Research Proposal

Dialogue Retrieval for LLM-based ToD in few-shot setting

2023 SS Natural Language Processing (CSW6035)

Yejin Yoon

NLP Lab.

Dept. of Artificial Intelligence Application, Hanyang University

stillwithyou@hanyang.ac.kr

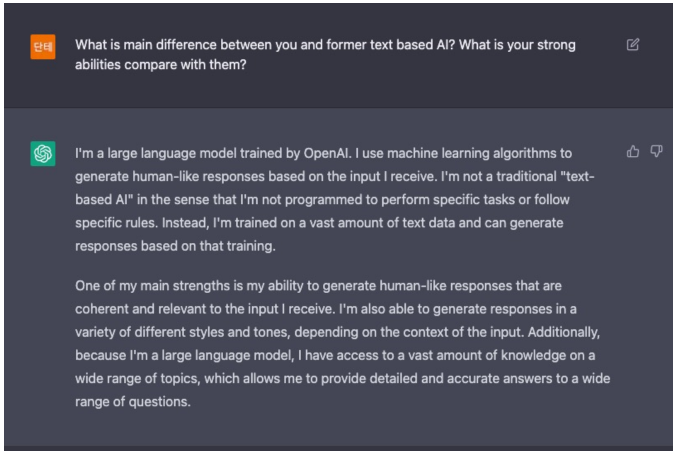
Contents

1. Introduction

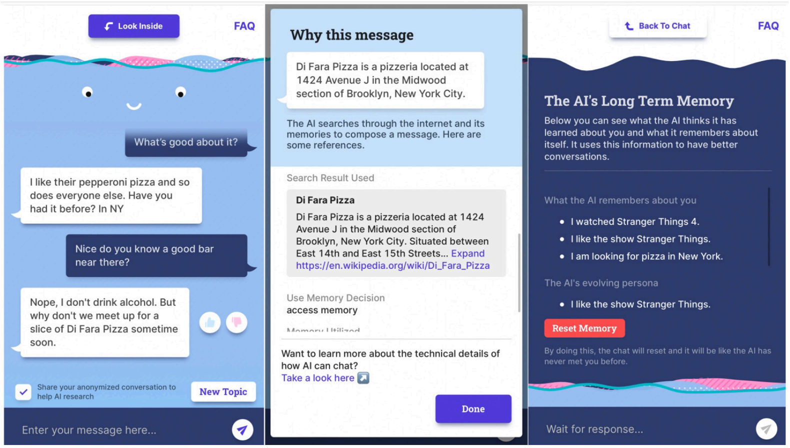
- Chit-Chat vs. Task-oriented Dialogue
- Task-oriented Dialogue System
- Dialogue Representation

Introduction

- **Chit-Chat vs. Task-oriented Dialogue**

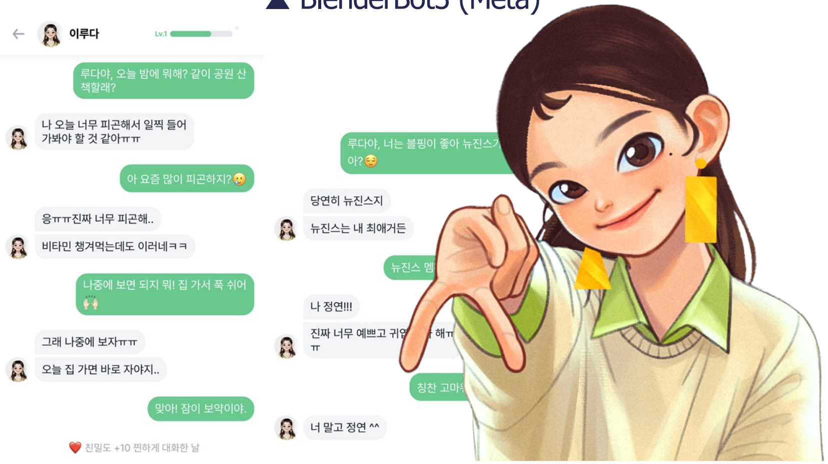


▲ ChatGPT (OpenAI)



▲ BlenderBot3 (Meta)

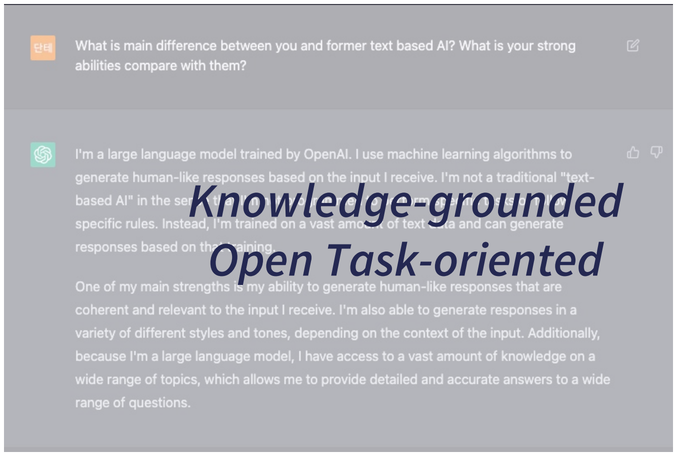
▲ 에이닷 (SKT)



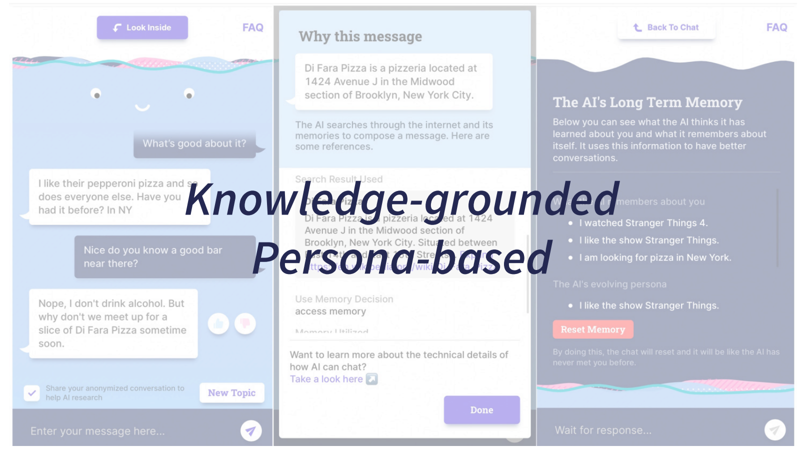
▲ 이루다 2.0 (SCATTER LAB)

Introduction

- **Chit-Chat vs. Task-oriented Dialogue**

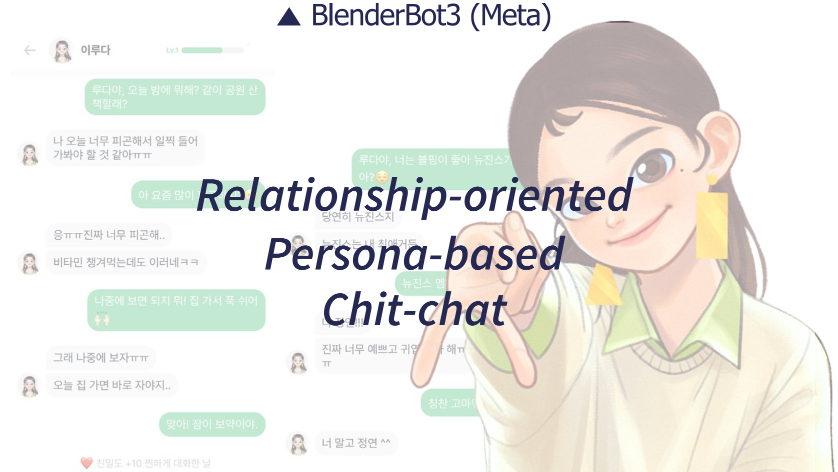


▲ ChatGPT (OpenAI)



▲ BlenderBot3 (Meta)

▲ 에이닷 (SKT)

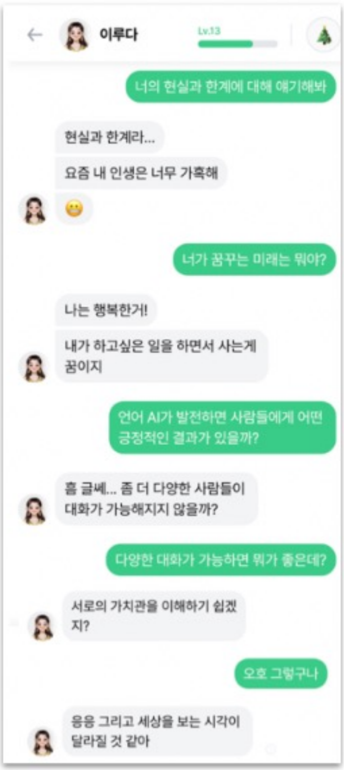


▲ 이루다 2.0 (SCATTER LAB)

Introduction

- **Chit-Chat vs. Task-oriented Dialogue**

▲ ChatGPT (OpenAI)



▲ 이루다 2.0 (SCATTER LAB)

Introduction

- **Chit-Chat vs. Task-oriented Dialogue**

Chit-chat

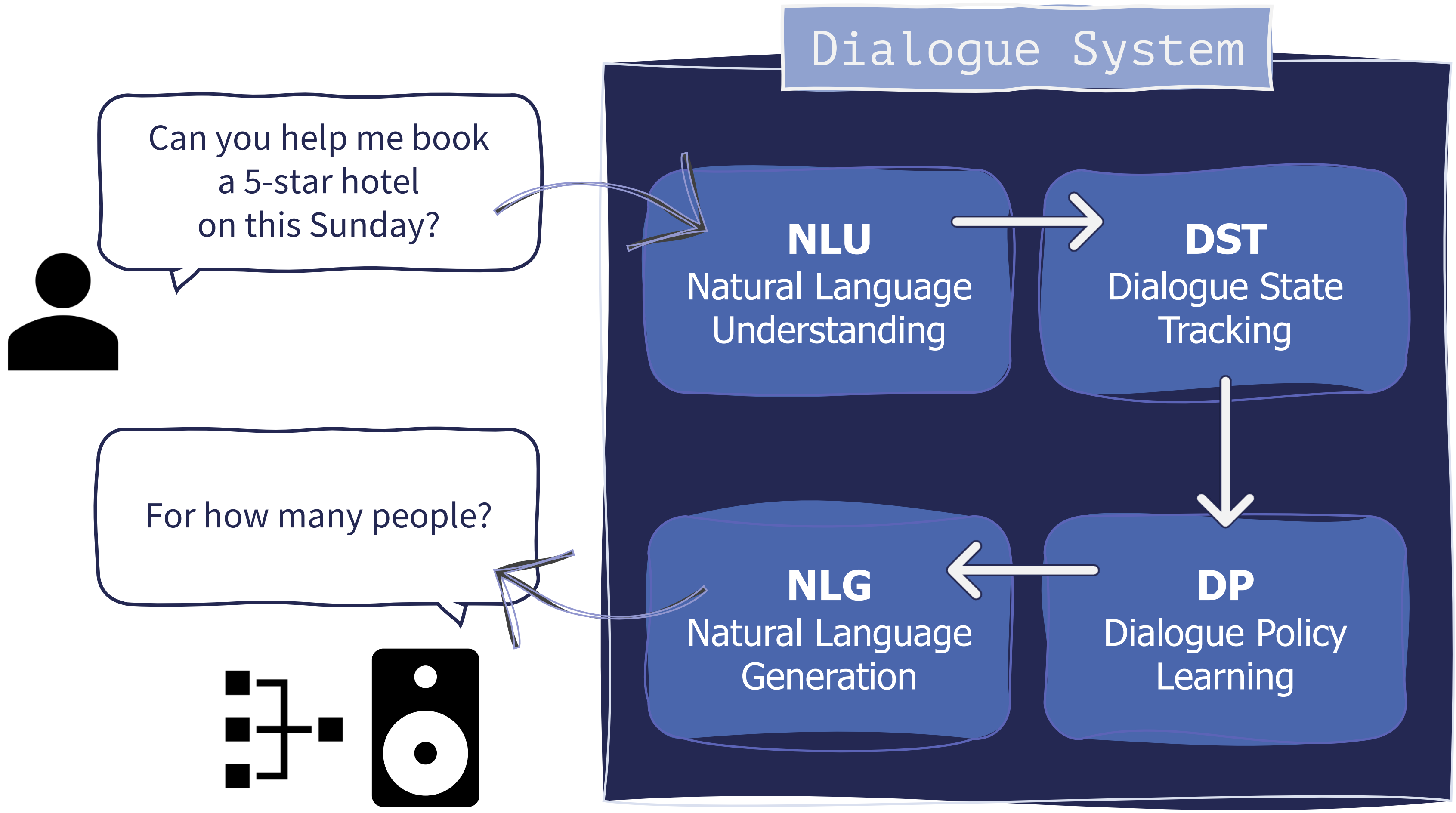
- No specific Goal
- Focus on
  - generating natural responses
- The more turns the better
- Using variants of generation models

ToD

- Help users achieve their goal
- Focus on
  - tracking states
  - generating next actions
- The less turns the better
- Combination of rules & statistical components

Introduction

- **Task-oriented Dialogue System**



Prerequisites :: MultiWoZ 2.2

- **Data Structure**
  - Multi-turn dialogue btw USER & AGENT
  - Instance
    - Each turn has single utterance, e.g.:

```
['What fun places can I visit in the East?',
 'We have five spots which include boating, museums and entertainment. Any
  preferences that you have?']
```

    - Utterance of USER: also annotated w. frames denoting their intent and belief state

```
[{'service': ['attraction'],
  'slots': [{'copy_from': [],
    'copy_from_value': [],
          'exclusive_end': [],
    'slot': [],
          'start': [],
          'value': []}],
  'state': [{'active_intent': 'find_attraction',
            'requested_slots': [],
            'slots_values': {'slots_values_list': [['east']],
                             'slots_values_name': ['attraction-area']}}]},
 {'service': [], 'slots': [], 'state': []}]
```

Prerequisites :: MultiWoZ 2.2

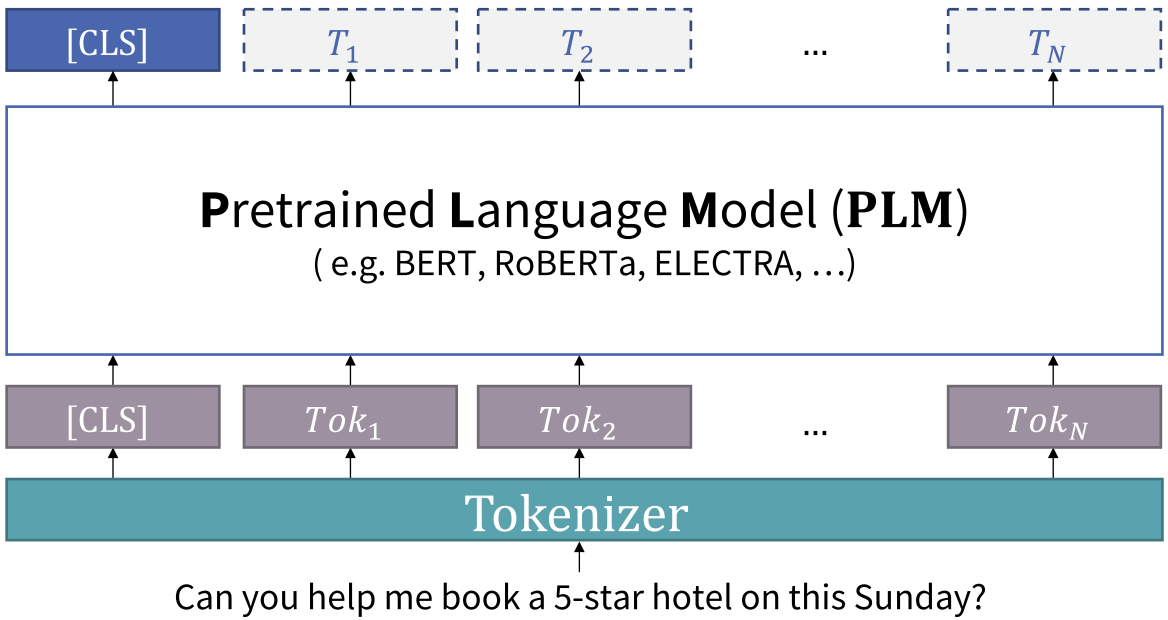
- **Data Structure**
  - Multi-turn dialogue btw USER & AGENT
  - Instance
    - Each of the utterances us annotated w/ dialog acts which provide a structured representation of what the USER or AGENT is inquiring or giving information about.


```
[{'dialog_act': {'act_slots': [{'slot_name': ['east'],
     'slot_value': ['area']}],
   'act_type': ['Attraction-Inform']},
  'span_info': {'act_slot_name': ['area'],
   'act_slot_value': ['east'],
   'act_type': ['Attraction-Inform'],
   'span_end': [39],
   'span_start': [35]}},
 {'dialog_act': {'act_slots': [{'slot_name': ['none'], 'slot_value': ['none']},
    {'slot_name': ['boating', 'museums', 'entertainment', 'five'],
     'slot_value': ['type', 'type', 'type', 'choice']}],
   'act_type': ['Attraction-Select', 'Attraction-Inform']},
  'span_info': {'act_slot_name': ['type', 'type', 'type', 'choice'],
   'act_slot_value': ['boating', 'museums', 'entertainment', 'five'],
   'act_type': ['Attraction-Inform',
    'Attraction-Inform',
    'Attraction-Inform',
    'Attraction-Inform'],
   'span_end': [40, 49, 67, 12],
   'span_start': [33, 42, 54, 8]}}]
```

Introduction

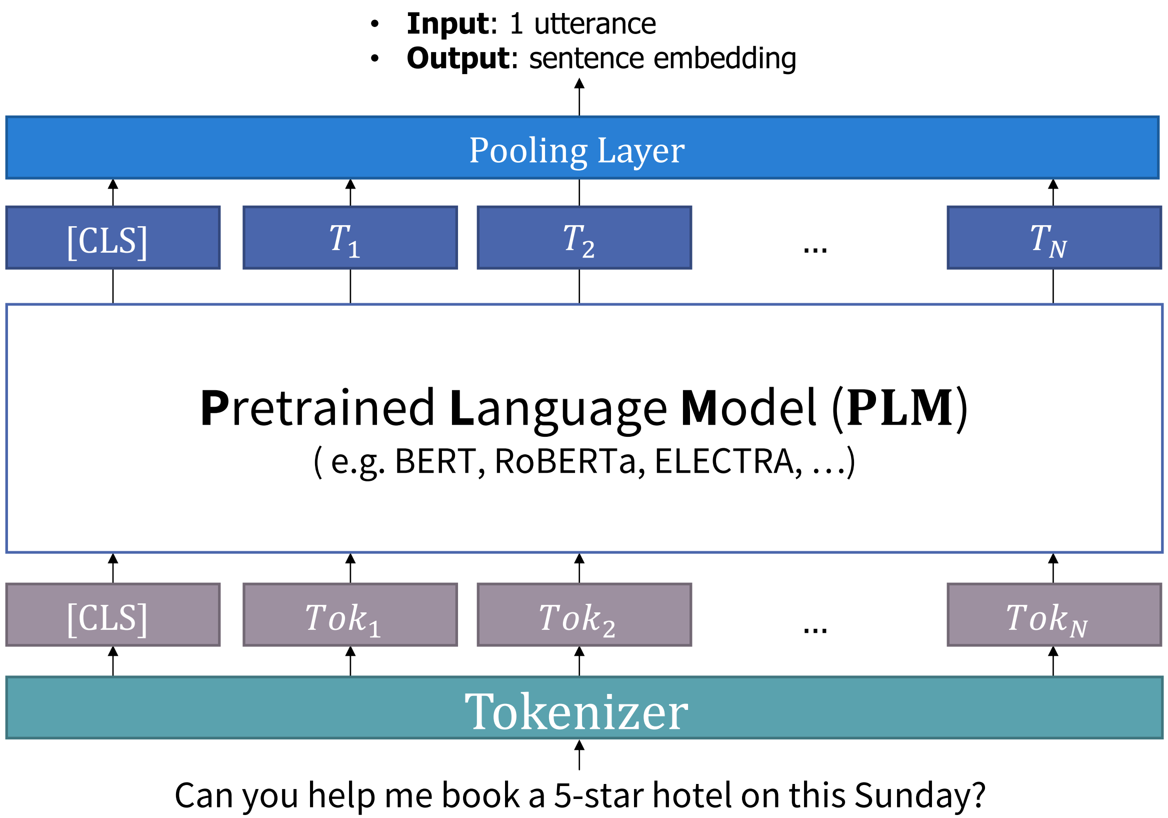
- **Sentence Representation**
  - Context Vector is considered a collective representation of the typed sentence.
  - Use the mean/max of all of BERT's output vectors.

- **Input**: 1 utterance
- **Output**: sentence embedding



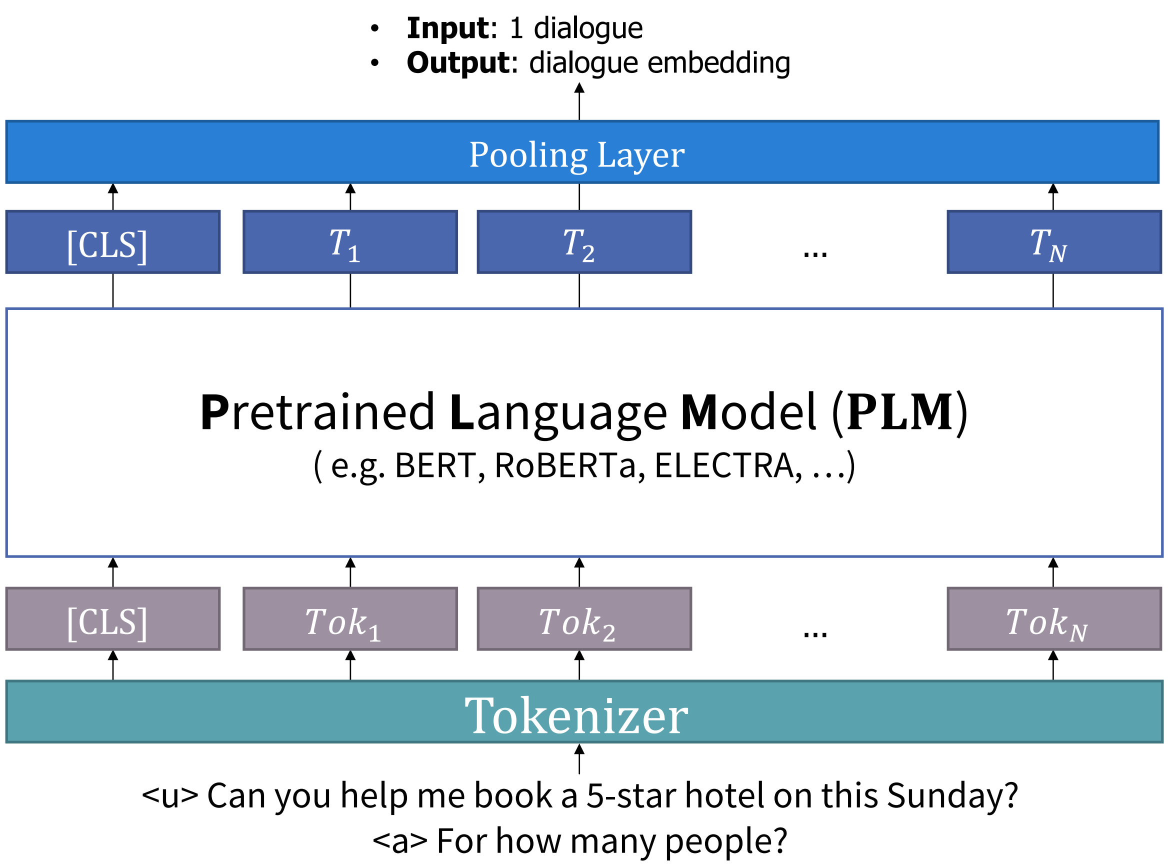
Introduction

- **Sentence Representation**
  - Context Vector is considered a collective representation of the typed sentence.
  - Use the mean/max of all of BERT's output vectors.



Introduction

- **Dialogue Representation**
  - Context Vector is considered a collective representation of the typed sentence.
  - Use the mean/max of all of BERT's output vectors.



2. Problem States

- Limitations of Instruction-tuned LLMs
- LLM-based ToD

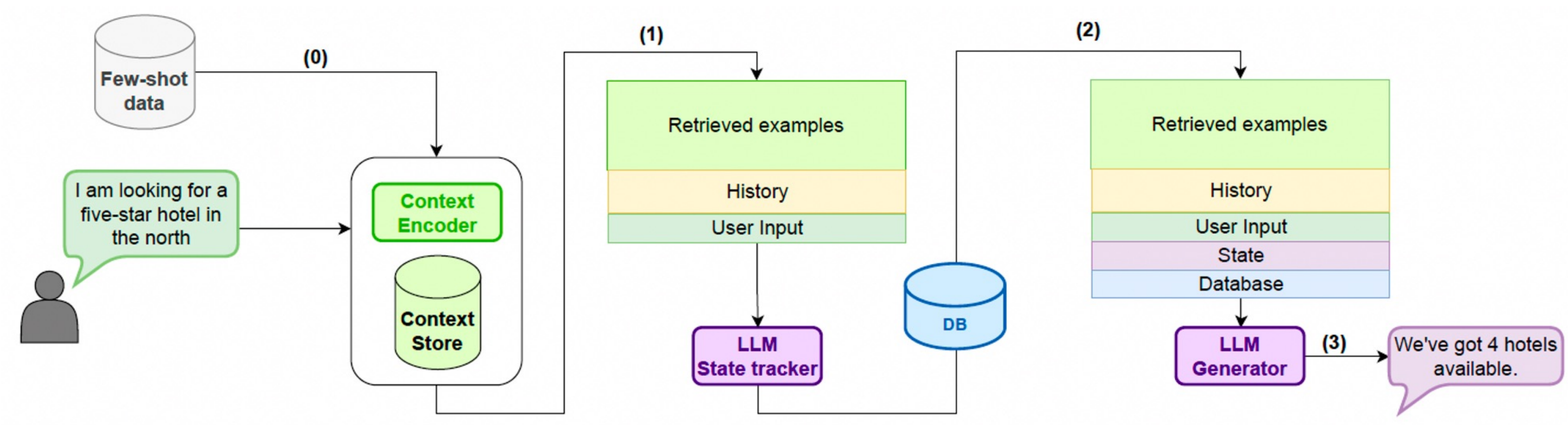
Problem States

- **Limitation of Instruction-tuned LLMs**
  - Vojtech Hudecek and Ondrej Dusek. "Are LLMs All You Need for Task-Oriented Dialogue?" arXiv: 2304.06556v1 (2023) **#TOD #Prompt #DST #Retrieval**

model	few shot	oracle BS	Schema Guided Dialogues BLEU	JGA	Slot-F1	Success	MultiWOZ 2.2 BLEU	JGA	Slot-F1	Success
supervised SotA	✗	✗	29.90	0.30	0.60	–	19.90	0.58	–	0.80
Alpaca-LoRA-7B-zs-gbs	✗	✗	2.66	0.06	0.05	0.08	1.47	0.11	0.11	0.05
Tk-Instruct-11B-zs-gbs	✗	✗	4.01	0.04	0.02	0.07	2.27	0.03	0.02	0.12
GPT-NeoXT-20B-zs-gbs	✗	✗	0.29	0.03	0.01	0.07	0.23	0.05	0.06	0.06
OPT-IML-30B-zs-gbs	✗	✗	1.85	0.04	0.01	0.07	0.44	0.08	0.16	0.03
ChatGPT-zs-gbs	✗	✗	–	–	–	–	2.99	0.24	0.37	0.15
Alpaca-LoRA-7B-zs-obs	✗	✓	2.68	–	–	0.23	1.64	–	–	0.09
Tk-Instruct-11B-zs-obs	✗	✓	4.95	–	–	0.24	2.62	–	–	0.24
GPT-NeoXT-20B-zs-obs	✗	✓	0.61	–	–	0.22	0.26	–	–	0.05
OPT-IML-30B-zs-obs	✗	✓	1.90	–	–	0.23	0.29	–	–	0.06
ChatGPT-zs-obs	✗	✓	–	–	–	–	3.58	–	–	0.34
Alpaca-LoRA-7B-fs-gbs	✓	✗	4.92	0.04	0.02	0.08	3.83	0.08	0.07	0.08
Tk-Instruct-11B-fs-gbs	✓	✗	8.17	0.04	0.02	0.07	6.24	0.05	0.04	0.05
GPT-NeoXT-20B-fs-gbs	✓	✗	2.83	0.07	0.07	0.08	2.59	0.11	0.18	0.08
OPT-IML-30B-fs-gbs	✓	✗	0.82	0.06	0.06	0.07	3.90	0.06	0.11	0.05
ChatGPT-fs-gbs	✓	✗	–	–	–	–	7.17	0.21	0.36	0.20
Alpaca-LoRA-7B-fs-obs	✓	✓	5.09	–	–	0.23	4.52	–	–	0.30
Tk-Instruct-11B-fs-obs	✓	✓	8.87	–	–	0.24	7.38	–	–	0.36
GPT-NeoXT-20B-fs-obs	✓	✓	3.06	–	–	0.23	2.83	–	–	0.23
OPT-IML-30B-fs-obs	✓	✓	0.56	–	–	0.22	4.16	–	–	0.24
ChatGPT-fs-obs	✓	✓	–	–	–	–	8.06	–	–	**0.58**

Problem States

- **LLM-based ToD**
  - Vojtech Hudecek and Ondrej Dusek. “Are LLMs All You Need for Task-Oriented Dialogue?” arXiv: 2304.06556v1 (2023) **#ToD #Prompt #DST #Retrieval**



(0) Encode dialogues for using few-shot examples

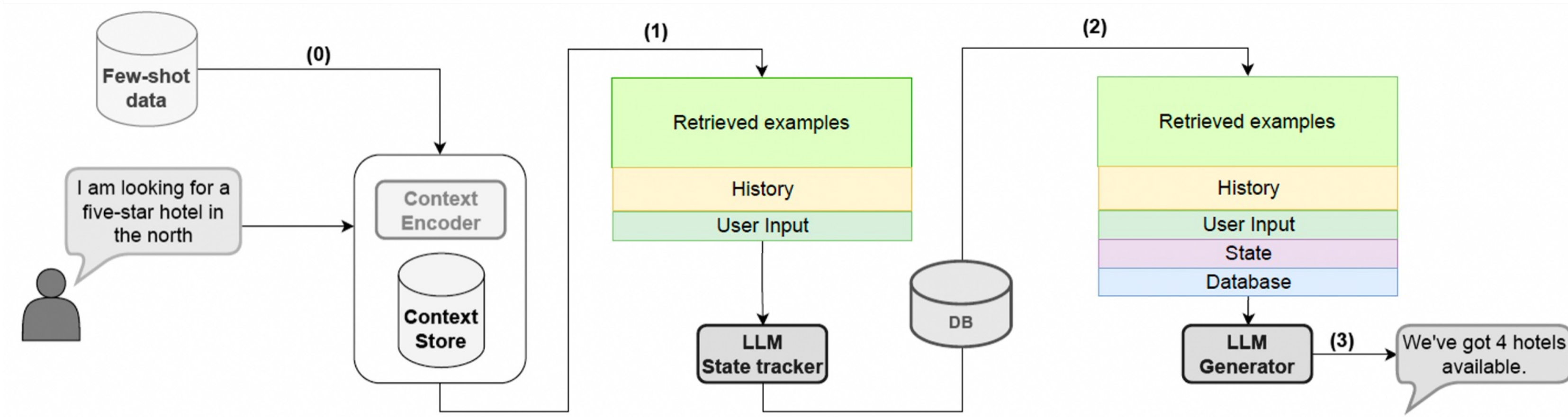
(1) Retrieve the relevant examples w/user query & construct initial prompt

(2) Track dialogue state by calling the instruction-tuned LLM
→ DB query results + DST + current dialogue context + user query

(3) Generate final response

Problem States

- **Improve LLM-based ToD**
  - Vojtech Hudecek and Ondrej Dusek. “Are LLMs All You Need for Task-Oriented Dialogue?” arXiv: 2304.06556v1 (2023) **#ToD #Prompt #DST #Retrieval**

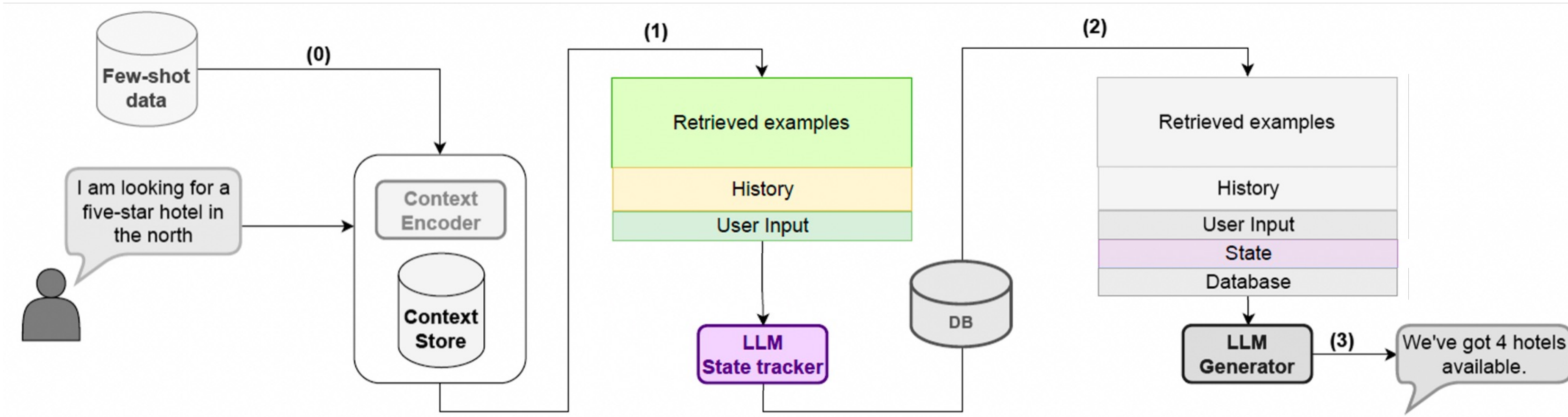


1. Prompt Engineering
   - More specific instruction
   - Set a role of the model
2. Adapt better DST approach
   - Retrieve better examples for Few-shot DST
   - Make LLMs to interpret the structured manner
   - Use FFT SOTA DST model

Prompt	Definition: Capture values from a conversation about hotels in JSON. Values that should be captured are: - "price": the price of the hotel ... [history] Customer: "I want a cheap place to stay." Output:
Output:	{"price": "cheap"}

Problem States

- **Improve LLM-based ToD**
  - Vojtech Hudecek and Ondrej Dusek. "Are LLMs All You Need for Task-Oriented Dialogue?" arXiv: 2304.06556v1 (2023) **#ToD #Prompt #DST #Retrieval**

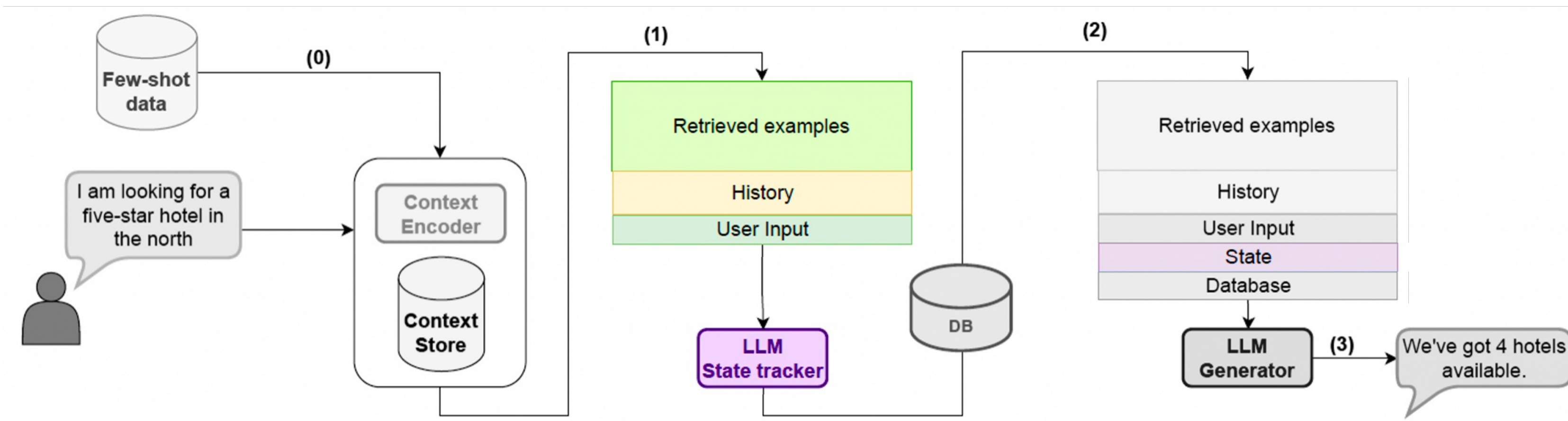


1. Prompt Engineering
   - More specific instruction
   - Set a role of the model
2. Adapt better DST approach
   - Retrieve better examples for Few-shot DST
   - Make LLMs to interpret the structured manner
   - Use FFT SOTA DST model

Prompt	Definition: Capture values from a conversation about hotels in JSON. Values that should be captured are: - "price": the price of the hotel ... [history] Customer: "I want a cheap place to stay." Output:
Output:	{"price": "cheap"}

Problem States

- **Improve LLM-based ToD**
  - Vojtech Hudecek and Ondrej Dusek. "Are LLMs All You Need for Task-Oriented Dialogue?" arXiv: 2304.06556v1 (2023) **#ToD #Prompt #DST #Retrieval**



1. Prompt Engineering
   - More specific instruction
   - Set a role of the model
2. Adapt better DST approach
   - Retrieve better examples for Few-shot DST
   - Make LLMs to interpret the structured manner
   - Use FFT SOTA DST model

Prompt	Definition: Capture values from a conversation about hotels in JSON. Values that should be captured are: - "price": the price of the hotel ... [history] Customer: "I want a cheap place to stay." Output:
Output:	{"price": "cheap"}

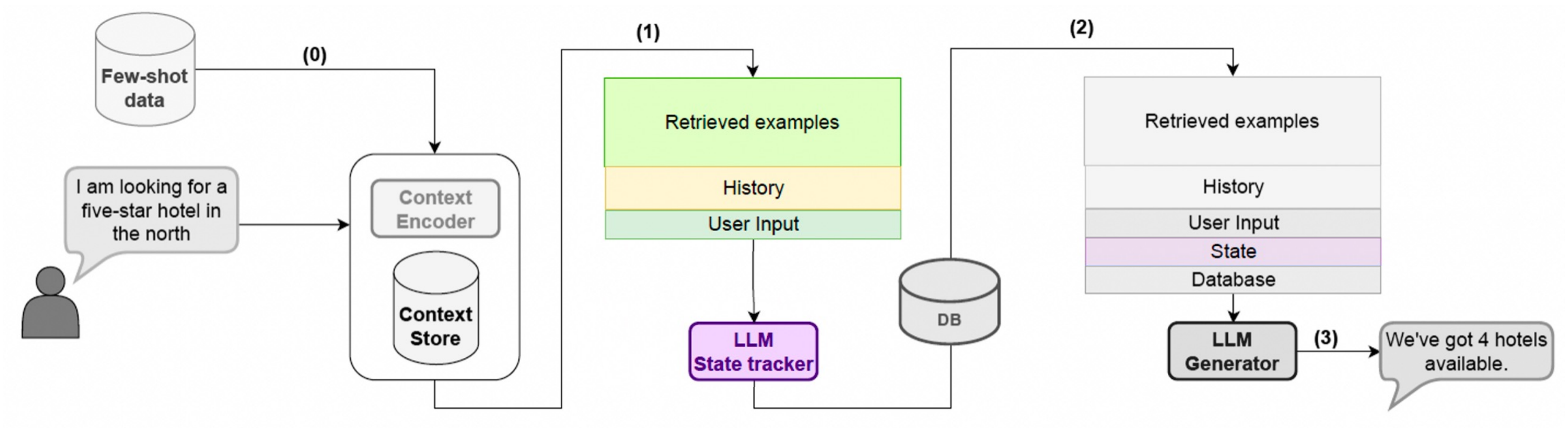
3. Suggestions

- RQ1) Does using good examples improve few-shot DST performance?
- RQ2) What is a good dialogue example?
- RQ3) Can we expect better performance using relevant dialog examples in other tasks?

Suggestions

RQ1) Does using good examples improve few-shot DST performance?

Vojtech Hudecek and Ondrej Dusek. “Are LLMs All You Need for Task-Oriented Dialogue?” arXiv: 2304.06556v1 (2023) **#TOD #Prompt #DST #Retrieval**



1. **Select Dialogues to reproduce the experiment from LLM-based ToD paper**
2. **Complete the few-shot prompt given a proper or random dialogue example**
3. **Compare the performance of few-shot DST**
4. Evaluate the benchmarks on ToD or multi-turn Dialogue and verify robustness and performance improvements (may take into OOD issues)

Suggestions :: Toy Experiments

RQ1) Does using good examples improve few-shot DST performance?

1. **Select Dialogues to reproduce the experiment from LLM-based TOD paper**

Vojtech Hudecek and Ondrej Dusek. Are LLMs All You Need for Task-Oriented Dialogue? arXiv: 2304.06556v1 (2023) **#TOD #Prompt #DST #Retrieval**

- Experiment details are not publicly available.
- Sample of zero-shot prompt

Task Def.
Domain Info.
Dialogue History
Query

```
Definition: Capture values from conversation about hotels in JSON.
Values that should be captured are:
 - "price": the price of the hotel
 - "area": area where the hotel is located
 - "internet": if the hotel has wifi internet. Capture yes/no
 - "parking": if the hotel has free parking. Capture yes/no
Customer: "I want a cheap place to stay.”
Assistant: "Sure, what part of town do you prefer?”
Output: I need it to be in the city centre
```

> It said it had trouble updating the entire BST,
so it only generate the slots to update. (The example prompt didn't show that)

- Output / Updated State

```
{"area": "center"}
```

```
hotel {"price": "cheap", "area": "center"}
```

Suggestions :: Toy Experiments

RQ1) Does using good examples improve few-shot DST performance?

1. **Select Dialogues to reproduce the experiment from LLM-based TOD paper**
   1) Select Dialogue to test: MultiWOZ 2.2 test.json; MUL0484.json

```
Customer: "Hello, I am looking for a restaurant in Cambridge. I believe it is called Golden Wok."
Assistant: "It is located at 191 Histon Road Chesterton"
Customer: "Can you book me a table for 11:00 on Friday?"
Assistant: "Yes I can! Table for 1?"
Customer: "Actually, for 4, please."
Assistant: "Okay, your booking was successful! The reference number is MUFCMYFF . The table will be
  reserved for 15 minutes."
Customer: "Great, can you also get me information or architecture in the area"
Assistant: "Sure. There are several churches and an old schools attraction, all in the centre area. Do
  you have a preference?"
Customer: "What do you recommend?"
Assistant: "old schools is lovely, they are on trinity lane and free admission"
Customer: "Can I get the postcode for that? I also need to book a taxi to the Golden Wok."
Assistant: "The postcode is cb21tt. Are you looking for a taxi from Old Schools to the Golden Wok?"
Customer: "Yes I do. I'd like to make sure I arrive at the restaurant by the booked time. Can you
  check?"
Assistant: "What time do you want to leave?"
Customer: "Actually all you have to do is set the taxi so it arrives by the arrived time. Am I better
  off booking it myself?"
Assistant: "I have booked you a taxi for Friday to arrive at Old Schools at 10:45. Is there anything
  else I may help you with?"
Customer: "I need the contact number and car type also."
Assistant: "it is a Lexus white in colour and contact number is 07381884388"
Customer: "Thanks for the service, good day."
Assistant: "You're welcome! Have a great day!"
```

Restaurant

Attraction

Taxi

Suggestions :: Toy Experiments

RQ1) Does using good examples improve few-shot DST performance?

1. **Select Dialogues to reproduce the experiment from LLM-based TOD paper**
   1) Select Dialogue to test: MultiWOZ 2.2 test.json; MUL0484.json
   2) Select Dialogue to use as a proper example: same all domains, same slot to update

```
Customer: "What is the address for Holy Trinity Church?"
Assistant: "The address for holy trinity church is market street postcode cb23nz and the phone number
  is 01223355397."
Customer: "thank you. i wish to get entrance fee"
Assistant: "Holy trinity church is free. May I help you with anything else today?"
```

Attraction

```
Customer: "Yes, I am looking for an Indian restaurant in the east."
Assistant: "The four choices range from moderate to expensive in their price range. Do you have a
  preference?"
Customer: "How about an expensive one. What are my choices in that range?"
Assistant: "Pipasha Restaurant and Sitar Tandoori are the 2 expensive Indian restaurants in the area.
  Do you want the information on those?"
Customer: "i want o book for 7 people at 18:00 on monday"
Assistant: "I booked you for pipasha restaurant. The table will be reserved for 15 minutes. Reference
  number is : 3W8KKLOJ . Can i help you with anything else?"
```

Restaurant

```
Customer: "Yes, one more thing. I will need a taxi from Holy Trinity Church to pipasha restaurant."
Assistant: "Sure! And what time do you prefer?"
Customer: "I need to arrive by the booked time at the restaurant."
Assistant: "For how many people?"
Customer: "There will be 7 people travelling."
Assistant: "A grey volvo will pick you up at Holy Trinity Church for your arrival at Pipasha by 18:00.
  Should you need to contact the driver, the number is: "07746897279"
Customer: "Thank you. That is all I need."
Assistant: "Thank you for using the Cambridge towneinfo centre. Goodbye"
```

Taxi

Suggestions :: Toy Experiments

RQ1) Does using good examples improve few-shot DST performance?

1. **Select Dialogues to reproduce the experiment from LLM-based TOD paper**
   1) Select Dialogue to test: MultiWOZ 2.2 test.json; MUL0484.json
   2) Select Dialogue to use as a proper example: same all domains, same slot to update
   3) Select Dialogue to use as a random example: same some domains, not same slot

```
Customer: "Hello. I'm going out for an expensive dinner and need to find a restaurant. I'd like
  somewhere in the north, and I'm willing to pay quite a bit of money."
Assistant: "I'd be happy to help you with that. I have 5 expensive restaurants in the north. Did you
  have a type of cuisine in mind?"
                                   ... Skip the middle ...                          Restaurant
Customer: "How about at 16:45?"
Assistant: "Booking was successful. The table will be reserved for 15 minutes. The reference number is:
  "0PK0TW8F"
Customer: "I'm also looking for a place to stay that offers both internet and free parking."
Assistant: "There are 29 places in Cambridge which offer both, any other preferences?"
Customer: "Hmm, yes. Someplace expensive."
Assistant: "I can recommend the University Arms hotel in the centre of the city."   Hotel
                                   ... Skip the middle ...
Assistant: "Alright! I was able to book your room for 1 night at the University Arms Hotel for 3 people.
  Your reference number is F13028XH . Can I help with anything else?"
Customer: "I'm going to need a taxi as well from the hotel to the restaurant that arrives at 16:45.
  Could I get the contact number and car type for that?”
Assistant: "Sure! The contact number is 07648846859 and the car is a black audi. Is there anything else
  you need help with?”                                                               Taxi
Customer: "No, you have been most helpful. Have a nice day."
Assistant: "Thanks, have a nice day as well.”
```

Suggestions :: Toy Experiments

RQ1) Does using good examples improve few-shot DST performance?

2. **Complete the few-shot prompt given a proper or random dialogue example**

Task Def.

```
Definition: Capture values from conversation about restaurant and taxi in JSON.
Values that should be captured are (some may not need to be captured):
- "r-name": the name of the restaurant
- "r-bookpeople": number of people booked in the restaurant
- "r-bookday": the day of the week that the restaurant is booked
- "r-booktime": the time that the restaurant is booked. Format is 00:00.
- "t-departure": where the taxi will depart from
- "t-destination": where the taxi will arrive
- "t-arriveby": the time that the taxi is arrived at the destination
```

Example 1

```
example conversation:
Customer: "Yes, one more thing. I will need a taxi from Holy Trinity Church to pipasha restaurant."
Assistant: "Sure! And what time do you prefer?"

correct answer:
{"t-departure": "holy trinity church"}, {"t-destination":"pipasha restaurant"}
```

Example 2

```
example conversation:
Customer: "I want to depart the restaurant at 05:00 to go to the attraction"
Assistant: "A yellow Tesla will pick you up, and the contact number is 07267272725.
         Can I help with anything else today?"

wrong answer:
{"h-internet": "yes"}, {"t-destination": "05:00"}
```

Suggestions :: Toy Experiments

RQ1) Does using good examples improve few-shot DST performance?

2. **Complete the few-shot prompt given a proper or random dialogue example**

```
Definition: Capture values from conversation about restaurant and taxi in JSON.
Values that should be captured are (some may not need to be captured):
- “r-name”: the name of the restaurant
- “r-bookpeople”: number of people booked in the restaurant
- “r-bookday”: the day of the week that the restaurant is booked
- “r-booktime“: the time that the restaurant is booked. Format is 00:00.
- “t-departure”: where the taxi will depart from
- “t-destination”: where the taxi will arrive
- “t-arriveby”: the time that the taxi is arrived at the destination

example conversation:
Customer: "Hmm, yes. Someplace expensive."
Assistant: "I can recommend the University Arms hotel in the centre of the city."

correct answer:
{"h-pricerange": "expensive"}

example conversation:
Customer: "That is all. Thank you for your help."
Assistant: "Thank you for using Cambridge TownInfo centre!
         Have a good day!"

wrong answer:
{"h-internet": "yes"}, {"t-destination": "05:00"}
```

Task Def.

Example 1

Example 2

Suggestions :: Toy Experiments

RQ1) Does using good examples improve few-shot DST performance?

2. **Complete the few-shot prompt given a proper or random dialogue example**

```
Definition:                                                         Task Def.

example conversation:                                               Example 1
correct answer:

example conversation:                                               Example 2
wrong answer:

the conversation:                                                   History
Customer: "Hello, I am looking for a restaurant in Cambridge. I believe it is called Golden Wok."
Assistant: "It is located at 191 Histon Road Chesterton"
Customer: "Can you book me a table for 11:00 on Friday?"
Assistant: "Yes I can! Table for 1?"
Customer: "Actually, for 4, please."
Assistant: "Okay, your booking was successful! The reference number is MUFCMYFF . The table will be
reserved for 15 minutes."
Customer: "Great, can you also get me information or architecture in the area"
Assistant: "Sure. There are several churches and an old schools attraction, all in the centre area.
Do you have a preference?"
Customer: "What do you recommend?"
Assistant: "old schools is lovely, they are on trinity lane and free admission"

Output: "Can I get the post code for that? I also need to book a taxi to the Golden Wok."   Query
```

Suggestions :: Toy Experiments

RQ1) Does using good examples improve few-shot DST performance?

3. Compare the performance of few-shot DST

- LLM: ChatGPT (gpt-3.5-turbo)
  - Model applied to ChatGPT currently in service
  - Equal in completeness to the davinci model,
    but more efficient in terms of price (1/10 of the davinci)
  - Optimized for conversation and text generation
  - Cannot be fine-tuned

Dialogue History

```
... Skip the previous ...
Assistant: “old schools is lovely, they are on trinity lane and free admission”
Output: “Can I get the post code for that? I also need to book a taxi to the Golden Wok.”
```

Proper

{ "r-name": "Golden Wok", "r-bookpeople": 4, "r-bookday": "Friday", "r-booktime": "11:00", "t-destination": "Golden Wok" }

Random

{ "r-name": "Golden Wok", "r-bookpeople": 4, "r-bookday": "Friday", "r-booktime": "11:00", "t-destination": "191 Histon Road Chesterton" }

Suggestions

RQ2) What is a good dialogue example?

- **Initial Opinion**

1. Query/passage design
   - 1 or 2 last utterances: (Customer +) Assistant
   - Entire dialogue embedding vs. turn embedding vs. hybrid
2. Using Label Semantics
   - Last utterances + BST
   - All dialogue + BST
3. Similarity Score
   1) Vanilla similarity

   $$\mathrm{sim}(q,p) = E_Q(q)^\top E_P(p)$$

   2) Utterance-wise similarity

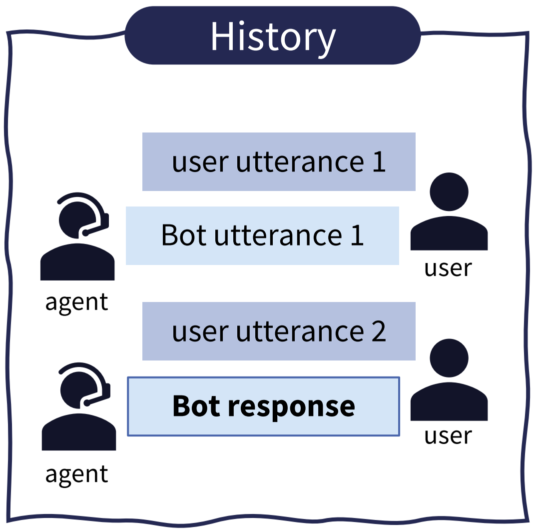
   $$\mathrm{sim}(q,p) = \lambda\, \mathrm{sim}\big(utt_{q(n-1)}, utt_{p(i)}\big) + (1-\lambda)\, \mathrm{sim}\big(utt_{q(n)}, utt_{p(j)}\big)$$

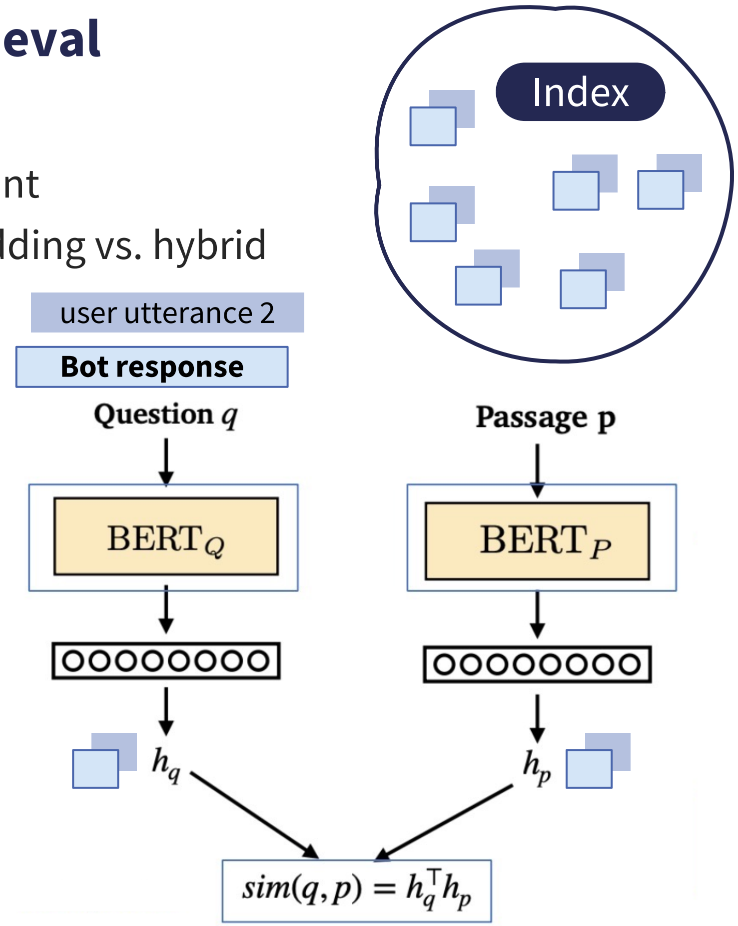
   3) Weighted similarity

Suggestions

RQ2) What is a good dialogue example?

- **Initial Opinion: Dialogue-specific Retrieval**
  1. Query/passage design
     - 1 or 2 last utterances: (Customer +) Assistant
     - Entire dialogue embedding vs. turn embedding vs. hybrid

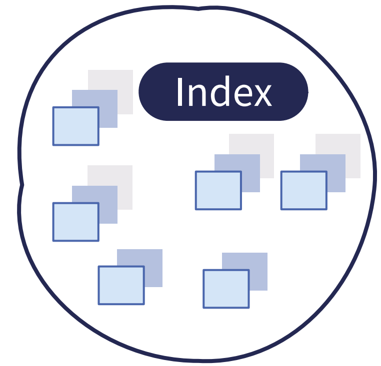


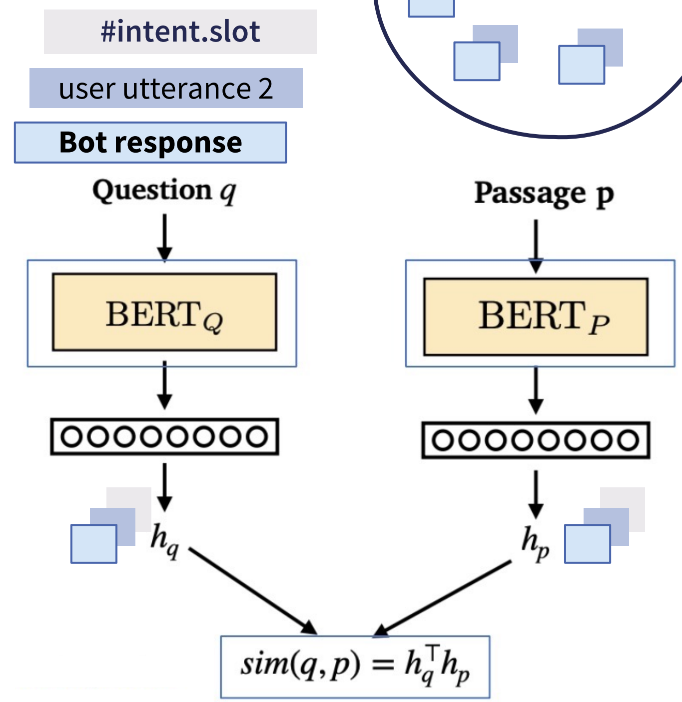


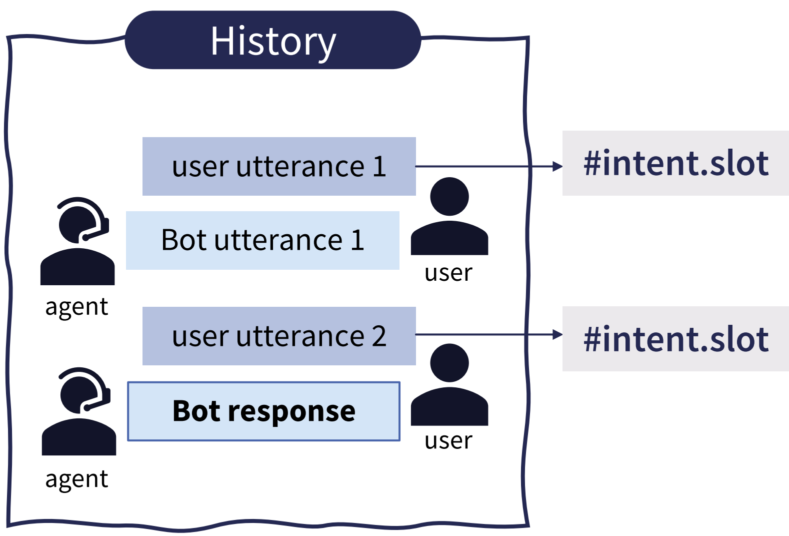
Suggestions

RQ2) What is a good dialogue example?

- **Initial Opinion: Dialogue-specific Retrieval**
  2. Using Label Semantics
     - Last utterances + BST
     - All dialogue + BST







Suggestions

RQ2) What is a good dialogue example?

- **Initial Opinion: Dialogue-specific Retrieval**
  - 3. Similarity Score
    - 1) Vanilla similarity

$$\mathrm{sim}(q,p) = E_Q(q)^\top E_P(p)$$

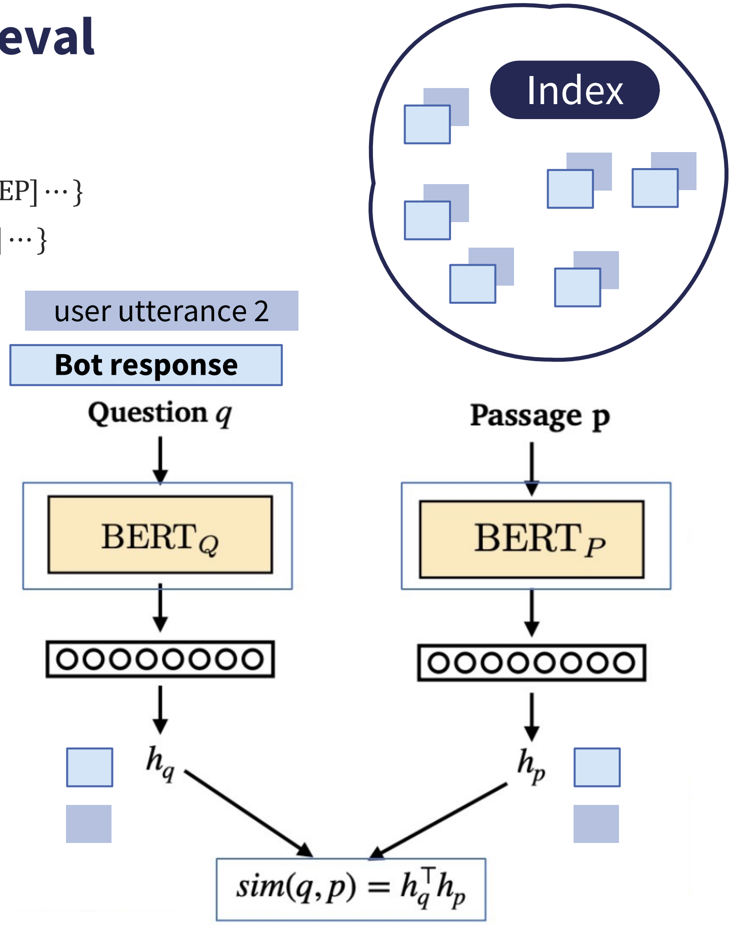
* q: $\{utt_{q(n-1)}$ [SEP] $utt_{q(n)}$ [SEP] $\cdots\}$

* p: $\{utt_{p(i)}$ [SEP] $utt_{p(j)}$ [SEP] $\cdots\}$

    - 2) Utterance-wise similarity

$$\begin{aligned}\mathrm{sim}(q,p) &= \lambda\, \mathrm{sim}\big(utt_{q(n-1)}, utt_{p(i)}\big) \\ &\quad +(1-\lambda)\, \mathrm{sim}\big(utt_{q(n)}, utt_{p(j)}\big) \\ &= \lambda\, E_Q\big(utt_{q(n-1)}\big)^\top E_P\big(utt_{p(i)}\big) \\ &\quad +(1-\lambda)\, E_Q\big(utt_{q(n)}\big)^\top E_P\big(utt_{p(j)}\big)\end{aligned}$$

    - 3) Weighted similarity
      - Domain
      - Intent / slot (value)
      - Speaker



Suggestions

- **RQ3)** Can we expect better performance
  using relevant dialog examples in other tasks?

- **Various Tasks**
  - Dialogue Response Selection
  - DST
    - Intent Detection
    - Dialogue Acts Detection
    - Slot-Filling

- **Various Domain**
  - Chit-chat
  - Knowledge-grounded Dialogue (FAQ)

4. Related Works

- BERT-FP
- DialogueCSE
- Dialog-BERT (**이루다**), ToD-BERT

Related Works

Han, Hong, et al. "Fine-grained Post-training for Improving Retrieval-based Dialogue Systems." NAACL-HLT 2021, 1549-1558 (2021). *** NAACL 2021, LG AI Research, SGU, SKKU**

- **BERT-FP**

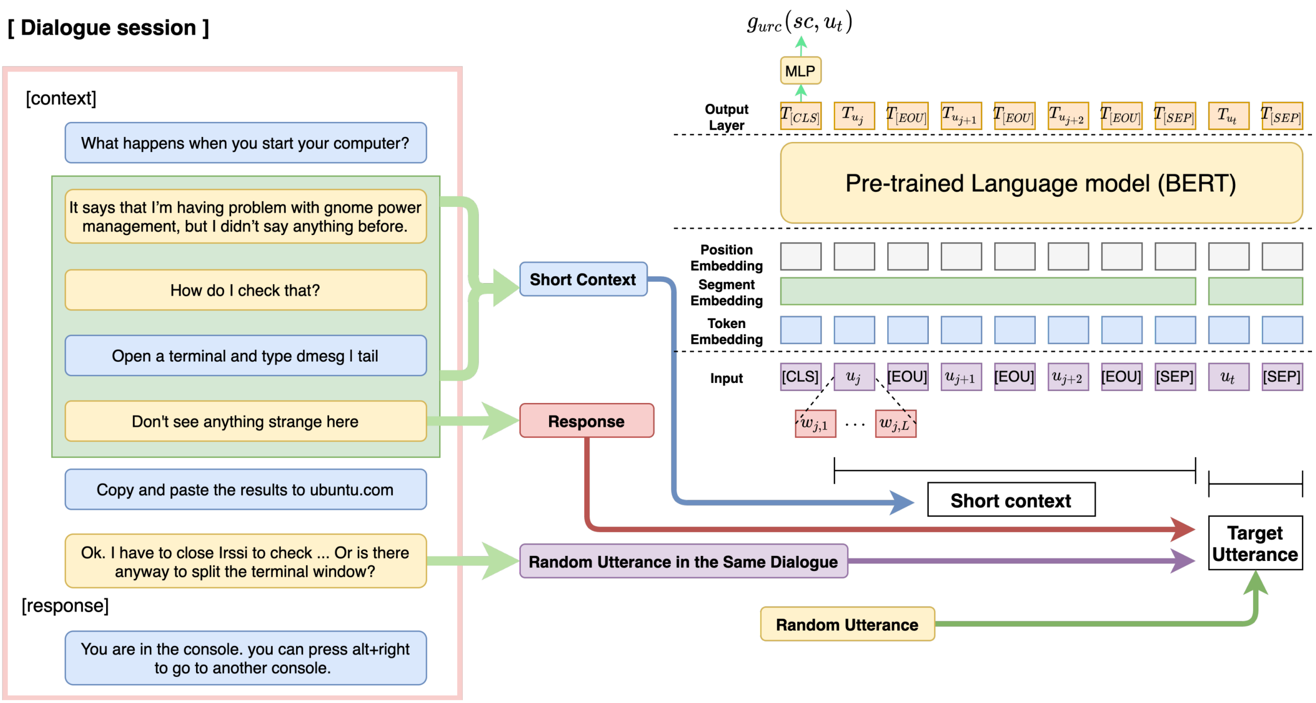
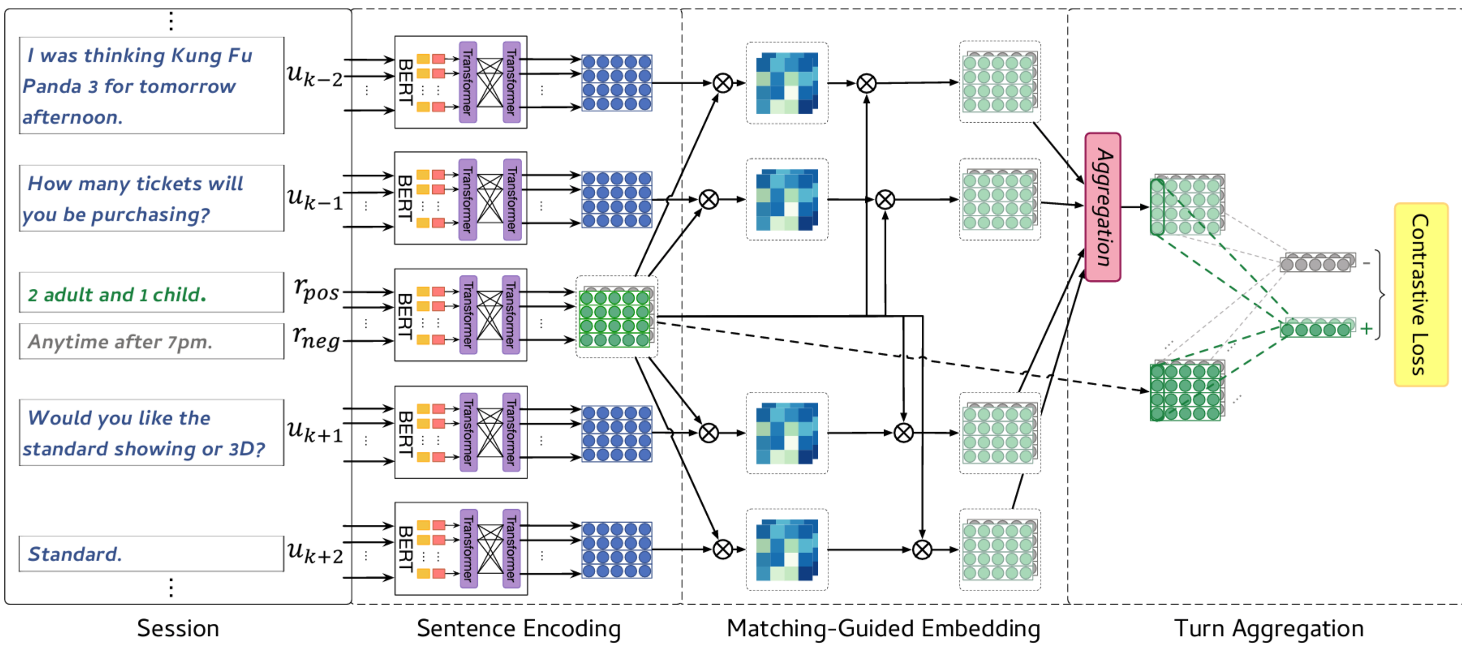


Figure 1: Architecture of fine-grained post-training. The short context length k is three.

Related Works

Liu, Che, et al. (Alibaba Group) "DialogueCSE: Dialogue-based contrastive learning of sentence embeddings." EMNLP 2021.

- **DialogueCSE**
  - DialogueCSE's positive pair
    - Context-free Embedding: Response Embedding (Independent of context)
    - Context-aware Embedding: Context + Response Embedding

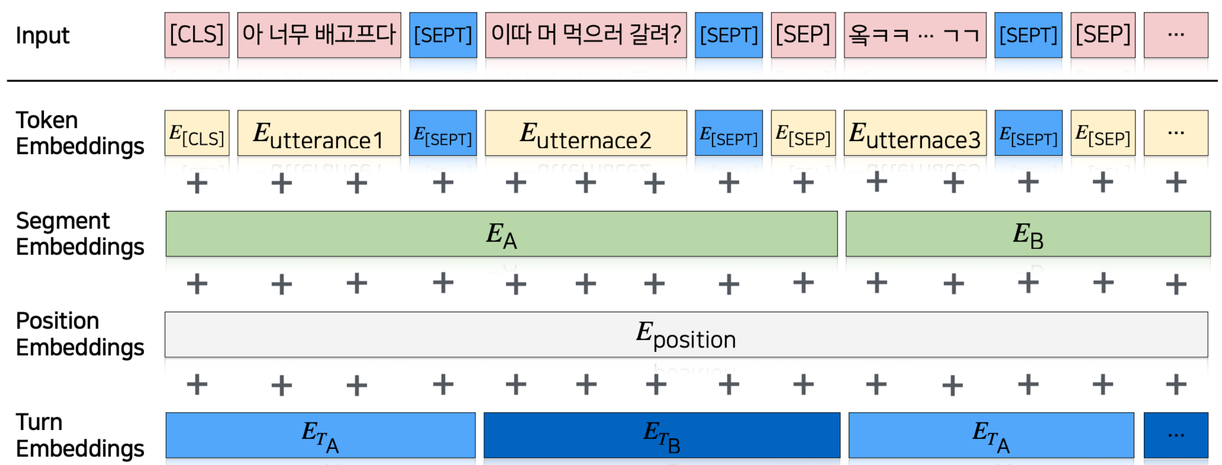


Related Works

SCATTERLABS, Dialog-BERT: 100억 건의 메신저 대화로 일상대화 인공지능 서비스하기, DEVIEW 2019.

Wu, Hoi, et al. (Salesforce) “ToD-BERT: Pre-trained Natural Language Understanding for Task-Oriented Dialogue" EMNLP 2020.

- **Dialog-BERT**
  - Pre-training strategies: Turn-separate Token & Turn embedding
  - Fine-tuning strategies



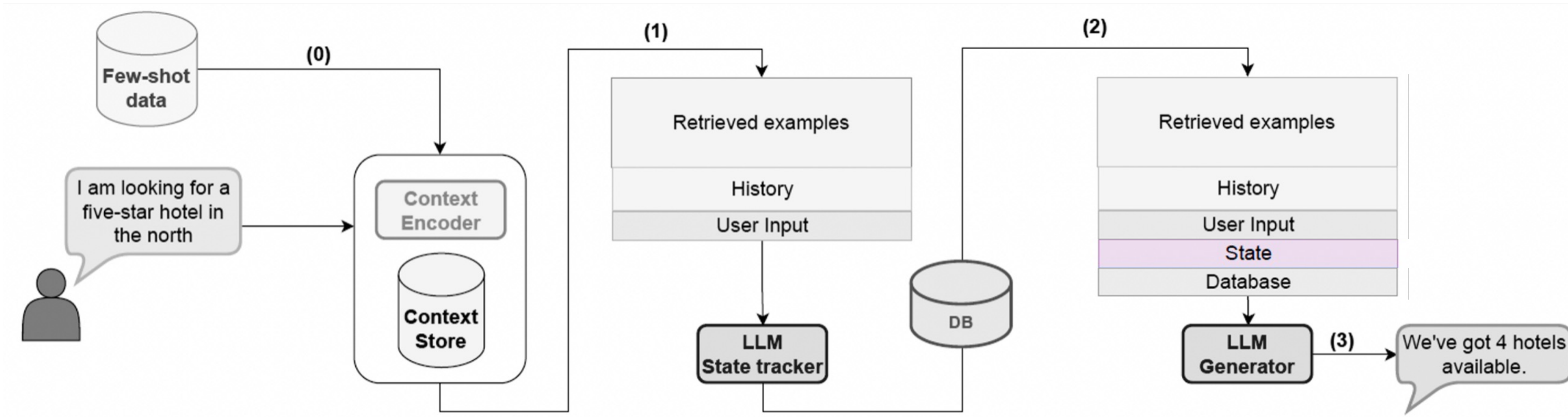
5. Experiments

- Backbone: BERT, T5, ToD-BERT, …
- Datasets: MultiWOZ, SGD, …

6. Future Works

Future Works

- **Improve LLM-based ToD**
  - Vojtech Hudecek and Ondrej Dusek. "Are LLMs All You Need for Task-Oriented Dialogue?" arXiv: 2304.06556v1 (2023) **#ToD #Prompt #DST #Retrieval**



1. Prompt Engineering
   - More specific instruction
   - Set a role of the model
2. Adapt better DST approach
   - Retrieve better examples for Few-shot DST
   - Make LLMs to interpret the structured manner
   - Use FFT SOTA DST model

Prompt	Definition: Capture values from a conversation about hotels in JSON. Values that should be captured are: - "price": the price of the hotel ... [history] Customer: "I want a cheap place to stay." Output:
Output:	{"price": "cheap"}

Conclusion

- Instruction-tuned LLMs are poor at ToD,
  because they have poor DST performance.

- To improve performance,
  we need to feed it appropriate examples.

- Suggest a Dialogue Retriever to find
  the proper dialog context for in-context learning

{ End Page }

Thank you :D

Yejin Yoon

HYU NLP Lab.
Dept. of Artificial Intelligence Application,
Hanyang University

stillwithyou@hanyang.ac.kr