Dialog State Tracking and Action Selection Using Deep Learning Mechanism for Interview Coaching

**Ming-Hsiang Su**, Kun-Yi Huang, Tsung-Hsien Yang, Kuan-Jung Lai and Chung-Hsien Wu

Department of Computer Science and Information Engineering National Cheng Kung University, Taiwan
Outline

- Introduction
  - Motivation and goal
- Data collection and annotation
- System framework
  - Word embedding
  - Dialog state tracking
  - Dialog action selection
- Experimental results
- Conclusions
Introduction

- While in school, students are busy in studying for a better future.
  - Students rarely have the opportunity to practice interview for work or study.
- If students are afraid or nervous during the interview,
  - they can not answer questions asked by the interviewers properly.
Motivation and Goal

- In this study, we proposed a **coaching system** to improve user’s interview skills.
Motivation and Goal

- How to select a **proper question** plays an important role in a coaching system.
  - **Dialog state tracker** and **dialog action selection** model need to be constructed first.
  - According to the information provided, a coaching system can select a proper question to ask users.
Data collection and annotation

- Interview dialog corpus collection:
  - The domain of the corpus is chosen as the College Admission Interview.
  - 12 participants were invited.
  - During corpus collection, two participants completed the interview without prior design questions and answers.
  - A total of 75 dialogs with 540 question and answer pairs were collected.
  - Average number of sentences for each answer is 3.95.
According to the collected corpus, 6 categories and 10 semantic slots were defined.

<table>
<thead>
<tr>
<th>Category</th>
<th>Semantic Slot</th>
</tr>
</thead>
<tbody>
<tr>
<td>Studying experiences</td>
<td>Community and cadres</td>
</tr>
<tr>
<td></td>
<td>Score and other achievement</td>
</tr>
<tr>
<td>Interests, and strengths and weaknesses</td>
<td>Interest</td>
</tr>
<tr>
<td></td>
<td>Strength and weakness</td>
</tr>
<tr>
<td>Learning motivation and future prospects</td>
<td>Motivation</td>
</tr>
<tr>
<td></td>
<td>Reading plans and future plans</td>
</tr>
<tr>
<td>Domain knowledge</td>
<td>Professional field and curriculum</td>
</tr>
<tr>
<td></td>
<td>Programming language and specialized terms</td>
</tr>
<tr>
<td>Personality trait</td>
<td>Personality</td>
</tr>
<tr>
<td>Others</td>
<td>Others</td>
</tr>
</tbody>
</table>
System Framework

Training phase

- Interview Corpus
  - Word Embedding Model Training
  - LSTM+ANN-based State Tracking Model Training

Testing phase

- Answer
  - Word Embedding
  - Dialog State

Dialog States

- Dialog State Classification Model Training
  - Dialog Action Selection Model Training
  - RL-based Action Selection Model Training
  - Dialog State Classification Model

- Interview Question Selection
  - Interview Question
Word embedding model

- Each word is mapped to its corresponding word vector $w_i$ by using word2vec.
  - Word2vec creates vectors that are distributed numerical representations of word features, such as the context of individual words.
  - The purpose and usefulness of word2vec is to group the vectors of similar words together in the vector space.
  - Word2vec encodes each word in a vector and trains words against other words that neighbor them in the input corpus.
This study uses the Skip-gram model.

- We use **Chinese Gigaword corpus** to train word2vector.
- Totally, **42619** words were obtained.
- The word vector is connected to the vector representation of the sentence.
Dialogue State Tracking

- Sentence and answer hidden vector representation:
  - Considering semantic representation of each sentence in an answer.

\[ \text{Word Vectors: } v_1, v_2, \ldots, v_n \]

\[ \text{Word embedding: } w_1, w_2, \ldots, w_n \]

\[ \text{Sentence LSTM: } h \]

\[ \text{Sentence Hidden Vector: } s_1, s_2, \ldots, s_m \]

\[ \text{Answer LSTM: } h \]

\[ \text{Answer Hidden Vector: } u_1, u_2, \ldots, u_m \]
Dialog state representation:
- Using 10 ANN models, each for one slot.
- The input of ANNs is the answer hidden vector.
- Compose all slot values to form a dialog state
Dialog Action Selection

- Training action selection model and question generation model.
Dialog Action Selection

- Deep Q network Pre-training

Interview Dialogue Corpus

(sates, actions)

Restrict input state and only learn the Q-value with the corresponding action.

Deep Q-network

\[ Q(s_t, a_t, \theta) \]
Experimental setup

- Action selection model
  - **Reward curve** was approximate to 1
    - Deep Q-network nodes: 100
    - Mini-batch: 64
    - Experience size: 10000
    - Number of Training Simulated Dialogs: 1000

![Avg. Reward curve](Avg. Reward curve)
Experimental results (1)

- Evaluation the action selection model with different reward function
  - 1-0 reward function:
    \[ r_t^i = \begin{cases} 
    1, & \text{if } Na_t^i = 0 \text{ or } 1 \\
    0, & \text{otherwise} 
  \end{cases} \]
  - Number of completed actions:
    \[ r_t^i = \begin{cases} 
    AR, & \text{if } Na_t^i < 2 \\
    0, & \text{otherwise} 
  \end{cases} \]
  - SimpleDS reward function:
    \[ r_t^i = CR \times 0.5 + DR \times 0.5 \]

- $Na_t$ is the number of the $i$-th action which has been positively confirmed at time $t$
- $AR$ is the number of actions positively confirmed divided by the actions to confirm
- $CR$ is the number of slots positively confirmed divided by the slots to confirm, $DR$ as $p(a_t|s_t)$

<table>
<thead>
<tr>
<th>Reward function</th>
<th>Slots-turns ratio (slots/turn)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-0 reward</td>
<td>0.92</td>
</tr>
<tr>
<td>Completed actions</td>
<td>0.89</td>
</tr>
<tr>
<td>SimpleDS</td>
<td>0.86</td>
</tr>
</tbody>
</table>
Experimental results (2)

- We analyze the average number of completed slots, the average dialog turns and their ratio with different architectures.
- Testing with 1000 simulated dialogs.

<table>
<thead>
<tr>
<th></th>
<th>Avg. Number of Completed slots</th>
<th>Avg. turns</th>
<th>Standard Deviation (turns)</th>
<th>Slots-turns ratio (slots/turn)</th>
</tr>
</thead>
<tbody>
<tr>
<td>w-Restriction + w/Pre-training</td>
<td>7.77</td>
<td>8.44</td>
<td>0.75</td>
<td>0.92</td>
</tr>
<tr>
<td>w-Restriction + w/o-Pre-training</td>
<td>7.74</td>
<td>8.51</td>
<td>0.74</td>
<td>0.91</td>
</tr>
<tr>
<td>w/o-Restriction + w/Pre-training</td>
<td>7.04</td>
<td>8.40</td>
<td>4.04</td>
<td>0.84</td>
</tr>
<tr>
<td>w/o-Restriction + w/o-Pre-training</td>
<td>7.82</td>
<td>10.56</td>
<td>2.63</td>
<td>0.74</td>
</tr>
</tbody>
</table>

w : with, w/o : without
Experimental results (3)

- Question selection model evaluation:
  - We use questionnaire to evaluate Naturalness and Utility
    - Naturalness: The semantic content of question from system is ideal
    - Utility: The system flow is applicable
  - 5 subjects were invited
  - 5-level Likert scale

<table>
<thead>
<tr>
<th></th>
<th>Mean</th>
<th>Standard Deviation</th>
<th>t-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Naturalness</td>
<td>3.80</td>
<td>0.447</td>
<td>4.00**</td>
</tr>
<tr>
<td>Utility</td>
<td>3.60</td>
<td>0.548</td>
<td>2.45**</td>
</tr>
</tbody>
</table>
Conclusions

- We propose an approach for dialog state tracking and dialog action selection in an interview conversation.
  - The word2vec model is employed for word distributed representation.
  - The LSTM+ANN-based model is used to predict dialog states.
  - The deep RL-based model is used to learn the relation between dialog states and actions.
- In the future, a richer interview corpus and a robust DST model are helpful to improve system performance.
Questions?