# Learning Multimodal Clarification Strategies

Verena Rieser[1] Ivana Kruijff-Korbayová[1] Oliver Lemon[2]

[1]Department of Computational Linguistics,
Saarland University, Germany

[2]School of Informatics,
University of Edinburgh, GB

In affiliation with: IGK and TALK Project
`http://www.talk-project.org/`

# Outline

# Outline

# Outline

# Outline

# 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? `ambig 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]` `ambig 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]` `ambig 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

# 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**
  $\longrightarrow$ 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**
  $\longrightarrow$ 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**.
  $\longrightarrow$ We don't know how to set function features in dialogue systems!

# Outline

# Approach

Assumptions

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

# Approach

<u>Assumptions</u>

- 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.
   $\rightarrow$ 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).

# 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).**

# Framework for learning multimodal CRs

1. Collect data on possible strategies in WOZ experiment.
   $\longrightarrow$ Identify possible state-action mappings
2. Bootstrap an initial policy using supervised learning in the ISU approach.
   $\longrightarrow$ Learn wizards' decisions in context
3. **Optimise the learnt policy for dialogue systems using reinforcement learning (RL).**
   $\longrightarrow$ How should the performance function (reward) look like?

# Outline

# The SAMMIE-2[1] Data Collection

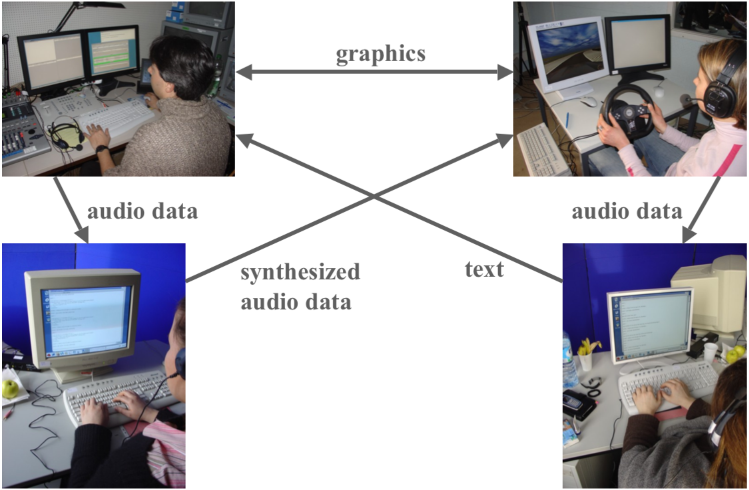



Figure: Multimodal Wizard-of-Oz data collection setup for an in-car

# Experimental Setup

6 wizards, 24 subjects

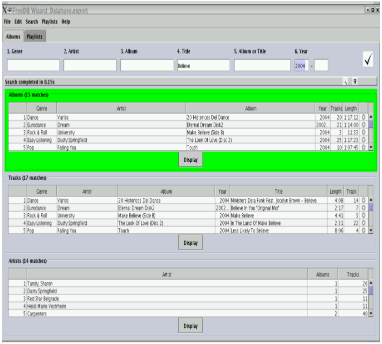
Wizard:

- Screen output options pre-computed, wizard freely talking
- Wizard "sees what the system sees" (corrupted transcriptions) $\rightarrow$ "clarification pop-up"

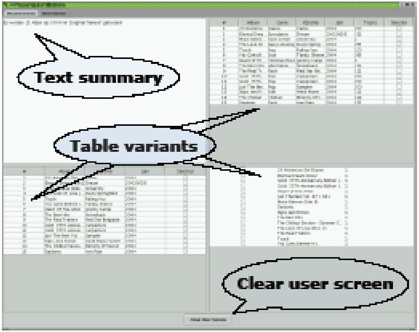
User:

- 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

# 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

# Reinforcement Learning

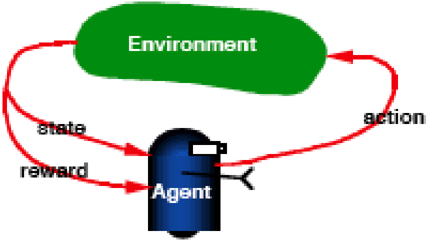



Figure: [Sutton and Barto], 1998.

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

# RL and PARADISE

Performance modelling for RL in PARADISE [Walker], 2000.

# RL and PARADISE

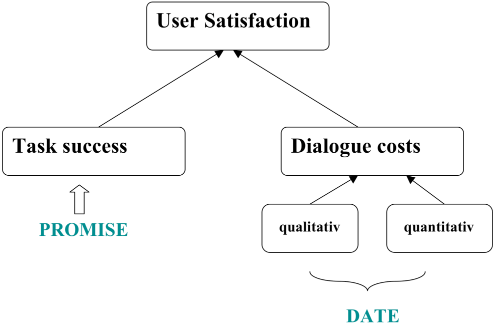
Performance modelling for RL in PARADISE [Walker], 2000.

UserSatisfaction(max TaskSuccess, min Costs)

# RL and PARADISE

Performance modelling for RL in PARADISE [Walker], 2000.

UserSatisfaction(max TaskSuccess, min Costs)



# Outline

# 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)
$\rightarrow$ `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)
$\rightarrow$ `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)
$\rightarrow$ `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)
$\rightarrow$ `present-info` indicates task success

# Costs of Multimodal Dialogue Acts

| ID | Utterance | Speaker | Modality | Speech act |
|---|---|---|---|---|
| 1 | Please play “Nevermind". | user | speech | request |
| 2a | Does this list contain the song? | wizard | speech | request info |
| 2b | [shows list with 20 DB matches] | wizard | graphic | present info |
| 3a | Yes. It’s number 4. | user | speech | provide info |
| 3b | [selects item 4] | user | graphic | provide info |

- Simultaneous actions
- Redundant actions

# Costs of Multimodal Dialogue Acts

| ID | Utterance | Speaker | Modality | Speech act |
|---|---|---|---|---|
| 1 | Please play “Nevermind". | user | speech | request |
| 2a | Does this list contain the song? | wizard | speech | request info |
| 2b | [shows list with 20 DB matches] | wizard | graphic | present info |
| 3a | Yes. It’s number 4. | user | speech | provide info |
| 3b | [selects item 4] | user | graphic | provide info |

- Simultaneous actions
- Redundant actions

# Costs of Multimodal Dialogue Acts

| ID | Utterance | Speaker | Modality | Speech act |
|---|---|---|---|---|
| 1 | Please play “Nevermind". | user | speech | request |
| **2a** | **Does this list contain the song?** | wizard | speech | request info |
| **2b** | **[shows list with 20 DB matches]** | wizard | graphic | present info |
| 3a | Yes. It’s number 4. | user | speech | provide info |
| 3b | [selects item 4] | user | graphic | provide info |

- Simultaneous actions
- Redundant actions

# Costs of Multimodal Dialogue Acts

| ID | Utterance | Speaker | Modality | Speech act |
|---|---|---|---|---|
| 1 | Please play “Nevermind". | user | speech | request |
| 2a | Does this list contain the song? | wizard | speech | request info |
| 2b | [shows list with 20 DB matches] | wizard | graphic | present info |
| 3a | Yes. It’s number 4. | user | speech | provide info |
| 3b | [selects item 4] | user | graphic | provide info |

- Simultaneous actions
- Redundant actions

# Costs of Multimodal Dialogue Acts

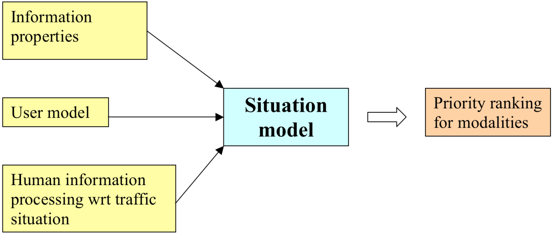
| ID | Utterance | Speaker | Modality | Speech act |
|---|---|---|---|---|
| 1 | Please play “Nevermind". | user | speech | request |
| 2a | Does this list contain the song? | wizard | speech | request info |
| 2b | [shows list with 20 DB matches] | wizard | graphic | present info |
| 3a | **Yes. It’s number 4.** | user | speech | provide info |
| 3b | **[selects item 4]** | user | graphic | provide info |

- Simultaneous actions
- Redundant actions

# Outline

# Cognitive load of primary and secondary task

[Salmen], PhD thesis, 2002)): *Multi-modale Menüausgabe im Fahrzeug*.




Can we utilise these rankings for our reward measure?

# Outline

# Task success

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

| attribute | possible values | info flow |
|---|---|---|
| <depart-city> | {Milano, Roma, Torino, Trento} | to agent |
| <arrival-city> | {Milano, Roma, Torino, Trento} | to agent |
| <depart-range> | {morning, evening} | to agent |
| <depart-time> | {6am, 8am, 6pm, 9pm} | to user |

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

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

Example: "Plan an evening watching TV": film = [channel, time] $\vee$ [title, time] $\vee$ [title, channel] $\vee$ ...

# Task success

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

| attribute | possible values | info flow |
|---|---|---|
| <depart-city> | {Milano, Roma, Torino, Trento} | to agent |
| <arrival-city> | {Milano, Roma, Torino, Trento} | to agent |
| <depart-range> | {morning, evening} | to agent |
| <depart-time> | {6am, 8am, 6pm, 9pm} | to user |

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

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

Example: "Plan an evening watching TV": film = [channel, time] $\vee$ [title, time] $\vee$ [title, channel]$\vee$ . . .

# Task success

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

| attribute | possible values | info flow |
|---|---|---|
| <depart-city> | {Milano, Roma, Torino, Trento} | to agent |
| <arrival-city> | {Milano, Roma, Torino, Trento} | to agent |
| <depart-range> | {morning, evening} | to agent |
| <depart-time> | {6am, 8am, 6pm, 9pm} | to user |

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

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

Example: "Plan an evening watching TV": film = [channel, time] $\vee$ [title, time] $\vee$ [title, channel]$\vee$ ...

# 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*?

# 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*?

# 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"] ∧ [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”*] $\wedge$ [*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”] ∧ [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”]* $\wedge$ *[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.**

<u>*Example:*</u>

*item1= [title= “Enter Sandman"]*

*If item1 has several matches in the DB:*

*item1= [title= “Enter Sandman"]* $\wedge$ *[album]*

$\longrightarrow$ Recursive definition of task success based on ambiguity.

# Outline

# 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?

$\longrightarrow$ Recognise **“emotions"** as immediate positive/negative feedback

$\longrightarrow$ 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*.

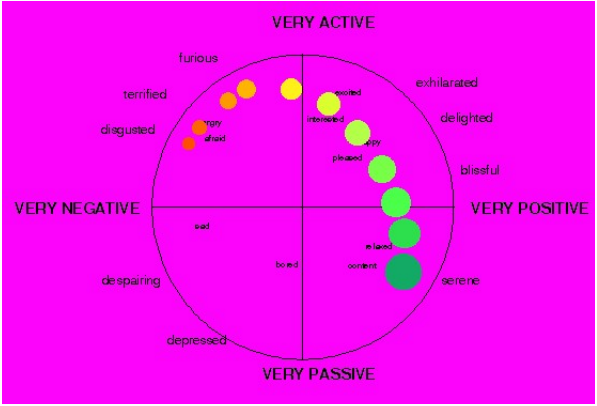
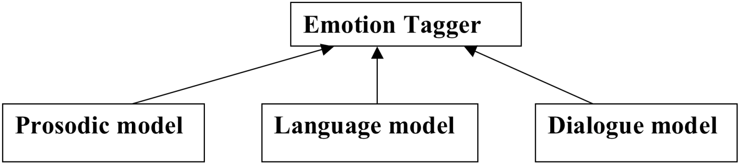



Figure: Feeltrace, [Cowie et al.], 2000

# 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".*

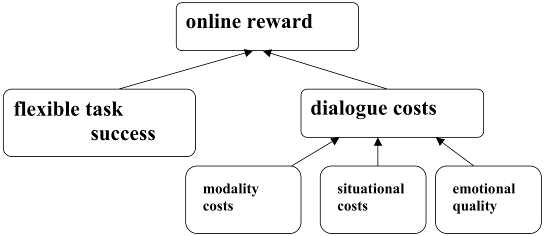



Figure: Performance modelling for multi-modal in-car dialogues

# 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:

- Verena Rieser and Johanna Moore. Implications for Generating Clarification Requests in Task-oriented Dialogues. *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL-05)*, 2005.
- Ivana Kruijff-Korbayová, Nate Blaylock, Ciprian Gerstenberger, Verena Rieser, Tilman Becker, Michael Kaisser, Peter Poller, Jan Schehl. An Experimental Setup for Collecting Data for Adaptive Output Planning in a Mutlimodal Dailogue System.*Proceedings of European Natural Language Generation Workshop*, 2005.
- Verena Rieser, Ivana Kruijff-Korbayová, Oliver Lemon: A Framework for Learning Multimodal Clarification Strategies. Submitted.

# For Further Reading I


Marylin Walker and Rebecca Passoneau.
DATE: A dialogue act tagging scheme for evaluation.
*Proceedings of the Human Language Technology Conference*, 2001.

Nicole Beringer and Ute Kartal and Katerina Louka and Florian Schiel and Uli Türk.
PROMISE: A Procedure for Multimodal Interactive System Evaluation.
*Proceedings of the Workshop Multimodal Resources and Multimodal Systems Evaluation*, 2002.

# For Further Reading II

Marylin Walker.
An Application of Reinforcement Learning to Dialogue Strategy Selection in a Spoken Dialogue System for Email.
*Journal of Artificial Intelligence Research*, 2000.

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
*Proceedings of the ISCA Workshop on Speech and Emotion: A Conceptual Framework for Research*, 2000.

# 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) $\wedge \ldots \wedge$ subtask(itemN)
item1, ..., itemN = ValueList
ValueList = constraint1 $\vee$ constraint2 $\vee \ldots \vee$ 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:**

task = makePlaylist
makePlaylist = subtask(item1) $\wedge \ldots \wedge$ subtask(itemN)
item1, ..., itemN = ValueList
ValueList = constraint1 $\vee$ constraint2 $\vee \ldots \vee$ 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:**

task = makePlaylist
makePlaylist = subtask(item1) $\wedge \ldots \wedge$ subtask(itemN)
item1, . . . , itemN = ValueList
ValueList = constraint1 $\vee$ constraint2 $\vee \ldots \vee$ 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:**
task = makePlaylist
makePlaylist = subtask(item1) $\wedge \ldots \wedge$ subtask(itemN)
item1, . . . , itemN = ValueList
ValueList = constraint1 $\vee$ constraint2 $\vee \ldots \vee$ 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:**
  task = makePlaylist
  makePlaylist = subtask(item1) $\wedge \ldots \wedge$ subtask(itemN)
  item1, ..., itemN = ValueList
  ValueList = constraint1 $\vee$ constraint2 $\vee \ldots \vee$ 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