A Unifying View On Task-oriented Dialogue Annotation

Vojtěch Hudeček
Léon-Paul Schaub
Daniel Štancík
Patrick Paroubek
Ondřej Dušek

hudecek@ufal.mff.cuni.cz
schaub@lisn.fr
stancl@ufal.mff.cuni.cz
pap@lisn.fr
odusek@ufal.mff.cuni.cz

Overview

- Unifying annotation & ontologies in 4 task-oriented dialogue datasets:
  - MultiWOZ, SGD, DSTC2, CamRest
  - one of the largest annotated sets to date
- Data analysis and visualization
- Baseline model training and comparison

Success/failure analysis

Conditional entropy

Analysis

- 3 dialogue phases (based on conditional entropy)
  - information growth, stagnation, information deprecation
- Human-human vs. human-machine: entropy evolution differs
- Most dialogue failures due to missing information
  - Recoverable
  - It is correlated with entropy evolution.
- Best model = trained on full data
  - SGD → better BLEU
  - MultiWOZ → better state tracking

Data Statistics

<table>
<thead>
<tr>
<th></th>
<th>SGD</th>
<th>MultiWOZ</th>
<th>DSTC2</th>
<th>Camrest</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domains</td>
<td>18</td>
<td>7</td>
<td>1</td>
<td>1</td>
<td>19</td>
</tr>
<tr>
<td>Slots</td>
<td>145</td>
<td>29</td>
<td>10</td>
<td>7</td>
<td>166</td>
</tr>
<tr>
<td>Dialogues</td>
<td>22.8k</td>
<td>10.4k</td>
<td>3.2k</td>
<td>700</td>
<td>37.1k</td>
</tr>
<tr>
<td>Avg. utt. length</td>
<td>9.9</td>
<td>13.2</td>
<td>8.5</td>
<td>10.7</td>
<td>10.5</td>
</tr>
<tr>
<td>Entropy</td>
<td>4.8</td>
<td>4.4</td>
<td>2.1</td>
<td>3.0</td>
<td>4.8</td>
</tr>
</tbody>
</table>

Example

dialogue_id: MUL0674.json, original_dataset: multiwoz

domains: train, hotel

goal: hotel: { book: { day: thursday, people: 7, stay: 3 } info: { name: aylesbry lodge guest house } }

train: { book: { people: 7 }, info: { arriveBy: ... } }

message: You are looking for information in Cambridge. You are looking for a train. The train should arrive by ...

uttentances: ...

[ ... actor: ... I am looking for a train arriving by 21:45 and departing from cambridge ... turn: 1, intent: train ],
[ ... actor: ... What day are you making this trip, and where would you like to travel to? ... turn: 1, intent: train ],

state: train-day: Sunday, train-departure: cambridge, train-destination: kings lynn, train-end_time: 21:45

turn: 2

intent: train, ...

[ actor: system, utternance: Train TR1600 leaves Cambridge at 20:58 and arrives in kings lynn at 21:45. Would that work? delexicalization: [day] = thursday, [people] = 7, ...

nlu: Train-inform([day]=thursday, [destination]=kings lynn)

... & general-require()]

Baseline results

Baseline results

<table>
<thead>
<tr>
<th></th>
<th>SGD</th>
<th>DIASER</th>
<th>GPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLEU</td>
<td>0.9</td>
<td>0.9</td>
<td>0.9</td>
</tr>
<tr>
<td>Slot F1</td>
<td>0.9</td>
<td>0.9</td>
<td>0.9</td>
</tr>
</tbody>
</table>

Joint-goal accuracy

- Train: 4 subcorpora mixes
- Eval.: SGD MW DIASER

Supported by: AXID and the ANRT CIFRE #2017/1542, Human-E-IA-Net project / EC Horizon 2020, Grant Agreement H2020-FETFLAG-2018-2020 no-952026 and Charles University projects PRIMUS/19/SCI/10, GA UK No-382120 and SVV No-260575. Using resources provided by the LINDAT/CLARIAH-CZ Research Infrastructure (Czech Ministry of Education, Youth and Sports project No-1LM2018701)

Presented at LREC 2022, Marseille.

https://github.com/ufal/diaser