Abstract

This work focuses on incorporating sentiment information into task-oriented dialogue systems. Current end-to-end approaches only consider user semantic inputs in learning and under-utilize other user information. But the ultimate evaluator of dialogue systems is the end-users and their sentiment is a direct reflection of user satisfaction and should be taken into consideration. Therefore, we propose to include user sentiment obtained through multimodal information (acoustic, dialogic and textual) in the end-to-end learning framework to make systems more user-adaptive and effective. We incorporated user sentiment information in both supervised and reinforcement learning settings. In both settings, adding sentiment information reduced the dialogue length and improved the task success rate on a bus information search task.

Multimodal Sentiment Detector

We manually annotated 50 dialogs with 517 conversation turns to train this sentiment detector. The annotated set is open to public.

Prediction made by the detector will be used in the supervised learning and reinforcement learning.

Three sets of features: 1) Acoustic features; 2) Dialogic features; 3) Textual features.

Dialogic features include: 1) Interruptions; 2) Button usage; 3) Repetitions; 4) Start over. These four categories of dialog features are chosen based on the previous literature and the observed statistics in the dataset.

Model Architecture

A sentiment detector is built on an annotated subset and is used to predict sentiment labels and sentiment scores for the supervised and reinforcement learning.

Supervised learning uses the predicted sentiment labels from the sentiment detector as additional context features for the training.

Reinforcement learning simulates the dialogs and uses the predicted sentiment scores from the sentiment detector as immediate rewards to guide the training.

The whole model is end-to-end trainable and user-adaptive.

Reinforcement Learning

User simulator

- Reinforcement learning requires feedback from the environments. So we created a user simulator and simulated user sentiment by sampling from the real data.
- Summary statistics, e.g. how many times one entity has been asked, are used to compare different dialogs.

Sentiment scores used in the reward functions

- Four different rewards functions with sentiment scores: 1) baseline; 2) SRRS: baseline + sentiment score from random samples; 3) SRRP: baseline + penalty for repetitions; 4) SRRP: baseline + penalties for both repetition and interruption.
- Dialog length: By adapting to user sentiment, all models with sentiment reward reduces the average dialogue length.
- Success rate: SRRP performs the best. By adding penalties, the model covers more data points, and improves the success rate and convergence speed.

Dataset & Discussion

Dataset: DSTC1, a bus information search task.
1) Sentiment Detector: trained with a subset of 50 dialogs from DSTC1, with sentiment annotated under context.
2) Supervised Learning: trained and tested with the entire DSTC1 set, with sentiment features predicted by the sentiment detector.
3) Reinforcement Learning: dialogs of the same task are simulated. The user sentiments are simulated by sampling from a subset of DSTC1.

The learned dialogue policy is more sentiment adaptive

The intuition behind the good performance of models with user sentiment is that the model learns to adapt to user sentiment.

Further improvements
1) Include more channels, such as vision, to improve the sentiment detector;
2) Create a similarity measure for the dialogue states vectors and improve sentiment simulation
3) Reward shaping in reinforcement learning.

Conclusion

We proposed to detect user sentiment from multimodal channels and incorporate the detected sentiment as feedback into adaptive end-to-end dialogue system training.

We included sentiment information directly as a context feature in the supervised learning framework and used sentiment scores as immediate rewards in the reinforcement learning setting.

Experiments suggest that incorporating user sentiment is helpful in reducing the dialog length and increasing the task success rate in both SL and RL settings.

We believe this approach can be easily generalized to other domains given its end-to-end training procedure and task independence.

References


Table 1. Sentiment detector performance.

<table>
<thead>
<tr>
<th>Model</th>
<th>Avg. F-1</th>
<th>Std. F-1</th>
<th>Max F-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acoustic features only</td>
<td>0.635</td>
<td>0.027</td>
<td>0.648</td>
</tr>
<tr>
<td>Dialogic features only</td>
<td>0.596</td>
<td>0.001</td>
<td>0.596</td>
</tr>
<tr>
<td>Textual features only</td>
<td>0.664</td>
<td>0.010</td>
<td>0.665</td>
</tr>
<tr>
<td>Textual + Dialogic</td>
<td>0.672</td>
<td>0.011</td>
<td>0.700</td>
</tr>
<tr>
<td>Acoustic + Dialogic</td>
<td>0.680</td>
<td>0.019</td>
<td>0.707</td>
</tr>
<tr>
<td>Acoustic + Textual</td>
<td>0.647</td>
<td>0.023</td>
<td>0.666</td>
</tr>
<tr>
<td>Acoustic + Dialogic + Text</td>
<td>0.608</td>
<td>0.028</td>
<td>0.736</td>
</tr>
</tbody>
</table>

Table 2. Feature importance ranking.

<table>
<thead>
<tr>
<th>Model</th>
<th>Weighted F-1</th>
<th>Dialog Arc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCN</td>
<td>0.639</td>
<td>6.00%</td>
</tr>
<tr>
<td>HCN + raw dialogic features</td>
<td>0.619</td>
<td>5.79%</td>
</tr>
<tr>
<td>HCN + predicted sentiment</td>
<td>0.642</td>
<td>6.55%</td>
</tr>
</tbody>
</table>

Table 3. Supervised learning performance.

Table 4. RL convergent success rate.