Stanford CS224v Course

Conversational Virtual Assistants with Deep Learning

Lecture 4

Knowledge Assistants Grounded on Structured & Free-text Data

Monica Lam, Shicheng Liu, Sina Semnani

Yelp: DB + Free text

Name TEXT	**Cuisine** TEXT[]	**Rating** NUM(2,1)	...	**Popular_dishes** FREE TEXT	**Reviews** FREE TEXT
Hummus Mediterranean Kitchen	mediterranean,halal, salad	4.0	...	Chicken Kebab Plate, Lamb Beef Gyro, Marinated Chicken Gyros, ...	| t l ; d r | This is your place if you're looking for a healthy and filling meal whether it's a quick pick-me-up or casual dining, ...
Penny Roma	italian, venues & event spaces	4.0	...	Cacio E Pepe, Agnolotti Dal Plin, Albacore Tartare, ...	My girlfriend was craving pasta on a Monday night. ... We were not expecting such an intimate and romantic dining experience. The restaurant was candle lit, modern, and perfect for a date night. ...

- Blue columns contain structured data
- Yellow columns are free text
- A lot of rows (thousands)

DB + Free-Text Combo Example

I need a family-friendly restaurant

→ "family-friendly" in reviews

I need a family-friendly restaurant in Palo Alto

→ "family-friendly" in reviews; Palo Alto in DB

Hybrid Data Sources

Domain Corpus

Schema
Free-Text → **Framework** → **Knowledge Chatbot**

	Structured	Free Text
Yelp	Cuisines, opening hours, address, rating	Reviews, popular dishes
Products	Product ID, cost, ratings, sizes, physical dimensions, color	Descriptions, reviews
Electronic Health Records	Patient demographics Prescription	Diagnoses
Wiki	Wikidata, Wikipedia tables	Wikipedia

Lecture Goal

Domain Corpus

Knowledge Bases
Free-Text → **Framework** → **Knowledge Chatbot**

- (Non-hybrid) multi-hop retrieval
- Prior work on hybrid data sources
- SUQL: a proposal

Text-Based QA Research Topics

1. Frequently asked questions (FAQ)
2. Reading comprehension (RC): answer giving a paragraph
   - Dataset: SQuAD (100s of papers)
3. Open domain QA (e.g. Wikipedia)
   - Create index with DR (dense retrieval): Colbert, Coco-DR
4. Closed book question answering
   - Answering questions after reading the internet/Wikipedia

Quiz: What is the status today on these topics?

5. Multi-Hop Question Answering

- So far, answers can be retrieved from one paragraph
- Many questions need "multi-hops":

Which university has the bigger campus,
Stanford or UC Berkeley?

When was the flag bearer of Rio Olympic born?

- What we need to do:
  - Multiple (cascading) retrievals; formulate/compute answer

Quiz

FOR STRUCTURED DATA,

DO WE NEED MULTI-HOP RETRIEVAL?

If Info is in Databases

Which university has the bigger campus, Stanford or UC Berkeley?

SELECT university FROM Campus-Table
where university='Stanford' or university='Berkeley'
ORDER BY size DESC

When was the flag bearer of Rio Olympic born?

SELECT year FROM Birthdates-Table
where Name in
(SELECT bearer-name FROM Olympic-Table WHERE location == 'Rio')

Multi-hop: composition of query

Paper on Multi-Hop QA for Free-Text

Inspired by semantic parsing for databases

BREAK It Down: A Question Understanding Benchmark

Tomer Wolfson[1,3], Mor Geva[1,3], Ankit Gupta[1], Matt Gardner[3], Yoav Goldberg[2,3], Daniel Deutch[1], Jonathan Berant[1,3]

[1]Tel Aviv University [2]Bar-Ilan University [3]Allen Institute for AI

{tomerwol,morgeva}@mail.tau.ac.il, ankitgupta.iitkanpur@gmail.com, mattg@allenai.org, yoav.goldberg@gmail.com, danielde@post.tau.ac.il, joberant@cs.tau.ac.il

Break-it-Down!

Question
Which university has the bigger campus, Stanford or UC Berkeley?

QDMR Decomposition
1. Return the campus size of UC Berkeley.
2. Return the campus size of Stanford.
3. Return the greater of #1, #2

- Question Decomposition Meaning Representation (QDMR)
  - Neural: Complex questions as a series of atomic English questions
  - Deterministic: convert the series to pseudo-SQL
  - Collected with crowdsourcing using a special UI
- Dataset: Collected "breakdown" of 84K questions

Wolfson et al. "Break It Down: A Question Understanding Benchmark." TACL (2020).

QDMR

SQL primitives in English

Add high-level combos:
- select+project
- select+filter
- filter+group+comparative

Composition with #s

Operator	Template / Signature	Question	Decomposition
Select	Return [entities] $w \rightarrow S_e$	How many touchdowns were scored overall?	1. **Return touchdowns** 2. Return the number of #1
Filter	Return [ref] [condition] $S_o, w \rightarrow S_o$	I would like a flight from Toronto to San Diego please.	1. Return flights 2. **Return #1 from Toronto** 3. **Return #2 to San Diego**
Project	Return [relation] of [ref] $w, S_e \rightarrow S_o$	Who is the head coach of the Los Angeles Lakers?	1. Return the Los Angeles Lakers 2. **Return the head coach of #1**
Aggregate	Return [aggregate] of [ref] $w_{agg}, S_o \rightarrow n$	How many states border Colorado?	1. Return Colorado 2. Return border states of #1 3. **Return the number of #2**
Group	Return [aggregate] [ref1] for each [ref2] $w_{agg}, S_o, S_e \rightarrow S_n$	How many female students are there in each club?	1. Return clubs 2. Return female students of #1 3. **Return the number of #2 for each #1**
Superlative	Return [ref1] where [ref2] is [highest / lowest] $S_e, S_n, w_{sup} \rightarrow S_e$	What is the keyword, which has been contained by the most number of papers?	1. Return papers 2. Return keywords of #1 3. Return the number of #1 for each #2 4. **Return #2 where #3 is highest**
Comparative	Return [ref1] where [ref2] [comparison] [number] $S_e, S_n, w_{com}, n \rightarrow S_e$	Who are the authors who have more than 500 papers?	1. Return authors 2. Return papers of #1 3. Return the number of #2 for each of #1 4. **Return #1 where #3 is more than 500**
Union	Return [ref1] , [ref2] $S_o, S_o \rightarrow S_o$	Tell me who the president and vice-president are?	1. Return the president 2. Return the vice-president 3. **Return #1 , #2**
Intersection	Return [relation] in both [ref1] and [ref2] $w, S_e, S_e \rightarrow S_o$	Show the parties that have representatives in both New York state and representatives in Pennsylvania state.	1. Return representatives 2. Return #1 in New York state 3. Return #1 in Pennsylvania state 4. **Return parties in both #2 and #3**
Discard	Return [ref1] besides [ref2] $S_o, S_o \rightarrow S_o$	Find the professors who are not playing Canoeing.	1. Return professors 2. Return #1 playing Canoeing 3. **Return #1 besides #2**
Sort	Return [ref1] sorted by [ref2] $S_e, S_n \rightarrow \langle e_1 \ldots e_k \rangle$	Find all information about student addresses, and sort by monthly rental.	1. Return students 2. Return addresses of #1 3. Return monthly rental of #2 4. **Return #2 sorted by #3**
Boolean	Return [if / is] [ref1] [condition] [ref2] $S_o, w, S_o \rightarrow b$	Were Scott Derrickson and Ed Wood of the same nationality?	... 3. Return the nationality of #1 4. Return the nationality of #2 5. **Return if #3 is the same as #4**
Arithmetic	Return the [arithmetic] of [ref1] and [ref2] $w_{ari}, n, n \rightarrow n$	How many more red objects are there than blue objects?	... 3. Return the number of #1 4. Return the number of #2 5. **Return the difference of #3 and #4**

Table 1: The 13 operator types of QDMR steps. Listed are, the natural language template used to express the operator, the operator signature and an example question that uses the query operator in its decomposition.

BreakRC: Break-it-down Reading Comprehension

1. Parse the question to its QDMR representation
   - using a seq2seq model trained on dataset
2. Execute each step of QDMR algorithmically
   - **Retrieve** top paragraphs e.g. using **TF-IDF**
   - **Reader**: paragraphs, question → answer
     - interpret the operation and execute

TF-IDF: term frequency–inverse document frequency, measure of importance of a word to a document in a collection or corpus, adjusted for the fact that some words appear more frequently in general

Example

Question
Which university has the bigger campus, Stanford or UC Berkeley?

Step 1

QDMR Decomposition
1. Return the campus size of UC Berkeley.
2. Return the campus size of Stanford.
3. Return which is the greater of #1, #2

Step 2: For each question

1. Answer("campus size of UC Berkeley?")
   - 1,232 acres
2. Answer("campus size of Stanford?")
   - 8,180 acres
3. argmax("1,232 acres", "8,180 acres")
   - Stanford

HotpotQA Dataset (Yang et al.)

Paragraph A: The 2015 Diamond Head Classic was a college basketball tournament ... *Buddy Hield was named the tournament's MVP*.
Paragraph B: *Chavano Rainier "Buddy" Hield* is a Bahamian professional basketball player for the **Sacramento Kings** of the NBA...
Q: Which team does the player named 2015 Diamond Head Classic's MVP play for?

- Crowdworkers
  - read the 1st paragraph of 2 Wikipedia articles with the same entity
  - ask a question that requires both paragraphs to answer
- 112K question-answer pairs

Yang et al. "HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering." ACL (2018).

Evaluation of Break It Down

- Baseline: BERTQA is BERT trained on HotpotQA (no breakdown)
  - Find the answer in the concatenation of 10 paragraphs
  - TF-IDF is applied to the user question

	EM	F1
BERTQA	33.6	43.3

Wolfson et al. "Break It Down: A Question Understanding Benchmark." TACL (2020).

Evaluation of Break It Down

- BreakRC uses Break-it-Down to generate subquestions
  - Retrieve each subquestion with TF-IDF;
    Gold is the upperbound – assume no errors in breakdown

	EM	F1
BERTQA	33.6	43.3
BreakRC (gold)	34.6	44.6
BreakRC (generated)	28.8	37.7

Wolfson et al. "Break It Down: A Question Understanding Benchmark." TACL (2020).

Evaluation of Break It Down

- Combined:
  - Uses BreakRC to fetch data on subquestions
  - There are errors in finding the answers using BreakRC
  - Feeds all the fetched paragraphs to BERTQA

	EM	F1
BERTQA	33.6	**43.3**
BreakRC (gold)	34.6	44.6
BreakRC (generated)	28.8	37.7
Combined (gold)	41.2	52.4
Combined	38.3	**49.3**

Wolfson et al. "Break It Down: A Question Understanding Benchmark." TACL (2020).

Discussion

- Breaking down the question is a good idea
- QDMR: Question Decomposition Meaning Representation uses informal English version of "SQL" primitives
  - If LLMs understand SQL queries, can we just use SQL syntax?
    - Does that simplify the implementation?

Lecture Goal

Domain Corpus

Knowledge Bases
Free-Text → **Framework** → **Knowledge Chatbot**

- (Non-hybrid) multi-hop retrieval
- **Prior work on hybrid data sources**
- SUQL: a proposal

Hybrid Data QA

Combining
Knowledge Bases & Textual QA

Quiz

Knowledge Bases & Text

What is your approach?

1. Classify the Question: KB or Text

"What do others think?": Task-Oriented Conversational Modeling with Subjective Knowledge

Chao Zhao [1] Spandana Gella [2] Seokhwan Kim [2] Di Jin [2]
Devamanyu Hazarika [2] Alexandros Papangelis [2] Behnam Hedayatnia [2]
Mahdi Namazifar [2] Yang Liu [2] Dilek Hakkani-Tur [2]
zhaochao@cs.unc.edu {sgella,seokhwk,djinamzn}@amazon.com
{dvhaz,papangea,behnam,mahdinam,yangliud,hakkanit}@amazon.com
[1] UNC Chapel Hill [2] Amazon, Alexa

Published 10/3/2023

https://browse.arxiv.org/pdf/2305.12091.pdf

SK-TOD Dataset

Subject-Knowledge Task-Oriented Dataset

"The first dataset, which contains subjective knowledge-seeking dialogue contexts and manually annotated responses grounded in subjective knowledge sources."

- MultiWOZ restaurant and hotel (in Cambridge area)
  - User goal: search and book restaurants and hotels
  - Structure: 13 slots (easier than SQL)
  - Text: 143 entities and 1,430 reviews (8,013 sentences)

Largest TOD Dataset with Subjective Knowledge

	Size	Manual	Dial	TOD	Query	Aspect	Senti	Mul-Knwl	Senti-%
Semeval/MAMS (2016; 2019)	5K/22K	✓	✗	n/a	✗	✓	✓	✗	n/a
Space (2021)	1K	✓	✗	n/a	✗	✓	✓	✓	✗
Yelp/Amazon (2019; 2020)	200/180	✓	✗	n/a	✗	✗	✓	✓	✗
Justify-Rec (2019)	1.3M	✗	✗	n/a	✗	✓	✗	✓	✗
AmazonQA (2016)	309K	✗	✗	n/a	✓	✗	✗	✗	n/a
SubjQA (2020)	10K	✗	✗	n/a	✓	✓	✓	✗	n/a
Holl-E (2018)	9K	✓	✓	✗	✗	✗	✗	✓	✗
Foursquare (2018)	1M	✗	✓	✗	✗	✗	✗	✓	n/a
SK-TOD (Ours)	20K	✓	✓	✓	✓	✓	✓	✓	✓

Table 1: Comparison between SK-TOD and other benchmarks based on the subjective content. We consider if the dataset is manually annotated, dialogue-based, task-oriented, and query-focused. We also list if it considers aspect and sentiment, multiple knowledge snippets (Mul-Knwl), and the proportion of two-sided sentiments (Senti-%).

Binary Classifier (IR vs Semantic Parsing)

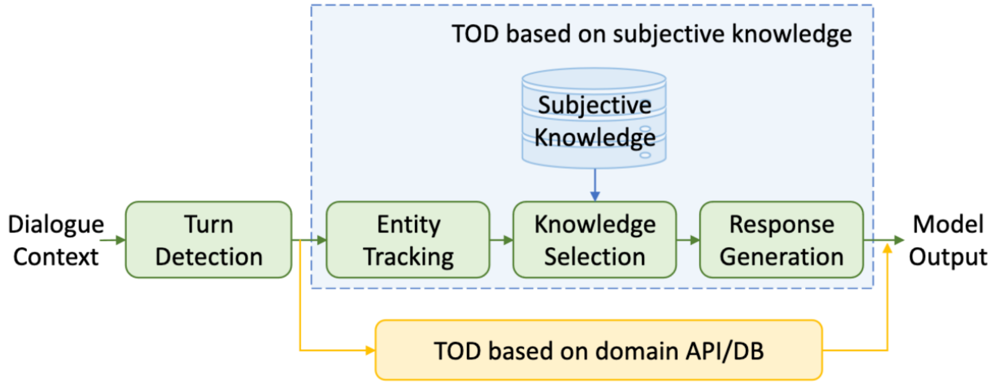


Figure 2: The pipeline architecture of SK-TOD.

Turn detection: binary classification

INFORMATION RETRIEVAL (IR)

Entity Tracking
Fuzzy n-gram matching between dialogue &entities. (A few entities in MultiWOZ)

Knowledge Selection
Calculate relevant knowledge score between dialogue context & knowledge snippets

Response Generation
Use pretrained GPT-2/Bart

2. Choose Later: Stanford Chirpy Cardinal

Neural Generation Meets Real People: Building a Social, Informative Open-Domain Dialogue Agent

Ethan A. Chi*, **Ashwin Paranjape***, **Abigail See***, **Caleb Chiam***, **Trenton Chang**, **Kathleen Kenealy**, **Swee Kiat Lim**, **Amelia Hardy**, **Chetanya Rastogi**, **Haojun Li**, **Alexander Iyabor**, **Yutong He**, **Hari Sowrirajan**, **Peng Qi**, **Kaushik Ram Sadagopan**, **Nguyet Minh Phu**, **Dilara Soylu**, **Jillian Tang**, **Avanika Narayan**, **Giovanni Campagna**, and **Christopher D. Manning**

- Alexa Social Bot Prize Runner up, 2021
- Different modules to handle different kinds of questions
- Run generators in parallel
- Choose at the end

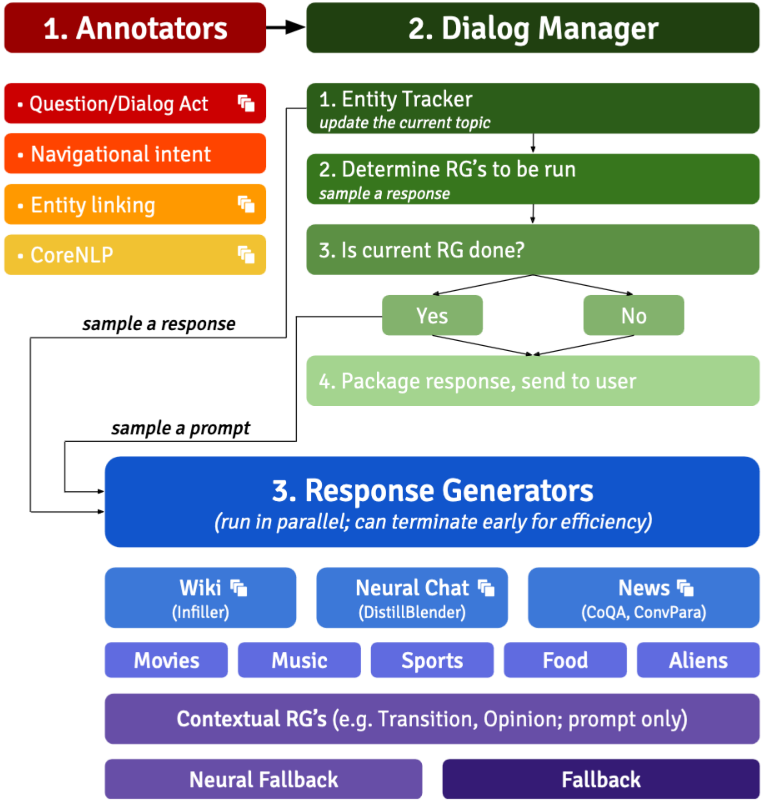


Figure 4: Overall system design.

Proceedings of the SIGdial 2022 Conference, pages 376–395

Problem with Choosing Between Modules

Real User Questions

naturally span across both structured and free text columns.

Hey! Can you recommend me an Italian restaurant with a romantic atmosphere?

Name TEXT	Cuisine TEXT[]	Rating NUM(2,1)	...	Popular_dishes FREE TEXT	Reviews FREE TEXT
Hummus Mediterranean Kitchen	mediterranean,halal, salad	4.0	...	Chicken Kebab Plate, Lamb Beef Gyro, Marinated Chicken Gyros, ...	| t l ; d r | This is your place if you're looking for a healthy and filling meal whether it's a quick pick-me-up or casual dining, ...
Penny Roma	italian, venues & event spaces	4.0	...	Cacio E Pepe, Agnolotti Dal Plin, Albacore Tartare, ...	My girlfriend was craving pasta on a Monday night. ... We were not expecting such an intimate and romantic dining experience. The restaurant was candle lit, modern, and perfect for a date night. ...

- Real-user queries about restaurants: 55/100 need a combo
- Separate modules cannot answer these questions

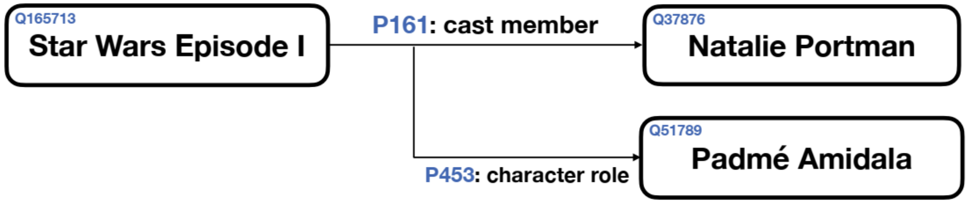
3. Converting All to Text

- Unified Knowledge Question Answering (UniK-QA)
- Converts everything to text
  - Text
  - Semi-structured data e.g. Wikipedia lists, tables and info boxes
  - Knowledge base
- Then uses the usual pipeline designed for textual QA

Oguz et al. "UniK-QA: Unified Representations of Structured and Unstructured Knowledge for Open-Domain Question Answering.“ arXiv preprint arXiv:2012.14610 (2021).

Wikipedia Tables / Wikidata

- Wikipedia tables are linearized simply by concatenating cell values with special tokens to separate rows
- Wikidata:



Converted Text:

Star Wars Episode I cast member Natalie Portman, and character role Padmé Amidala .

Oguz et al. "UniK-QA: Unified Representations of Structured and Unstructured Knowledge for Open-Domain Question Answering.“ Findings of NAACL (2022).

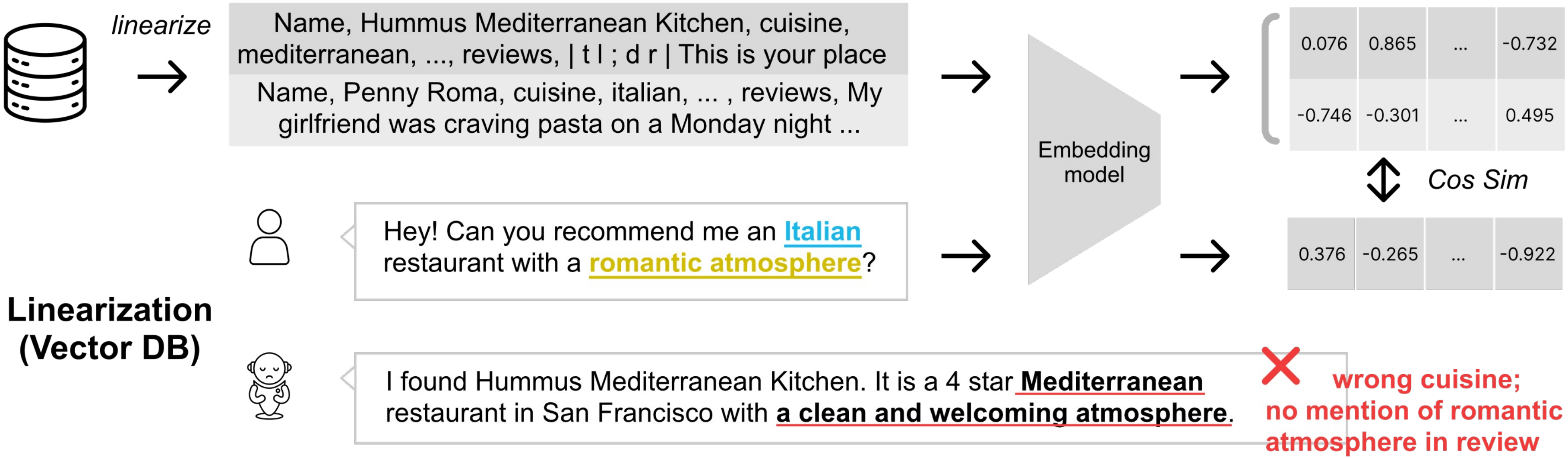
Unified Knowledge Question Answering

- Retrieve and read QA pipeline with SOTA components
  - Neural retriever; T5 reader
- Exact match scores:

Knowledge Source	Natural Questions	TriviaQA	WebQuestions
Wikipedia Text	49.0	64.0	50.6
Wikipedia Tables	36.0	34.5	41.0
Wikidata	27.9	35.4	55.6
Text + Tables	**54.1**	**65.1**	50.2
Text + Tables + Wikidata	54.0	64.1	**57.8**

Quiz: What do you observe?

Testing with Yelp



Problems with Linearization

- How to combine different clauses together?
  (Algebraic operators handle them naturally)
    - e.g. multiple clauses: "Are there any restaurants that have **a great view of the Golden Gate bridge** in **San Francisco**?"

- Hard to do column computations
  (rank, max, min, average, count) on flattened data
    - e.g. "what is the **top-rated** Chinese restaurant?"

4. Hybrid: Semantic Parsing + IR

HybridQA: A Dataset of Multi-Hop Question Answering over Tabular and Textual Data

Wenhu Chen, Hanwen Zha, Zhiyu Chen, Wenhan Xiong, Hong Wang, William Wang
University of California, Santa Barbara, CA, USA
{wenhuchen, hwzha, xwhan}@cs.ucsb.edu,
{zhiyuchen, hongwang600, william}@cs.ucsb.edu

Findings of the Association for Computational Linguistics: EMNLP 2020, pages 1026–1036
November 16 - 20, 2020.

Earlier than SK-TOD. Note: this is not conversational

HybridQA Dataset

Wikipedia Tables *hyperlinked* Wikipedia Pages

The 2016 Summer Olympics officially known as the Games of the XXXI Olympiad (Portuguese : Jogos da XXXI Olimpíada) and commonly known as **Rio** 2016 , was an international multi-sport event ……

Flag bearers of Myanmar at the Olympics

Name	Year	Season	Flag bearer
XXXI	2016	Summer	Yan Naing Soe
XXX	2012	Summer	Zaw Win Thet
XXIX	2008	Summer	Phone Myint Tayzar
XXVIII	2004	Summer	Hla Win U
XXVII	2000	Summer	Maung Maung Nge
XX	1972	Summer	Win Maung

Yan Naing Soe (born **31 January 1979**) is a Burmese judoka . He competed at the 2016 Summer Olympics in the **men 's 100 kg event** , …… He was the flag bearer for Myanmar at the **Parade of Nations** .

Zaw Win Thet (born **1 March 1991** in Kyonpyaw , Pathein District , Ayeyarwady Division , Myanmar) is a Burmese runner who ……

Myint Tayzar Phone (Burmese : မြင့်တေဇာဖုန်း) born **July 2 , 1978**) is a sprint canoer from Myanmar who competed in the late 2000s .

……

Win Maung (born **12 May 1949**) is a Burmese footballer . He competed in the men 's tournament at the 1972 Summer Olympics …

Hardness

Q: In which year did the judoka bearer participate in the Olympic opening ceremony?	A: 2016
Q: Which event does the does the XXXI Olympic flag bearer participate in?	A: men's 100 kg event
Q: Where does the Burmesse jodoka participate in the Olympic opening ceremony as a flag bearer?	A: Rio
Q: For the Olympic event happening after 2014, what session does the Flag bearer participate?	A: Parade of Nations
Q: For the XXXI and XXX Olympic event, which has an older flag bearer?	A: XXXI
Q: When does the oldest flag Burmese bearer participate in the Olympic ceremony?	A: 1972

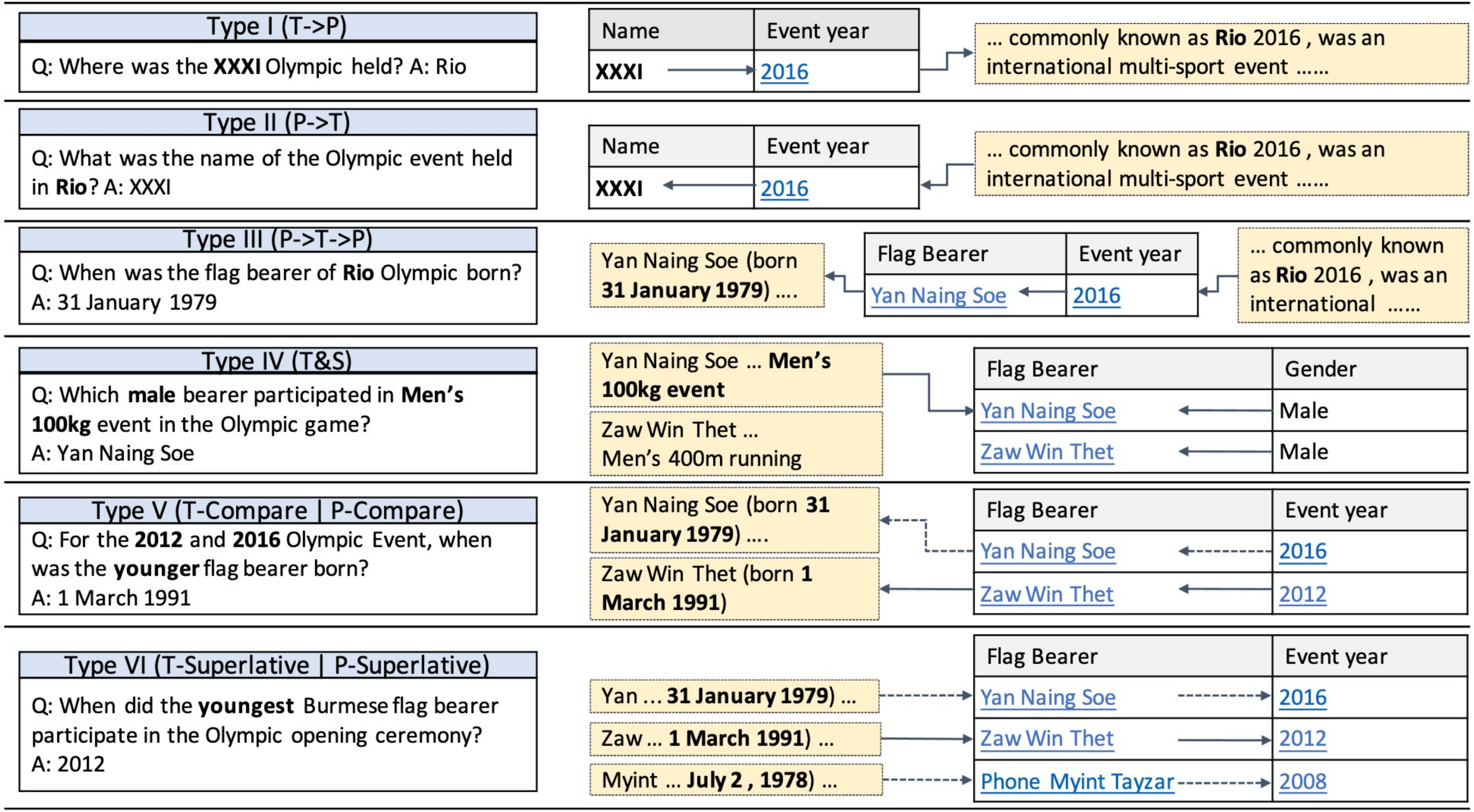


Figure 3: Illustration of different types of multi-hop questions.

T: Table
P: Passage

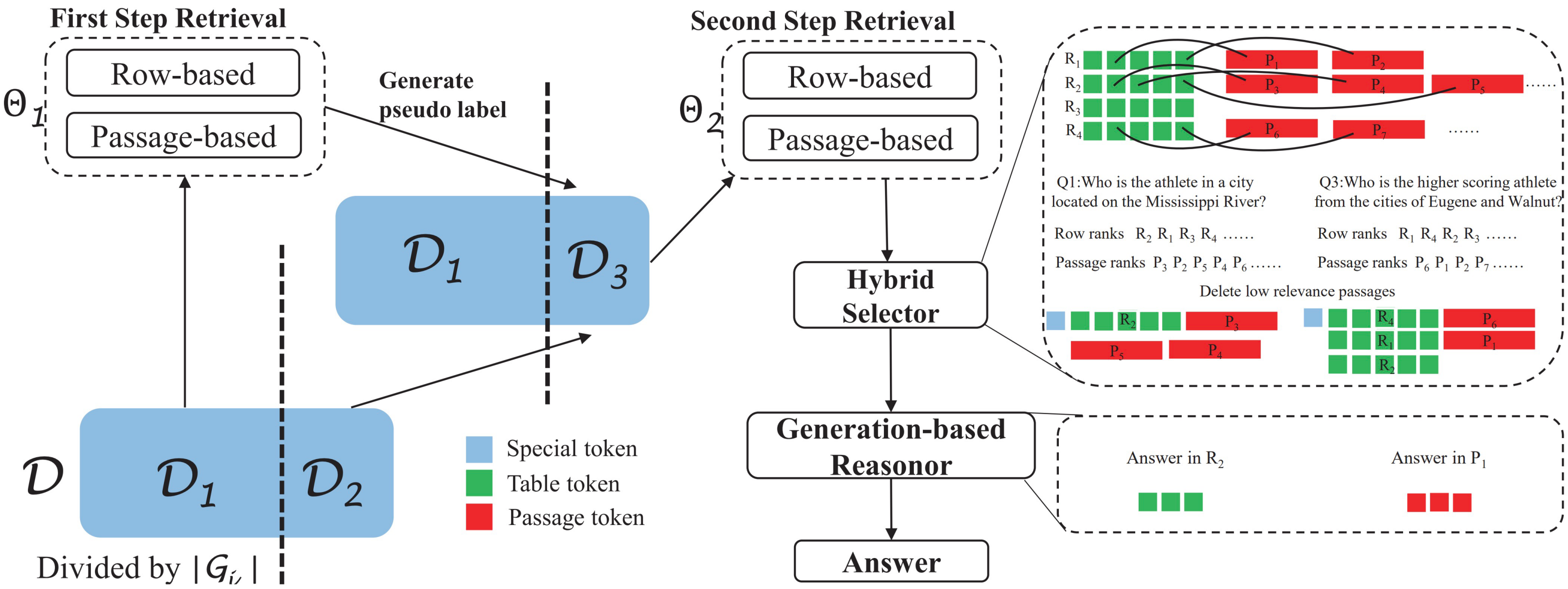
SOTA Model

S[3]HQA: A Three-Stage Approach for Multi-hop Text-Table Hybrid Question Answering

Fangyu Lei[1,2], Xiang Li[1,2], Yifan Wei[1,2],
Shizhu He[1,2], Yiming Huang[1,2], Jun Zhao[1,2], Kang Liu[1,2]
[1]The Laboratory of Cognition and Decision Intelligence for Complex Systems
Institute of Automation, Chinese Academy of Sciences
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences
{leifangyu2022, lixiang2022, weiyifan2021, huangyiming2023}@ia.ac.cn
{shizhu.he, jzhao, kliu}@nlpr.ia.ac.cn

https://browse.arxiv.org/pdf/2305.11725.pdf

Hybrid Retrievers



Only uses LLM in the "generation-based reasoner" step

https://github.com/wenhuchen/HybridQA#recent-papers
SOTA model: https://arxiv.org/abs/2305.11725

3 Steps

1. Retriever with Refinement Training:
   - To recover from annotation errors
2. Hybrid selector
   - Sorting and filtering based on relevance
3. LLM-Based Reasoner
   - Question types: Count, compare
   - Use lexical analysis to identify the question type
   - Chain-of-thought prompting

SOTA Result

	Table				Passage				Total			
	Dev		Test		Dev		Test		Dev		Test	
	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1
Unsupervised-QG (Pan et al., 2021)	-	-	-	-	-	-	-	-	25.7	30.5	-	-
HYBRIDER (Chen et al., 2020b)	54.3	61.4	56.2	63.3	39.1	45.7	37.5	44.4	44.0	50.7	43.8	50.6
DocHopper (Sun et al., 2021)	-	-	-	-	-	-	-	-	47.7	55.0	46.3	53.3
$MuGER^2$ (Wang et al., 2022b)	60.9	69.2	58.7	66.6	56.9	68.9	57.1	68.6	57.1	67.3	56.3	66.2
POINTR (Eisenschlos et al., 2021)	68.6	74.2	66.9	72.3	62.8	71.9	62.8	71.9	63.4	71.0	62.8	70.2
DEHG (Feng et al., 2022)	-	-	-	-	-	-	-	-	65.2	**76.3**	63.9	75.5
MITQA (Kumar et al., 2021)	68.1	73.3	68.5	74.4	66.7	75.6	64.3	73.3	65.5	72.7	64.3	71.9
MAFiD (Lee et al., 2023)	69.4	75.2	68.5	74.9	66.5	75.5	65.7	75.3	66.2	74.1	65.4	73.6
S^3HQA	**70.3**	**75.3**	**70.6**	**76.3**	**69.9**	**78.2**	**68.7**	**77.8**	**68.4**	75.3	**67.9**	**75.5**
Human	-	-	-	-	-	-	-	-	-	-	88.2	93.5

Table 1: Performance of our model and related work on the HybridQA dataset.

Quiz: What do you observe?

Problem

- Data are retrieved in just one hop
  - Cannot solve cascading multi-hop questions, where the data to retrieve depends on the 1st hop answer
- Imprecision with the lexical analysis
- Lacks completeness and compositionality
  - Just one “count” or one “compare” operation

Summary

- **Many questions require information from hybrid data sources**
- **Structure OR Text: is inadequate**
  - Binary classifier up front (SK-TOD, 2023)
  - Pick afterwards (Stanford Chirpy Cardinal, 2021)
- **Different approaches to combine structures and free-text**
  - Structures → Text: Linearization (one hop)
  - Text → Structure: Semantic parser (Hard to represent free text in KB)
  - Hybrid: Retrieve from both and combine (one hop each)
- **Hybrid questions are multi-hop questions**
  - Break-it-Down: the only truly multi-hop solution for text only

Quiz: What should we do given prior results?

Desired Solution

1. Keep hybrid data sources
   - Don't convert KB → Text, or Text → KB
2. Hybrid multi-hop: Compose IR and KB accesses arbitrarily
3. Break-it-down decomposes questions into "English-like" SQL operations
   - Has issues with executing "English" operations
   - Can't support DBes

Since LLMs understand SQL

- SUQL (Structured & Unstructured Query Language) **NEW!**
- Extends SQL to include free-text operation with structured access
- Create an optimizing compiler to support full SUQL (all compositions)

Lecture Goal

Domain Corpus

Knowledge Bases
Free-Text → **Framework** → **Knowledge Chatbot**

- (Non-hybrid) multi-hop retrieval
- Prior work on hybrid data sources
- **SUQL: a proposal**

SUQL Free-Text Support

- Add free-text primitives into SQL, implemented with IR & LLM
- Two functions: summary, answer

“I want a family-friendly restaurant in Palo Alto”

↓

SELECT *, summary(reviews) FROM restaurants
WHERE location = 'Palo Alto’
AND answer(reviews, ‘is it a family friendly restaurant’) = 'Yes' LIMIT 1;

Quiz

How do you implement summary and answer?

HybridQA Questions in SUQL

```
CREATE TABLE Flag (
"Name" TEXT,
"Flag Bearer" TEXT,
"Flag Bearer_Info" TEXT[],
"Gender" TEXT,
"Event year" TEXT,
"Event year_Info" TEXT[]);
```

T: Table
P: Paragraph

Type I (T->P)

Where was the XXXI Olympic held?

```
SELECT answer("Event year_Info",
                    'where is this event held?')
        FROM "Flag" WHERE "Name" = 'XXXI'
```

Type II (P->T) What was the name of the Olympic event held in Rio?

```
SELECT "Name" FROM "Flag"
WHERE answer("Event year_Info",
        'where is this event held?') = 'Rio'
```

HybridQA Questions in SUQL

```
CREATE TABLE Flag (
"Name" TEXT,
"Flag Bearer" TEXT,
"Flag Bearer_Info" TEXT[],
"Gender" TEXT,
"Event year" TEXT,
"Event year_Info" TEXT[]);
```

T: Table
P: Paragraph

Type III (P->T->P) When was the flag bearer of Rio Olympic born?

```
SELECT answer("Flag Bearer_Info", 'when is this person born?')
FROM "Flag"
WHERE answer("Event year_Info", 'where is this event held?') = 'Rio'
```

Type IV (T&P) Which **male** bearer participated in **Men's 100kg** event in the Olympic game?

```
SELECT "Flag Bearer" FROM "Flag" WHERE "Gender" = 'Male' AND
answer("Flag Bearer_Info",
        'what event did this person participate in?')
        = "Men's 100kg event"
```

HybridQA Questions in SUQL

```
CREATE TABLE Flag (
"Name" TEXT,
"Flag Bearer" TEXT,
"Flag Bearer_Info" TEXT[],
"Gender" TEXT,
"Event year" TEXT,
"Event year_Info" TEXT[]);
```

T: Table
P: Paragraph

Type V (T-Compare | P-Compare)

For the **2012** and 2016 Olympic Event, when was the younger flag bearer born?

```
SELECT MAX
(answer("Flag Bearer_Info",  'when is this person born?')::date)
FROM "Flag" WHERE "Event year" IN ('2016', '2012')
```

Type VI (T-Superlative | P-Superlative) When did the **youngest** Burmese flag bearer participate in the Olympic opening ceremony?

```
SELECT "Event year" FROM "Flag" ORDER BY
answer("Flag Bearer_Info", 'when is this person born?')::date
DESC LIMIT 1;
```

Pros and Cons of SUQL

- **Pros: Formal representation & semantic parsing**
  - Compositionality
  - Domain independence
  - Interpretability
  - Allows query optimization over the whole expression (better than Break-it-down)
- **Cons: It is new – many unknowns**
  - Can LLMs generate the right formal representation?
    - Choosing between the different fields
    - Can it generate complex queries
  - What is the speed?

Examples

Restaurants

Do you have a recommendation for a first date restaurant in Palo Alto?
We're thinking sushi but not sure what's good around here.

↓

```
SELECT *, summary(reviews) FROM restaurants
WHERE 'sushi' = ANY (cuisines) AND location = 'Palo Alto' AND rating >= 4.0
AND answer(reviews, 'is this restaurant good for a first date?') = 'Yes’
ORDER BY num_reviews DESC LIMIT 1;
```

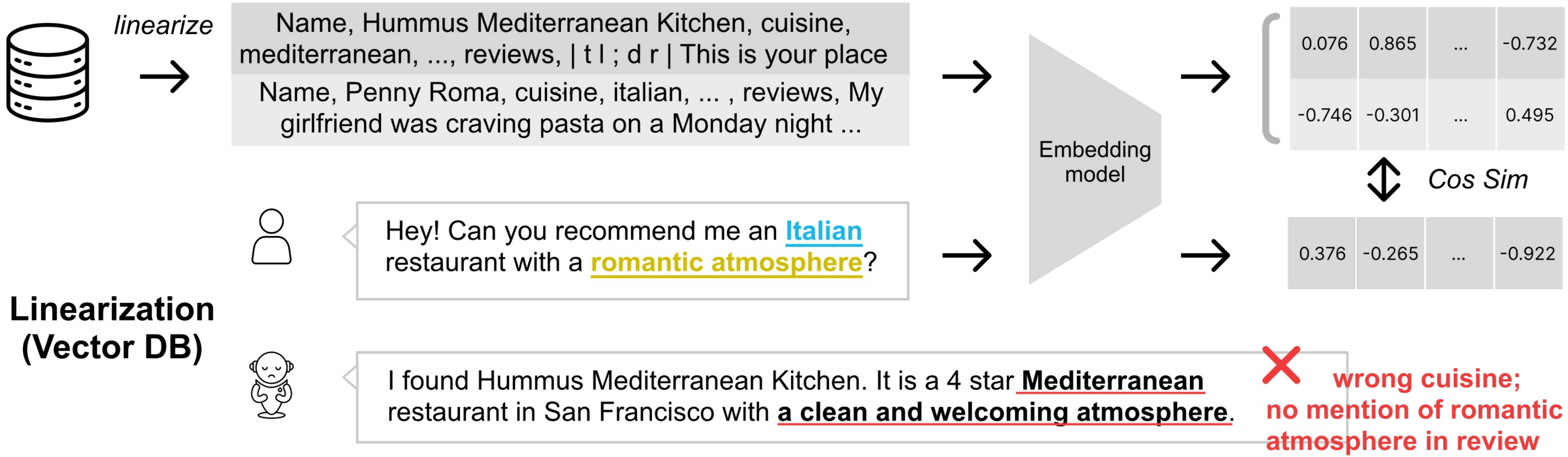
Laptops

I need a laptop with a Thunderbolt 3 port and at least 16GB RAM for my workstation setup.

↓

```
SELECT *, summary(reviews) FROM laptops
WHERE ram >= 16 AND
(answer(about, 'does this laptop have Thunderbolt 3?') = 'Yes’
OR answer(description, 'does this laptop have Thunderbolt 3?') = 'Yes’)
LIMIT 3;
```

Previous Example



Using SUQL

Hey! Can you recommend me an Italian restaurant with a romantic atmosphere?

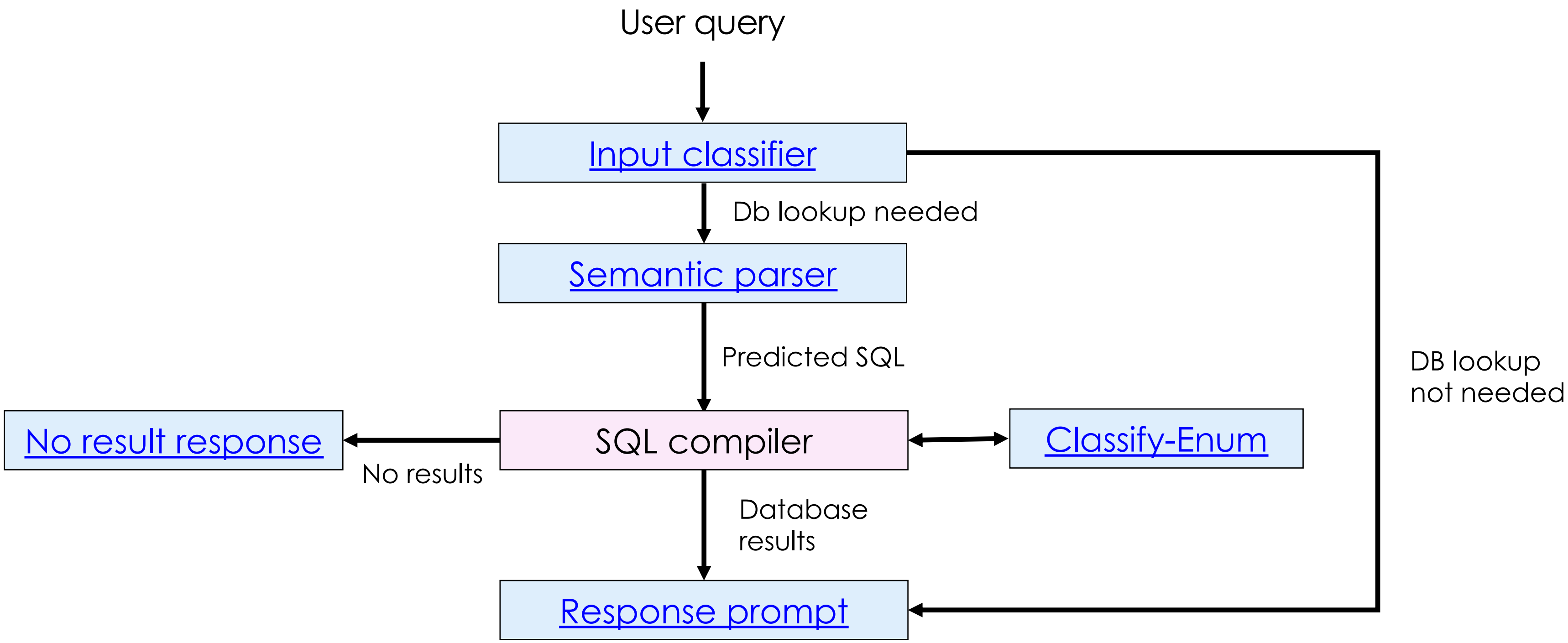
↓ *Semantic Parser*

```
SELECT *, summary(reviews) FROM restaurants
WHERE 'italian' = ANY (cuisines) AND
answer(reviews, 'is this restaurant romantic?') = 'Yes'  LIMIT 1;
```

SUQL Compiler →

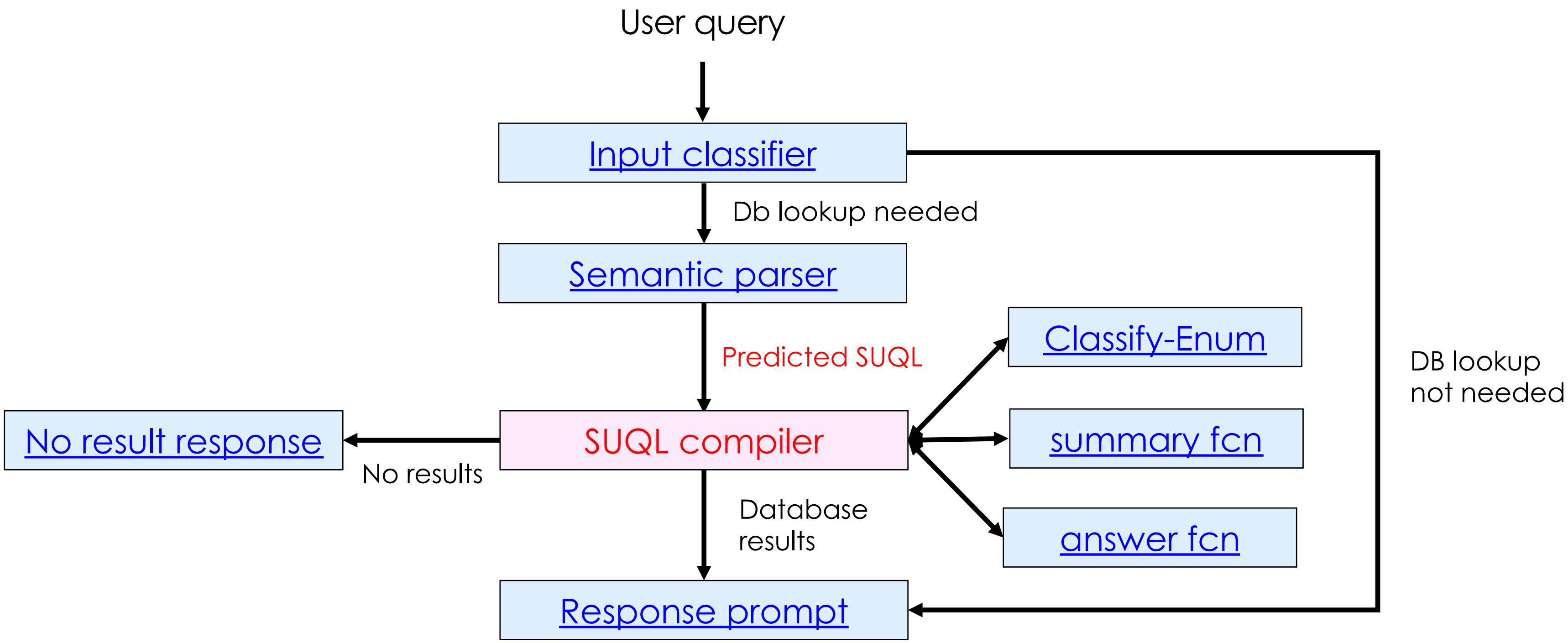
I found Penny Roma, which has a 4.0 rating on our database and offers a variety of Italian dishes. Overall, the atmosphere is described as delightful, authentic, and perfect for a date spot.

Agent Design for SQL



Click the links to see the prompts (written in jinja syntax)

Agent Design: SQL Updated with SUQL



7 prompts with few-shot examples.   Templates written in jinja syntax

Can We Build a Good SUQL Parser?

Semantic Parsing with Prompting LLMs

- LLMs know SQL
- Teach LLMs SUQL with few-shot examples with text fields

Find me a place with pasta.

↓

```
SELECT *, summary(reviews) FROM restaurants
WHERE answer(popular_dishes, 'does this restaurant serve pasta') = 'Yes' LIMIT 1;
```

I love Chinese food. Find me a restaurant that doesn't have a long wait time.

↓

```
SELECT *, summary(reviews), answer(reviews, 'what is the wait time?’) FROM restaurants
WHERE 'chinese' = ANY (cuisines)
AND answer(reviews, 'does this restaurant have short wait time?') = 'Yes' LIMIT 1;
```

Experiment

- Yelp on San Francisco, Palo Alto, Cupertino, Sunnyvale
  - A real, large dataset
- Scraped reviews and popular dishes information
- LLM: gpt-3.5-turbo
- Information retrieval on text fields are indexed with Coco-DR

- SQL can run SUQL programs, but it is too slow
  - Developed an optimizing SUQL compiler

COCO-DR reference: https://arxiv.org/abs/2210.15212

Preliminary Evaluation (in Restaurants)

- Dataset
  - Assessment: iterate on 50 simulated questions (dev set)
  - Evaluation: 100 real-user questions (test set)
- Parsing accuracy: 96.0% on dev and 94.0% on test.
- Query *precision*: #correct results / #results

	Dev set (N = 50)	Test set (N = 100)
Linearization (return 3 results)	58.0 %	52.2%
SUQL	**95.6%**	**93.8%**

Quiz: how can you get incorrect results with semantic parsing on SUQL?

Preliminary Query Recall: Missed All the Answers

Assistant fails to find an answer → verbalizes the user query

I searched for 5-star restaurants in Sunnyvale that serve kids food. Unfortunately, I couldn't find any search results. Is there anything else I can help you with?

In case of a parse error

14 false negatives on our test set (100 questions)

- Parsing: 2 errors (one syntactic, one enum field confusion) ✓
- Query evaluation
  - Structured: Can be improved
    - 4 Unsupported location service
    - 2 Opening hours errors
    - 2 dishes: searched in popular dishes, but results are in reviews
  - Free-text: 4 false negatives from ChatGPT call in 'answer' Quiz: Is this OK?

Conclusion

- Databases are an important source of knowledge
  - Query languages: formal, expressive, compositional
  - Natural conversations on DB content: series of queries
- Need for hybrid: databases and free-text
- SUQL: extends semantic parsing to hybrid data sources
  - Consider using SUQL for your project (small domain)
- Open questions for open, large data and knowledge bases
  - Possible solution: data synthesis

Future Topic

Appendix: Performance

Text Search over Entire Corpus is Too Slow

Query Optimization

Large Free-text Corpus

- Speed of query resolution

answer(reviews, ‘is it a family friendly restaurant’) = ‘Yes’

Problem: Takes too long to query over every review

Solution: Create a new field “if_family_friendly” in the database

answer(reviews, ‘is it a family friendly restaurant’) = ‘Yes’
→ if_family_friendly = True

Creating Databases from Free-Text

1. Use LLM to simulate users' common questions
2. Encourage LLM semantic parser on user questions to generate new fields (with few-shot examples)
3. Stuff the database
   - Give LLM the free-text and the DB form (JSON)
   - Ask LLM to fill in the form

Query Execution Optimization

Developed an optimizing SUQL compiler
to optimize the execution for all queries!

1. Return only necessary results
2. Order filtering to reduce slow operations
3. Lazy evaluation: produce results only when needed

1. Return Only Necessary Results

answer(reviews, ‘is it a family friendly restaurant’) = ‘Yes’

- For applications such as recommendation, it is not necessary to return all the answers
- IR uses embedding model (vector similarity) to return top candidates
- Return only top results to LLM-based answer functions

2. Order Filtering

answer(reviews, ‘is it a family friendly restaurant’) = ‘Yes’
AND
‘french’ = ANY(cuisines)

- Execution of structured predicates is much cheaper
- Always execute structured predicates first

3. Lazy Evaluation

answer(reviews, ‘is it a family friendly restaurant’) = ‘Yes’ AND ‘french’ = ANY(cuisines) **LIMIT 1**

- Lazy evaluation: Evaluate only when the result is needed
- No need to keep calling answer as soon as LIMIT 1 is reached

SUQL Compiler Overview

- For each SELECT statement with answer in it
  (begin with bottom node with no sub-queries)
  - Apply optimizations to the SELECT statement
  - Store the processed results in a temporary table *temp*
  - Substitute this statement with *SELECT from temp*
- SQL compiler handles the final processing