Towards scalable multi-domain conversational agents

Xiaoxue Zang, Abhinav Rastogi, Srinivas Sunkara, Raghav Gupta, Pranav Khaitan

Google Research
The 4th International Workshop on Search-Oriented Conversational AI
Overview

- Introduction
- The Schema-guided approach
- The Schema-guided dialogue dataset
- Data collection methodology
- Dialogue state tracking challenge (DSTC 8)
Overview

- Introduction
- The Schema-guided approach
- The Schema-guided dialogue dataset
- Data collection methodology
- Dialogue state tracking challenge (DSTC 8)
Book a table at Cascal for 6 pm today.

Sure. For how many people?

```
get_availability(
    restaurant="Cascal",
    time = "18:00:00",
    date = "2018-11-28",
)

{
    "restaurant" : "Cascal",
    "time" : "18:00:00",
    "date" : "2018-11-28",
    "seats" : 5
}
```
Book a table at Cascal for 6 pm today.

Sure. For how many people?

Intent: RESERVE_RESTAURANT
restaurant: "Cascal"
time: "6 pm"
date: "today"

[ AFFIRM, REQUEST(SEATS), ]
Challenges from Real World APIs

- The number of APIs to support is large
- No uniform schema or entity names
  - Slot names
    - "origin" = "from", "departing from" = "where from", ....
  - Slot values
    - "San Francisco" = "SFO" = "SF", ....
- Don't allow arbitrary calls
  - find_restaurant(city="San Francisco", date="today") ✓
  - find_restaurant(city, date="today") ✗
- The set of supported entities may be large and dynamic
  - May not expose the list of all available entities
  - Robustness to OOV problem
Requirements of Virtual Assistants

- Facilitate dialogues across multiple Services/APIs
  - Handle large universe of services spanning multiple domains
  - Data efficient: Zero-shot or few-shot learning for tail services

<table>
<thead>
<tr>
<th>Domain</th>
<th>Hotel</th>
<th>Movie</th>
<th>Flight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Services</td>
<td><img src="Hotels.com" alt="" /></td>
<td><img src="FANDANGO" alt="" /></td>
<td><img src="Expedia" alt="" /></td>
</tr>
<tr>
<td></td>
<td>Booking.com</td>
<td></td>
<td>skyscanner</td>
</tr>
<tr>
<td></td>
<td>IMDb</td>
<td></td>
<td>Google</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Expedia</td>
</tr>
</tbody>
</table>
Requirements of Virtual Assistants

- Can handle multi-domain conversations
  - Seamlessly switching domains
  - Carrying over relevant slots among APIs

Book a table at Cascal at 6 pm today and find a taxi to go there.

```python
get_availability(
    restaurant="Cascal",
    time = "18:00:00",
    date = "2018-11-28",
)

get_cab(
    destination="Cascal",
)
```
Requirements of Virtual Assistants

- Robust to changes in schema
  - No retraining if the interface of an API changes
  - Robust to new/unseen slot values

Free-form input
## Existing Dialogue Datasets

<table>
<thead>
<tr>
<th>Metric</th>
<th>DSTC2</th>
<th>WOZ2.0</th>
<th>FRAMES</th>
<th>M2M</th>
<th>MultiWOZ</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Domains</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>7</td>
</tr>
<tr>
<td>#Dialogues</td>
<td>1,612</td>
<td>600</td>
<td>1,369</td>
<td>1,500</td>
<td>8,438</td>
</tr>
<tr>
<td>#Turns</td>
<td>23,354</td>
<td>4,472</td>
<td>19,986</td>
<td>14,796</td>
<td>113,556</td>
</tr>
<tr>
<td>#Slots</td>
<td>8</td>
<td>4</td>
<td>61</td>
<td>13</td>
<td>24</td>
</tr>
<tr>
<td>#Values</td>
<td>212</td>
<td>99</td>
<td>3,871</td>
<td>138</td>
<td>4,510</td>
</tr>
</tbody>
</table>

- Datasets not large enough for training generic virtual assistants
Existing Dialogue Datasets

<table>
<thead>
<tr>
<th>Metric</th>
<th>DSTC2</th>
<th>WOZ2.0</th>
<th>FRAMES</th>
<th>M2M</th>
<th>MultiWOZ</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Domains</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>7</td>
</tr>
<tr>
<td>#Dialogues</td>
<td>1,612</td>
<td>600</td>
<td>1,369</td>
<td>1,500</td>
<td>8,438</td>
</tr>
<tr>
<td>#Turns</td>
<td>23,354</td>
<td>4,472</td>
<td>19,986</td>
<td>14,796</td>
<td>113,556</td>
</tr>
<tr>
<td>#Slots</td>
<td>8</td>
<td>4</td>
<td>61</td>
<td>13</td>
<td>24</td>
</tr>
<tr>
<td>#Values</td>
<td>212</td>
<td>99</td>
<td>3,871</td>
<td>138</td>
<td>4,510</td>
</tr>
</tbody>
</table>

- Datasets not large enough for training generic virtual assistants
- Very few slots and possible values
### Existing Dialogue Datasets

<table>
<thead>
<tr>
<th>Metric</th>
<th>DSTC2</th>
<th>WOZ2.0</th>
<th>FRAMES</th>
<th>M2M</th>
<th>MultiWOZ</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Domains</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>7</td>
</tr>
<tr>
<td>#Dialogues</td>
<td>1,612</td>
<td>600</td>
<td>1,369</td>
<td>1,500</td>
<td>8,438</td>
</tr>
<tr>
<td>#Turns</td>
<td>23,354</td>
<td>4,472</td>
<td>19,986</td>
<td>14,796</td>
<td>113,556</td>
</tr>
<tr>
<td>#Slots</td>
<td>8</td>
<td>4</td>
<td>61</td>
<td>13</td>
<td>24</td>
</tr>
<tr>
<td>#Values</td>
<td>212</td>
<td>99</td>
<td>3,871</td>
<td>138</td>
<td>4,510</td>
</tr>
</tbody>
</table>

- Datasets not large enough for training generic virtual assistants
- Very few domains, slots and possible values
- Make simplified assumptions on underlying APIs/Services

Existing dialogue datasets do not expose all the challenges
Overview

- Introduction
- The Schema-guided approach
- The Schema-guided dialogue dataset
- Data collection methodology
- Dialogue state tracking challenge (DSTC 8)
The Principal Idea

Problems
Assuming the set of slots and intents to be handled by the dialogue system are fixed

- Slots and intents are treated as merely labels
- Cannot handle unseen/new intents or slots

Solution
Associate a semantic representation with each label using their natural language description

- Predictions are made over dynamic sets of labels using embedded representations
- Unseen/new slots and intents can be handled
The Schema-guided Approach

Service

- service_name: "RideSharing"
  description: "On-demand taxi calling service"

Slots

- name: "destination"
  description: "Destination for taxi ride"
- name: "number_of_riders"
  description: "Number of riders to call taxi for"
- name: "shared_ride"
  description: "Whether ride is shared or not"
- name: "ride_fare"
  description: "Total cost of the ride"

Intents

- name: "GetRide"
  description: "Call a taxi to head to a given destination"
  required_slots: ["destination", "number_of_riders"]
  optional_slots: ["shared_ride" = "False"]
The Schema-guided Approach

Descriptions can be used to semantically identify related services and slots

Service

- service_name: "RideSharing_1"
  description: "On-demand taxi calling service"

- service_name: "RideSharing_2"
  description: "App to book a cab to any destination"

Slots

- name: "number_of_riders"
  description: "Number of riders to call taxi for"

- name: "number_of_seats"
  description: "Number of seats to reserve in the cab"

Intents

- name: "GetRide"
  description: "Call a taxi to head to a given destination"

- name: "GetRide"
  description: "Book a cab for any destination, number of seats and ride type"
The Schema-guided Approach

Annotations dependent on the schema of API under consideration

<table>
<thead>
<tr>
<th>Flight Booking Service A</th>
<th>Flight Booking Service B</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Intents</strong></td>
<td><strong>Intents</strong></td>
</tr>
<tr>
<td>SearchFlight, ReserveFlight</td>
<td>FindFlight, ReserveFlight</td>
</tr>
<tr>
<td><strong>Slots</strong></td>
<td><strong>Slots</strong></td>
</tr>
<tr>
<td>origin, destination, depart, return, number_stops, trip_type, ...</td>
<td>depart, arrive, depart_date, return_date, direct_only, trip_type, ...</td>
</tr>
</tbody>
</table>

**SearchFlight:**  
origin = Baltimore  
destination = Seattle  
number_stops = 0

User: Find direct round trip flights from Baltimore to Seattle.

System: What dates are you looking for?

User: Flying out May 16 and returning May 20.

System: I found a Delta itinerary for 302 dollars.

**FindFlight:**  
depart = Baltimore  
arrive = Seattle  
direct_only = True

depart = May 16  
return = May 20
Overview

● Introduction
● The Schema-guided approach
● The Schema-guided dialogue dataset
● Data Collection Methodology
● Dialogue State Tracking Challenge (DSTC 8)
The Schema-Guided Dialogue Dataset (SGD)

<table>
<thead>
<tr>
<th>Metric</th>
<th>DSTC2</th>
<th>WOZ2.0</th>
<th>FRAMES</th>
<th>M2M</th>
<th>MultiWOZ</th>
<th>SGD</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Domains</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>7</td>
<td>16</td>
</tr>
<tr>
<td>#Dialogues</td>
<td>1,612</td>
<td>600</td>
<td>1,369</td>
<td>1,500</td>
<td>8,438</td>
<td>16,142</td>
</tr>
<tr>
<td>#Turns</td>
<td>23,354</td>
<td>4,472</td>
<td>19,986</td>
<td>14,796</td>
<td>113,556</td>
<td>329,964</td>
</tr>
<tr>
<td>#Slots</td>
<td>8</td>
<td>4</td>
<td>61</td>
<td>13</td>
<td>24</td>
<td>214</td>
</tr>
<tr>
<td>#Values</td>
<td>212</td>
<td>99</td>
<td>3,871</td>
<td>138</td>
<td>4,510</td>
<td>14,139</td>
</tr>
</tbody>
</table>

- **Largest** publicly available dataset for task-oriented dialogues
- Contains multi-domain conversations involving **34 services** over **16 domains**
- Test set contains few unseen services to test zero shot generalization
## Covered Domains (Training set)

For more details, please go to dataset link:  
https://github.com/google-research-datasets/dstc8-schema-guided-dialogue

<table>
<thead>
<tr>
<th>Domain</th>
<th>Weather</th>
<th>Banks</th>
<th>Buses</th>
<th>Calendar</th>
<th>Events</th>
<th>Flights</th>
<th>Homes</th>
<th>Hotels</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Dialogues</td>
<td>951</td>
<td>727</td>
<td>2280</td>
<td>1602</td>
<td>3509</td>
<td>2747</td>
<td>847</td>
<td>3353</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Domain</th>
<th>Media</th>
<th>Movies</th>
<th>Music</th>
<th>Rental car</th>
<th>Restaurant</th>
<th>Ride sharing</th>
<th>Services</th>
<th>Travel</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Dialogues</td>
<td>1113</td>
<td>1617</td>
<td>1290</td>
<td>1585</td>
<td>2419</td>
<td>1703</td>
<td>1889</td>
<td>1871</td>
</tr>
</tbody>
</table>
Okay, it's family movie night. We want to see a good movie that's family oriented.

Well, my first choice is Captain Marvel.

Oh. No, I want us to see a movie with subtitles tonight. I'm thinking The Curse of La Llorona movie might be a good choice if it has subtitles.

Do you know the title of the movie you want to see?

Okay, Captain Marvel is available. There are no subtitles with this movie. Please confirm if this is the movie you want to see.

Yes, The Curse of La Llorona film is available with subtitles. If you want to see that movie, please confirm.
Example - multi domain

Please tell me the weather forecast for March 8th.

Any particular city's weather forecast?

Please check the weather in **Menlo Park**.

The expected average temperature for the day is 83 degrees Fahrenheit, with a 9 percent chance of rain.

Sounds good! Please find me a restaurant offering Latin American cuisine there.

Cafe Del Sol Restaurant, a nice restaurant **Menlo Park** is among the 3 I found.

Weather Domain → **Restaurant Domain**
Overview

- Introduction
- The Schema-guided approach
- The Schema-guided dialogue dataset
- **Data collection methodology**
- Dialogue state tracking challenge (DSTC 8)
Data Collection Methodology

- Two commonly used approaches for data collection
  - Chat based
  - Simulation based
Chat-based Data Collection

1. Crowdworkers chat with each other given the high-level intent.
2. Annotate the dialogues.

Task: Book a restaurant in los altos and ask for its price range.

I want to eat in los altos today at 6pm.

May generate more realistic conversation.

- Coarse or no control over dialogue flow.
  → Hard to control the dialogue variations.

- Annotation can be noisy and expensive [MultiWOZ 2.1 - Eric et. al.]
Simulation-based Data Collection

Approach 2:
1. Simulate dialogue outlines.
2. Paraphrase a simulated conversation

INFORM(location="los altos")
INFORM(date="today")
INFORM(time="6pm")

I want to eat in los altos today at 6pm.

✓ Annotations are cleaner as they are automatically generated
✓ Cheaper and faster to collect
✓ Easy to ensure coverage of many dialogue flows
  → Control the dialogue variation
✗ Requires a powerful simulator that simulates the dialogue outlines
Data Collection Approach

1. Generate dialogue skeletons using a multi-domain simulator
   - REQUEST(location)
   - REQUEST(cuisine)
   - INFORM(location="los altos")
   - INFORM(date="tomorrow")
   - INFORM(cuisine="greek")
   - OFFER(restaurant="Opa!")

2. Use utterance templates to generate synthetic conversations
   - Where do you want to eat? What kind of food?
   - I want to eat in los altos.
   - I want to go there tomorrow.
   - I want to eat greek food.
   - Opa! is a wonderful restaurant.

3. Get paraphrased dialogues from crowdworkers
   - Where are you located and what is your preferred food?
   - I am looking for a greek restaurant in los altos tomorrow.
   - How does Opa! sound?
The Dialogue Simulator

- Multi-domain simulator generates dialogue outlines containing user and system actions and state annotations

- User Actions (11)
  - INFORM_INTENT,
  - AFFIRM_INTENT,
  - NEGATE_INTENT,
  - THANK_YOU,
  - AFFIRM,
  - NEGATE,
  - REQUEST_ALTS,
  - SELECT

- System Actions (10)
  - OFFER_INTENT,
  - INFORM_COUNT,
  - REQ_MORE,
  - OFFER,
  - CONFIRM,
  - NOTIFY_SUCCESS,
  - NOTIFY_FAILURE

- Implementation details to be released after conclusion of DSTC8
Utterance Templates

- User and system actions are converted into robot utterances using templates

User Actions

<table>
<thead>
<tr>
<th>INFORM(where_to=A)</th>
<th>I want to travel to A.</th>
</tr>
</thead>
<tbody>
<tr>
<td>INFORM(where_to=don't care)</td>
<td>Any place works for me, I just want to travel.</td>
</tr>
</tbody>
</table>

System Actions

<table>
<thead>
<tr>
<th>OFFER(title=A, B)</th>
<th>What about A or B?</th>
</tr>
</thead>
<tbody>
<tr>
<td>INFORM(number_stops=A)</td>
<td>The flight has A stops.</td>
</tr>
</tbody>
</table>

- Templates make it easier to convey the meaning of actions to workers
## Dialogue Paraphrasing

- Generate natural utterances from robot utterances
- Very simple, annotation preserving crowd task

<table>
<thead>
<tr>
<th>SYSTEM</th>
<th>Please confirm the following details: The name of the hair stylist is Great Clips, the appointment is at 2 pm and the appointment is on March 2nd.</th>
<th>Enter the utterance text here</th>
</tr>
</thead>
<tbody>
<tr>
<td>USER</td>
<td>Yes that sounds good. What is its contact number? What is its average rating?</td>
<td>Enter the utterance text here</td>
</tr>
<tr>
<td>SYSTEM</td>
<td>Your appointment is confirmed. Its average rating is 3.8. Its phone number is 408-532-1849.</td>
<td>Enter the utterance text here</td>
</tr>
<tr>
<td>USER</td>
<td>Thank you very much. I want to watch a movie around there on that day. I like Comedy movies.</td>
<td>Enter the utterance text here</td>
</tr>
</tbody>
</table>
Data Collection Challenge

- Some crowd workers try to get away with minimal work.

<table>
<thead>
<tr>
<th>Short unnatural utterances</th>
<th>No paraphrasing - copy from the given outlines</th>
</tr>
</thead>
<tbody>
<tr>
<td>U: Reserve hotel</td>
<td>U: I am looking for something to eat. I want it in</td>
</tr>
<tr>
<td>S: name</td>
<td>Fremont. I am looking for some intermediate priced</td>
</tr>
<tr>
<td>U: one hotel andra 4th of</td>
<td>food. S: What kind of food are you looking for?</td>
</tr>
<tr>
<td>march nine</td>
<td>U: Some Asian Fusion kind of food will be perfect.</td>
</tr>
<tr>
<td>S: Ok</td>
<td></td>
</tr>
<tr>
<td>U: Ok</td>
<td></td>
</tr>
<tr>
<td>S: Ok</td>
<td></td>
</tr>
</tbody>
</table>

Solution

- Create a whitelist of skilled crowdworkers.
- Automated checks to ensure alignment with annotations and variability in utterances.
Whitelist Creation

Initial paraphrasing tasks open to all workers, followed by a verification task for obtaining scores for each paraphrased dialogue.

<table>
<thead>
<tr>
<th>Speaker</th>
<th>Outline</th>
<th>Dialogue</th>
<th>Is dialogue valid and natural?</th>
</tr>
</thead>
<tbody>
<tr>
<td>USER</td>
<td>I am looking for a place to eat. I want it in Fremont. I am looking for something intermediate priced.</td>
<td>I'm looking for somewhere I can grab some food in Fremont. Something intermediate priced would be perfect.</td>
<td>True False</td>
</tr>
<tr>
<td>SYSTEM</td>
<td>What kind of food are you looking for?</td>
<td>Is there any kind of food you want to eat specifically today?</td>
<td>True False</td>
</tr>
</tbody>
</table>

Do you think the overall dialogue sounds like a natural conversation?
- True
- False
How to ensure data quality?

Use verification task and scripts to find skilled workers

- Give paraphrasing work only to the qualified crowd workers.
- Update the whitelist consistently
  - Run scripts to compute the similarities between outlines and paraphrased utterances.
  - Remove workers whose dialogues are of high similarity score and their dialogues.
Overview

- Introduction
- The Schema-guided approach
- The Schema-guided dialogue dataset
- Data Collection Methodology
- Dialogue State Tracking Challenge (DSTC 8)
DSTC 8 Task

● Aim
  ○ Dialogue state tracking in a virtual assistant

● New Challenges
  ○ Zero-shot learning of unseen services
  ○ Handling OOV slot values
  ○ Transfer of slot values in multi-domain dialogues

● Submission deadline: Oct 13, 2019
Thank You!

Questions?

xiaoxuez@google.com
abhirast@google.com