Learning Multimodal Clarification Strategies

Verena Rieser\textsuperscript{1}  Ivana Kruijff-Korbayová\textsuperscript{1}  Oliver Lemon\textsuperscript{2}

\textsuperscript{1}Department of Computational Linguistics, Saarland University, Germany

\textsuperscript{2}School of Informatics, University of Edinburgh, GB

In affiliation with: IGK and TALK Project

http://www.talk-project.org/
Outline

Motivation
The basic problem
Previous work

Framework
The Learning Approach
The Data Collection
Results

Performance modelling
RL and Performance modelling
Dialogue costs and multimodality
Modality costs and situations
Ambiguity and task success
Dialogue quality and “emotions"
Outline

Motivation
The basic problem
Previous work

Framework
The Learning Approach
The Data Collection
Results

Performance modelling
RL and Performance modelling
Dialogue costs and multimodality
Modality costs and situations
Ambiguity and task success
Dialogue quality and “emotions"
Outline

Motivation
The basic problem
Previous work

Framework
The Learning Approach
The Data Collection
Results

Performance modelling
RL and Performance modelling
Dialogue costs and multimodality
Modality costs and situations
Ambiguity and task success
Dialogue quality and “emotions"
Outline

Motivation
   The basic problem
   Previous work

Framework
   The Learning Approach
   The Data Collection
   Results

Performance modelling
   RL and Performance modelling
   Dialogue costs and multimodality
   Modality costs and situations
   Ambiguity and task success
   Dialogue quality and "emotions"
Clarification Requests in Multimodal Dialogue

User: Add “American Pie" to this list.

CRs:

Pardon?
Add what?
The album or the song?
By Madonna or Don McLean?
Any of the songs here?
Any of these playlists?
Clarification Requests in Multimodal Dialogue

User: Add "American Pie" to this list.

CRs:

Pardon?

Add what?

The album or the song?

By Madonna or Don McLean?

Any of the songs here?

Any of these playlists?
Clarification Requests in Multimodal Dialogue

User: Add “American Pie” to this list.

CRs:
- Pardon? no acoustic hypothesis
- Add what?
- The album or the song?
- By Madonna or Don McLean?
- Any of the songs here?
- Any of these playlists?
Clarification Requests in Multimodal Dialogue

User: Add “American Pie” to this list.

CRs:

Pardon?

Add what?

The album or the song?

By Madonna or Don McLean?

Any of the songs here?

Any of these playlists?
Clarification Requests in Multimodal Dialogue

User: Add “American Pie” to this list.

CRs: Pardon?
Add what? partial acoustic hypothesis
The album or the song?
By Madonna or Don McLean?
Any of the songs here?
Any of these playlists?
Clarification Requests in Multimodal Dialogue

User: Add “American Pie” to this list.

CRs:

Pardon?

Add what?

The album or the song?

By Madonna or Don McLean?

Any of the songs here?

Any of these playlists?
Clarification Requests in Multimodal Dialogue

User: Add “American Pie” to this list.

CRs:

Pardon?
Add what?
The album or the song? ambig lexical interpretation
By Madonna or Don McLean?
Any of the songs here?
Any of these playlists?
Clarification Requests in Multimodal Dialogue

**User:** Add “American Pie" to this list.

**CRs:**
- Pardon?
- Add what?
- The album or the song?
- By Madonna or Don McLean?
- Any of the songs here?
- Any of these playlists?
Clarification Requests in Multimodal Dialogue

User: Add “American Pie” to this list.

CRs:

Pardon?
Add what?
The album or the song?
By Madonna or Don McLean? ambiguous
reference
Any of the songs here?
Any of these playlists?
Clarification Requests in Multimodal Dialogue

User: Add "American Pie" to this list.

CRs:

Pardon?
Add what?
The album or the song?
By Madonna or Don McLean?
Any of the songs here?
Any of these playlists?
Clarification Requests in Multimodal Dialogue

User: Add “American Pie” to this list.

CRs:

Pardon?
Add what?
The album or the song?
By Madonna or Don McLean?
Any of the songs here? [display list]
Any of these playlists?
Clarification Requests in Multimodal Dialogue

User: Add “American Pie” to this list.

CRs:

Pardon?
Add what?
The album or the song?
By Madonna or Don McLean?
Any of the songs here? [display list] ambiguous reference
Any of these playlists?
Clarification Requests in Multimodal Dialogue

User: Add “American Pie” to this list.

CRs:

Pardon?
Add what?
The album or the song?
By Madonna or Don McLean?
Any of the songs here? [display list]
Any of these playlists?
Clarification Requests in Multimodal Dialogue

User: Add “American Pie” to this list.

CRs:

Pardon?
Add what?
The album or the song?
By Madonna or Don McLean?
Any of the songs here? [display list]
Any of these playlists? [display list]
Clarification Requests in Multimodal Dialogue

User: Add “American Pie” to this list.

CRs:

Pardon?
Add what?
The album or the song?
By Madonna or Don McLean?
Any of the songs here? [display list]
Any of these playlists? [display list] ambiguous reference
Clarification Requests in Multimodal Dialogue

User: Add “American Pie” to this list.

CRs:
- Pardon?
- Add what?
- The album or the song?
- By Madonna or Don McLean?
- Any of the songs here? [display list]
- Any of these playlists? [display list]

CRs indicate a problem with “understanding” (part of) an utterance.
Clarification Requests in Multimodal Dialogue

User: Add “American Pie" to this list.

CRs:
- Pardon?
- Add what?
- The album or the song?
- By Madonna or Don McLean?
- Any of the songs here? [display list]
- Any of these playlists? [display list]

CRs indicate a problem with “understanding" (part of) an utterance.

How to generate CRs indicating different types of errors?
Outline

Motivation
  The basic problem
  Previous work

Framework
  The Learning Approach
  The Data Collection
  Results

Performance modelling
  RL and Performance modelling
  Dialogue costs and multimodality
  Modality costs and situations
  Ambiguity and task success
  Dialogue quality and “emotions"
Generating CRs in task-oriented dialogues

[Rieser and Moore], ACL 2005: Implications for generating clarification requests in task-oriented dialogues.

- Form-function mappings
- Human decision making on function features was influenced by dialogue type, modality and channel quality.
Generating CRs in task-oriented dialogues

[Rieser and Moore], ACL 2005: Implications for generating clarification requests in task-oriented dialogues.

- Form-function mappings
  - Human decision making on function features was influenced by dialogue type, modality and channel quality.
Generating CRs in task-oriented dialogues

[Rieser and Moore], ACL 2005: *Implications for generating clarification requests in task-oriented dialogues.*

- **Form-function mappings**
  - We know how to generate surface forms of CRs once we have the functions

- Human decision making on function features was influenced by dialogue type, modality and channel quality.
Generating CRs in task-oriented dialogues

[Rieser and Moore], ACL 2005: *Implications for generating clarification requests in task-oriented dialogues.*

- **Form-function mappings**
  → We know how to generate surface forms of CRs once we have the functions

- Human decision making on function features was influenced by **dialogue type, modality and channel quality.**
Generating CRs in task-oriented dialogues

[Rieser and Moore], ACL 2005: *Implications for generating clarification requests in task-oriented dialogues.*

- **Form-function mappings**
  → We know how to generate surface forms of CRs once we have the functions

- Human decision making on function features was influenced by **dialogue type, modality and channel quality**.
  → We don’t know how to set function features in dialogue systems!
Outline

Motivation
   The basic problem
   Previous work

Framework
   The Learning Approach
   The Data Collection
   Results

Performance modelling
   RL and Performance modelling
   Dialogue costs and multimodality
   Modality costs and situations
   Ambiguity and task success
   Dialogue quality and “emotions"
Approach

Assumptions

- Clarification strategies involve complex decision making over a variety of contextual factors
- and exhaustive planning towards reaching a “goal".
Assumptions

• Clarification strategies involve **complex decision making over a variety of contextual factors**
• and **exhaustive planning towards reaching a “goal”**.
Approach

Assumptions

- Clarification strategies involve **complex decision making over a variety of contextual factors**
- and **exhaustive planning towards reaching a “goal”**.

→ Apply reinforcement learning (RL) in the information state update (ISU) approach.
Framework for learning multimodal CRs

1. Collect data on possible strategies in WOZ experiment.
2. Bootstrap an initial policy using supervised learning in the ISU approach.
3. Optimise the learnt policy for dialogue systems using reinforcement learning (RL).
Framework for learning multimodal CRs

1. Collect data on possible strategies in WOZ experiment.
2. Bootstrap an initial policy using supervised learning in the ISU approach.
3. Optimise the learnt policy for dialogue systems using reinforcement learning (RL).
Framework for learning multimodal CRs

1. Collect data on possible strategies in WOZ experiment.  
   → Identify possible state-action mappings
2. Bootstrap an initial policy using supervised learning in the ISU approach.
3. Optimise the learnt policy for dialogue systems using reinforcement learning (RL).
Framework for learning multimodal CRs

1. Collect data on possible strategies in WOZ experiment.  
   → Identify possible state-action mappings

2. Bootstrap an initial policy using supervised learning in the ISU approach.

3. Optimise the learnt policy for dialogue systems using reinforcement learning (RL).
1. Collect data on possible strategies in WOZ experiment.  
   → Identify possible state-action mappings
2. Bootstrap an initial policy using supervised learning in the ISU approach.  
   → Learn wizards’ decisions in context
3. Optimise the learnt policy for dialogue systems using reinforcement learning (RL).
Framework for learning multimodal CRs

1. Collect data on possible strategies in WOZ experiment.  
   → Identify possible state-action mappings

2. Bootstrap an initial policy using supervised learning in the ISU approach.  
   → Learn wizards’ decisions in context

3. Optimise the learnt policy for dialogue systems using reinforcement learning (RL).
Framework for learning multimodal CRs

1. Collect data on possible strategies in WOZ experiment.
   → Identify possible state-action mappings

2. Bootstrap an initial policy using supervised learning in the ISU approach.
   → Learn wizards’ decisions in context

3. Optimise the learnt policy for dialogue systems using reinforcement learning (RL).
   → How should the performance function (reward) look like?
Outline

Motivation
   The basic problem
   Previous work

Framework
   The Learning Approach
   The Data Collection
   Results

Performance modelling
   RL and Performance modelling
   Dialogue costs and multimodality
   Modality costs and situations
   Ambiguity and task success
   Dialogue quality and “emotions"
The SAMMIE-2\textsuperscript{1} Data Collection

Figure: Multimodal Wizard-of-Oz data collection setup for an in-car music player application, using the Lane Change driving simulator.
Experimental Setup

6 wizards, 24 subjects

**Wizard:**
- Screen output options pre-computed, wizard freely talking
- Wizard “sees what the system sees” (corrupted transcriptions) → “clarification pop-up"

**User:**
- User’s primary task is driving
- Secondary MP3 selection task:
  - (a) searching for a title either in the database or in an existing playlist
  - (b) building a playlist satisfying a number of constraints (“10 songs from the 70s”)
Wizards’ choice for graphical presentation (2 steps)

1. Choose content: album, tracks or artists.

2. Choose graphical presentations
Outline

Motivation
   The basic problem
   Previous work

Framework
   The Learning Approach
   The Data Collection

Results

Performance modelling
   RL and Performance modelling
   Dialogue costs and multimodality
   Modality costs and situations
   Ambiguity and task success
   Dialogue quality and “emotions"
Wizards’ Performance

- User Satisfaction fairly high across wizards (15.0, $\delta=2.9$, range 5 to 25)$^2$
  - “Most helpful" presentation strategy was showing a table with most information.
  - Graphical display was judged distracting the driver.
  - Amount of graphical information was judged too much while driving.

---

$^2$US as the sum of 5 different aspects probed by a survey following [Walker et al.], 2002.
Wizards’ Performance

- User Satisfaction fairly high across wizards (15.0, $\delta=2.9$, range 5 to 25)$^2$
- “Most helpful” presentation strategy was showing a table with most information.
- Graphical display was judged distracting the driver.
- Amount of graphical information was judged too much while driving.

$^2$US as the sum of 5 different aspects probed by a survey following [Walker et al.], 2002.
Wizards’ Performance

- User Satisfaction fairly high across wizards (15.0, $\delta=2.9$, range 5 to 25)$^2$
- “Most helpful” presentation strategy was showing a table with most information.
- Graphical display was judged distracting the driver.
- Amount of graphical information was judged too much while driving.

$^2$US as the sum of 5 different aspects probed by a survey following [Walker et al.], 2002.
Wizards’ Performance

- User Satisfaction fairly high across wizards (15.0, $\delta=2.9$, range 5 to 25)$^2$
- “Most helpful" presentation strategy was showing a table with most information.
- Graphical display was judged distracting the driver.
- Amount of graphical information was judged too much while driving.

$^2$US as the sum of 5 different aspects probed by a survey following [Walker et al., 2002.]
Consequences for Performance Modelling

- “Costs" caused by multi-modal dialogue acts.
- Vague task success by non directed task definition and high ambiguity.
- In-car environment: cognitive workload on primary task.
- All features should be available at runtime (RL).
Consequences for Performance Modelling

• “Costs" caused by multi-modal dialogue acts.
• Vague task success by non directed task definition and high ambiguity.
• In-car environment: cognitive workload on primary task.
• All features should be available at runtime (RL).
Consequences for Performance Modelling

- “Costs" caused by multi-modal dialogue acts.
- Vague task success by non directed task definition and high ambiguity.
- In-car environment: cognitive workload on primary task.
- All features should be available at runtime (RL).
Consequences for Performance Modelling

- “Costs" caused by multi-modal dialogue acts.
- Vague task success by non directed task definition and high ambiguity.
- In-car environment: cognitive workload on primary task.
- All features should be available at runtime (RL).
Outline

Motivation
   The basic problem
   Previous work

Framework
   The Learning Approach
   The Data Collection
   Results

Performance modelling
   RL and Performance modelling
   Dialogue costs and multimodality
   Modality costs and situations
   Ambiguity and task success
   Dialogue quality and “emotions”
Reinforcement Learning

The reward/performance function defines the “goal" of the RL agent.

Figure: [Sutton and Barto], 1998.
RL and PARADISE

RL and PARADISE


UserSatisfaction\(\text{max TaskSuccess, min Costs}\)
RL and PARADISE

UserSatisfaction(max TaskSuccess, min Costs)
Outline

Motivation
- The basic problem
- Previous work

Framework
- The Learning Approach
- The Data Collection
- Results

Performance modelling
- RL and Performance modelling
- Dialogue costs and multimodality
- Modality costs and situations
- Ambiguity and task success
- Dialogue quality and “emotions"
Dialogue costs and dialogue acts

**PARADISE:**

- turn duration, elapsed time, number of turns, . . .

**DATE:**

- accounts for relations between cost features and features indicating task success
- multiple views on one turn: *conversational domain, task/sub-task level, speech act*

Example: For certain speech acts turn duration is positively related to US [Walker and Passonneau, 2001) → present-info indicates task success
Dialogue costs and dialogue acts

**PARADISE:**
- turn duration, elapsed time, number of turns, . . .

**DATE:**
- accounts for relations between cost features and features indicating task success
- multiple views on one turn: *conversational domain, task/sub-task level, speech act*

Example: For certain speech acts turn duration is positively related to US [Walker and Passonneau, 2001] → present-info indicates task success
Dialogue costs and dialogue acts

**PARADISE:**
- turn duration, elapsed time, number of turns, ...

**DATE:**
- accounts for relations between cost features and features indicating task success
- multiple views on one turn: conversational domain, task/sub-task level, speech act

Example: For certain speech acts turn duration is positively related to US [Walker and Passonneau, 2001] → present-info indicates task success
Dialogue costs and dialogue acts

**PARADISE:**
- turn duration, elapsed time, number of turns, . . .

**DATE:**
- accounts for relations between cost features and features indicating task success
- multiple views on one turn: conversational domain, task/sub-task level, speech act

Example: For certain speech acts turn duration is positively related to US [Walker and Passonneau, 2001] → present-info indicates task success
Motivation

Framework

Performance modelling

Summary

Dialogue costs and dialogue acts

PARADISE:

- turn duration, elapsed time, number of turns, . . .

DATE:

- accounts for relations between cost features and features indicating task success
- multiple views on one turn: conversational domain, task/sub-task level, speech act

Example: For certain speech acts turn duration is positively related to US [Walker and Passonneau, 2001]

→ present-info indicates task success
Dialogue costs and dialogue acts

**PARADISE:**
- turn duration, elapsed time, number of turns, ...

**DATE:**
- accounts for relations between cost features and features indicating task success
- multiple views on one turn: *conversational domain, task/sub-task level, speech act*

**Example:** For certain speech acts turn duration is positively related to US [Walker and Passonneau], 2001)
  → *present-info indicates task success*
Costs of Multimodal Dialogue Acts

<table>
<thead>
<tr>
<th>ID</th>
<th>Utterance</th>
<th>Speaker</th>
<th>Modality</th>
<th>Speech act</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Please play “Nevermind”.</td>
<td>user</td>
<td>speech</td>
<td>request</td>
</tr>
<tr>
<td>2a</td>
<td>Does this list contain the song?</td>
<td>wizard</td>
<td>speech</td>
<td>request info</td>
</tr>
<tr>
<td>2b</td>
<td>[shows list with 20 DB matches]</td>
<td>wizard</td>
<td>graphic</td>
<td>present info</td>
</tr>
<tr>
<td>3a</td>
<td>Yes. It’s number 4.</td>
<td>user</td>
<td>speech</td>
<td>provide info</td>
</tr>
<tr>
<td>3b</td>
<td>[selects item 4]</td>
<td>user</td>
<td>graphic</td>
<td>provide info</td>
</tr>
</tbody>
</table>

- **Simultaneous actions**
- **Redundant actions**
## Costs of Multimodal Dialogue Acts

<table>
<thead>
<tr>
<th>ID</th>
<th>Utterance</th>
<th>Speaker</th>
<th>Modality</th>
<th>Speech act</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Please play “Nevermind”. Does this list contain the song?</td>
<td>user</td>
<td>speech</td>
<td>request</td>
</tr>
<tr>
<td>2a</td>
<td>[shows list with 20 DB matches]</td>
<td>wizard</td>
<td>speech</td>
<td>request info</td>
</tr>
<tr>
<td>2b</td>
<td>[shows list with 20 DB matches]</td>
<td>wizard</td>
<td>graphic</td>
<td>present info</td>
</tr>
<tr>
<td>3a</td>
<td>Yes. It’s number 4.</td>
<td>user</td>
<td>speech</td>
<td>provide info</td>
</tr>
<tr>
<td>3b</td>
<td>[selects item 4]</td>
<td>user</td>
<td>graphic</td>
<td>provide info</td>
</tr>
</tbody>
</table>

- **Simultaneous actions**
- **Redundant actions**
Costs of Multimodal Dialogue Acts

<table>
<thead>
<tr>
<th>ID</th>
<th>Utterance</th>
<th>Speaker</th>
<th>Modality</th>
<th>Speech act</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Please play “Nevermind”.</td>
<td>user</td>
<td>speech</td>
<td>request</td>
</tr>
<tr>
<td>2a</td>
<td>Does this list contain the song?</td>
<td>wizard</td>
<td>speech</td>
<td>request info</td>
</tr>
<tr>
<td>2b</td>
<td>[shows list with 20 DB matches]</td>
<td>wizard</td>
<td>graphic</td>
<td>present info</td>
</tr>
<tr>
<td>3a</td>
<td>Yes. It’s number 4.</td>
<td>user</td>
<td>speech</td>
<td>provide info</td>
</tr>
<tr>
<td>3b</td>
<td>[selects item 4]</td>
<td>user</td>
<td>graphic</td>
<td>provide info</td>
</tr>
</tbody>
</table>

- Simultaneous actions
- Redundant actions
## Costs of Multimodal Dialogue Acts

<table>
<thead>
<tr>
<th>ID</th>
<th>Utterance</th>
<th>Speaker</th>
<th>Modality</th>
<th>Speech act</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Please play “Nevermind”.</td>
<td>user</td>
<td>speech</td>
<td>request</td>
</tr>
<tr>
<td>2a</td>
<td>Does this list contain the song?</td>
<td>wizard</td>
<td>speech</td>
<td>request info</td>
</tr>
<tr>
<td>2b</td>
<td>[shows list with 20 DB matches]</td>
<td>wizard</td>
<td>graphic</td>
<td>present info</td>
</tr>
<tr>
<td>3a</td>
<td>Yes. It’s number 4.</td>
<td>user</td>
<td>speech</td>
<td>provide info</td>
</tr>
<tr>
<td>3b</td>
<td>[selects item 4]</td>
<td>user</td>
<td>graphic</td>
<td>provide info</td>
</tr>
</tbody>
</table>

- **Simultaneous actions**
- **Redundant actions**
## Costs of Multimodal Dialogue Acts

<table>
<thead>
<tr>
<th>ID</th>
<th>Utterance</th>
<th>Speaker</th>
<th>Modality</th>
<th>Speech act</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Please play “Nevermind”.</td>
<td>user</td>
<td>speech</td>
<td>request</td>
</tr>
<tr>
<td>2a</td>
<td>Does this list contain the song?</td>
<td>wizard</td>
<td>speech</td>
<td>request info</td>
</tr>
<tr>
<td>2b</td>
<td>[shows list with 20 DB matches]</td>
<td>wizard</td>
<td>graphic</td>
<td>present info</td>
</tr>
<tr>
<td>3a</td>
<td>Yes. It’s number 4.</td>
<td>user</td>
<td>speech</td>
<td>provide info</td>
</tr>
<tr>
<td>3b</td>
<td>[selects item 4]</td>
<td>user</td>
<td>graphic</td>
<td>provide info</td>
</tr>
</tbody>
</table>

- **Simultaneous actions**
- **Redundant actions**
Outline

Motivation
  The basic problem
  Previous work

Framework
  The Learning Approach
  The Data Collection
  Results

Performance modelling
  RL and Performance modelling
  Dialogue costs and multimodality
  Modality costs and situations
  Ambiguity and task success
  Dialogue quality and “emotions”
Cognitive load of primary and secondary task


Can we utilise these rankings for our reward measure?
Outline

Motivation
- The basic problem
- Previous work

Framework
- The Learning Approach
- The Data Collection
- Results

Performance modelling
- RL and Performance modelling
- Dialogue costs and multimodality
- Modality costs and situations
- Ambiguity and task success
- Dialogue quality and “emotions”
Task success

**PARADISE:** AVM-style definition of task success

<table>
<thead>
<tr>
<th>attribute</th>
<th>possible values</th>
<th>info flow</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;depart-city&gt;</td>
<td>{Milano, Roma, Torino, Trento}</td>
<td>to agent</td>
</tr>
<tr>
<td>&lt;arrival-city&gt;</td>
<td>{Milano, Roma, Torino, Trento}</td>
<td>to agent</td>
</tr>
<tr>
<td>&lt;depart-range&gt;</td>
<td>{morning, evening}</td>
<td>to agent</td>
</tr>
<tr>
<td>&lt;depart-time&gt;</td>
<td>{6am, 8am, 6pm, 9pm}</td>
<td>to user</td>
</tr>
</tbody>
</table>

**PROMISE:** [Beringer et al.], 2002

- *information bits* to measure (sub-)task success

Example: "Plan an evening watching TV": film = [channel, time] ∨ [title, time] ∨ [title, channel] ∨ ...
Task success

**PARADISE:** AVM-style definition of task success

<table>
<thead>
<tr>
<th>attribute</th>
<th>possible values</th>
<th>info flow</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;depart-city&gt;</td>
<td>{Milano, Roma, Torino, Trento}</td>
<td>to agent</td>
</tr>
<tr>
<td>&lt;arrival-city&gt;</td>
<td>{Milano, Roma, Torino, Trento}</td>
<td>to agent</td>
</tr>
<tr>
<td>&lt;depart-range&gt;</td>
<td>{morning, evening}</td>
<td>to agent</td>
</tr>
<tr>
<td>&lt;depart-time&gt;</td>
<td>{6am, 8am, 6pm, 9pm}</td>
<td>to user</td>
</tr>
</tbody>
</table>

**PROMISE:** [Beringer et al.], 2002

- *information bits* to measure (sub-)task success

Example: "Plan an evening watching TV": film = [channel, time] ∨ [title, time] ∨ [title, channel] ∨ ...
Task success

**PARADISE:** AVM-style definition of task success

<table>
<thead>
<tr>
<th>attribute</th>
<th>possible values</th>
<th>info flow</th>
</tr>
</thead>
<tbody>
<tr>
<td>⟨depart-city⟩</td>
<td>{Milano, Roma, Torino, Trento}</td>
<td>to agent</td>
</tr>
<tr>
<td>⟨arrival-city⟩</td>
<td>{Milano, Roma, Torino, Trento}</td>
<td>to agent</td>
</tr>
<tr>
<td>⟨depart-range⟩</td>
<td>{morning, evening}</td>
<td>to agent</td>
</tr>
<tr>
<td>⟨depart-time⟩</td>
<td>{6am, 8am, 6pm, 9pm}</td>
<td>to user</td>
</tr>
</tbody>
</table>

**PROMISE:** [Beringer et al.], 2002

- *information bits* to measure (sub-)task success

**Example:** "Plan an evening watching TV": film = [channel, time] ∨ [title, time] ∨ [title, channel] ∨ ...
Ambiguity in PROMISE

Your little brother likes to listen to heavy metal music. You want to build him a playlist including three metal songs. Make sure you have “Enter Sandman" on the playlist! Save the playlist under the name “heavy guys".

**main task** (makePlaylist)

**sub-tasks:** search(item1), search(item2), search(item3), playlist(name), add(item1, name), add(item2, name), add(item3, name)

What to do when “Enter Sandman" has several matches in the DB? How to measure task success *online*?
Ambiguity in PROMISE

Your little brother likes to listen to heavy metal music. You want to build him a playlist including three metal songs. Make sure you have “Enter Sandman" on the playlist! Save the playlist under the name “heavy guys".

main task (makePlaylist)
sub-tasks: search(item1), search(item2), search(item3), playlist( name), add(item1, name), add(item2, name), add(item3, name)

What to do when “Enter Sandman" has several matches in the DB? How to measure task success online?
Ambiguity in PROMISE

Your little brother likes to listen to heavy metal music. You want to build him a playlist including three metal songs. Make sure you have “Enter Sandman" on the playlist! Save the playlist under the name “heavy guys".

**main task** (makePlaylist)

**sub-tasks:** search(item1), search(item2), search(item3), playlist(name), add(item1, name), add(item2, name), add(item3, name)

What to do when “Enter Sandman" has several matches in the DB? How to measure task success *online*?
Ambiguity in PROMISE

Your little brother likes to listen to heavy metal music. You want to build him a playlist including three metal songs. Make sure you have “Enter Sandman” on the playlist! Save the playlist under the name “heavy guys”.

**main task** (makePlaylist)

**sub-tasks**: search(item1), search(item2), search(item3), playlist( name), add(item1, name), add(item2, name), add(item3, name)

What to do when “Enter Sandman" has several matches in the DB? How to measure task success online?
Your little brother likes to listen to heavy metal music. You want to build him a playlist including three metal songs. Make sure you have “Enter Sandman" on the playlist! Save the playlist under the name “heavy guys".

**main task** (makePlaylist)

**sub-tasks**: search(item1), search(item2), search(item3), playlist(name), add(item1, name), add(item2, name), add(item3, name)

What to do when “Enter Sandman" has several matches in the DB? How to measure task success *online*?
Algorithm for flexible task success definition

Extend the information bit set until the description is precise.

Example:

\[ \text{item1} = [\text{title} = \text{“Enter Sandman”}] \]

If item1 has several matches in the DB:

\[ \text{item1} = [\text{title} = \text{“Enter Sandman”}] \land [\text{album}] \]

→ Recursive definition of task success based on ambiguity.
Algorithm for flexible task success definition

Extend the information bit set until the description is precise.

**Example:**

\[item1 = \text{[title= "Enter Sandman"]}\]

If \(item1\) has several matches in the DB:
\[item1 = \text{[title= "Enter Sandman"]} \land \text{[album]}\]

→ Recursive definition of task success based on ambiguity.
Algorithm for flexible task success definition

Extend the information bit set until the description is precise.

**Example:**

```
item1 = [title= "Enter Sandman"]
```

If item1 has several matches in the DB:

```
item1 = [title= "Enter Sandman"] \land [album]
```

→ Recursive definition of task success based on ambiguity.
Algorithm for flexible task success definition

Extend the information bit set until the description is precise.

Example:

\[ \text{item1} = [\text{title} = \text{"Enter Sandman"}] \]

If item1 has several matches in the DB:

\[ \text{item1} = [\text{title} = \text{"Enter Sandman"}] \land [\text{album}] \]

→ Recursive definition of task success based on ambiguity.
Algorithm for flexible task success definition

Extend the information bit set until the description is precise.

*Example:*

\[
\text{item1} = \{\text{title} = "Enter Sandman"\}
\]

*If item1 has several matches in the DB:*

\[
\text{item1} = \{\text{title} = "Enter Sandman"\} \land \{\text{album}\}
\]

→ Recursive definition of task success based on ambiguity.
Outline

Motivation
  The basic problem
  Previous work

Framework
  The Learning Approach
  The Data Collection
  Results

Performance modelling
  RL and Performance modelling
  Dialogue costs and multimodality
  Modality costs and situations
  Ambiguity and task success
  Dialogue quality and “emotions"
Subjective evaluation using “emotions”

- **PARADISE**: user questionnaires
- How to get these measures at system runtime?
  → Recognise “emotions” as immediate positive/negative feedback
  → Hope to learn a strategy which reacts to user frustration/stress more quickly (following [Litman et al.])
Subjective evaluation using “emotions”

- **PARADISE**: user questionnaires
- How to get these measures at system runtime?
  → Recognise “emotions” as immediate positive/negative feedback
  → Hope to learn a strategy which reacts to user frustration/stress more quickly (following [Litman et al.])
Subjective evaluation using “emotions”

- **PARADISE**: user questionnaires
- How to get these measures at system runtime?
  → Recognise “emotions” as immediate positive/negative feedback
  → Hope to learn a strategy which reacts to user frustration/stress more quickly (following [Litman et al.])
Subjective evaluation using “emotions”

- **PARADISE**: user questionnaires
- How to get these measures at system runtime?
  → Recognise “emotions” as immediate positive/negative feedback
  → Hope to learn a strategy which reacts to user frustration/stress more quickly (following [Litman et al.])
Detecting emotions

IGK project, July 2005 (Hofer, Rieser): *Emotion tagging for the COMMUNICATOR corpus.*
Detecting emotions

IGK project, July 2005 (Hofer, Rieser): *Emotion tagging for the COMMUNICATOR corpus.*

**Figure:** Feeltrace, [Cowie et al.], 2000.
Detecting emotions

IGK project, July 2005 (Hofer, Rieser): Emotion tagging for the COMMUNICATOR corpus.
Summary

Hypothesis

- Multi-modal clarification strategies involve complex planning over a variety of contextual factors while maximising user satisfaction.

Method

- Apply RL in the ISU update approach and model user satisfaction by assigning local rewards.

Expected outcome

- Learn **flexible, context-adaptive** strategy for clarification subdialogues
- While following a **user centred** approach.
In other words . . .

Asking the “right” clarification depends on the context and the “goal”.

Figure: Performance modelling for multi-modal in-car dialogues
In other words . . .

Asking the “right” clarification depends on the context and the “goal”.

- Help to accomplish the task!
- Save costs!
- Don’t distract the driver!
- Don’t frustrate the driver!
In other words . . .

*Asking the “right” clarification depends on the context and the “goal”.*

- Help to accomplish the task!
- Save costs!
- Don’t distract the driver!
- Don’t frustrate the driver!
Papers associated with this talk:


For Further Reading I


Marylin Walker.  
An Application of Reinforcement Learning to Dialogue Strategy Selection in a Spoken Dialogue System for Email.  

Angelika Salmen.  
Multi-modale Menüausgabe im Fahrzeug.  
*(PhD thesis, University of Regensburg, 2002).*
For Further Reading III

Cowie, Roddy and Douglas-Cowie, Ellen and Savvidou, Suzie and McMahon, Edelle and Sawey, Martin and Schröder, Marc

’FEELTRACE’: An Instrument for Recording Perceived Emotion in Real Time

Algorithm for flexible task success definition

Constraints are sets of information bits
U is user input string
F field searched by wizard
DB is number of matches in the database

Initialize:
\[
\text{task} = \text{makePlaylist} \\
\text{makePlaylist} = \text{subtask(item1)} \land \ldots \land \text{subtask(itemN)} \\
\text{item1, \ldots, itemN} = \text{ValueList} \\
\text{ValueList} = \text{constraint1} \lor \text{constraint2} \lor \ldots \lor \text{constraintN}
\]

Repeat:
\[
\text{value} = \text{Parse(U)} \\
\text{If (value \neq F): "error; needs manual annotation"} \\
\text{Else:} \\
\quad \text{For constraint in ValueList:} \\
\quad \quad \text{If (DB \neq 0): refineConstraintDefinition} \\
\text{Until: Task success is precisely defined}
\]
Algorithm for flexible task success definition

Constraints are sets of information bits
U is user input string
F field searched by wizard
DB is number of matches in the database

Initialize:
\[
\text{task} = \text{makePlaylist} \\
\text{makePlaylist} = \text{subtask(item1)} \land \ldots \land \text{subtask(itemN)} \\
\text{item1}, \ldots, \text{itemN} = \text{ValueList} \\
\text{ValueList} = \text{constraint1} \lor \text{constraint2} \lor \ldots \lor \text{constraintN}
\]

Repeat:
\[
\text{value} = \text{Parse(U)} \\
\text{If} (\text{value} \neq \text{F}): "error; needs manual annotation" \\
\text{Else:} \\
\text{For constraint in ValueList:} \\
\text{If} (\text{DB} \neq 0): \text{refineConstraintDefinition}
\]

Until: Task success is precisely defined
Algorithm for flexible task success definition

Constraints are sets of information bits
U is user input string
F field searched by wizard
DB is number of matches in the database

Initialize:

\[
\text{task} = \text{makePlaylist} \\
\text{makePlaylist} = \text{subtask(item1)} \land \ldots \land \text{subtask(itemN)} \\
\text{item1, \ldots, itemN} = \text{ValueList} \\
\text{ValueList} = \text{constraint1} \lor \text{constraint2} \lor \ldots \lor \text{constraintN}
\]

Repeat:

\[
\text{value} = \text{Parse(U)} \\
\text{If (value} \neq \text{F): "error; needs manual annotation"}
\]

Else:

\[
\text{For constraint in ValueList:} \\
\text{If (DB} \neq 0): \text{refineConstraintDefinition}
\]

Until: Task success is precisely defined
Algorithm for flexible task success definition

Constraints are sets of information bits
U is user input string
F field searched by wizard
DB is number of matches in the database

Initialize:
- task = makePlaylist
- makePlaylist = subtask(item1) ∧ ... ∧ subtask(itemN)
- item1, ... , itemN = ValueList
- ValueList = constraint1 ∨ constraint2 ∨ ... ∨ constraintN

Repeat:
- value = Parse(U)
  - If (value != F): "error; needs manual annotation"
  - Else:
    - For constraint in ValueList:
      - If (DB != 0): refineConstraintDefinition

Until: Task success is precisely defined
Algorithm for flexible task success definition

Constraints are sets of information bits
U is user input string
F field searched by wizard
DB is number of matches in the database

Initialize:
\[
\text{task} = \text{makePlaylist} \\
\text{makePlaylist} = \text{subtask}(\text{item1}) \land \ldots \land \text{subtask}(\text{itemN}) \\
\text{item1}, \ldots, \text{itemN} = \text{ValueList} \\
\text{ValueList} = \text{constraint1} \lor \text{constraint2} \lor \ldots \lor \text{constraintN}
\]

Repeat:
\[
\text{value} = \text{Parse(U)} \\
\text{If} \ (\text{value} \neq \text{F}): \ "\text{error; needs manual annotation}\" \\
\text{Else}: \\
\quad \text{For constraint in ValueList:} \\
\quad \quad \text{If} \ (\text{DB} \neq 0): \ \text{refineConstraintDefinition}
\]

Until: Task success is precisely defined