Dialog State Tracking Challenge 4

Handbook v.3.0

http://www.colips.org/workshop/dstc4

Seokhwan Kim[1], Luis Fernando D'Haro[1], Rafael E. Banchs[1],
Jason Williams[2], and Matthew Henderson[3]

[1] Human Language Technologies, Institute for Infocomm Research (I2R - A*STAR)
One Fusionopolis Way, #21-01 Connexis (South Tower), Singapore 138632
[2] Microsoft Research, One Microsoft Way, Redmond, WA, USA - 98052-6399
[3] Google, 1-13 St Giles High St, London WC2H 8LG, United Kingdom

July 31, 2015

TABLE OF CONTENTS

1. Motivation

Dialog state tracking is one of the key sub-tasks of dialog management, which defines the representation of dialog states and updates them at each moment on a given on-going conversation. To provide a common test bed for this task, the first Dialog State Tracking Challenge (DSTC) was organized[1]. More recently, Dialog State Tracking Challenges 2 & 3 have been successfully completed[2].

In this fourth edition of the Dialog State Tracking Challenge, we will focus on a dialog state tracking task on human-human dialogs. In addition to this main task, we also propose a series of pilot tracks for the core components in developing end-to-end dialog systems based on the same dataset. We expect these shared efforts on human dialogs will contribute to progress in developing much more human-like systems.

This document is distributed as follows: section 2 gives detailed information about the challenge (i.e. how to register, the competition schedule, main and optional tasks). In section 3 the data used during the challenge is described. Examples of the dialogs included are also given. Section 4 describes the evaluation metrics and format for the main task submissions. Then, in section 5, description of several tools included with the data is provided. These tools are intended to allow participants to check the data, to have a baseline system that participants can modify or combine with their proposed systems. In section 6, the JSON data formats used for the annotations are described in detail. Finally, in section 7 several frequent questions are addressed regarding the data and participation in the challenge.

2. Participation

In this challenge, participants will be provided with labelled human-computer dialogs to develop dialog state tracking algorithms. Algorithms will then be evaluated on a common set of held-out dialogs, offline, to enable comparisons. As well as a corpus of labelled dialogs, participants will be given code that implements the evaluation measurements and a baseline tracker.

2.1. *Rules*

Participation is welcomed from any research team (academic, corporate, non profit, government). Members of the organizational committee and advisory committee are permitted to participate. In general, the identity of participants will not be published or made public. In written results, teams will be identified as team1, team2, etc. There are 2 exceptions to this: (1) the organizers will verbally indicate the identities of all teams at the conference/workshop chosen for communicating results; and (2) participants may identify their own team label (e.g. team5), in publications or presentations, if they desire, but may not identify the identities of other teams. On submission of results, teams will be required to fill in a questionnaire which gives some broad details about the approach they adopted.

In particular, there is interest in writing trackers that would be feasible for use in live systems. We provide a few recommendations for participants to follow so trackers may be directly comparable to others. These recommendations are not enforced, but will be addressed in the questionnaire at evaluation time.

- A tracker should be able to run fast enough to be used in a real-time dialog system
- A tracker should not make multiple runs over the test set before outputting its results

[1] http://research.microsoft.com/en-us/events/dstc/
[2] http://camdial.org/~mh521/dstc/

- A tracker should not use information from the future of a dialog to inform its output at a given turn

2.2. *Schedule*

Below we provide the proposed schedule for the evaluation as well as the conference submissions.

Shared Tasks

01 Apr 2015	Registration opens
15 Apr 2015	Labeled training and development data is released
17 Aug 2015	Unlabeled test data is released
31 Aug 2015	Entry submission deadline
04 Sep 2015	Evaluation results are released

Task Papers

23 Sep 2015	Paper submission deadline
31 Oct 2015	Paper acceptance notifications
16 Nov 2015	Camera-ready submission deadline
13-16 Jan 2016	Workshop is held @ IWSDS2016

Announcements and discussion about the challenge will be conducted on the group mailing list. Participants should be on the mailing list. Instructions for joining can be found on the DSTC homepage and at section 8 in this document.

2.3. *Registration*

The procedure to register to the Fourth Dialog State Tracking Challenge is as follows:

- **STEP 1:** Complete the registration form (available online). Please allow for few days while your request is being processed.
- **STEP 2:** Print the End-User-License Agreement for the TourSG dataset that you will receive.
- **STEP 3:** Complete, sign and submit back the End-User-License Agreement: the scanned version to luisdhe@i2r.a-star.edu.sg and the original form to the following address:

  Mr Teo Poh Heng
  Exploit Technologies Pte Ltd
  30 Biopolis Street, #09-02 Matrix
  Singapore 138671
  Tel: +65-6478 8452

- **STEP 4:** You will receive a one-time password for downloading the dataset.

2.4. *Tasks*

2.4.1. *Main task*

The goal of the main task of the challenge is to track dialog states for sub-dialog segments. For each turn in a given sub-dialog, the tracker should fill out a frame of slot-value pairs considering all dialog history prior to the turn. The performance of a tracker will be evaluated by comparing its outputs with reference annotations.

In the development phase, participants will be provided with both training and development sets of dialogs with manual annotations over frame structures. In the test phase, each tracker will be evaluated on the results generated for a test set of unlabeled dialogs. A baseline system and evaluation script will be provided along with the training data. Participation in the main track is mandatory for all teams and/or individuals registered in the DSTC4.

2.4.2. *Pilot tasks*

Four pilot tasks are available in DSTC4. These pilot tasks are optional for all participants in the challenge. However all participants are encouraged to participate in any or all of them.

- **Spoken language understanding:** Tag a given utterance with speech acts and semantic slots.
- **Speech act prediction:** Predict the speech act of the next turn imitating the policy of one speaker.
- **Spoken language generation:** Generate a response utterance for one of the participants.
- **End-to-end system:** Develop an end-to-end system playing the part of a guide or a tourist. This task will be conducted only if at least one team and/or individual registers for each of the pilot tasks above.

2.4.3. *Open track*

DST4 registered teams and/or individuals are free to work and report results on any proposed task of their interest over the provided dataset.

3. Data

3.1. *General Characteristics*

In this challenge, participants will use TourSG corpus to develop the components. TourSG consists of 35 dialog sessions on touristic information for Singapore collected from Skype calls between three tour guides and 35 tourists. These 35 dialogs sum up to 31,034 utterances and 273,580 words. All the recorded dialogs with the total length of 21 hours have been manually transcribed and annotated with speech act and semantic labels for each turn level.

Since each subject in these dialogs tends to be expressed not just in a single turn, but through a series of multiple turns, dialog states are defined in these conversations for each sub-dialog level. A full dialog session is divided into sub-dialogs considering their topical coherence and then they are categorized by topics. Each sub-dialog assigned to one of major topic categories will have an additional frame structure with slot value pairs to represent some more details about the subject discussed within the sub-dialog (see an example of reference annotations).

3.2. *Example of Dialog Annotations for the Main Task*

Speaker	Transcription	Annotation
Tourist	Can you give me some uh- tell me some cheap rate hotels, because I'm planning just to leave my bags there and go somewhere take some pictures.	**Topic:** Accommodation **Tourist_ACT:** REQ **Guide_ACT**: ACK **Type:** Hostel
Guide	Okay. I'm going to recommend firstly you want to have a backpack type of hotel, right?	
Tourist	Yes. I'm just gonna bring my backpack and my buddy with me. So I'm kinda looking for a hotel that is not that expensive. Just gonna leave our things there and, you know, stay out the whole day.	
Guide	Okay. Let me get you hm hm. So you don't mind if it's a bit uh not so roomy like hotel because you just back to sleep.	
Tourist	Yes. Yes. As we just gonna put our things there and then go out to take some pictures.	
Guide	Okay, um-	
Tourist	Hm.	

Table 1. Example of a transcription and annotation for a sub-dialog segment #1

Speaker	Transcription	Annotation
Guide	Let's try this one, okay?	**Topic:** Accommodation **GuideAct:** RECOMMEND **TouristAct:** ACK **INFO:** Pricerange **Name:** InnCrowd Backpackers Hostel
Tourist	Okay.	
Guide	It's InnCrowd Backpackers Hostel in Singapore. If you take a dorm bed per person only twenty dollars. If you take a room, it's two single beds at fifty nine dollars.	
Tourist	Um. Wow, that's good.	
Guide	Yah, the prices are based on per person per bed or dorm. But this one is room. So it should be fifty nine for the two room. So you're actually paying about ten dollars more per person only.	
Tourist	Oh okay. That's- the price is reasonable actually. It's good.	

Table 2. Example of a transcription and annotation for a sub-dialog segment #2

3.3. *Data Available for DSTC4*

For the purposes of the DSTC4 Challenge, the SGTour corpus has been divided in the following three parts:

1. **Train data:** manual transcriptions and annotations at both utterance and sub-dialog levels will be provided for 14 dialogs (7 from tour guide-1 and 7 from tour guide-2) for training the trackers.
2. **Dev data:** similar to the training data. In this case, 6 dialogs (3 from tour guide-1 and 3 from tour guide-2) for optimizing the trackers.
3. **Test data:** manual transcriptions will be provided for 15 dialogs (5 from tour guide-1, 5 from tour guide-2 and 5 from tour guide-3) for evaluating the trackers.

The three datasets will be released free of charge to all registered challenge participants after signing a license agreement with ETPL-A*STAR. The dataset will include transcribed and annotated dialogs, as well as ontology objects describing the annotations.

4. Evaluation

A system for the main task should generate the tracking output for every utterance in a given log file described in Section 6.1. While all the transcriptions and segment details provided in the log object from the beginning of the session to the current turn can be used, any information from the future turns are not allowed to be considered to analyze the state at a given turn.

Although the fundamental goal of this tracking is to analyze the state for each sub-dialog level, the execution should be done in each utterance level regardless of the speaker from the beginning to the end of a given session in sequence. It aims at evaluating the capabilities of trackers not only for understanding the contents mentioned in a given segment, but also for predicting its dialog states even at an earlier turn of the segment.

To examine these both aspects of a given tracker, two different 'schedules' are considered to select the utterances for the target of evaluation:

- Schedule 1 - all turns are included
- Schedule 2 - only the turns at the end of segments are included

If some information is correctly predicted or recognized at an earlier turn in a given segment and well kept until the end of the segment, it will have higher accumulated scores than the other cases where the same information is filled at a later turn under schedule 1. On the other hand, the results under schedule 2 indicate the correctness of the outputs after providing all the turns of the target segment.

In this challenge, the following two sets of evaluation metrics are used for the main task:

- Accuracy: Fraction of segments in which the tracker's output is equivalent to the gold standard frame structure
- Precision/Recall/F-measure
  - Precision: Fraction of slot-value pairs in the tracker's outputs that are correctly filled
  - Recall: Fraction of slot-value pairs in the gold standard labels that are correctly filled
  - F-measure: The harmonic mean of precision and recall

While the first metric is to check the equivalencies between the outputs and the references in whole frame-level, the others can show the partial correctness in each slot-value level.

Regarding operational aspects of the main track evaluation, it will be run as a CodaLab[3] competition. Every participant should create a CodaLab account first and then register for participating at DSTC4 competition page. Once the registration request is confirmed by organizers, the participants will be able to make submissions of the outputs from their trackers. Each submission should be done by uploading a single zip file that consists of a JSON file named 'answer.json' generated following the definition in Section 6.3. If there's no error on the submitted file, the results can be posted to the leaderboard and compared to the results from other participants. More details regarding participating in CodaLab competitions can be found from the CodaLab Wiki page[4].

5. Included Scripts and Tools

As in the previous DSTC 2 and 3 evaluations, the DSTC 4 evaluation includes a set of useful scripts and tools for dealing with the provided data. Below a brief description of the available tools is provided.

5.1. *Baseline Tracker*

A simple baseline tracker is included with the data. The baseline tracker determines the slot values by fuzzy string matching between the entries in the ontology and the transcriptions of the utterances mentioned from the beginning of a given segment to the current turn. If a part of given utterances is matched with an entry for a slot in the ontology with over a certain level of similarity, the entry is simply assigned as a value for the particular slot in the tracker's output. Since this baseline doesn't consider any semantic or discourse aspects from given dialogs, its performance is very limited and there is much room for improvement. The source code for the baseline tracker is included in *baseline.py*, please look there for full details on the implementations.

5.2. *Running and Evaluating Baseline*

This section serves as an introduction to using the baseline tracker and the evaluation scripts. For running the baseline tracker, FuzzyWuzzy[5] package should be installed. And you should have a scripts directory with a config directory within it. The config directory contains the definitions of the datasets, e.g. *dstc4_dev.flist* which enumerates the sessions in the development set of DSTC 4. It also contains the ontology objects in *ontology_dstc4.json*. You can run the baseline tracker like so:

```
python scripts/baseline.py --dataset dstc4_dev --dataroot data --trackfile
baseline_dev.json --ontology scripts/config/ontology_dstc4.json
```

This will create a file *baseline_dev.json* with a tracker output object. The structure and contents of the output can be checked using *check_track.py*:

```
python scripts/check_track.py --dataset dstc4_dev --dataroot data --ontology
scripts/config/ontology_dstc4.json --trackfile baseline_dev.json
```

This should output `Found no errors, trackfile is valid'. The checker is particularly useful for checking the tracker output on an unlabelled test set, before submitting it for evaluation in the challenge.

The evaluation script, *score.py* can be run on the tracker output like so:

```
python scripts/score.py --dataset dstc4_dev --dataroot data --trackfile
```

[3] https://www.codalab.org/

[4] https://github.com/codalab/codalab/wiki/User_Participating-in-a-Competition

[5] https://pypi.python.org/pypi/fuzzywuzzy

```
baseline_dev.json --scorefile baseline_dev.score.csv --ontology
scripts/config/ontology_dstc4.json
```

This creates a file *baseline_dev.score.csv* which lists all the metrics and we can use report.py to format these results:

```
python scripts/report.py --scorefile baseline_dev.score.csv
```

5.3. *Other Tools*

There are a few other scripts included which may be of use for participants:

- **dataset_walker.py:** A Python script which makes it easy to iterate through a dataset specified by file list (.flist) in scripts/config. When the script is called without arguments it outputs the content of all the training data on the terminal. In case you want to check the content of the development data, you need to modify the parameter value from dstc4_train to dstc4_dev.
- **ontology_reader.py**: A Python script which makes it easy to get the information from the ontology.

6. JSON Data Formats

The datasets are distributed as collections of dialogs, where each dialog has a log.json file containing a Log object in JSON, and possibly a label.json containing a Label object in JSON representing the annotations. Also distributed with the data is an Ontology JSON object, which describes the ontology/domain of the sessions. The below sections describe the structure of the Log, Label and Ontology objects.

6.1. *Log Objects*

The *log.json* file includes the information for each session between a given tourist and a given guide. The JSON files were generated following below the specification:

- session_id: a unique ID for this session (integer)
- session_date: the date of the call, in yyyy-mm-dd format (string)
- session_time: the time the call was started, in hh:mm:ss format (string)
- guide_id: a unique ID for the guide participated in this session (string)
- tourist_id: a unique ID for the tourist participated in this session (string)
- tourist_age: the age of the tourist (integer)
- tourist_sex: the gender of the tourist (string: “F”/“M”)
- tourist_visited_sg: whether the tourist has visited or not Singapore in the past (string: “Y”/“N”)
- utterances: [
    - utter_index: the index of this utterances in the session starting at 0 (integer)
    - speaker: the speaker of this utterance (string: “GUIDE”/“TOURIST”)
    - transcript: the transcribed text of this utterance (string). Filler disfluencies in the recorded utterance are annotated with preceding percent sign (%) like “%ah”, “%eh”, “%uh”, or “%um”.
    - segment_info: [

        - topic: the topic category of the dialog segment that this utterance belongs to (string: “OPENING” / “CLOSING” / “ITINERARY” / “ACCOMMODATION” / “ATTRACTION” / “FOOD” / “SHOPPING” / “TRANSPORTATION”)
        - target_bio: the indicator with BIO scheme whether this utterance belongs to a segment considered as a target for the main task or not. The value for this key should be ‘B’ if this utterance is located at the beginning of a target session or ‘I’ if the utterance is not at the beginning but inside the target session. Otherwise, it is assigned to ‘O’. (string: “B”/“I”/“O”)
        - guide_act: the dialog act of the guide through this segment (string: “QST” / “ANS” / “REQ” / “REQ_ALT” / “EXPLAIN” / “RECOMMEND” / “ACK” / “NONE”)
        - tourist_act: the dialog act of the tourist through this segment (string: “QST” / “ANS” / “REQ” / “REQ_ALT” / “EXPLAIN” / “RECOMMEND” / “ACK” / “NONE”)
        - initiativity: whether this segment is initiated by the guide or the tourist (string: “GUIDE” / “TOURIST”)
    ]
]

6.2. *Label Objects*

The annotations for each segment are given in the *label.json* file. The json object in the label file consists of three different types of labels: frame structures for the main task and speech acts and semantics for the other pilot tasks. Below is the specification of the object:

- session_id: a unique ID for this session (integer)
- utterances: [
    - utter_index: a unique ID for this session (integer)
    - frame_label
        - SLOT: [list of values (string)]
    - speech_act: [
        - act: speech act category (string)
        - attributes: [list of attributes (string)]

      ]
    - semantic_tagged: [list of tagged utterances (string)]

]

6.2.1. *Frame labels*

The gold standard frame structure for the dialog segment that the current utterance belongs to is given as the object value of the ‘frame_label’ key. Each object consists of a set of a slot and a list of values pairs defined for the topic category of a given segment. Slots can be categorized into two different types: regular slots and ‘INFO’ slot. Each regular slot represents a major subject defined for a given topic and it should be filled with particular values mainly discussed in the current segment. Below is the list of regular slots for every topic category and their descriptions.

- ACCOMMODATION

    - PLACE: It refers to the names of accommodations discussed in a given segment
    - TYPE_OF_PLACE: It refers to the types of accommodations discussed in a given segment
    - NEIGHBOURHOOD: It refers to the geographic areas where the accommodations are located
- ATTRACTION
    - PLACE: It refers to the names of attractions discussed in a given segment
    - TYPE_OF_PLACE: It refers to the types of attractions discussed in a given segment
    - NEIGHBOURHOOD: It refers to the geographic areas where the attractions are located
    - ACTIVITY: It refers to the touristic activities discussed in a given segment
    - TIME: It refers to the discussed time slots to visit the attractions
- FOOD
    - PLACE: It refers to the names of places for eating discussed in a given segment
    - TYPE_OF_PLACE: It refers to the types of places for eating discussed in a given segment
    - NEIGHBOURHOOD: It refers to the geographic areas where the eating places are located
    - CUISINE: It refers to the cuisine types discussed in a given segment
    - DISH: It refers to the names of dishes discussed in a given segment
    - DRINK: It refers to the names of drinks discussed in a given segment
    - MEAL_TIME: It refers to the discussed time slots for eating
- SHOPPING
    - PLACE: It refers to the names of places for shopping discussed in a given segment
    - TYPE_OF_PLACE: It refers to the types of places for shopping discussed in a given segment
    - NEIGHBOURHOOD: It refers to the geographic areas where the shopping places are located
    - TIME: It refers to the discussed time slots for shopping
- TRANSPORTATION
    - TYPE: It refers to the types of transportation discussed in a given segment
    - TO: It refers to the destinations discussed in a given segment
    - FROM: It refers to the origins discussed in a given segment
    - LINE: It refers to the MRT lines discussed in a given segment
    - STATION: It refers to the train stations discussed in a given segment
    - TICKET: It refers to the types of tickets for transportation

In addition to the regular slots, a frame could have a special slot named 'INFO' to indicate the subjects that are discussed in a given segment but not directly related to any particular values of other slots. For example, 'INFO' slot in a frame for 'FOOD' topic could be filled in with 'DISH' value if the segment deals with some general contents regarding dishes. But, when the speakers are talking about a specific dish, the frame has the corresponding value for the 'DISH' slot instead of a 'DISH' value for 'INFO' slot. Below is the list of 'INFO' slot values for each topic category and the descriptions about the target contents to be annotated with them.

ACCOMMODATION	
Amenity	amenities of accommodations
Architecture	architectural aspects of accommodations
Booking	booking for accommodations

Check-in	checking in for accommodations
Check-out	checking out of accommodations
Cleanness	cleanness of accommodations
Facility	facilities of accommodations
History	history of accommodations
Hotel rating	rating of accommodations
Image	dialogs with showing some images of accommodations
Itinerary	itinerary focusing on accommodations
Location	locations of accommodations
Map	dialogs with showing maps of the areas near accommodations
Meal included	meal plans provided by accommodations
Name	names of accommodations
Preference	tourists' preferences in looking for accommodations
Pricerange	room charges for accommodations
Promotion	discount promotions for accommodations
Restriction	any restrictions in accommodations
Room size	room sizes in accommodations
Room type	room types in accommodations
Safety	safety issues in accommodations

ATTRACTION	
Activity	tourist activities
Architecture	architectural aspects of tourist attractions
Atmosphere	atmosphere of tourist attractions
Audio guide	audio guide provided by tourist attractions
Booking	booking for tourist attractions
Dresscode	dress code for tourist attractions
Duration	time durations for visiting tourist attractions
Exhibit	exhibits shown in tourist attractions
Facility	facilities of tourist attractions
Fee	admission charges for tourist attractions
History	history of tourist attractions
Image	dialogs with showing some images of tourist attractions
Itinerary	itinerary focusing on tourist attractions
Location	locations of tourist attractions
Map	dialogs with showing maps of the areas near tourist attractions
Name	names of tourist attractions
Opening hour	operation hours of tourist attractions
Package	package tours or tickets for tourist attractions
Place	general discussion about tourist places without specifying target attractions
Preference	tourists' preferences in visiting tourist attractions
Promotion	discount promotions for tourist attractions
Restriction	any restrictions in tourist attractions

Safety	safety issues in tourist attractions
Schedule	schedules for exhibitions or shows in tourist attractions
Seat	seat information for shows in tourist attractions
Ticketing	ticketing information for tourist attractions
Tour guide	guided tour for tourist attractions
Type	types of tourist attractions
Video	dialogs with showing some video clips of tourist attractions
Website	dialogs with showing websites of tourist attractions

FOOD	
Cuisine	cuisine type for foods
Delivery	delivery services of foods
Dish	general discussion about dishes without specifying targets
History	history of foods or restaurants
Image	dialogs with showing some images of foods or restaurants
Ingredient	ingredients for foods
Itinerary	itinerary focusing on dining
Location	locations of restaurants
Opening hour	operation hours of restaurants
Place	general discussion about dining places without specifying targets
Preference	tourists' preferences for dining
Pricerange	price ranges for dining
Promotion	discount promotions for dining
Restriction	any restrictions in dining
Spiciness	spiciness about foods
Type of place	types of food places

SHOPPING	
Brand	brands of goods
Duration	time durations for shopping
Image	dialogs with showing some images of goods or shopping places
Item	shopping items
Itinerary	itinerary focusing on shopping
Location	locations of shopping places
Map	dialogs with showing maps of the areas near shopping places
Name	names of shopping places
Opening hour	operation hours of shopping places
Payment	payment options available at shopping places
Place	general discussion about shopping places without specifying targets
Preference	tourists' preferences for shopping
Pricerange	price ranges for shopping
Promotion	discount promotions for shopping
Tax refund	information about tax refund for tourists

Type	types of shopping places

TRANSPORTATION	
Deposit	information about deposits in tickets
Distance	traveling distance between origin and destination
Duration	travel time between origin and destination
Fare	transportation expenses
Itinerary	itinerary focusing on local transportation
Location	locations of train stations, bus stops, or terminals
Map	dialogs with showing train or bus route maps
Name	names of train stations, bus stops, or terminals
Preference	tourists' preferences in travelling with local transportations
Schedule	schedules for public transportations
Service	services related to transportations
Ticketing	ticketing information for local public transportations
Transfer	information about transit transfer to another line or another type of transportation
Type	types of transportation

The set of candidate values for each slot can be found in the ontology object (Section 6.4). Since each slot has a list of string values in its JSON object, multiple values can be assigned to a single slot, if more than one subject regarding the particular slot type are discussed in a given segment. All the annotations have been done considering not only the occurrences of relevant words for each candidate value in the surface of the segment, but also the correlations with the main subject of conversation at the moment that the sub-dialog was going on.

6.2.2. *Speech acts*

Since the speech acts were originally analyzed for each sub-utterance unit divided based on the pauses in the recordings and then combined into the full utterance level, each utterance could have more than one speech act objects if it was generated by concatenating its multiple sub-utterances. Thus, a list of speech act annotations is taken as the value for the 'speech_act' key of a given utterance.

Each object has two types of information: speech act category and attributes. Every sub-utterance should belong to one of the four basic speech act categories that denote the general role of the utterance in the current dialog flow. More specific speech act information can be annotated by combination with attributes. By contrast to act category, there's no constraint on the number of attributes for a single utterance. Thus, a sub-utterance can have no attribute or more than one attributes in the list object. Below are the list of speech act categories and attributes with their descriptions.

- Speech act categories
    - QST (QUESTION) used to identify utterances that pose either a question or a request
    - RES (RESPONSE) used to identify utterances that answer to a previous question or a previous request

    - INI (INITIATIVE) used to identify utterances that constitute new initiative in the dialog, which does not constitute either a question, request, answer or follow up action to a previous utterance
    - FOL (FOLLOW) a response to a previous utterance that is not either a question or a request
- Speech act attributes
    - ACK: used to indicate acknowledgment, as well as common expressions used for grounding
    - CANCEL: used to indicate cancelation
    - CLOSING: used to indicate closing remarks
    - COMMIT: used to identify commitment
    - CONFIRM: used to indicate confirmation
    - ENOUGH: used to indicate/request that no more information is needed
    - EXPLAIN: used to indicate/request an explanation/justification of a previous stated idea
    - HOW_MUCH: used to indicate money or time amounts
    - HOW_TO: used to request/give specific instructions
    - INFO: used to indicate information request
    - NEGATIVE: used to indicate negative responses
    - OPENING: used to indicate, opening remarks
    - POSITIVE: used to indicate positive responses
    - PREFERENCE: used to indicate/request preferences
    - RECOMMEND: used to indicate/request recommendations
    - THANK: used to indicate thank you remarks
    - WHAT: used to indicate concept related utterances
    - WHEN: used to indicate time related utterances
    - WHERE used to indicate location related utterances
    - WHICH: used to indicate entity related utterances
    - WHO: used to indicate person related utterances and questions

6.2.3. *Semantic tags*

Similarly to speech acts, semantic tags were also annotated for each sub-utterance level. Thus it takes a list of tagged sub-utterances as its value, and the number of items in the list should be the same with the one for speech acts.

We defined below the main categories for semantic annotation:

- AREA: It refers to a geographic area but not a specific spot or location
- DET: It refers to user's criteria used or reasons why the user would like to decide spot.
- FEE: It refers to admission fees, price of services or any other fare.
- FOOD: It refers to any type of food or drinks.
- LOC: It refers to specific touristic spots or commerce/services locations.
- TIME: It refers to time, terms, dates, etc.
- TRSP. It refers to expressions related to transportation and transportation services.
- WEATHER: It refers to any expression related to weather conditions.

Some of them include also subcategories, relative modifiers and from-to modifiers (Table 3).

MAIN	SUBCAT	REL	FROM-TO
AREA	COUNTRY, CITY, DISTRICT, NEIGHBORHOOD	NEAR, FAR, NEXT, OPPOSITE, NORTH, SOUTH, EAST, WEST	FROM, TO
DET	ACCESS, BELIEF, BUILDING, EVENT, PRICE, NATURE, HISTORY, MEAL, MONUMENT, STROLL, VIEW	-	-
FEE	ATTRACTION, SERVICES, PRODUCTS	-	-
FOOD	-	-	-
LOC	TEMPLE, RESTAURANT, SHOP, CULTURAL, GARDEN, ATTRACTION, HOTEL, WATERSIDE, EDUCATION, ROAD, AIRPORT	NEAR, FAR, NEXT, OPPOSITE, NORTH, SOUTH, EAST, WEST	FROM, TO
TIME	DATE, INTERVAL, START, END, OPEN, CLOSE	BEFORE, AFTER, AROUND	-
TRSP	STATION, TYPE	NEAR, FAR, NEXT, OPPOSITE, NORTH, SOUTH, EAST, WEST	FROM, TO
WEATHER	-	-	-

Table 3. List of Categories and Modifiers for Semantic Annotations

The semantic tags and their categories are indicated as follows:

- <MAIN CAT=”SUBCAT” REL=”REL” FROM-TO=”FROM_TO”> at the beginning of the identified word or compound
- </TAG> at the end of the identified word or compound

When either no specific subcategory exists for a given semantic tag or it is not possible to select among the available subcategories, the ‘CAT’ field is assigned to cat=”MAIN”.

6.3. *Tracker Output Objects*

Tracker outputs should be organized following below the JSON specification:

- dataset: the name of the dataset over which the tracker has been run (string)
- wall_time: the time in seconds it took to run the tracker (float)
- sessions: a list of results corresponding to each session in the dataset [
    - session_id: the unique ID of this session (integer)
    - utterances: [

        - utter_index: a unique ID for this session (integer)
        - frame_label: the tracker output for the segment that this utterance belongs to. The expected format for this object is same as the ones in the reference label objects. (Section 6.2.1)

    ]

6.4. *Ontology Object*

The ontology object in *ontology_dstc4.json* describes the definitions of the frame structures for the main task and some additional domain knowledges in the following format:

- tagsets
    - TOPIC
        - SLOT: [list of possible values]
- knowledge
    - MRT_LINE
        - CODE: string
        - NAME: string
        - COLOR: string
    - SHOPPING
        - NAME: string
        - TYPE_OF_PLACE: [list of shopping place types]
    - RESTAURANT
        - NAME: string
        - TYPE_OF_PLACE: [list of restaurant types]
        - NEIGHBOURHOOD: [list of neighbourhoods]
        - CUISINE: [list of cuisines]
        - PRICERANGE: integer (from 1 to 5)
    - FOOD
        - NAME: string
        - CUISINE: string
    - MRT_STATION
        - NAME: string
        - CODE: [list of codes]
        - NEIGHBOURHOOD: [list of neighbourhoods]
    - HOTEL
        - NAME: string
        - TYPE_OF_PLACE: [list of hotel types]
        - NEIGHBOURTHOOD: [list of neighbourhoods]
        - RATING: integer (from 1 to 5)
        - PRICERANGE: integer (from 1 to 5)
    - ATTRACTION
        - NAME: string
        - TYPE_OF_PLACE: [list of attraction types]
        - NEIGHBOURHOOD: [list of neighbourhoods]

- ROAD
  - NAME: string
  - NEIGHBOURHOOD: [list of neighbourhoods]
- NEIGHBOURHOOD
  - REGION: string
  - DISTRICT: string
  - SUBDISTRICT: string

The slots and their values in the frame labels in both label outject and tracker output object should be chosen following the definitions in the ‘tagset’ object. For each target topic category, a set of slots in the frame structure are given in the object. And the list of valid values are also specified as the value of the slot in this object. Two different types of values can be described in this list: the first type is for the static values and the other type is for referring to some information in the ‘knowledge’ object. For example, the ‘MEAL_TIME’ slot in the the ‘FOOD’ topic frame could take some values from a list of static values:

“MEAL_TIME”: [“Breakfast”, “Dinner”, “Lunch”].

On the other hand, the ‘DISH’ slot in the same frame has below the object as its value:

"DISH": [{ "slot": "NAME", "source": "DISH", "type": "knowledge" }],

which means that the list of ‘NAME’ values of ‘DISH’ objects in ‘knowledge’ part should be considered as the candidate values for the ‘DISH’ slot in the frame.

7. Frequently Asked Questions (FAQ)

7.1. *Do I or my company need to pay a license fee for getting the TourSG dataset?*

No, the TourSG dataset will be provided under a free of charge end-user-license to all participants in the Fourth Dialog State Tracking Challenge.

7.2. *Can I get the TourSG dataset without participating in the Challenge?*

Yes, but no free of charge end-user-license is available to non-participants. You or your company will need to pay a license fee for getting the TourSG dataset without participating in the Fourth Dialog State Tracking Challenge.

7.3. *Is participation in the main task of the Challenge mandatory?*

Yes, participation in the main task of the Challenge is mandatory for registered participants.

7.4. *Are baseline systems and evaluation scripts going to be provided?*

A baseline system and evaluation scripts will be provided only for the main task of the Challenge. Baselines and evaluation protocols for pilot tasks and open track are to be agreed directly with participants on such tasks.

7.5. *Is participation in the pilot tasks and open track of the Challenge mandatory?*

No, participation in the pilot tasks and open track of the Challenge is optional for registered participants.

7.6. *Do I need to participate in the first three pilot tasks in order to be able to participate in the "end-to-end system" task?*

No, but you need that at least one participant participates in each of the other pilot tasks. Otherwise you will not be able to set an end-to-end system, as no baselines are provided for any of the pilot tasks.

8. Subscription to mailing list

To join the mailing list, send an email to listserv@lists.research.microsoft.com with 'subscribe DSTC' in the body of the message (without quotes). Joining the list is encouraged for those with an interest in the challenge, and is a necessity for those participating.

Post to the list using the address: dstc@lists.research.microsoft.com.

9. Committees

9.1. *Organizing Committee*

Seokhwan Kim - I2R A*STAR
Luis F. D'Haro - I2R A*STAR
Rafael E Banchs - I2R A*STAR
Matthew Henderson - Google
Jason Williams - Microsoft Research

9.2. *Advisory Committee*

Paul Crook - Microsoft Research
Maxine Eskenazi - Carnegie Mellon University
Milica Gasic - University of Cambridge
Sungjin Lee - Carnegie Mellon University
Oliver Lemon - Herriot Watt
Olivier Pietquin - SUPELEC
Joelle Pineau - McGill University
Steve Young - University of Cambridge

10. References

[1] AW Black, S Burger, A Conkie, H Hastie, S Keizer, O Lemon, N Merigaud, G Parent, G Schubiner, B Thomson, JD Williams, K Yu, SJ Young, and M Eskenazi. Spoken dialog challenge 2010: Comparison of live and control test results. In In Proceedings 12th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL), Portland, USA, 2011.

[2] J. D. Williams. (2012) A belief tracking challenge task for spoken dialog systems. NAACL Workshop on Future directions and needs in the Spoken Dialog Community: Tools and Data. NAACL 2012.

[3] Jason D Williams, Antoine Raux, Deepak Ramachandran, and Alan Black. The dialog state tracking challenge. In In Proceedings 14th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL), Metz, France, 2013.

[4] Matthew Henderson, Blaise Thomson, and Jason Williams. The Second Dialog State Tracking Challenge. In Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue, p. 263. 2014.

[5] Henderson, Matthew, Blaise Thomson, and Jason Williams. The Third Dialog State Tracking Challenge. In Proceedings of IEEE Spoken Language Technology (2014).

[6] DSTC1 website: http://research.microsoft.com/en-us/events/dstc/

[7] DSTC 2 and 3 website: http://camdial.org/~mh521/dstc/