End-to-End Joint Learning of Natural Language Understanding and Dialogue Manager

XUESONG YANG  YUN-NUNG (VIVIAN) CHEN  DILEK HAKKANI-TÜR

ICASSP 2017  March 9th

HLT-L2: Spoken Language Understanding I

xyang45@illinois.edu  y.v.chen@ieee.org
What is a Dialogue System?

A dialogue system is a computer agent that interacts with human via natural languages.

Category:
- Chit Chat
- Task-Oriented

Pipelined Task-Oriented Dialogue System

Automatic Speech Recognition

Hypothesis
Any action movies recommended this weekend?

Text-To-Speech

Text response
Which theater do you prefer?

Natural Language Generation (NLG)

System Action/Policy
request_location

Natural Language Understanding (NLU)
- User Intent Detection
- Slot Filling

Semantic Frame
request_movie
(genre=action, date=this weekend)

Dialogue Management (DM)
- Dialogue State Tracking (DST)
- Dialogue Policy Decision

Motivation: The pipelined system (NLU → DM) results in error propagation issues.
Proposed Approach

- End-to-end model
  - Mitigate the effects of noisy output from NLU
  - Refine NLU by supervised signals from DM
- Multi-task jointly learning
  - NLU - User intent classification
  - NLU - User slot tagging
  - DM - System action prediction
- Contextual understanding
  - Access to the user history
  - Monitor user behavior states over turns
Hi, how may I help you?

Are there any cheap rate hotels to put my bags?

Do you want to have a backpack type of hotel?

Yes. Just gonna leave our things there and stay out the whole day.

So you don’t mind if it is not roomy, right?

Okay. These hotels are available for you: ...

Thanks, goodbye.

Yes.

Ok, thank you, bye!

Idea: predicting the next system action given the current user utterance together with the aggregated observations
Natural Language Understanding (NLU)

**Utterance:** BOS % um how much is a taxi cab there ? EOS

**Slot Tags:**

- B-det_PRICE
- I-det_PRICE
- B-trsp_TYPE
- I-trsp_TYPE
- B-area_CITY

**User Intents:** QST_HOW_MUCH; QST_INFO

**System Actions:**

- RES_EXPLAIN
- RES_INFO
- FOL_EXPLAIN
- FOL_HOW_MUCH
- FOL_INFO

---

**Task 1: Slot Tagging**

**Task 2: Multi-Label User Intent Prediction**

\[
h_i(nlu) = h_T^{2(nlu)}, \quad h_t^{2(nlu)} = \mathcal{H}(h_{t-1}^{2(nlu)}, h_t^{1(i)}, h_t^{1(i)})
\]

\[
\begin{align*}
\overrightarrow{h_t^{1(i)}} &= \mathcal{H}(w_t, \overrightarrow{h_{t-1}^{1(i)}}), \\
\overleftarrow{h_t^{1(i)}} &= \mathcal{H}(w_t, \overleftarrow{h_{t+1}^{1(i)}}), \\
y_t^{(tag_i)} &= \arg \max \left( \text{softmax} \left( \overrightarrow{W_{hy}^{(tag)}} h_t^{1(i)} + \overleftarrow{W_{hy}^{(tag)}} h_t^{1(i)} \right) \right)
\end{align*}
\]

\[
h_t^{2(int_i)} = \mathcal{H}(h_{t-1}^{2(int_i)}, \overrightarrow{h_t^{1(i)}}, \overleftarrow{h_t^{1(i)}})
\]

\[
p_t^{(int_i)} = \text{sigmoid} \left( W_{hy}^{2(int_i)} h_t^{1(i)} \right)
\]

\[
y_t^{(int_i)} = \begin{cases} 1, & p_t^{(int_i)} \geq \text{threshold} \\ 0, & \text{otherwise} \end{cases}
\]
**NLU+DM 1: Pipelined BLSTMs**

**Utterance:** BOS % um how much is a taxi cab there ? EOS

**Slot Tags:** 0 0 0 B-det_Price I-det_Price O O B-trsp_Type I-trsp_Type B-area_City O O

**User Intents:** QST_HOW_MUCH; QST_INFO

**System Actions:** RES_EXPLAIN; RES_INFO; FOL_EXPLAIN; FOL_HOW_MUCH; FOL_INFO

---

**Task 1+2: Natural Language Understanding**

---

**Task 3: Multi-Label System Action Prediction**

---

**System Actions at j+1**

---
NLU+DM 2: End-to-End Model (JointModel)

**Utterance:** BOS % um how much is a taxi cab there? EOS

**Slot Tags:** 0 0 0 B-det_PRICE I-det_PRICE O 0 B-trsp_TYPE I-trsp_TYPE B-area_CITY O 0

**User Intents:** QST_HOW_MUCH; QST_INFO

**System Actions:** RES_EXPLAIN; RES_INFO; FOL_EXPLAIN; FOL_HOW_MUCH; FOL_INFO

**Task 1+2+3: End-to-End Joint NLU+DM**

supervised learning with three tasks

DM output signal refines NLU for better robustness
Data

- Dialogue State Tracking Challenge 4
  - Human-human dialogues: 21-hour dialogue sessions on touristic information collected via Skype between tour guides and tourists

<table>
<thead>
<tr>
<th>Domains</th>
<th>Speech Act</th>
<th>Speech Act Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accommodation</td>
<td>QST (QUESTION)</td>
<td>ACK, CLOSING, COMMIT, THANK</td>
</tr>
<tr>
<td>Attraction</td>
<td>RES (RESPONSE)</td>
<td>CANCEL, CONFIRM, ENOUGH, WHAT</td>
</tr>
<tr>
<td>Food</td>
<td>INI (INITIATIVE)</td>
<td>EXPLAIN, HOW_MUCH, HOW_TO, WHEN</td>
</tr>
<tr>
<td>Shopping</td>
<td>FOL (FOLLOW)</td>
<td>INFO, NEGATIVE, OPENING, WHERE</td>
</tr>
<tr>
<td>Transportation</td>
<td></td>
<td>POSITIVE, PREFERENCE, RECOMMEND, WHO</td>
</tr>
</tbody>
</table>

User Intent = Speech Act + Attributes
System Action = Speech Act + Attributes

<table>
<thead>
<tr>
<th></th>
<th>Train</th>
<th>Dev</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>#utter</td>
<td>5,648</td>
<td>1,939</td>
<td>3,178</td>
</tr>
<tr>
<td>#intent</td>
<td>68</td>
<td>54</td>
<td>58</td>
</tr>
</tbody>
</table>
**DM Result – System Action Prediction (SAP)**

- **Metric:** frame-level accuracy (FrmAcc)

<table>
<thead>
<tr>
<th>Model</th>
<th>FrmAcc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline (CRF+SVMs)</td>
<td>7.7</td>
</tr>
<tr>
<td>Pipeline-BLSTM</td>
<td>12.0</td>
</tr>
<tr>
<td>JointModel</td>
<td>22.8</td>
</tr>
</tbody>
</table>

- **Pipeline-BLSTM** and **JointModel** outperform the baseline

- **JointModel** improves **Pipeline-BLSTM** about 10% accuracy, indicating the importance of mitigating downside of pipeline

Human-human conversations are complicated, so predicting system actions for DM is difficult.
DM Result – System Action Prediction (SAP)

- Metric: frame-level accuracy (FrmAcc)

<table>
<thead>
<tr>
<th>Model</th>
<th>FrmAcc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline (CRF+SVMs)</td>
<td>7.7</td>
</tr>
<tr>
<td>Pipeline-BLSTM</td>
<td>12.0</td>
</tr>
<tr>
<td>JointModel</td>
<td>22.8</td>
</tr>
<tr>
<td>Oracle-SAP (SVM)</td>
<td>7.7</td>
</tr>
<tr>
<td>Oracle-SAP (BLSTM)</td>
<td>19.7</td>
</tr>
</tbody>
</table>

**Oracle models** show the upper-bound of the SAP performance, since it transfers the errors from NLU to SAP.

Contextual user turns make significant contribution to DM performance.

**JointModel** achieves the best DM performance (FrmAcc) with richer latent representations.
NLU Result – Slot Filling & Intent Prediction

- Metrics: frame-level accuracy (FrmAcc)

<table>
<thead>
<tr>
<th>Models</th>
<th>Slot</th>
<th>Intent</th>
<th>NLU</th>
</tr>
</thead>
<tbody>
<tr>
<td>NLU-Baseline (CRF+SVM)</td>
<td>77.3</td>
<td>37.2</td>
<td>33.1</td>
</tr>
<tr>
<td>NLU-Pipeline-BLSTM</td>
<td>76.8</td>
<td>40.0</td>
<td>36.4</td>
</tr>
<tr>
<td>NLU-JointModel</td>
<td>76.5</td>
<td>42.2</td>
<td>37.4</td>
</tr>
</tbody>
</table>

CRF+SVMs baseline maintains strong NLU performance with 33.1%

Pipeline-BLSTM and JointModel outperformed the baseline

Extra supervised DM signal helps refine the NLU by back-propagating the associated errors

DM signal (system action prediction) helps more on user intent prediction than slot filling, and NLU is significantly improved
Conclusion

- First propose an end-to-end deep hierarchical model for joint NLU and DM with limited contextual dialogue memory
- Leverage multi-task learning using three supervised signals
  - NLU: User intent classification
  - NLU: Slot tagging
  - DM: System action prediction
- Outperform the state-of-the-art pipelined NLU and DM models
  - Better DM due to the contextual dialogue memory
  - Robust NLU fine-tuned by supervised signal from DM
Thanks for Your Attention! 😊

Code Available at
https://github.com/XuesongYang/end2end_dialog
Appendix - Baseline Model

Predicting system actions at the next turn as responses to the current user behaviors by pipelining NLU and SAP together

NLU: CRF for slot tagging, and One-Vs-All SVMs for intent classification
SAP: One-Vs-All SVMs

Appendix: Configuration

- **Optimizer:** a mini-batch stochastic gradient descent method Adam
- **Contextual history:** five user turns
- **Dimension of word embedding:** 512
- **Dropout ratio:** 0.5
- **No early stopping, but use 300 training epochs**
- **Best models for three tasks are selected individually under different metrics**
  - Token-level micro-average F1 score is used for slot filling
  - Frame-level accuracy (it counts only when the whole frame parse is correct) is used for both user intent prediction and system action prediction
  - The decision thresholds are tuned on dev set