Natural Language and Dialog

Artificial Intelligence for Interactive Media and Games

Professor Charles Rich
Computer Science Department
rich@wpi.edu

Outline

- Computational theory of human language and dialog
  - background
  - terminology
- Language and dialog in games
  - current common industry practice
  - emerging trends
- Speech
What is Dialog?

- a conversation between two participants
  - verbal communication
    - spoken or written
    - what about non-verbal components?
  - at least two turns
    - each turn consists of one or more utterances
    - not necessarily complete sentences
    - backchannels (uh-huh), overlapping, interruptions
  - what about more than two participants?
    - more complex turn-taking rules
    - dialog is a two-person discourse

- in a shared context
  - not just any random utterances
  - e.g., a story or collaboration

What is the Purpose of Dialog?

- contributes to participants’ goals in the context
  - example
    - my goal (desired world state) is for the window to be open
    - I say “Please open the window” to person standing next to the window
    - if person is cooperative, she says “Ok”
    - she opens the window
  - is it still a dialog if she skips saying “Ok”? 
    - yes, if she opens the window (nonverbal response)
    - notice interleaving/coordination of communication (utterances) and action (world state changes)
What is the *Immediate* Purpose of Dialog?

- the *speaker* is trying to achieve a change in the *mental state* of the *hearer*, including:
  - emotional state
    - "your mother wears army boots"
    - "I love you"
  - beliefs
    - "roses are red"
    - "I'm scared"
  - goals (intentions)
    - "please open the window"
    - "don't look in there"
What about Questions?

- “What time is it?”
  - speaker’s goal is to change his own mental state
    - to one in which he knows the time
  - speaker could achieve goal by looking at clock
  - but if no clock, can achieve goal indirectly
    - by changing mental state of hearer (with wrist watch)
    - to include goal of telling speaker the time
  - in other words: “Please tell me the time.”

Levels of Language Representation

1. Sound Waves (Speech)
2. Surface Form (Words)
3. Syntax
4. Semantics
5. Pragmatics

many-to-one mappings from each level to next
- multiple surface forms with same syntax
- multiple syntactic forms with same semantics
- etc.
1. Surface Form

- The sequence of words that are actually written, read, spoken or heard
- Two utterances, e.g., in two different languages, may differ in their surface forms, but have the same meaning:
  - *English*: “the roses are red”
  - *French*: “les roses sont rouges”
- Or even in the *same* language:
  - *Active*: “John kissed Mary”
  - *Passive*: “Mary was kissed by John”

2. Syntax

- sequence of words $\rightarrow$ tree
- *Parsing* (“diagramming”) a sentence in terms of:
  - *part of speech tags*: adjective, preposition, noun, etc.
  - *syntactic roles*: subject, verb, (direct/indirect) object, etc.

```
art  adj  adj  n  v  prep  art  adj  n
```

“The quick brown fox jumped over the lazy dog.”

```
[S
  [NP The quick brown fox]
  [VP jumped
    [PP over
      [NP the lazy dog]]]]
```
3. Semantics

- The meaning of an utterance in isolation
- Much less standardized than syntax
  - frame-based semantics
  - logical (axiomatic) semantics
  - probabilistic semantics
  - etc., etc.
- Two sentences with different surface form and different syntax may have same semantics
  - “John kissed Mary.”
    \[S [NP John] [VP kissed [NP Mary]]\]
  - “Mary was kissed by John.”
    \[S [NP Mary] [VP was kissed [PP by [NP John]]]\]
  - frame semantics:
    \{ action: kiss, agent: John, theme: Mary, time: past \}

4. Pragmatics

- Everything else about how the utterance functions in its context
- Even less standardized than semantics
  - E.g., goal/belief modification pragmatics
(Spoken) Language Understanding

- Start with (sound wave or) words
- Compute pragmatic function

Speech Recognition

Speech Understanding

- (Perhaps) mapping through syntactic and semantic forms along the way...

Language Generation

- Start with pragmatic (deep) representation
- Output surface form

- (Perhaps) mapping through semantic and syntactic form along the way...
State of the Art in Academic Research

• unrestricted language (speech) understanding input
• constrained domain
• full syntactic parsing, semantic interpretation
• pragmatics
• general-purpose language generation

http://www.cs.rochester.edu/research/cisd/projects/plow/

VIDEO

State of the Art in Academic Research

• unrestricted (spoken) language understanding input
• constrained domain
• statistical approach
• canned language generation (voice acting)

http://ict.usc.edu/projects/responsive_virtual_human_museum_guides/C40

VIDEO


Language Understanding Challenges

- **Coverage**
  - you can make almost anything work if you restrict the domain enough
    - know all the words that will be used
    - know all the purposes (pragmatics)
  - e.g., airline reservation system
  - but not the Turing Test

- **Semantics**
  - lack of agreement inhibits generalization and sharing of results

Language Generation Challenges

- **Expressiveness**
  - how to say the same thing with different styles, emotional content, etc.
  - e.g., “Hello” vs. “Yo, dude”
  - need computational theory which separates *style* and *content*

- **Coherence**
  - generation needs to have wider window than single utterance
  - planning a sequence of utterances (anaphora, etc.)
Dialog in Games

- In what genres is dialog most important?
  - role playing games (RPG)
  - text adventure (interactive fiction - IF)
  - first person shooters (FPS)
  - real-time strategy (RTS)
  - sports? casual? serious? ...

Dialog between Whom?

- player ⇔ NPC
  - main challenge and research focus
  - “dialog trees” commonly used
- NPC ⇔ NPC
  - player is bystander
- player ⇔ player
  - e.g., in MMO’s
  - no problem for humans on both ends
  - system/NPC as bystander?
Player-NPC Dialog

- Two computational problems to solve
  - generating NPC utterances
  - understanding player utterances

- Dialog trees
  - common solution to both at the same time
  - all possible player and NPC utterances authored in advance
  - decision tree based on user choices

Dialog Tree implemented as script

```python
Speak("Welcome stranger. What brings thee among us gentle folk?")

reply = player.SpeakOption(
    1, "Yo dude, wazzup?",
    2, "I want your money, your woman and that chicken")

if reply == 1 then
    Speak("Wazzuuuuup!")
else if reply == 2 then
    Speak("Well, well. A fight ye wants, is it? Ye can't just go around these parts demandin' chickens from folk. Yer likely to get that ugly face smashed in. Be off with thee!")
end
```

[From Buckland, Chapter 6]
Dialog Trees

- Advantages
  - *fast & flexible* – code can do anything
  - *reliable* – no misunderstandings
  - *expressive* – author has complete control to create desired style, character, atmosphere, etc.

- Disadvantages
  - restricts player
  - very labor intensive
  - doesn’t scale well to complex interactions
    - must keep variability down to keep labor down
    - leads to lack of replayability
    - can help somewhat by designing special editors and engines for executing dialog trees

XML Dialog Tree

```xml
<say id="Shelter" actor="sidekick" text="Not so fast. I can't walk much further today, and the weather's getting worse">
  <say actor="player" text="Okay. What should we do, then?">
    <say actor="sidekick" text="We need to build a shelter for the night">
      <say id="Floor" actor="player" text="Let's use pieces of that wreck to build a hut">
        <say actor="sidekick" text="Okay, the floor is flat already, so what should we build first?">
          <do task="BuildWalls" apply="world.get('shack').hasPillar('right')" />
            <say actor="sidekick" text="Do we want pillars at the front?" />
              <say actor="player" text="Sure, let's go for it!">
                <do task="BuildPillars" />
                  <say actor="sidekick" text="Now all that's left is the roof!">
                    <do task="BuildRoof" />
                      <say actor="sidekick" ref="Roof" />
                        <do task="Roof_tree" apply="world.get('shack').hasPillar('right')" />
                          <say actor="sidekick" text="Glad that's over!">
                            <do task="BuildPillars" />
                              <say actor="sidekick" text="They look cool, but we definitely need walls">
                                <do task="BuildWalls" />
                                  <say actor="sidekick" ref="Roof" />
                                    ...
```

[Disco for Games (D4g)]
Morrowind (The Elder Scrolls) Dialog Editor

- database of utterances
  - organized around extensible set of “topics”
  - each utterance has conditions and results
  - including menu choices
  - simple templating via variables (name, etc.)
  - sort of like programming a rule-based system

http://tommyshideout.net/files/srikandi/DialogueTut
Text Adventure Games

- also called “interactive fiction” (IF)
  - confusing with “interactive drama” and “interactive storytelling”
- started with *Adventure* in 1975
  - Infocom’s *Zork* series in 70’s and 80’s
- annual Interactive Fiction Competition
- restricted syntax and semantics
  - few hundred standard verbs
  - open-ended nouns
  - simple (learned) command syntax
  - highly evolved set of ad hoc techniques

---

> WAIT
Time passes...

The scooter glides into the station’s docking port. The retro-thrusters bring the scooter to a halt. As the docking port fills with air the scooter's hatch opens.

> LOOK
Scooter, in the pilot's couch
You are in the cramped one-man space scooter. Through the viewport, you see the docking port of Space Station BG-12. The scooter's hatch is open. You can see a toolbox here.

> OPEN THE TOOLBOX
You can't reach it from the pilot's couch.
Text Adventures Games

>`STAND UP
You are standing again.

>`OPEN THE TOOLBOX
It's locked.

>`LEAVE THE SCOOTER
Docking Port
This is the huge docking port of the space station. Only one ship is here at the moment, a one-man scooter, so the docking port seems unusually empty. A tube leads down toward the heart of the station,

>`DOWN
Main Hallway, Sector M
This is the station's main corridor, which continues to port and starboard. A tube leads up toward the docking port.

Emerging Trends in Game Dialog

- Natural language understanding
  - replacing fixed menu choices
  - give player more flexibility to express herself

- Natural language generation
  - generating NPC utterances procedurally
  - reduces authoring labor

- Speech
Façade

“Classic” (but still state of the art) game experiment in text natural language understanding

- unrestricted text input
- micro-domain (very constrained)
- go directly from surface form to pragmatic effect
- broad, shallow, author-intensive techniques
- cheating strategies when doesn’t understand

Façade – Surface Text Rules

- word spotting and pattern matching rules
  - dialog acts (pragmatic)
    - (“hello” | “hi”) [ “there” ] ➔ Hello
    - “grace” ➔ Character(Grace)
    - Hello && Character(?char) ➔ Greet(?char)

- example dialog acts:
  - Agree(?char), Disagree(?char)
  - Express(?char, ?emotion)
  - ReferTo(?char, ?object)
ANDI-Land

http://www.andi-land.com
“Logical Agents for Language and Action”,
M. Magnusson & P. Doherty,
Linkoping U., Sweden, AIIDE’08

- restricted natural language text input
  - using context-free grammar
  - shows user possible syntactic completions as player types
  - underlying logical theorem-prover
  - all output generated procedurally

ANDI-Land

Magni: “Who owns the axe?”
  ↓ parsing
  [S Who [VP owns [NP the axe]]]
  ↓ semantic interpretation
  informRef(magni, value(12:15, owner(axe)))
  ↓ theorem proving
  inform(magni, Id(value(12:15, owner(axe)), smith))
  ↓ reversible grammar

Smith: “I own the axe.”
**ANDI-Land**

**Magni**: “Sell the axe to me.”

- **parsing**
  \[S \ [VP \ sell \ [NP \ the \ axe] \ [PP \ to \ me]]\]

- **semantic interpretation**
  \[\exists t_1, t_2 \ [\text{Occurs}(\text{smith}, (t_1, t_2), \text{sell}(\text{axe, magni})))\]

- **theorem proving**

- Committed(\text{smith, } t_1, \text{Occurs}... ) \land
- Executable(\text{smith, } (t_1,t_2), \text{sell(axe,magni)}) \land
- Believes(\text{smith, } t_1, \text{ActionId}(\text{sell(axe, magni)}, \text{sell(axe, magni)})) \Rightarrow
  \text{Occurs}(\text{smith, } (t_1,t_2), \text{sell(axe, magni)}))

---

**Natural Language Generation**

- Generating **NPC to NPC** dialog for Interactive Storytelling
  - no pre-authored dialog
  - situations generated by autonomous planning agents
  - using logic and templates to generate surface forms

http://www-scm.tees.ac.uk/f.charles

---

http://www-scm.tees.ac.uk/f.charles
Speech

- Speech recognition
- Speech generation
- Speech in games
  - experiments with player speech input
  - NPC speech output almost always recorded

Speech Recognition

- widely available commercial systems
  - all based on HMM (Hidden Markov Models) trained on large corpora
  - built into PCs and smartphones
- easier vs. harder versions
  - isolated word vs. continuous
  - speaker trained vs. speaker independent
  - small vs. large vocabulary
  - grammar-based vs. dictation
  - push-to-talk vs. open-microphone (keyword spotting)
Speech Generation

- text to speech
- widely available commercial systems
  - many different “voices”
  - never sounds as good as recorded voices
  - built into PCs and smartphones
- two approaches
  - concatenative
    - chops up and stitches back together recorded voices
    - usually sounds pretty good
    - a lot of labor to produce each voice
  - model-based
    - uses mathematical model of vocal tract
    - easy to adjust parameters to get different voices
    - less natural sounding

Emotional Speech Generation

- research of Catherine Pelachaud
- same words but different sounds (and gestures) for different emotional states
Clancy’s EndWar

- Ubisoft 2009
- Andi-Land style menu, but using voice

Mass Effect 3

- Ubisoft 2011
- Kinect voice recognition
- Voice selection from regular dialog menus
Alelo Tactical Language

- spinoff of USC research
- very successful serious game

http://tactillanguage.com

Summary

- Natural language and dialog in games
  - academic research techniques mature
  - a lot of interest at points of overlap between academia and industry (e.g., AIIDE)
  - initial experimentation in games mixed
  - potential for breakthrough application in games in next few years