Litebody And DTaSk

Daniel Schulman
Northeastern University

Conversational Agents for Behavioral Healthcare

• Goals:
  – Easily useable by anyone, anywhere.
  – Reusable for different applications.
  – Modular and modifiable.
Litebody and DTask

Litebody is a framework for developing and deploying lightweight embodied conversational agents on the web.

DTask is a dialogue manager designed for creating complex, system-directed dialogues which can be extended and reused.
Applications

• A reusable/extensible system for building ontology-based health behavior change interventions.
• Web-based at-home follow-up for recently discharged hospital patients.
• Health education for young women.

Demo

Relational Agent Loaded,
Press Start to Continue

start
The Litebody Suite

Server
- Text-to-Speech Engine
- Litebody Framework
- Dialogue Manager

Litebody Client
Litebody Client

- Implemented in Adobe Flash
- Less than ½ MB download
- Supports a core set of animations:
  - Eyebrow raises
  - Posture shifts
  - Headnod
  - Gaze towards/away
  - 3 facial expressions
  - 12 visemes

Litebody Client

- Input methods:
  - Multiple choice
  - Free text
  - Custom input widgets
- Displaying virtual documents and other artifacts
  - With deictic gestures
- Background images
The Litebody Suite

- Litebody is a modular, with several pieces that can be modified or replaced to suit different needs.

The Litebody Suite

- The TTS engine generates audio and animation timing.
  - Implementations:
    - Loquendo
      - Proprietary
      - High voice quality
    - FreeTTS
      - Open source
      - Lower voice quality
The Litebody Suite

- A dialogue manager is responsible for generating the agent’s utterances, and processing user input.
- Interfaces with Litebody through a simple automata interface.

DTask

- DTask is designed primarily for system-directed dialogue with multiple-choice input.
- Dialogue is specified as a hierarchical task decomposition.
  - Based on the CEA-2018 standard for XML description of a task decomposition, with some dialogue-specific extensions.
Dialogue as Task Decomposition

- Task: Conversation
- Subtasks:
  1. Opening
  2. Ritual Social
  3. Review Tasks
  4. Assessment
  5. Counseling
  6. Assign Tasks
  7. Pre-closing
  8. Closing

A conversation is divided into discourse segments.
- Each segment is a task which can be decomposed into subtasks.
- Any task can have multiple decompositions ("recipes").
Extensibility and Reuse

• This representation allows a separation of the high-level structure of a conversation from individual utterances.

Extensibility and Reuse

• This representation allows a separation of the high-level structure of a conversation from individual utterances.
• This allows easier reuse of effort:
  – We authored dialogue to perform a 60-day intervention for physical activity promotion.
Extensibility and Reuse

- This representation allows a separation of the high-level structure of a conversation from individual utterances.
- This allows easier reuse of effort:
  - We authored dialogue to perform a 60-day intervention for physical activity promotion.
  - Extended to promote fruit and vegetable consumption in 10% of the person-hours, with over 90% reuse of high-level recipes.

DTask Models: Task Recipes

```
<task id="Conversation">
  <input name="user" type="string"/>
  <input name="topic" type="string"/>
  <output name="liked" type="boolean"/>
</task>

<subtasks id="ShortConverse" goal="Conversation">
  <step name="greet" task="Greeting"/>
  <step name="ask" task="AskAbout"/>
  <step name="bye" task="Bye"/>
  <binding slot="$greet.user" value="$this.user"/>
  <binding slot="$ask.topic" value="$this.topic"/>
  <binding slot="$this.liked" value="$ask.positive"/>
</subtasks>
```
DTask Models: Dialogue

```xml
<task id="AskAbout">
  <input name="topic" type="string"/>
  <output name="positive" type="boolean"/>
</task>

<d:turn task="AskAbout">
  <applicable>$this.topic=='weather'</applicable>
  <d:agent>How's the weather out there today?</d:agent>
  <d:user>
    <d:say>Good.</d:say>
    <d:result slot="positive" value="true"/>
  </d:user>
  <d:user>
    <d:say>Terrible.</d:say>
    <d:result slot="positive" value="false"/>
  </d:user>
</d:turn>
```

Acknowledgements

- Grant funding from NIH National Library of Medicine.
- Litebody and DTask incorporate work from many past and current members of the Relational Agents Group at Northeastern University.
- DTask is partially based on code originally supplied by Charles Rich.
<table>
<thead>
<tr>
<th>Questions?</th>
</tr>
</thead>
<tbody>
<tr>
<td>• <a href="http://www.relationagents.com/litebody.html">www.relationagents.com/litebody.html</a></td>
</tr>
<tr>
<td>• <a href="http://litebodysuite.sourceforge.net">litebodysuite.sourceforge.net</a></td>
</tr>
<tr>
<td>• <a href="http://dtask.sourceforge.net">dtask.sourceforge.net</a></td>
</tr>
</tbody>
</table>