

Language Technology II

Dialogue Management

Ivana Kruijff-Korbayová
 korbay@coli.uni-sb.de
 www.coli.uni-sb.de/~korbay/ → Teaching

Outline

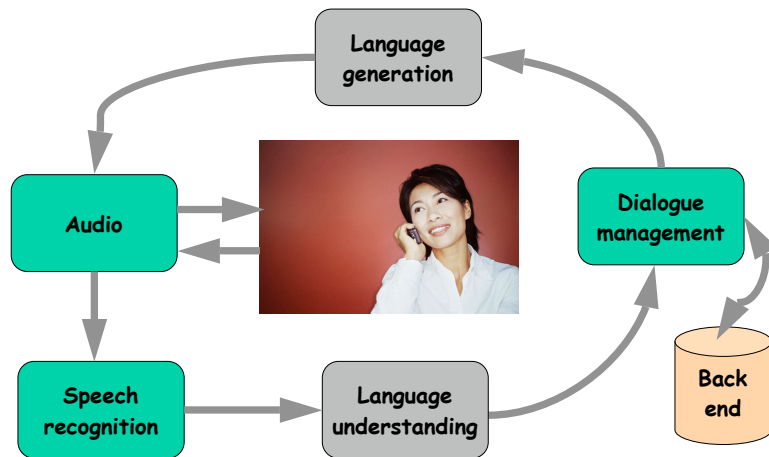
- Tasks of dialogue management
- Finite State-Based DM
- Frame-Based DM
- Agent-Based DM
- Grounding and Verification
- Future Perspectives

6/14/05

Ivana Kruijff-Korbayová
 Language Technology II: Dialogue Management

1

Pipeline Architectures



6/14/05

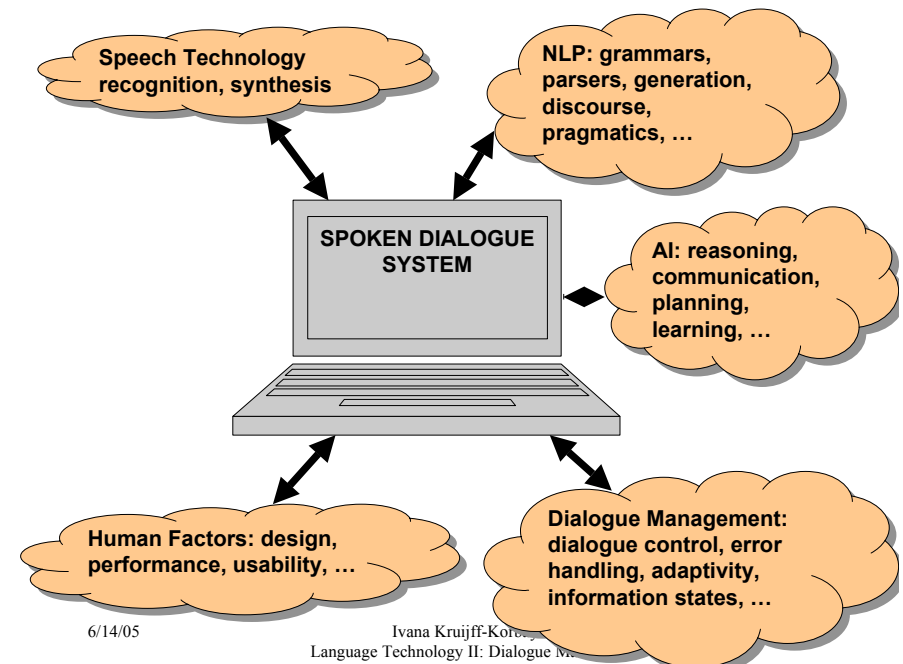
Ivana Kruijff-Korbayová
 Language Technology II: Dialogue Management

3

6/14/05

Ivana Kruijff-Korbayová
 Language Technology II: Dialogue Management

2



6/14/05

Ivana Kruijff-Korbayová
 Language Technology II: Dialogue Management

Task of Dialogue Management

- Dialogue flow control
- Dialogue modeling

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

5

Dialogue Flow Control

when to say something,
when to stop
⇒ turn taking (last lecture)

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

6

Dialogue Modeling

Where we are &
What to say next

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

7

Why We Need Dialogue Models?

- System and user are working on a task
(vs. Eliza-style conversation)
task structure --> dialogue structure
- But:
 - Dialogue does not need to follow task-steps
 - There is more to dialogue than task, e.g.,
grounding

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

8

Methods of DM

- Finite state machine-based
 - Sequence of pre-defined steps (dialogue script)
- Frame-based (also: form-filling)
 - Set of slots to be filled (task template)
- Agent-based
 - Joint problem solving by collaborating agents
 - Recognition, construction and execution of plans



6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

9

Finite State Machine Based Models

- Set of states and transitions
- State determines system utterance
- User utterance determines transition to next state (deterministic)
- Fixed dialogue script
- System-driven interaction

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

10

Finite State Machine (Automaton)

- $\langle \text{States, Init-State, Alphabet, Transition-fctn} \rangle$
- Variants: machines having
 - actions associated with states (Moore machine)
 - actions associated with transitions (Mealy machine)
 - multiple start states
 - transitions conditioned on no input symbol (a null)
 - more than one transition for a given symbol and state (nondeterministic finite state machine)
 - states designated as accepting states (recognizer)
 - etc.

See, e.g., NIST <http://www.nist.gov/dads/HTML/finiteStateMachine.html>

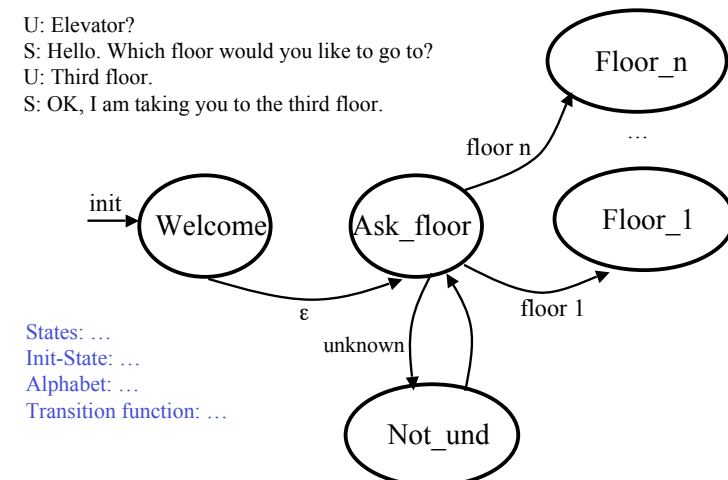
6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

11

FSM-Based Models

U: Elevator?
S: Hello. Which floor would you like to go to?
U: Third floor.
S: OK, I am taking you to the third floor.



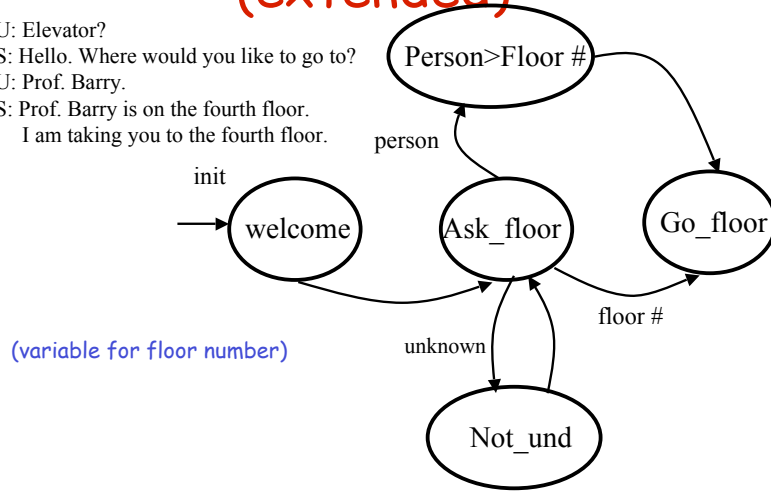
6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

12

FSM-Based Models (extended)

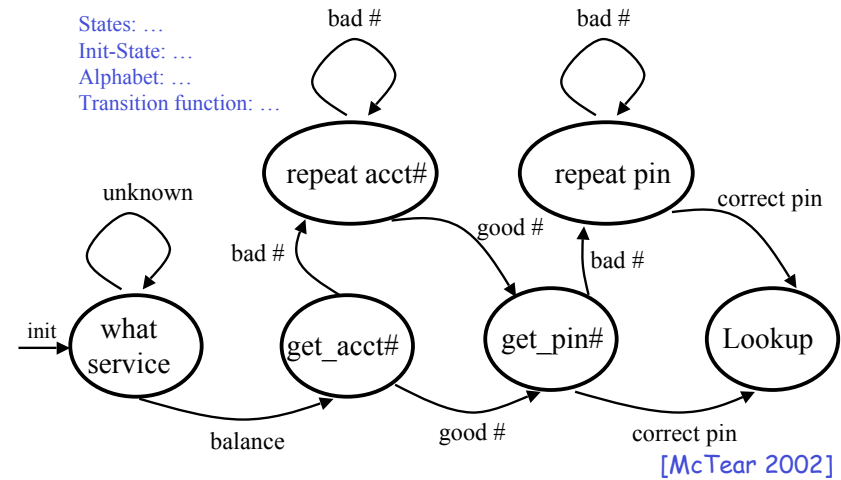
U: Elevator?
 S: Hello. Where would you like to go to?
 U: Prof. Barry.
 S: Prof. Barry is on the fourth floor.
 I am taking you to the fourth floor.



(variable for floor number)

FSM-Based Models

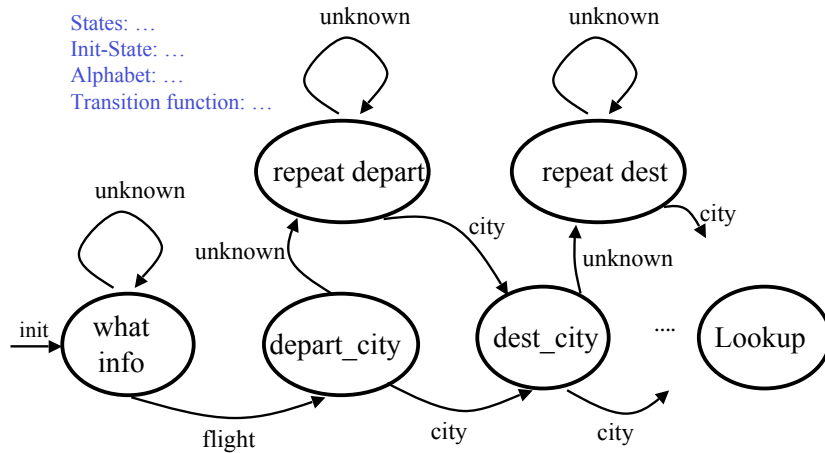
States: ...
 Init-State: ...
 Alphabet: ...
 Transition function: ...



[McTear 2002]

FSM-Based Models

States: ...
 Init-State: ...
 Alphabet: ...
 Transition function: ...



FSM-Based DM: Sum Up

- Advantages
 - Fixed prompts can be pre-recorded
 - Speech recognition and input interpretation can be tuned for each state
- Disadvantages
 - Very rigid dialogue flow
 - Inhibiting user initiative
 - In principle can make more flexible, but it quickly gets very complex
 - Only suitable for simple fixed tasks

Frame-Based Models

- Frame (form): what info should be supplied by user
- Dialogue states: which slots are filled
- General routines for what system should do next (given which slots are filled)

departure_city	?
departure_date	?
destination_city	?
return_date	?
...	

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

17

Frame-Based Models

S: What can I do for you?
U: I want to fly to Paris

departure_city	?
departure_date	?
destination_city	Paris
return_date	?
...	

S: Where will you fly from?
U: From Berlin on August 1st.

departure_city	Berlin
departure_date	1/8/05
destination_city	Paris
return_date	?
...	

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

18

Frame-Based Models

- Deciding what to do next
 - Next unfilled slot (similar to FS-based)
 - Slot-combination weighting
 - Ontology-based coherence
- Database lookup
 - Delayed (typically; after certain slots filled)
 - Immediate (can be "expensive", but helpful)

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

19

Ontology-based coherence

1. S: What is the patient's sex?
U: Female with severe nipple discharge
S: What is the patient's age?
U: Fifty five
S: Is the discharge bilateral?
U: No
2. S: What is the patient's sex?
U: Female with severe nipple discharge
S: Is the discharge bilateral?
U: No
S: What is the patient's age?
U: Fifty five

[Milward&Beveridge 2003]

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

20

Delayed vs. Immediate Lookup

S: What can I do for you?
U: I want to fly from Berlin to Paris.
S: When would you like to fly?
U: April 1st.
S: At what time would like to fly?
U: In the morning.
S: Sorry. There is no flight from Berlin to Paris on April 1st in the morning.
Would you like to fly later?
U: OK.
S: There is one flight. It leaves Berlin at 3 PM and arrives to Paris at 5 PM.

S: What would you like to hear?
U: Play Yesterday.
S: There are 75 songs called Yesterday. Which artist would you like?
U: The Beatles.
S: Sorry, I do not have Yesterday by the Beatles. Would you like another artist?
U: No. Thank you.

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

21

VoiceXML

- VoiceXML is a web-based markup language for representing spoken dialogs
- VoiceXML application collects and processes info, and plays back info
- VoiceXML assumes a voice browser
 - Info conveyed to user by audio output (synthesized and/or recorded)
 - Info received from user as audio input (voice and/or telephone keypad tones)
 - (analogical to HTML)

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

22

VoiceXML

- Main elements of a VoiceXML document
 - Form: basic unit of functionality
 - Field: prompts for and accepts user input
 - Prompt: sequence of audio elements or TTS messages
 - Audio: audio file or TTS message to play
 - Filled: processes input, can pass control to other forms
- Form Interpretation Algorithm
 - Defines how fields in a form are filled in , and how the fill ordering can be modified
- Global event handlers for (e.g., error handling, help)
 - Define behavior when predefined global conditions occur
- Control transfer conditional and subroutine constructs (= special-purpose programming language)

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

23

VoiceXML Example

See VoiceXML tutorials

<http://www.palowireless.com/voicexml/tutorials.asp>

e.g.,

http://www.vocomosoft.com/voicexml_tutorial.htm

Or Chapters 1 and 2 of

<http://cafe.bevocal.com/docs/tutorial/index.html>

give good first steps

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

24

Frame-Based DM: Sum Up

- Advantages
 - Enables some user initiative
 - More flexible
- Disadvantages
 - Speech recognition more difficult, because user input less restricted
 - Not every task can be modeled by a frame

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

25

Agent-Based DM

- (Collaborative) problem solving by (rational) agents
- Agents have knowledge about solving tasks
 - deciding on goals (objectives): adopt, select, defer, abandon, release
 - forming plans to achieve goals (recipes)
 - executing those plans (acting)
 - revising decisions (replanning, abandoning goals, etc.)
- Agents communicate to establish common ground
- Agents reason about beliefs and actions
- Intention recognition

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

26

Collaboration

- Communication is a joint activity: agents collaborate to establish and achieve their goals
- Neither agent can accomplish the task alone
--> mixed initiative
- Need joint goals and mutual understanding
--> cooperation
--> grounding

6/14/05

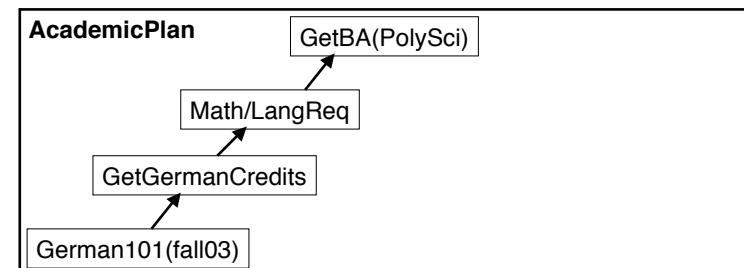
Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

27

Intention Recognition

Given: plan for getting a BA

U: I'll take German 101 fall semester.



6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

28

Interleaved Planning&Acting

User: Send ambulance one to Parma right away.
(initiate (c-adopt (action (send amb1 Parma))))
(initiate (c-select (action (send amb1 Parma))))
System: OK. [sends ambulance]
(complete (c-adopt (action (send amb1 Parma))))
(complete (c-select (action (send amb1 Parma))))
System: Where should we take the victim once we pick them up?
(initiate (c-adopt (resource (hospital ?x))))
User: Rochester General Hospital.
(continue (c-adopt (resource (hospital RocGen))))
System: OK.
(complete (c-adopt (resource (hospital RocGen))))

[Blaylock et al. 2003]

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

29

Agent-based

- Advantages
 - Flexibility and adaptivity
 - Any task can be modeled
- Disadvantages
 - Intention recognition
 - Lots of reasoning
 - > see QUD-based for "shortcuts"

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

30

Dialogue Modeling as Information State Update

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

31

Information State (TRINDI)

- Representation of current state of dialogue
- Used by system to
 - Interpret user's turn
 - Decide which external actions to take
 - Decide what to say (and when to say it)
 - Store information (dialogue context representation)
- Utterances update information state
- Approaches to DM differ in how IS is represented, what role it plays, what it contains

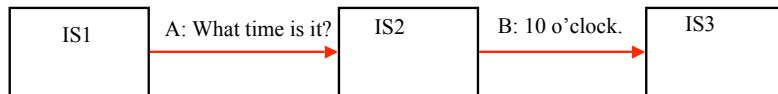
6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

32

IS Update Rules

- Describe possible transitions from one information state to the next
 - If <conditions-on-IS-values>
 - then <changes-to-IS-values>



- When applicable
- What IS change

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

33

FS as ISU

- IS: current-state; input
- Update rules:
 - If [state] & [input]
 - then [output]; [next-state]
- Example for elevator or account-balance

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

34

Frame-Based as ISU

- IS: task-frame; user's move; system move
- Update rules: e.g.,
 - If [user move slot X value V] then [fill X with V]
 - If <conditions-on-frame-values>
 - then <ask-slot-value Y>
- Decision about next system move is also a rule
- Example for travel dialogue

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

35

Frame-Based as ISU

S: What can I do for you?
U: I want to fly to Paris

departure_city	?
departure_date	?
destination_city	Paris
return_date	?
...	

S: Where will you fly from?
U: From Berlin on August 1st.

departure_city	Berlin
departure_date	1/8/05
destination_city	Paris
return_date	?
...	

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

36

General ISU-Based Models

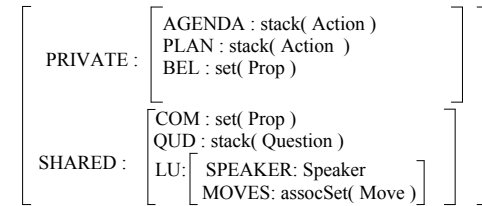
- Projects: Trindi, Siridus, D'Homme, BEETLE, TALK, ...
- Task- vs. Dialogue-Structure
 - Task --> dialogue
 - But, dialogue does not have to follow task (execution) structure
- "Dialogue planning" (agenda)
 - Task model fills agenda with task-related goals
 - Dialogue manager can add more goals, e.g., for grounding
- Approaches:
 - QUD-based (Godis)
 - Obligation-based (Edis)

6/14/05

37

QUD-Based ISU Modeling

- Information state:



+ module interface variables

INPUT : String
LATEST-MOVES: Set(Move)
LATEST-SPEAKER: Speaker
NEXT-MOVES: Set(Move)
OUTPUT: String

6/14/05

38

QUD-Based ISU Modeling

- U: "how much does a flight cost?"
- if user asks Q, push respond(Q) on AGENDA
 - if respond(Q) on AGENDA and PLAN empty, find plan for Q and load to PLAN
 - if findout(Q) first on PLAN, ask Q
- S: "where do you want to go?"
- U: "Paris"
- if LM=answer(A) and A **about** Q, add P=Q[A] to SHARED.COM
 - if P in SHARED.COM and Q topmost on QUD and P **resolves** Q, pop QUD
 - if P in SHARED.COM and P **fulfils goal** of findout(Q) and findout(Q) on PLAN, pop PLAN

6/14/05

39

QUD-Based ISU Modeling

- Sample dialogue plan:

```
findout(?x.transport(x))
findout(?x.dest-city(x))
findout(?x.depart-city(x))
findout(?x.dept-month(x))
findout(?x.dept-day(x))
findout({?class(economy), ?class(business)})
consultDB(?x.price(x))
respond(?x.price(x))
```

⇒ system's agenda

6/14/05

40

QUD-Based ISU Modeling

- IS update rule for answer integration:
integrateAnswer

```
pre: {
  in($SHARED.LU.MOVES, answer(A))
  fst($SHARED.QUD, Q)
  $DOMAIN:about(A, Q)
}
eff: {
  DOMAIN: combine(Q, A, P)
  add($SHARED.COM, P)
}
```

- Before an answer can be integrated by the system, it must be matched to a question on QUD

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

41

QUD-Based ISU Modeling

...

S: "what class did you have in mind?"

U: "cheap"

- if consultDB(Q) on PLAN, consult database for answer to Q; store result in PRIVATE.BEL
- if Q on QUD and P in PRIVATE.BEL s.t. P resolves Q, answer(P)

S: "The price is £123"

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

42

QUD-Based ISU Modeling

- Dealing with multiple issues:
 - if user asks Q, push Q on QUD and load plan for dealing with Q
 - if users asks Q' while system is dealing with Q, **throw out plan for Q** but Q remains on QUD
 - when Q' resolved, Q topmost on QUD will trigger reloading plan for dealing with Q
 - general rule: if SHARED.COM contains info resolving Q, don't ask Q
 - so any resolved questions in plan will be thrown out

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

43

QUD-Based ISU Modeling

U: I want price information [raise ?x.price(x)]

S: Where do you want to go?

U: London

S: When do you want to travel?

QUD=<?x.dept-month(x), ?x.price(x)>

U: Do I need a Visa? [raise ?visa]

QUD=<?visa, ?x.dept-date(x), ?x.price(x)>

S: Where are you travelling from?

U: Gothenburg

S: No, you don't need a Visa.

ISSUES=<?x.dept-month(x), ?x.price(x)>

PLAN empty; QUD empty

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

44

QUD-Based ISU Modeling

S: No, you don't need a Visa.
QUD=<?x.dept-date(x), ?x.price(x)>
U: OK, I want to leave in April [answer dept-month(april)]
QUD=<?x.price(x)>
PLAN empty, so reload plan for dealing with ?x.price(x)
Throw out all question which have already been resolved; raise the first unresolved question on plan
S: What day do you want to leave?
...

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

45

QUD-Based ISU Modeling

- What all can be done...
 - Accommodation ("overanswering")
 - Reraising
 - Task switching, sharing information across tasks
 - ...
- But no planning: static dialogue plans, not much work done on those
--> integrate with ideas in agent-based, where focus on task planning = current research

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

46

Grounding

Establishing common ground

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

47

Grounding

- Grounding problems are due to
 - Lack of perception or understanding
 - Ambiguity
 - Conflicts
 - Misunderstanding
- Decision:
accept/reject/verify/clarify/repair/ignore ...
- Clarification and repair strategies, e.g., ask for repetition, rephrase, clarify

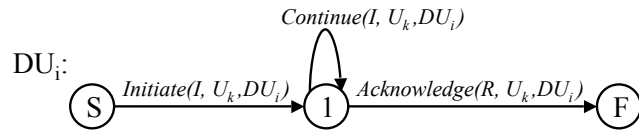
6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

48

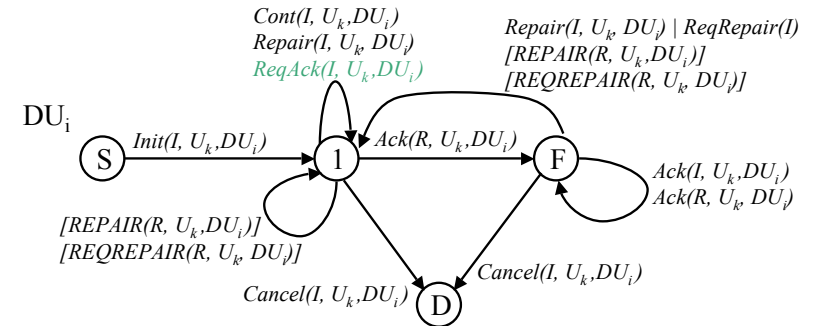
Grounding Acts (Traum 1998)

- What is the function of utterance U_k ?
- Does U_k initiate, continue or complete a discourse unit DU_i ?



Discourse unit (DU_i): unit of (to be) grounded content

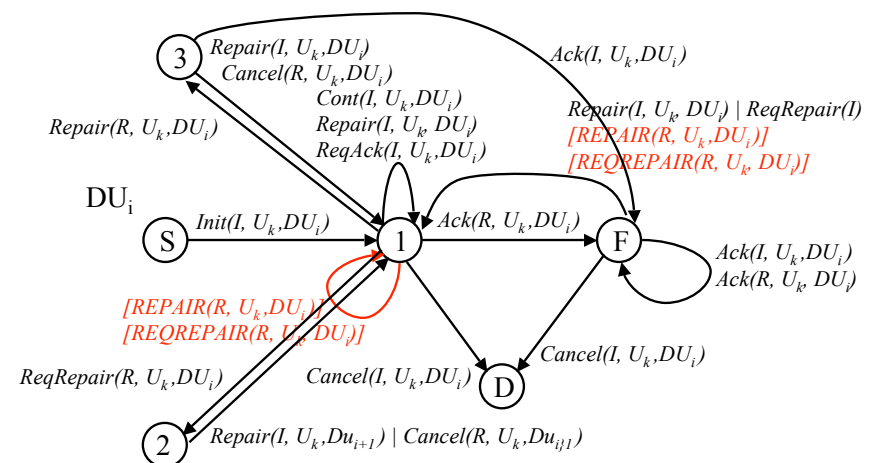
Grounding Acts



Grounding Act Examples

- 1:A: Move the boxcar to Corning
- 2:A: and load it with oranges
- 3:B: OK

Finite Model



Grounding Strategies

- Assuring correct understanding
 - Pessimistic strategy:
 - Immediate explicit verification
 - Terribly inefficient
 - Optimistic strategy
 - Delayed accumulated verification
 - Difficult to recover from errors
 - Error-chaining
 - Carefully optimistic strategy
 - "Implicit" verification by incorporating info to be grounded in next system turn

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

53

Verification Strategies in Systems

- Immediate explicit feedback (and verification request)
 - S: Where do you want to go?
 - U: Hamburg.
 - S: Traveling to Hamburg. (OK?)
 - U: Yes.
 - S: When do you want to go?
- Immediate "implicit" feedback by incorporating material to be grounded in the next system turn (see if user accepts or protests)
 - S: Where do you want to go?
 - U: Hamburg.
 - S: And when do you want to go to Hamburg?
- Delayed explicit feedback by summarizing at task end
 - ...
 - S: So. Traveling from Saarbrücken to Hamburg on Monday June 6
 - ...

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

54

Choice of Verification Strategy

- ASR confidence below/above threshold
- Pragmatic Plausibility
 - Combining ASR confidence with task interpretation confidence (plausible actions in context)
- Context-adapted strategies
 - Reinforcement learning: learn optimal strategies from annotated data, based on rewards for efficient dialogue and user satisfaction

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

55

Dialogue Move/Act Recognition

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

56

Dialogue Act Recognition

- Utterance features, e.g., keywords, syntactic mood, intonation, etc.
- Previous dialogue acts
- In finite state machine-based models, dialogue acts correspond to the arcs (transitions) for user turns, and to states for system turns
- In form filling, dialogue acts correspond to prompts and “answers” filling slots; additional DAs for error handling etc.
- In ISU-based approaches, there are update rules that correspond to DAs

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

57

Future Challenges

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

58

Key Issues for the Future

- Pervasive systems
 - Systems need to handle distributed dialogues (shifts to different dialogue situations / managers), concurrent dialogues (issues of co-ordination, synchronisation, redundancy); interaction model needs to be predominantly event-based (external events, opportunistic)
- Adaptivity:
 - Systems need to be dynamically adaptive in a number of different ways: to the environments in which they are used (modality), to their user's preferences and needs (personalisation), and to changes in task and context.
- Ability to learn:
 - Systems need to be able to learn from interactions with users in order to provide an optimally usable interface that matches the current environment and user.
- Standardisation:
 - There is a need for a common set of standards to support re-usability for developers and to support usability for the users of spoken dialogue systems.

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

59

Adaptivity and Learning

- Adaptivity (TOOT, Interact, Kyoto Bus, Circuit Fix-It Shop)
 - Dynamically adjust dialogue strategy e.g. change from mixed initiative to system initiative
 - Based on problems detected e.g. poor speech recognition, user corrections (dialogue history)
 - Based on prediction rules learned in a training phase
 - Based on user model and / or context model
- Reinforcement learning
 - learning optimal dialogue strategies e.g. constraining v open-ended prompts, explicit v implicit verification

6/14/05

Ivana Kruijff-Korbayová
Language Technology II: Dialogue Management

60

References

- D. Jurafsky and J. Martin (2000): *Speech and Language Processing*, Chapter 19.
- McTear (2002): Spoken Dialogue technology. In *ACM Surveys*, pp. 1-80
- VoiceXML Forum: <http://www.voicexml.org/>
- Traum and Larsson (2003): The Information State Approach to Dialogue Management. In Smith and Kuppevelt (eds.): *Current and New Directions in Discourse & Dialogue*, Kluwer Academic Publishers. (pp. 325-353, 28 pages)
- D. Traum (1998): A computational model of grounding. *AAAI Fall Symposium on Psychological Models of Communication in Collaborative Systems*.
- R. San-Segundo et al. (2001) *Designing Confirmation Mechanisms and Error Recover Techniques in a Railway Information System for Spanish*. *SigDial Workshop*.