Goal-Oriented Chatbot Dialog Management Bootstrapping with Transfer Learning

Vladimir Ilievski, Claudiu Musat, Andreea Hossmann, Michael Baeriswyl

IJCAI, Stockholm, July 2018
Agenda

1. Key elements of Goal-Oriented Chatbots

2. Problem statement and transfer learning solution

3. Model

4. Transfer Learning

5. Experiments and Results
## Key Elements of the Goal-Oriented (GO) Chatbots

<table>
<thead>
<tr>
<th>Domain</th>
<th>Slots And Intents</th>
<th>Predefined Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predefined domain of expertise:</td>
<td>User intent:</td>
<td>Remembering user’s choices.</td>
</tr>
<tr>
<td>• movie booking</td>
<td>• inform</td>
<td>Driving the conversation with towards achieving the goal.</td>
</tr>
<tr>
<td>• restaurant booking</td>
<td>• request</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Slots or intent parameters:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• date: tomorrow</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• count: 2 people</td>
<td></td>
</tr>
</tbody>
</table>
Paradigms of implementations

**Fully-Supervised**
- Sequence-to-Sequence Fashion
- Encode a user request and its context
- Decode a bot answer
- Mimicking an expert
- Require huge amounts of data

**Reinforcement Learning**
- Based on Deep Q-Nets (DQN)
- Simulate conversation
- Explore the dialogue space
- Limited number of dialogue turns
- Require less data

Our choice
Problem: Limited Data

Challenge
Non trivial data requirements
Limited in-domain data
Obtaining in-domain data is hard

Solution
Leverage the domain similarity
Use *Transfer Learning*
Use less data
Solution: Transfer Learning

Source Domain
User Utterance
Which *theater* can I book *3 tickets* for *Titanic*?
request(theater, num_people=3, movie=Titanic)

Domain: 
Movie 
Booking

Train with full dataset

Bot 
What *date* are you interested in?
request(date)

Target Domain
User Utterance
Which *restaurant* can I book for *2 people*?
request(restaurant, num_people=2)

Domain: 
Restaurant 
Booking

Train with less data & better performances

Bot 
What *date* are you interested in?
request(date)
Goal-Oriented Dialog

• At time $t$:
  • given the user utterance $u_t$
  • the system replies with action $a_t$

• User utterance:
  • user’s intent (e.g. inform, request info)
  • intent parameters or slots (e.g. date: today)

• System action:
  • Request a value for empty slot
  • Suggest a value based on a Knowledge Base
Goal-Oriented Dialog

• The entire dialog: slot-value pairs called semantic frames

• Two levels of execution:
  • Semantic level
  • Natural language level
Model

User Goal

inform_slots:
{
    movie_name: "Titanic",
    number_of_people: "3",
    date: "tomorrow"
},

request_slots:
{
    city,
    theater,
    start_time
}
\[ L(\theta) = \mathbb{E}_{s_t, a_t, r_t, s_{t+1}} \left[ \left( r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1} | \theta') - Q(s_t, a_t | \theta) \right)^2 \right] \]
No Transfer Learning

No shared slots and actions - no shared weights
With Transfer Learning

Shared slots and actions - shared weights

With Transfer Learning

Shared slots and actions - shared weights
Experiments

**Data**

Two hypotheses:
- Train with less data - compare success rate
- Train faster - compare learning rate

Pair of Domains:
- Source Domain
- Target Domain

For each domain:
- 120 training user goals
- 32 testing user goals

Two models:
- Transfer learning model
- Model from scratch

**Flow**

1. Train on Source Domain
2. Transfer Learning on Target Domain
3. Train on Target Domain
4. Compare Performance
Domain Cases

1. Domain Overlapping:
   • Source Domain: Movie Booking
   • Target Domain: Restaurant Booking

2. Domain Extension:
   • Source Domain: Restaurant Booking
   • Target Domain: Tourist Info
Train With Less Data

- For both models we do 100 iterations of:
  - Splitting the data set in portions: 5, 10, 20, 30, 50 and 120
  - Warm-start both models
  - Train on each subset and test on the test set of 32 user goals
  - Report the training and testing success rates
Train With Less Data - Results

Domain Overlapping

Training Performance Over User Goals

Testing Performance Over User Goals

Domain Extension

Training Performance Over User Goals

Testing Performance Over User Goals
Faster Learning

• For both models we do 100 iterations of:
  • Train using the full set of 120 user goals
  • Test on the set of 32 testing user goals
  • Transfer learning model: does not take warm-starting
  • Model from scratch: takes warm-starting
  • Report learning curve
Faster Learning - Results

Domain Overlapping

Learning curve over training data set
- blue: transfer learning, no warm-start
- red: no transfer learning, warm-start

Learning curve over testing data set
- blue: transfer learning, no warm-start
- red: no transfer learning, warm-start

Domain Extension

Learning curve over training data set
- blue: transfer learning, no warm-start
- red: no transfer learning, warm-start

Learning curve over testing data set
- blue: transfer learning, no warm-start
- red: no transfer learning, warm-start
Conclusion

• Training GO Chatbots with less data

• Better performances

• Faster learning
Thanks for your attention

Questions?