A Framework for Learning Multimodal Clarification Strategies

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

[1]Department of Computational Linguistics,
Saarland University

[2]School of Informatics,
University of Edinburgh

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

CRs in Spoken Dialogue Systems

System: What city are you leaving from?
User: Urbana Champaign.
System: Sorry, I'm not sure I understood what you said. Where are you leaving from?
User: Urbana Champaign.
System: I'm still having trouble understanding you. . . . What city are you leaving from?
User: Chicago.

[CMU Communicator – User-System]

→ System performs badly and sounds quite artificial.

CRs in Human-Human Dialogue

Cust:	I guess getting a car in London will not do me much good in /uh/ Spain is that right?
Agent:	I’m sorry? Getting a car . . . ?
Cust:	Yeah I’ll need a car in Madrid.
Agent:	OK.

Cust.:	I’ll be returning on Thursday the fifth.
Agent:	The fifth of February?
Cust.:	/UHU/

[CMU Communicator – Human-Human]

$\longrightarrow$ How to convert these kinds of clarification strategies to dialogue systems?

Outline

Outline

Outline

Outline

Outline

Outline

Generating CRs in task-oriented dialogues

[Rieser and Moore] *Implications for generating clarification requests in task-oriented dialogues*, ACL-05.

- **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] *Implications for generating clarification requests in task-oriented dialogues*, ACL-05.

- **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] *Implications for generating clarification requests in task-oriented dialogues*, ACL-05.

- **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] *Implications for generating clarification requests in task-oriented dialogues*, ACL-05.

- **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] *Implications for generating clarification requests in task-oriented dialogues*, ACL-05.

- **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**.
  For dialogue systems we still don't know:

Generating CRs in task-oriented dialogues

[Rieser and Moore] *Implications for generating clarification requests in task-oriented dialogues*, ACL-05.

- **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**.
  For dialogue systems we still don't know:
  → How to set the function features?

Generating CRs in task-oriented dialogues

[Rieser and Moore] *Implications for generating clarification requests in task-oriented dialogues*, ACL-05.

- **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**.
  For dialogue systems we still don't know:
  → How to set the function features?
  → How do these strategies perform?

Outline

Approach

Assumptions

- Clarification strategies involve **complex decision making over a variety of contextual factors**
- and **exhaustive planning towards maximising a desired outcome.**

$\rightarrow$ Apply reinforcement learning (RL) in the information state update (ISU) approach.

What is RL?

Framework for learning multimodal CRs

Overall approach:
$MDP = (S, A, T, R)$

1. Collect data on possible strategies in WOZ experiment.
   $\rightarrow$ Extract $\{A, S, R\}$
2. Bootstrap an initial policy using supervised learning in the ISU approach.
   $\rightarrow$ Learn wizards' decisions in context (T)
3. Optimise the learnt policy for dialogue systems using RL ($\pi^* \approx maxE[\sum_{j \geq i} r(d, j) | s_i, a]$).
   $\rightarrow$ How can we improve online reward measures $r(d, j)$?

Outline

The SAMMIE-2[1] Data Collection

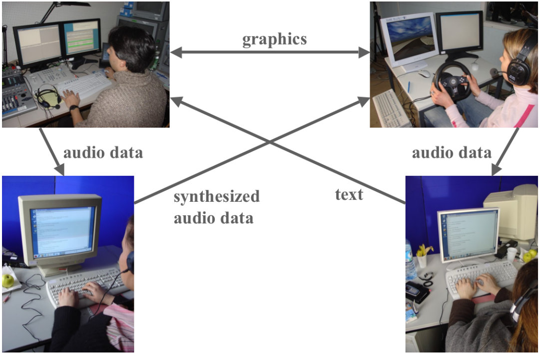


Figure: Multimodal Wizard-of-Oz data collection setup for an in-car music player application, using the Lane Change driving simulator. Top right: User, Top left: Wizard, Bottom: transcribers.

[1] SAMMIE stands for Saarbrücken Multimodal MP3 Player Interaction Experiment (cf. for more details [Kruijff-Korbayová et al.], ENLG 2005).

Experimental Setup

6 wizards, 24 subjects

User:

- User's primary task is driving
- Secondary MP3 selection task

Wizard:

- Screen output options pre-computed, wizard freely talking
- Wizard "sees what the system sees"

Introducing uncertainty:

- Corrupted transcriptions by "word killer" agent ($\approx$ acoustic problems)
- Lexical and reference ambiguities by task and DB
- Pop-up questionnaire window "CLARIE" agent

Outline

Evaluation

- 1772 turns and 17076 words.
- 774 wizard turns, 10.2% CRs (from CLARIE)
- User Satisfaction fairly high across wizards (15.0, δ=2.9, range 5 to 25)
- Multimodality: “Most helpful" vs. distracting

Corpus Requirements 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.
- Need to **explore** → online reward measure!

Outline

Currently applied (ad hoc) Reward Measures

- User satisfaction from questionnaires (offline)
  e.g. Final Reward = 14.94;
- Binary task success (online)
  e.g. Final Reward = +1|-1;
- Cost function of filled and confirmed slot values, dialogue length etc. (online)
  e.g. Final Reward = (expected length)+(filled slots)+(retrieving info)+. . .
- US as defined in PARADISE (online)
  e.g. Final Reward (US)=0.47(Mean Recognition Score)+0.21(Perception of task completion)+0.15*(elapsed time);*

→ Can we use existing (fine grained) evaluation schemes?

RL and PARADISE

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

RL and PARADISE

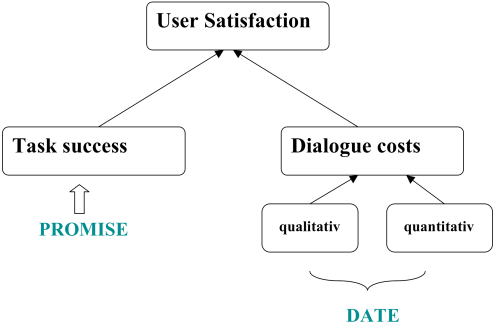
Performance modelling for RL in PARADISE [Walker], 2000.

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)
→ `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

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

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

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
  info bits are defined to describe when a task is completed;

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
  info bits are defined to describe when a task is completed;

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
  info bits are defined to describe when a task is completed;

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)`

info-bits: `item1=[title: “Enter Sandman"], item2=[title]` ∨ `[album,track]...`

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)`

info-bits: `item1=[title: "Enter Sandman"], item2=[title]` ∨ `[album,track]...`

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)`

info-bits: `item1=[title:“Enter Sandman"],` `item2=[title]` ∨ `[album,track]...`

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)`

info-bits: `item1=[title:“Enter Sandman"],` `item2=[title]` ∨ `[album,track]...`

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)`

info-bits: `item1=[title: “Enter Sandman"],` `item2=[title]` ∨ `[album,track]...`

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

Algorithm for flexible task success definition

1. **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*]

$\rightarrow$ Recursive online definition of task success based on ambiguity.

2. **Backing-off to evaluate final task success based on "user's goal".**

Outline

Policy shaping for immediate credit

Policy shaping:
argument the underlying reward structure with shaping function F (bias reflecting prior knowledge).

$$M' = (S, A, T, R + F) \tag{1}$$

- Task success: give credit for every (grounded) information bit.
- Mutlimodal cost function: F can be estimated with *dynamic shaping*.

Outline

What we haven't solved so far . . .

- How to account for more **user-centred** reward measures?
- What about more **qualitative measures**?
- What about **cognitive load** while driving?

$\longrightarrow$ Can we utilise “emotions" as continuos reward signal?

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 **continuous, local rewards** in combination with “delayed" rewards.

Expected outcome

- Learn **flexible, context-adaptive strategy** for clarification subdialogues
- Define a portable online reward measure.

In other words . . .

Asking the “right" clarification depends on the context and the reward as the “goal".

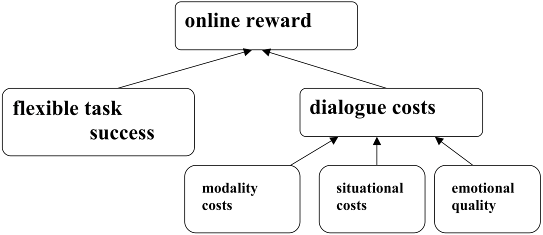


Figure: Performance modelling for multi-modal in-car dialogues

In other words . . .

Asking the “right" clarification depends on the context and the reward as 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 reward as 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, Ivana Kruijff-Korbayová, Oliver Lemon: **A Framework for Learning Multimodal Clarification Strategies**. To be published in: Proceedings of SIGDIAL, 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 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)*, 2005.

For Further Reading I

Richard S. Sutton and Anrew G. Barto.
Reinforcement Learning: An Introduction.
The MIT Press, 1998.

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.

Algorithm for flexible task success definition

U is user input string
DB is number of matches in the database
Initialize: task = makePlaylist
makePlaylist = subtask(item1) $\wedge \ldots \wedge$ subtask(itemN)
item1, . . . , item N = alternativeSetList
alternativeSetList =infoSet1 $\vee$ infoSet2 $\vee \ldots \vee$ infoSetN
infoSet1, infoSet2, . . . , infoSetN = infoBit1 $\wedge$ infoBit2 $\wedge$ infoBitN
For every U:
value = Parse(U)
If (DB != 0):
newSet = currentSet.add(infoBit)
alternativeSetList.add(newSet)
For every infoSet in alternativeSetList:
try to instantiate infoSet
currentUserGoal = infoSet instatiated

Outline

Implications for reward measures

Implications for a more informative reward

- Hypothesis1: Local reward measures lead to faster learning.

→ *Filled slots as local and task success as final reward*

- Hypothesis2: The reward measure is the place to incorporate complex domain knowledge

→ *Reflect the relation between costs and speech acts*

Policy shaping

Policy shaping:
argument the underlying reward structure with shaping function F (bias reflecting prior knowledge).

$$M' = (S, A, T, R + F) \tag{2}$$

F can be estimated with *dynamic shaping*.

Reinforcement Learning (RL)

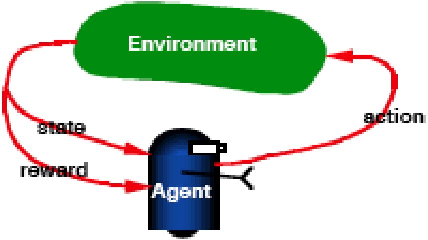


Figure: [Sutton and Barto], 1998.

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

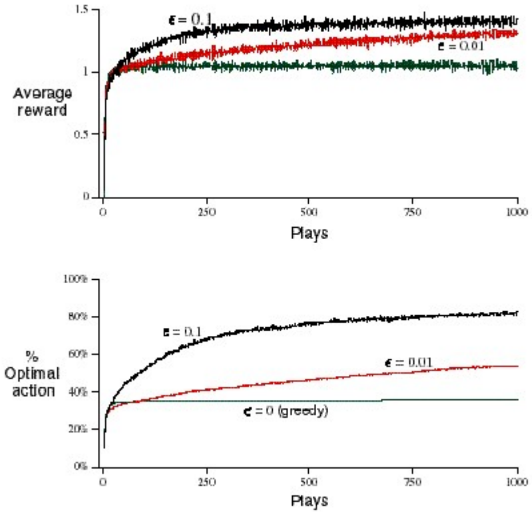
MDP model for RL

- Markov Decision Process:
  $MDP = (S, A, T, R)$
- Transition probability function:
  $P^a_{ss} = Pr\{s_{t+1} = s' | s_t = s, a_t = a\}$
- Reward signal:
  $R^a_{ss} = E\{r_{t+1} | s_t = s, a_t = a, s_{t+1}\}$
- Optimal policy π*:
  $Q(s_i, a) \approx E[\sum_{j \geq i} r(d, j) | s_i, a]$

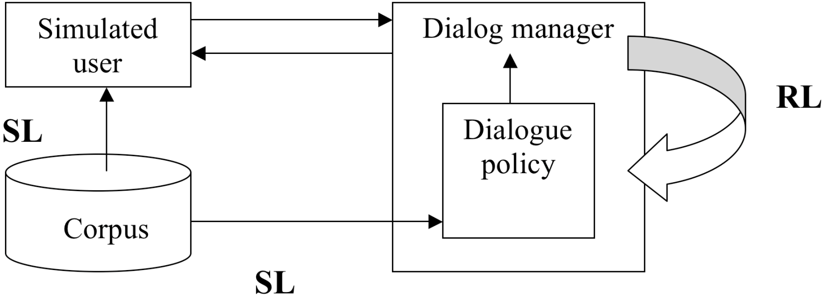
Major features of RL

- Adaptation
- Evaluative feedback
- Delayed reinforcement
- Exploitation vs. exploration

Greedy actions



RL for dialogue systems



How does this work for us?