Towards Conversational Human-Computer Interaction

James F. Allen, Donna K. Byron, Myroslava Dzikovska, George Ferguson, Lucian Galescu, Amanda Stent

AI Magazine, Winter 2002

Presented By
Cory Lueninghoener
March 13, 2002

Outline

- Background
- Complex interfaces
- Practical dialogue
  - Description
  - Example
  - Challenges
- TRIPS, an interactive planner
- Multimedia extravaganza
- Conclusion

Background

- Speech acts: communication actions
  - Speaker, hearer, utterance
  - Want ability to: inform, query, answer, command, promise, acknowledge, and share
- Speech acts need to be parsed
- Communication steps
  - Speaker: Intention, generation, synthesis
  - Hearer: Perception, analysis, disambiguation, incorporation

- Two communication models: encoded message and situated language

What is the best design for a complex interface?

- GUIs become cluttered

- Speech is much easier for a human user

- A 1-1 GUI to Speech translation doesn't work

File Edit View Search Bookmarks Help

- Solution: Dialogue-based interaction.

- Example: The Rochester Interactive Planning System (TRIPS)

Practical Dialogue

- Dialogue focused on accomplishing a concrete task
  - Task-Oriented
  - Information Seeking
  - Advice and Tutoring
  - Command and Control

Practical Dialogue Hypothesis: Conversational competence required for practical dialogues is simpler to achieve than general conversational competence.

Domain-Independence Hypothesis: The bulk of complexity in language interpretation and dialogue management is independent of the task being performed.

Practical Dialogue - Example

(1) USR: We need to get the woman in Penfield to Strong.

(2) SYS: OK

(3) USR: What vehicles are available?

(4) SYS: There are ambulances in Pittsford and Webster.

(5) USR: OK. Use the one from Pittsford.

(6) SYS: Do you know that Route 96 is blocked due to construction?

(7) USR: Oh

(8) USR: Let's use the interstate instead.

(9) SYS: OK. I'll dispatch the crew.

Four Challenges for Dialogue Systems

1. Parsing languages into practical dialogues

2. Integrating dialogue and task performance

3. Intention recognition

4. Mixed initiative dialogue

1. Parsing language into practical dialogues

- Difficult problem; Past work has focused on structure rather than meaning

- Ambiguity is bad
  - Helped through language constraints
  - “Eat chip” $\rightarrow$ corn chip, not computer chip

- Utterance based vs. Sentence based
  - Gives more context
  - “I went to the grocery store. I got chips.”

2. Integrating dialogue and task performance

- System needs to be flexible
- Separate agent-based backend from dialogue-based frontend.
- Generic underlying dialogue architecture.
  - Objectives
  - Solutions
  - Resources: objects, abstractions
  - Situations
- Utterances manipulate these concepts,

  manipulate = {create, modify, delete, evaluate, describe} $\times$ {objecties, solutions, resources, situations}

3. Intention recognition

- What is the intent behind an utterance?

- Example: Moving people from Abyss to Delta using truck one.

  - "Can we use a helicopter to get the people from Abyss?"

  - "Can we use a helicopter to get the people at Barnacle?"

  - "Can we use a helicopter to get the people from Delta?"

- TRIPS: two-level approach

  - Figure out possible intentions

  - Prune according to current situation

4. Mixed initiative dialogue

- People talk back and forth in real life

- Easy solutions:
  - Fixed system initiative
  - Fixed user initiative

- Easy solutions are too restrictive

- Mixed Initiative is more complex
  - Who is in control?
  - How do you handle interruptions?

TRIPS: The Rochester Interactive Planning System

- Prototype practical dialogue system developed by the authors

- First studied human-human interaction

- Started simple, incrementally adding features

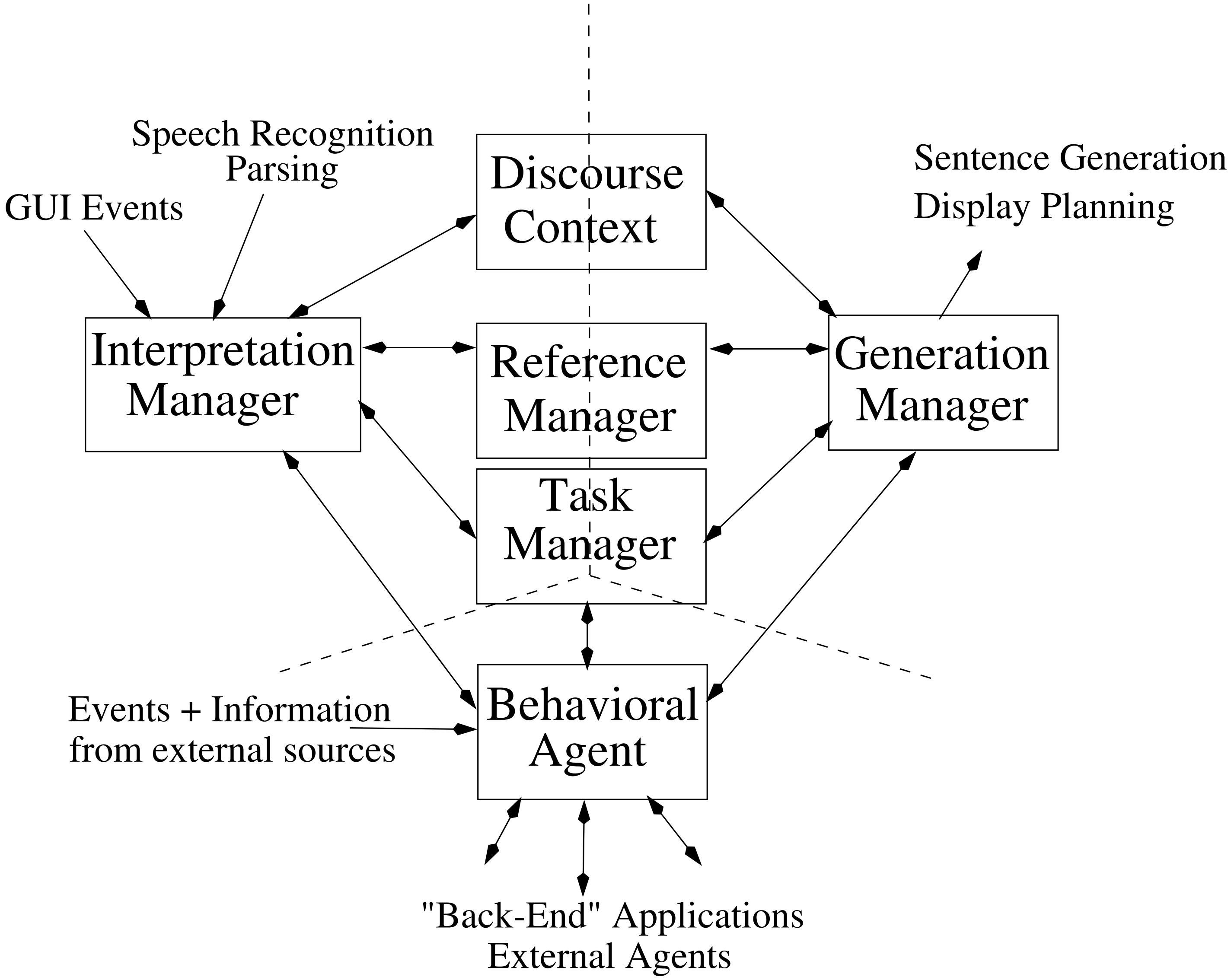
- Three areas of component functionality

  1. Interpretation

  2. Generation

  3. Behavior

1. Interpretation Manager
2. Behavioral Agent
3. Generation Manager



TRIPS: The Rochester Interactive Planning System

1. Parser takes in utterances, produces speech acts

2. Parsed info passed to Interpretation Manager

   (a) Reference Manager uses context/prior knowledge to choose referents

   (b) Task Manager helps figure out the intention

   (c) A response is formulated

3. Behavioral Agent comes up with actions

4. Generation Manager decides what to do with the new info

Multimedia Extravaganza

Conclusion

- For some applications, dialogue-based interfaces are the best kind

- Complex design issues

- Simplifications help

  - Focus on practical dialogue

  - Language constraints

  - Utterance based parsing

  - Generic dialogue architecture

  - Context recognition

- These methods can be used in realistic implementations and actually be useful