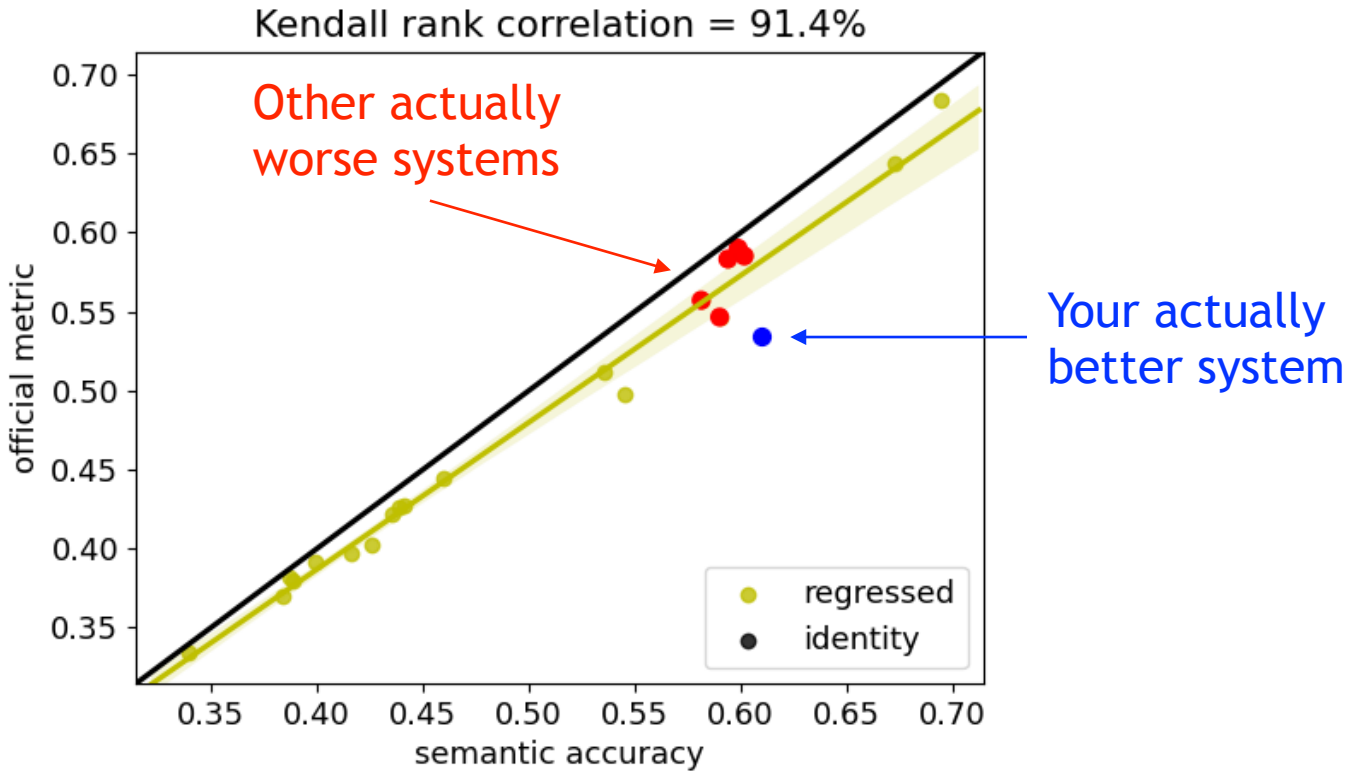
Statistics of the “false negative” fraction across 21 submissions.

Mean	Standard Deviation	Max
2.6%	1.7%	8.1%

Makes non-negligible fraction of errors.

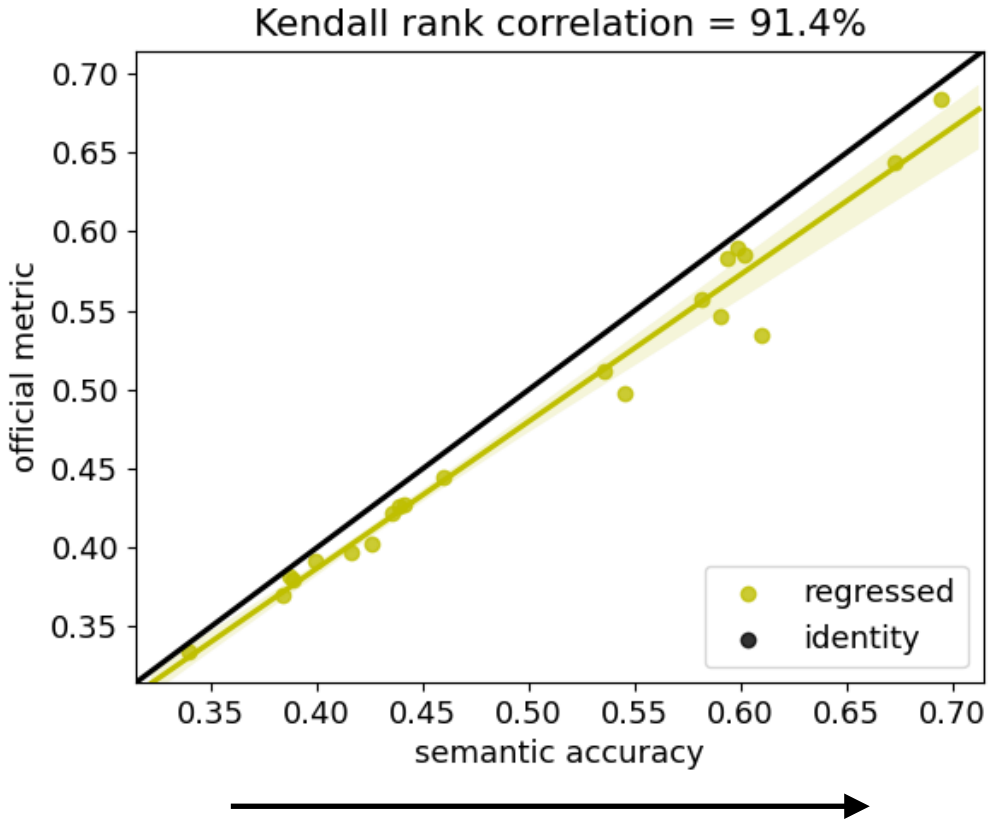
Official Metric in Hindsight





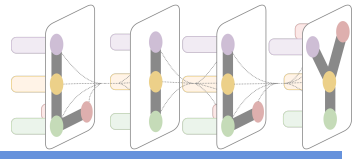
Does not reflect all semantic improvements.

Official Metric in Hindsight



Correlates less as systems become better.

New Metric Release



Test suite and our new metric implementation is now publicly available
for eleven Text-to-SQL datasets:
SPIDER, SParC, CoSQL, Academic, Advising, ATIS, GeoQuery, IMDB, Restaurants, Scholar and Yelp

Test suite evaluation is now the official metric of
SPIDER, SParC and CoSQL leaderboards.

Takeaways

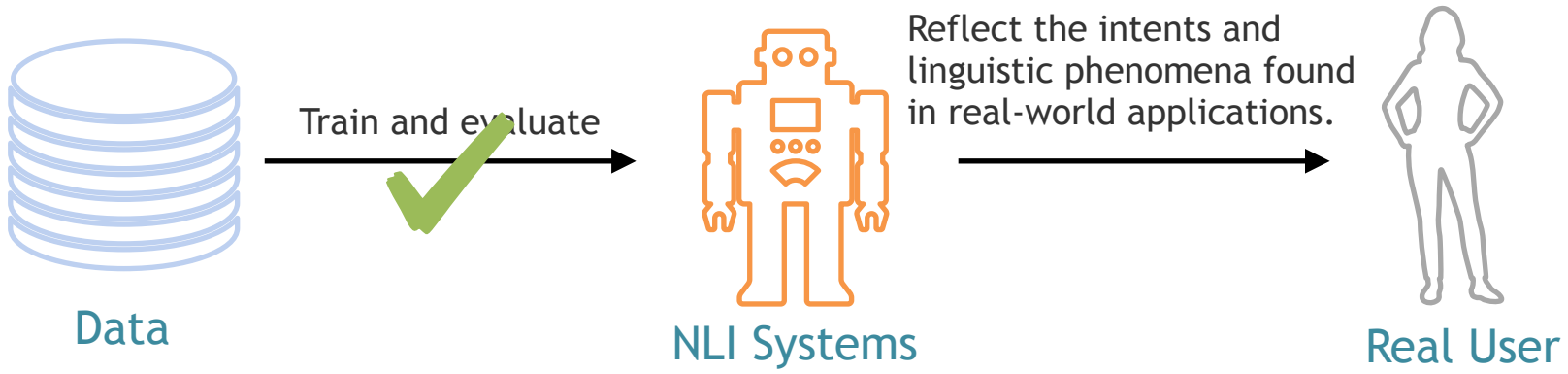
- Surface logical form matching poorly correlates with semantic accuracy.
- Evaluating on multiple inputs (test suite) can approximation semantic accuracy better.
- We propose a method to construct a test suite and evaluate its quality through neighbor queries.
- Looking forward to semantic evaluation with test suite in other domains!

Towards Ecologically Valid Research on Language User Interfaces

Harm de Vries, Dzmitry Bahdanau, Christopher Manning

Ecological Validity

Ecological validity is a special case of external validity, specifying the degree to which findings generalize to naturally occurring scenarios.



Google Assistant

Ecological Validity is Hard

Even though some existing interfaces could access real user data, but

- Data privacy needs
- The “real” data is still system dependent.

NLI Data Collection

- Earlier researchers collect small size data and attempt to closely simulate the NLI's anticipated use-case
- Many recent benchmarks opted for cheaper and more scalable method such as crowdsourcing.

Crowdsourcing

Crowd workers are not representative of the target user population.

Common Issues of Popular Benchmarks

- Synthetic language
- Artificial tasks
- Not working with prospective users
- The use of scripts and/or priming
- Single-turn interfaces

Deviation	Project
Synthetic language	BabyAI (Chevalier-Boisvert et al., 2019) CLEVR (Johnson et al., 2017) CFQ (Keysers et al., 2019) GQA (Hudson and Manning, 2019)
Artificial task	GuessWhat (De Vries et al., 2017) CerealBar (Suhr et al., 2019) CoDraw (Kim et al., 2019) VisionAndLanguage (Anderson et al., 2018)
Not working with prospective users	Visual Question Answering (Antol et al., 2015) Visual Dialog (Das et al., 2017) Spider (Yu et al., 2018) SQuAD (Rajpurkar et al., 2016)
Scripts and priming	MultiWOZ (Budzianowski et al., 2018) ALFRED (Shridhar et al., 2020) CoSQL (Yu et al., 2019a) Sparc (Yu et al., 2019b) AirDialogue (Wei et al., 2018) Overnight (Wang et al., 2015)
Single-turn interfaces	Advising (Finegan-Dollak et al., 2018) MS Marco (Bajaj et al., 2016) Natural Questions (Kwiatkowski et al., 2019) DuReader (He et al., 2018)

Synthetic Language

The key difficulty in designing a synthetic language is to obtain broad linguistic coverage while maintaining the natural aspect of language.

- It defines a context-free grammar to generate simple instructions (Vocab is small and features only a few dozen words.) such as BabyAI:

> *open the yellow door, then go to the key behind you.*

- In general, it is important for grammar-based approaches to carefully limit the operators that can lead to combinatorial explosion, as these are often the source of unnatural utterances. CFQ:

> *Did Patrick Scully's sibling marry Carolyn Zeifman, influence Tetsuo II: Body Hammer's art director, director, and executive producer, and influence Christophe Gans?*

Artificial Tasks

Artificial tasks do not correspond to or even resemble a practically relevant NLI setting.

- People would naturally use these LUIs such as GuessWhat

Not Working with Prospective Users

The population that would actually benefit from the language user interface rarely participates in the data collection effort.

- Spider: annotated by SQL experts. Some questions are quite literal translations of the SQL query.

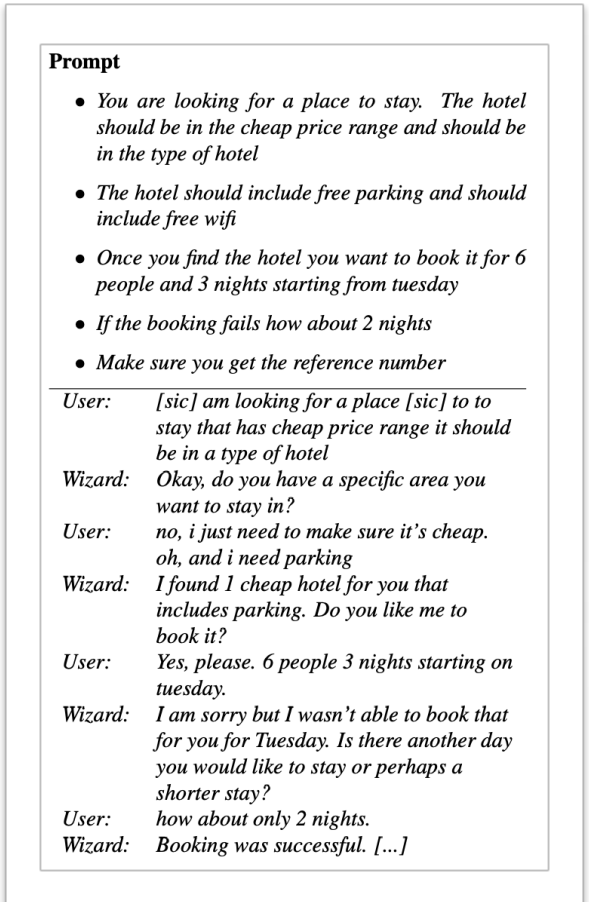
> *What are the names of the customers who bought product "food" at least once?*

- SQuAD: crowdworkers had no information need, which makes it unclear if the resulting questions match the ones from users looking for this information.

Scripts and Priming

To compensate for the lack of access to potential users and/or to capable wizards, many recent data collection efforts relied on scripts that constrained the flow of human-computer interaction

- MultiWoZ/CoSQL: Scripts limits the search scenarios, or questions often stay close to (parts of) the prompt.



Prompt

- *You are looking for a place to stay. The hotel should be in the cheap price range and should be in the type of hotel*
- *The hotel should include free parking and should include free wifi*
- *Once you find the hotel you want to book it for 6 people and 3 nights starting from tuesday*
- *If the booking fails how about 2 nights*
- *Make sure you get the reference number*

User:	*[sic] am looking for a place [sic] to to stay that has cheap price range it should be in a type of hotel*
Wizard:	*Okay, do you have a specific area you want to stay in?*
User:	*no, i just need to make sure it's cheap. oh, and i need parking*
Wizard:	*I found 1 cheap hotel for you that includes parking. Do you like me to book it?*
User:	*Yes, please. 6 people 3 nights starting on tuesday.*
Wizard:	*I am sorry but I wasn't able to book that for you for Tuesday. Is there another day you would like to stay or perhaps a shorter stay?*
User:	*how about only 2 nights.*
Wizard:	*Booking was successful. [...]*

Single-Turn Interfaces

Some datasets consider real and useful tasks and involve target users in the data collection effort

- Advising/MS MARCO/Natural Questions: These benchmarks much more ecologically valid than the ones we discussed earlier, yet the user is only allowed to ask a single question in these tasks.

Evaluation

- The key issue with the current evaluation procedure is that it does not account for errors that the system makes along the conversation.
- Evaluating under the assumption of ground-truth inputs does not measure how well the system is able to recover from its own mistakes.
- The only way to measure that is through a human-in-the-loop evaluation that assesses whether the interaction as a whole was successful.

Finding the key problems/bottlenecks is not that easy!

More discussion on Ecologically Valid Research on NLI (Tao Yu)

Key Issues

- **Data**: affects everything we are going to do next
- **System**: reliable, explainable, and generalized
- **Evaluation**: guides system development

NLI Data - Text-to-SQL

Datasets reflect different database usage scenarios

- Database access
- User backgrounds
- Query goal
  - Data analytics
  - Data retrieval
  - Data visualization
  - Task completion such as booking hotel
- Domain

NLI Data - Spider

Clear but complex questions, excludes domain knowledge

- Database access: users can access DB
- User backgrounds: users know SQL
- Query goal
  - Data analytics
  - Data retrieval
  - Data visualization
  - Task completion such as booking hotel
- Domain: general purpose cross domain

NLI Data - ATIS, IMDB, Yelp, Advising etc.

Requires domain knowledge reasoning, more diverse and ambiguous

- Database access: users cannot access DB
- User backgrounds: users don't know SQL
- Query goal
  - Data analytics
  - Data retrieval
  - Data visualization
  - Task completion such as booking hotel
- Domain: only work for a single domain

NLI Data - MultiWoZ

More conversational, simple semantic meaning, predefined task goal

- Database access: users cannot access DB
- User backgrounds: users don't know SQL
- Query goal
  - Data analytics
  - Data retrieval
  - Data visualization
  - Task completion such as booking hotel
- Domain: only work for multiple in domains

NLI System - Text-to-SQL

To build reliable, explainable, and generalized systems

- User intent classification
- Query suggestion based on partial user inputs
- Domain adaptation
- Result explanation for user verification
- Multimodal interactive learning
- Multi-turn query understanding

Domain Adaptation

- **Spider train vs. test**: question styles are similar, but on different databases
- **Spider train vs. ATIS/Yelp/IMDB etc.**: question styles are different (domain specific jargon, ambiguous, data conventions), and on different databases

Domain Adaptation

SQL experts (Spider model) + **Domain experts** (domain knowledge, data conventions)

Advising dataset:

Easiest course?

```
Workload = select min(wordload)
from course
```

Can I take CS50 as I need to leave for work at 5:00 everyday ?

```
Course_offering.end_time < 17:00
```

ATIS dataset:

Later afternoon?

```
Time between 16:00 and 18:00
```

Yelp dataset

Good restaurants?

```
Rating > 4.0
```

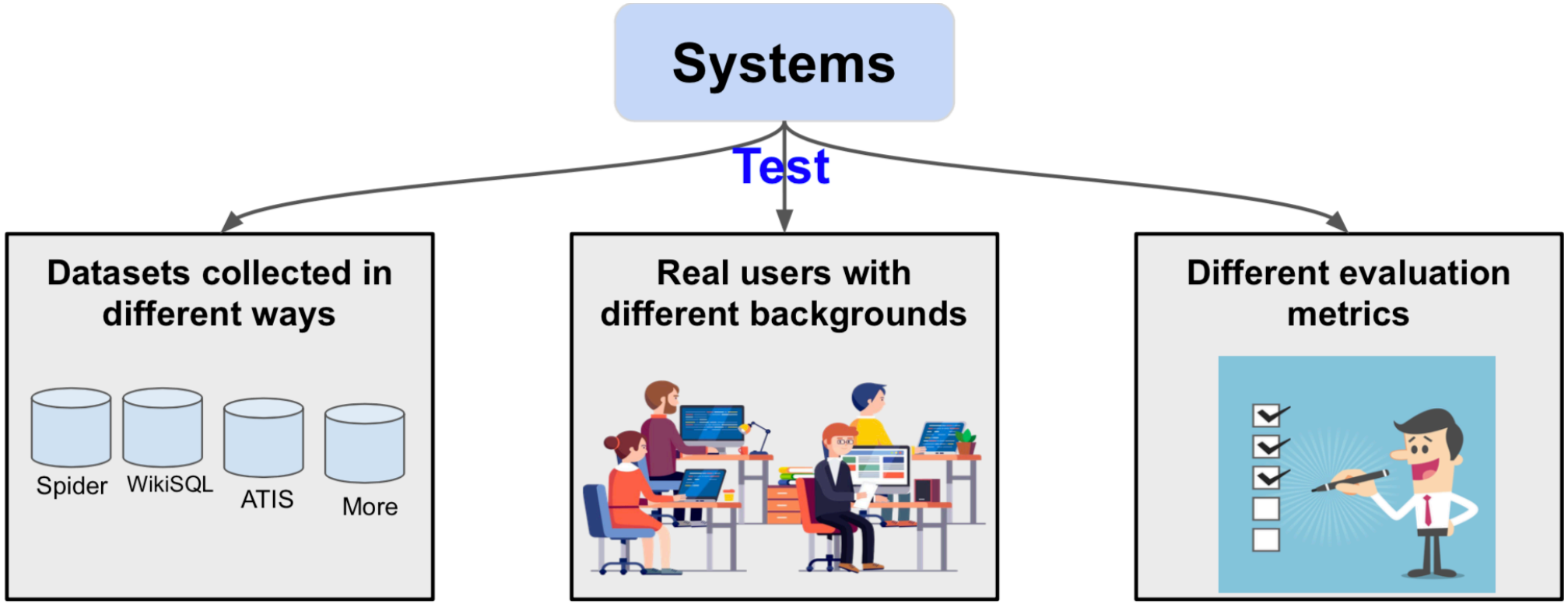
Geo dataset

Major cities?

```
Area > 750 or population > 150000
```

Evaluation

To build a unified evaluation platform for interactive NLIs



Discussion?

- The key component of all NLIs (including vision navigation, robots, databases, webs) is language grounding, could we unify their formal representation and study them together?
- What else do we have to focus on for developing the next generation NLI systems?

Thank you!