Dialogue Architecture & Project

AIMA: Chapter 22 AND Jurafsky & Martin
Chapter 24, Available in the Library.
Outline

- Project Considerations
- The Linguistics of Conversation
- Basic Conversational Agents
  - NLU
  - Generation
  - Dialogue Manager
- Dialogue Manager Design
  - Finite State
  - Frame-based
  - Initiative: User, System, Mixed
Project Considerations

- Project Teams/Proposals Due Wed Jan 19th
- Teams of 4 undergrads and 1 grad
- On Jan 19th
  - Which of 4 domains.
  - Has to be SMART in some way
  - Sample Dialogue (Goals for capability)
  - Some technical requirements
  - Why you think you can do it in the time
  - How should we evaluate your system?
THE PROJECT

- Domain for dialogue system
  - IMDB Movies
  - Mama Bot Recipe
  - OTTBot (Orc Turing Test for World of Warcraft DB)
  - LastFM Music Database

- Has to be SMART in some way
- Sample Dialogue (Goals for capability)
- Some technical requirements
- How should we evaluate your system?
Agents (AIMA)

- An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators.

- Human agent: eyes, ears, and other organs for sensors; hands, legs, mouth, and other body parts for actuators.

- Robotic agent: cameras and infrared range finders for sensors; various motors for actuators.
PEAS: Taxi Driver Agent

- **Performance measure**: Safe, fast, legal, comfortable trip, maximize profits
- **Environment**: Roads, other traffic, pedestrians, customers
- **Actuators**: Steering wheel, accelerator, brake, signal, horn
- **Sensors**: Cameras, sonar, speedometer, GPS, odometer, engine sensors, keyboard
Conversational Agents

- **AKA:**
  - Spoken Language Systems
  - Dialogue Systems
  - Speech Dialogue Systems

- **Applications:**
  - Travel arrangements (Amtrak, United airlines)
  - Telephone call routing
  - Tutoring
  - Communicating with robots
  - Anything with limited screen/keyboard
I would like to purchase a flight ...

@action=BUY
@origin=New York

Which airport in New York city?

# RequestAirport (NYC)
Text-Based Dialogue System

- Natural Language Understanding
- Dialog Manager
- Task / Database/Data
- Natural Language Generation
- Goal, Dialog Act
- Meaning

Words

NATURAL LANGUAGE AND DIALOGUE SYSTEMS LAB
UC SANTA CRUZ
PEAS: Performance measure, Environment, Actuators, Sensors

Performance measure: Task success, time to completion, user satisfaction

Environment: User, Database of information, Servers

Actuators: Natural Language Generation, Possibly a Screen display, Transactions on DB

Sensors: Keyboard or messaging (user specifies travel goals/requirements/parameters)
**Text-Based Dialogue System**

- **TEAMS:**
  - NLU: 1 person + crowd sourcing or whole team for corpus/testing?
  - NLG: 1 person
  - Task/DB: ½ person unless significant extension of DB or task model targeting DB
  - DM: 1 to 2 people
Agent types: (see AIMA)
Four basic types in order of increasing generality

- Simple reflex agents?
  - What would make it ‘smart’?
  - If it did complex language processing on inputs?
  - Paraphrase generator from any input could be interesting.

- State-based reflex agents?
  - What would make it ‘smart’?
  - Google: keeps some user state but not dialog or task state

- Goal-based agents
  - Model of user, model of task, state representation

- Utility-based agents
  - Often learn **OVER TIME**, keep history of previous dialogs and successes/failures
Simple reflex agents
ELIZA agent (1970’s)

- http://www-ai.ijs.si/eliza-cgi-bin/eliza_script

- Eliza uses pattern matching on user input to produce response, has very little if any state or goals.
Reflex agents with state
Goal-based agents
A travel dialog: Communicator

S1: Hello. You’ve reached the [Communicator]. Tell me your full name
U2: hi i’d like to fly to seattle tuesday morning
S3: Traveling to Seattle on Tuesday, August eleventh in the morning. Your full name?
U4: [John Doe]
S5: Hello, [John], I’m glad to hear from you again. You can now ask me for help.
S6: OK. Let’s see I have a United flight departing Pittsburgh at ten forty five a.m. arriving into Seattle at one fifty seven p.m. Is that OK?
U7: Yes
S8: Will you return to Pittsburgh from Seattle?
U9: no i will go to Los Angeles
S10: Traveling to Los Angeles. on what date
U11: Sorry can we go back to talk about the previous leg again
S12: For the first leg, you are on a United flight 1446 departing Pittsburgh at ten forty five A.M. tomorrow, arriving in Seattle at one fifty seven P.M. Do you want to take that?
U13: actually i want to leave on wednesday
S14: a flight on wednesday.
S15: I have a flight departing Pittsburgh at ten forty five a.m., arrives Seattle at one fifty seven p.m. Is that OK?
U16: Yes

Figure 19.1   The travel domain: a fragment from a successful conversation between a user (U) and the Communicator system (S) of Xu and Rudnicky (2000).
Utility-based agents (can learn utilities)
ELVIS (Walker et al. 1998), NJFun (Litman et al. 2001)

- **S:** Welcome to NJFun, How May I Help You? *(User Initiative)*
- **S’:** Welcome to NJFun, Please say an activity name or say list activities to get a list of activities I know about. *(System Initiative)*
- **U:** I’d like to find some winetasting in Lambertville this morning.
  - **S:** Did you say you are interested in Lambertville? *(Explicit Confirmation)*
  - **U:** Yes.
  - **S:** Did you say you want to go in the morning?
  - **U:** Yes
- **S:** I found a winery in Lambertville that is open in the morning. It is ....
Project Domains: Structured vs. Textual Data

- **Structured:** Restaurant, Air or Train Travel Schedules, Tourist Information
- **Textual:** Wikipedia, User Reviews, Twitter, Online Chat

- Processing unstructured data for backend requires some NLP “smarts”
- But data dynamically processed, up-to-date without ongoing maintenance
- Potentially much harder to make interaction and presentation of information “smart”
- => we will use structured data (but recipes would require a bit of parsing to get smarter interaction)
Outline

- Project Considerations
- The Linguistics of Conversation
- Basic Conversational Agents
  - NLU
  - Generation
  - Dialogue Manager
- Dialogue Manager Design
  - Finite State
  - Frame-based
  - Initiative: User, System, Mixed
Linguistics of Human Conversation

- Turn-taking
- Speech Acts
- Grounding
- Conversational Structure
- Implicature
Turn-taking

- Dialogue is characterized by turn-taking.
  - A:
  - B:
  - A:
  - B:
  - ...

- Resource allocation problem:

- How do speakers know when to take the floor?
  - Total amount of overlap relatively small (5% - Levinson 1983)
  - Don’t pause either
  - Must be a way to know who should talk and when.
Turn-taking rules

- At each transition-relevance place of each turn:
  a. If during this turn the current speaker has selected B as the next speaker then B must speak next.
  b. If the current speaker does not select the next speaker, any other speaker may take the next turn.
  c. If no one else takes the next turn, the current speaker may take the next turn.
Implications of subrule a

- For some utterances the current speaker selects the next speaker
  - Adjacency pairs
    - Question/answer
    - Greeting/greeting
    - Compliment/downplayer
    - Request/grant
  - Silence between 2 parts of adjacency pair is different than silence after
    - A: Is there something bothering you or not?
    - (1.0)
    - A: Yes or no?
    - (1.5)
    - A: Eh
    - B: No.
Speech Acts

- Austin (1962): An utterance is a kind of action
- Clear case: performatives
  - I name this ship the Titanic
  - I second that motion
  - I bet you five dollars it will snow tomorrow
- Performative verbs (name, second)
- Austin’s idea: not just these verbs
Each utterance is 3 acts

- **Locutionary act**: the utterance of a sentence with a particular meaning
- **Illocutionary act**: the act of asking, answering, promising, etc., in uttering a sentence.
- **Perlocutionary act**: the (often intentional) production of certain effects upon the thoughts, feelings, or actions of addressee in uttering a sentence.
Locutionary and illocutionary

- “You can’t do that!”
- Illocutionary force:
  - Protesting
- Perlocutionary force:
  - Intent to annoy addressee
  - Intent to stop addressee from doing something
## The 3 levels of act revisited

<table>
<thead>
<tr>
<th>Locutionary Force</th>
<th>Illocutionary Force</th>
<th>Perlocutionary Force</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Can I have the rest of your sandwich?</strong></td>
<td>Question</td>
<td>Request</td>
</tr>
<tr>
<td><strong>I want the rest of your sandwich</strong></td>
<td>Declarative</td>
<td>Request</td>
</tr>
<tr>
<td><strong>Give me your sandwich!</strong></td>
<td>Imperative</td>
<td>Request</td>
</tr>
</tbody>
</table>
Illocutionary Acts

- What are they?
5 classes of speech acts: Searle (1975)

- **Assertives**: committing the speaker to something’s being the case
  - suggesting, putting forward, swearing, boasting, concluding
- **Directives**: attempts by the speaker to get the addressee to do something
  - asking, ordering, requesting, inviting, advising, begging
- **Commissives**: committing the speaker to some future course of action
  - promising, planning, vowing, betting, opposing
- **Expressives**: expressing the psychological state of the speaker about a state of affairs
  - thanking, apologizing, welcoming, deploring
- **Declarations**: bringing about a different state of the world via the utterance
  - I resign; You’re fired
Grounding

- Dialogue is a collective act performed by speaker and hearer.
- Common ground: set of things mutually believed by both speaker and hearer.
- Need to achieve common ground, so hearer must ground or acknowledge speakers utterance.
- Clark (1996):
  - Principle of closure. Agents performing an action require evidence, sufficient for current purposes, that they have succeeded in performing it.
  - Need to know whether an action succeeded or failed.
Clark and Schaefer: Grounding

- **Continued attention**: B continues attending to A
- **Relevant next contribution**: B starts in on next relevant contribution
- **Acknowledgement**: B nods or says continuer like *uh-huh, yeah, assessment (great!)*
- **Demonstration**: B demonstrates understanding A by paraphrasing or reformulating A’s contribution, or by collaboratively completing A’s utterance
- **Display**: B displays verbatim all or part of A’s presentation
A human-human conversation

C₁: ...I need to travel in May.
A₁: And, what day in May did you want to travel?
C₂: OK uh I need to be there for a meeting that’s from the 12th to the 15th.
A₂: And you’re flying into what city?
C₃: Seattle.
A₃: And what time would you like to leave Pittsburgh?
C₄: Uh hmm I don’t think there’s many options for non-stop.
A₄: Right. There’s three non-stops today.
C₅: What are they?
A₅: The first one departs PGH at 10:00am arrives Seattle at 12:05 their time.
The second flight departs PGH at 5:55pm, arrives Seattle at 8pm. And the
last flight departs PGH at 8:15pm arrives Seattle at 10:28pm.
C₆: OK I’ll take the 5ish flight on the night before on the 11th.
A₆: On the 11th? OK. Departing at 5:55pm arrives Seattle at 8pm, U.S. Air
flight 115.
C₇: OK.
Grounding examples

- **Display:**
  - C: I need to travel in May
  - A: And, what day in May did you want to travel?

- **Acknowledgement**
  - C: He wants to fly from Boston
  - A: mm-hmm
  - C: to Baltimore Washington International
  - [Mm-hmm (usually transcribed “uh-huh”) is a backchannel, continuer, or acknowledgement token]
Grounding Examples (2)

- Acknowledgement + next relevant contribution
  - And, what day in May did you want to travel?
  - And you’re flying into what city?
  - And what time would you like to leave?

- The *and* indicates to the client that agent has successfully understood answer to the last question.
Grounding negative responses
From Cohen et al. (2004)

- System: Did you want to review some more of your personal profile?
- Caller: No.
- System: Okay, what’s next?

- System: Did you want to review some more of your personal profile?
- Caller: No.
- System: What’s next?
Grounding and Dialogue Systems

- Grounding is not just a tidbit about humans
- Is key to design of conversational agent
- Why?
Grounding and Dialogue Systems

- Grounding is not just a tidbit about humans
- Is key to design of conversational agent
- Why?
  - HCI researchers find users of speech-based interfaces are confused when system doesn’t give them an explicit acknowledgement signal
  - Stifelman et al. (1993), Yankelovich et al. (1995)
Conversational Structure

- **Telephone conversations**
  - Stage 1: Enter a conversation
  - Stage 2: Identification
  - Stage 3: Establish joint willingness to converse

<table>
<thead>
<tr>
<th>Stage</th>
<th>Speaker &amp; Utterance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A₁: (rings B’s telephone)</td>
</tr>
<tr>
<td>1,2</td>
<td>B₁: Benjamin Holloway</td>
</tr>
<tr>
<td>2</td>
<td>A₁: this is Professor Dwight’s secretary, from Polymania College</td>
</tr>
<tr>
<td>2,3</td>
<td>B₁: ooh yes –</td>
</tr>
<tr>
<td>4</td>
<td>A₁: uh:mm . about the: lexicology <em>seminar</em></td>
</tr>
<tr>
<td>4</td>
<td>B₁: <em>yes</em></td>
</tr>
</tbody>
</table>
Why is this customer confused?

- Customer: (rings)
- Operator: Directory Enquiries, for which town please?
- Customer: Could you give me the phone number of um: Mrs. um: Smithson?
- Operator: Yes, which town is this at please?
- Customer: Huddleston.
- Operator: Yes. And the name again?
- Customer: Mrs. Smithson
Conversational Implicature

- **A:** And, what day in May did you want to travel?
- **C:** OK, uh, I need to be there for a meeting that’s from the 12th to the 15th.

- Note that caller did not answer question.
- Meaning of caller’s sentence:
  - Meeting
    - Start-of-meeting: 12th
    - End-of-meeting: 15th
  - Doesn’t say anything about flying!!!!!
- What is it that licenses agent to infer that caller is mentioning this meeting so as to inform the agent of the travel dates?
Conversational Implicature (2)

- A: … there’s 3 non-stops today.
- This would still be true if 7 non-stops today.
- But no, the agent means: 3 and only 3.
- How can caller infer that agent means:
  - only 3
Grice: conversational implicature

- Implicature means a particular class of licensed inferences.
- Grice (1975) proposed that what enables hearers to draw correct inferences is:
- Cooperative Principle
  - This is a tacit agreement by speakers and listeners to cooperate in communication
4 Gricean Maxims

- Relevance: Be relevant
- Quantity: Do not make your contribution more or less informative than required
- Quality: try to make your contribution one that is true (don’t say things that are false or for which you lack adequate evidence)
- Manner: Avoid ambiguity and obscurity; be brief and orderly
Relevance

- A: Is Regina here?
- B: Her car is outside.
- Implication: yes
  - Hearer thinks: why would he mention the car? It must be relevant. How could it be relevant? It could since if her car is here she is probably here.
- Caller: I need to be there for a meeting that’s from the 12th to the 15th
  - Hearer thinks: Speaker is following maxims, would only have mentioned meeting if it was relevant. How could meeting be relevant? If client meant me to understand that he had to depart in time for the mtg.
Quantity

- A: How much money do you have on you?
- B: I have 5 dollars
  - Implication: not 6 dollars
- Similarly, 3 non stops can’t mean 7 non-stops (hearer thinks:
  - if speaker meant 7 non-stops she would have said 7 non-stops
- A: Did you do the reading for today’s class?
- B: I intended to
  - Implication: No
  - B’s answer would be true if B intended to do the reading AND did the reading, but would then violate maxim
Text-Based Dialogue System

- **TEAMS:**
  - NLU: 1 person + crowd sourcing or whole team for corpus/testing?
  - NLG: 1 person
  - Task/DB: ½ person unless significant extension of DB or task model targeting DB
  - DM: 1 to 2 people
Natural Language Understanding

- Or “NLU”
- Or “Computational semantics”
- There are many ways to represent the meaning of sentences
- For dialogue systems, most common is “Frame and slot semantics”.
An example of a frame

- Show me morning flights from Boston to SF on Tuesday.

**SHOW:**

**FLIGHTS:**

**ORIGIN:**

- **CITY:** Boston
- **DATE:** Tuesday
- **TIME:** morning

**DEST:**

- **CITY:** San Francisco

- **HOW TO GENERATE THIS SEMANTICS?**
- Different Methods (see NLTK, chaps 6,7,8)
Semantic Parsing approach

- Context Free Grammar = CFG
- Define a CFG with Semantic nonterminals and word sequence or semantic terminals

- CFG in which the LHS of rules is a semantic category:
  - LIST -> show me | I want | can I see…
  - DEPARTTIME -> (after | around | before) HOUR | morning | afternoon | evening
  - HOUR -> one | two | three…| twelve (am | pm)
  - FLIGHTS -> (a) flight| flights
  - ORIGIN -> from CITY
  - DESTINATION -> to CITY
  - CITY -> Boston | San Francisco | Denver | Washington
Semantics for a sentence

LIST   FLIGHTS   ORIGIN
Show me   flights   from Boston

DESTINATION   DEPARTDATE
To San Francisco   on   Tuesday

DEPARTTIME
Morning
Thought/Discussion Question

WHAT ARE THE PROS AND CONS OF THE SEMANTIC PARSING APPROACH?
Thought/Discussion Question

- WHAT ARE THE PROS AND CONS OF THE SEMANTIC PARSING APPROACH?

- Everything written by hand
- Hard to get good coverage?
- Write the same rules repeatedly for many applications?

  - HOUR -> one | two | three… | twelve (am | pm)
  - CITY -> Boston | San Francisco | Denver | Washington
Other Methods for NLU

- **Syntactic Parsing approach**
  - Parse it into a syntactic structure using an NL Grammar;
  - NLTK has a chart parser built in
  - For each rule of the grammar, define a ‘matching rule’ to convert the parse representation into semantics/meaning

- **Classification approach**
  - Define categories of utterances that require different interpretation mechanisms or different behavior from the dialog manager
  - Train a classifier on lots of examples of the kinds of input you expect using ‘crowd sourcing’ (see Rossen et al. paper)
**TEAMS:**

- **NLU:** 1 person + crowd sourcing or whole team for corpus/testing?
- **NLG:** 1 person
- **Task/DB:** ½ person unless significant extension of DB or task model targeting DB
- **DM:** 1 to 2 people
Generation

- Generation components
  - Choose concepts to express to user
  - Plan out how to express these concepts in words

- Content Planner
  - Decides what content to express to user
    - Decides on the Dialogue Act
    - Ask a question, present an answer
  - Often merged with dialogue manager
  - Can select a subset of a pool of content by targeting user or context

- Language Generation
  - Chooses syntactic structures and words to express meaning.
NLG: Four approaches (at least)

- All words in sentence are prespecified!
- Template Based Generation
  - Can have variables
  - Leaving on $\$DAY\$. What time do you want to leave?
  - What time do you want to leave CITY-ORIG?
  - Will you return to CITY-ORIG from CITY-DEST?
- Rule-Based NLG:
  - Simple NLG toolkit; see Ehud Reiter’s page
- Statistical NLG: Lots of different ways.
  - Sparky Generator (on my page)
  - PERSONAGE generator (on my page) is trainable.
Rule Based/Statistical Language generation

- This is a field, like Parsing, or Natural Language Understanding, or Speech Synthesis, with its own conferences
- International Conference on NLG; European NLG
- Approach:
  - Dialogue manager builds representation of meaning of utterance to be expressed
  - Passes this to a “generator”
  - Generators have three components
    - Sentence planner
    - Surface realizer
    - Prosody assigner
Architecture of a generator for a dialogue system (after Walker and Rambow 2002)
HCI constraints on generation for dialogue: “Coherence”, can’t do it without state

- Discourse markers and pronouns (“Coherence”):

  (1) Please say the date.
  Please say the start time.
  Please say the duration…
  Please say the subject…

  (2) First, tell me the date.
      Next, I’ll need the time it starts.
      Thanks. <pause> Now, how long is it supposed to last?
      Last of all, I just need a brief description
HCI constraints on generation for dialogue: coherence (II):

**tapered prompts** can’t do it without state

- Prompts which get incrementally shorter:

  - **System**: Now, what’s the first company to add to your watch list?
  - **Caller**: Cisco
  - **System**: What’s the next company name? (Or, you can say, “Finished”)
  - **Caller**: IBM
  - **System**: Tell me the next company name, or say, “Finished.”
  - **Caller**: Intel
  - **System**: Next one?
  - **Caller**: America Online.
  - **System**: Next?
  - **Caller**: …
Text-Based Dialogue System

**TEAMS:**
- **NLU:** 1 person + crowd sourcing or whole team for corpus/testing?
- **NLG:** 1 person
- **Task/DB:** ½ person unless significant extension of DB or task model targeting DB
- **DM:** 1 to 2 people
Dialogue Manager

- Controls the architecture and structure of dialogue
  - Takes input from NLU component
  - Maintains some sort of state
  - Interfaces with Task Manager
  - Passes output to NLG modules
- Decides what the system is going to do next
- Key aspect of dialogue system design
- How to design it?
Four architectures for dialogue management

- Finite State
- Frame-based
- Information State
  - Markov Decision Processes
- AI Planning
Finite-State Dialogue Mgmt

- Consider a trivial airline travel system
  - Ask the user for a departure city
  - For a destination city
  - For a time
  - Whether the trip is round-trip or not
Finite State Dialogue Manager

1. What city are you leaving from?
2. Where are you going?
3. What date do you want to leave?
4. Is it a one-way trip?
5. Do you want to go from <FROM> to <TO> on <DATE>?
   - Yes
   - No

   - Yes: What date do you want to return?
   - No: Do you want to go from <FROM> to <TO> on <DATE> returning on <RETURN>?

   - Yes: Book the flight
   - No
Finite-state dialogue managers

- System completely controls the conversation with the user.
- It asks the user a series of questions
- Ignoring (or misinterpreting) anything the user says that is not a direct answer to the system’s questions
Dialogue Initiative

- Systems that control conversation like this are **system initiative** or **single initiative**.
- “Initiative”: who has control of conversation
- In normal human-human dialogue, initiative shifts back and forth between participants.
System Initiative

- Systems which completely control the conversation at all times are called system initiative.

- Advantages:
  - Simple to build
  - User always knows what they can say next
  - System always knows what user can say next
    - Known words and Topic: Better performance from NLU
  - Ok for VERY simple tasks (entering a credit card, or login name and password)

- Disadvantage:
  - Too limited
User Initiative

- User directs the system
- Generally, user asks a single question, system answers
- System can’t ask questions back, engage in clarification dialogue, confirmation dialogue
- Used for simple database queries
- User asks question, system gives answer
- Web search is user initiative dialogue.
Problems with System Initiative

- Real dialogue involves give and take!
- In travel planning, users might want to say something that is not the direct answer to the question.
- For example answering more than one question in a sentence:
  - Hi, I’d like to fly from Seattle Tuesday morning
  - I want a flight from Milwaukee to Orlando one way leaving after 5 p.m. on Wednesday.
Single initiative + universals

- We can give users a little more flexibility by adding universal commands.
- Universals: commands you can say anywhere.
- As if we augmented every state of FSA with these:
  - Help
  - Start over
  - Correct
- This describes many implemented systems.
- But still doesn’t allow user to say what s/he wants to say.
Mixed Initiative

- Conversational initiative can shift between system and user
- Simplest kind of mixed initiative: use the structure of the frame itself to guide dialogue

**Slot**
- ORIGIN
- DEST
- DEPT DATE
- DEPT TIME
- AIRLINE

**Question**
- What city are you leaving from?
- Where are you going?
- What day would you like to leave?
- What time would you like to leave?
- What is your preferred airline?
Frames are mixed-initiative

- User can answer multiple questions at once.
- System asks questions of user, filling any slots that user specifies.
- When frame is filled, do database query.
- If user answers 3 questions at once, system has to fill slots and not ask these questions again!
- Anyhow, we avoid the strict constraints on order of the finite-state architecture.
Multiple frames

- flights, hotels, rental cars
- Flight legs: Each flight can have multiple legs, which might need to be discussed separately
- Presenting the flights (If there are multiple flights meeting users constraints)
  - It has slots like 1ST_FLIGHT or 2ND_FLIGHT so user can ask “how much is the second one”
- General route information:
  - Which airlines fly from Boston to San Francisco
- Airfare practices:
  - Do I have to stay over Saturday to get a decent airfare?
Multiple Frames

- Need to be able to switch from frame to frame
- Based on what user says.
- Disambiguate which slot of which frame an input is supposed to fill, then switch dialogue control to that frame.
- Main implementation: production rules
  - Different types of inputs cause different productions to fire
  - Each of which can flexibly fill in different frames
  - Can also switch control to different frame
Defining Mixed Initiative

- Mixed Initiative could mean
  - User can arbitrarily take or give up initiative in various ways
    - This is really only possible in very complex plan-based dialogue systems
    - No commercial implementations
    - Important research area
  - Something simpler and quite specific which we will define in the next few slides
True Mixed Initiative

C₁: ...I need to travel in May.
A₁: And, what day in May did you want to travel?
C₂: OK uh I need to be there for a meeting that’s from the 12th to the 15th.
A₂: And you’re flying into what city?
C₃: Seattle.
A₃: And what time would you like to leave Pittsburgh?
C₄: Uh hmm I don’t think there’s many options for non-stop.
A₄: Right. There’s three non-stops today.
C₅: What are they?
A₅: The first one departs PGH at 10:00am arrives Seattle at 12:05 their time. The second flight departs PGH at 5:55pm, arrives Seattle at 8pm. And the last flight departs PGH at 8:15pm arrives Seattle at 10:28pm.
C₆: OK I’ll take the 5ish flight on the night before on the 11th.
C₇: OK.
How mixed initiative is usually defined

- First we need to define two other factors
- Open prompts vs. directive prompts
- Restrictive versus non-restrictive grammar
Open vs. Directive Prompts

- **Open prompt**
  - System gives user very few constraints
  - User can respond how they please:
    - “How may I help you?” “How may I direct your call?”

- **Directive prompt**
  - Explicit instructs user how to respond
  - “Say yes if you accept the call; otherwise, say no”
Restrictive vs. Non-restrictive grammars

- **Restrictive grammar**
  - Language model which strongly constrains the NLU system, based on dialogue state

- **Non-restrictive grammar**
  - Open language model which is not restricted to a particular dialogue state
### Definition of Mixed Initiative

<table>
<thead>
<tr>
<th>Grammar</th>
<th>Open Prompt</th>
<th>Directive Prompt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Restrictive</td>
<td><em>Doesn’t make sense</em></td>
<td><em>System Initiative</em></td>
</tr>
<tr>
<td>Non-restrictive</td>
<td><em>User Initiative</em></td>
<td><em>Mixed Initiative</em></td>
</tr>
</tbody>
</table>
S: Welcome to NJFun, How May I Help You? (User Initiative)
S’ : Welcome to NJFun, Please say an activity name or say list activities to get a list of activities I know about. (System Initiative)
U: I’d like to find some winetasting in Lambertville this morning.
S: Did you say you are interested in Lambertville? (Explicit Confirmation)
U: Yes.
S: Did you say you want to go in the morning?
U: Yes
S: I found a winery in Lambertville that is open in the morning. It is ....
Outline

- The Linguistics of Conversation
- Basic Conversational Agents
  - ASR
  - NLU
  - Generation
  - Dialogue Manager
- Dialogue Manager Design
  - Finite State
  - Frame-based
  - Initiative: User, System, Mixed
- Information-State
  - Dialogue-Act Detection
  - Dialogue-Act Generation
- Evaluation
- Utility-based conversational agents
  - MDP, POMDP
If we want a dialogue system to be more than just form-filling

Needs to:

- Decide when the user has asked a question, made a proposal, rejected a suggestion
- Ground a user’s utterance, ask clarification questions, suggestion plans

Suggests:

- Conversational agent needs sophisticated models of interpretation and generation
  - In terms of speech acts and grounding
  - Needs more sophisticated representation of dialogue context than just a list of slots
Information-state architecture

- Information state
- Dialogue act interpreter
- Dialogue act generator
- Set of update rules
  - Update dialogue state as acts are interpreted
  - Generate dialogue acts
- Control structure to select which update rules to apply
Dialogue acts

• Also called “conversational moves”
• An act with (internal) structure related specifically to its dialogue function
• Incorporates ideas of grounding
• Incorporates other dialogue and conversational functions that Austin and Searle didn’t seem interested in
Verbmobil task

- Two-party scheduling dialogues
- Speakers were asked to plan a meeting at some future date
- Data used to design conversational agents which would help with this task
- (cross-language, translating, scheduling assistant)
Verbmobil Dialogue Acts

THANK: thanks
GREET: Hello Dan
INTRODUCE: It’s me again
BYE: Alright, bye
REQUEST-COMMENT: How does that look?
SUGGEST: June 13th through 17th
REJECT: No, Friday I’m booked all day
ACCEPT: Saturday sounds fine
REQUEST-SUGGEST: What is a good day of the week for you?
INIT: I wanted to make an appointment with you
GIVE_REASON: Because I have meetings all afternoon
FEEDBACK: Okay
DELIBERATE: Let me check my calendar here
CONFIRM: Okay, that would be wonderful
CLARIFY: Okay, do you mean Tuesday the 23rd?
Automatic Interpretation of Dialogue Acts

- How do we automatically identify dialogue acts?
- Given an utterance:
  - Decide whether it is a QUESTION, STATEMENT, SUGGEST, or ACK
- Recognizing illocutionary force will be crucial to building a dialogue agent
- Perhaps we can just look at the form of the utterance to decide?
Can we just use the surface syntactic form?

- YES-NO-Q’s have auxiliary-before-subject syntax:
  - Will breakfast be served on USAir 1557?

- STATEMENTs have declarative syntax:
  - I don’t care about lunch

- COMMAND’s have imperative syntax:
  - Show me flights from Milwaukee to Orlando on Thursday night
## Surface form != speech act type

<table>
<thead>
<tr>
<th></th>
<th>Locutionary Force</th>
<th>Illocutionary Force</th>
</tr>
</thead>
<tbody>
<tr>
<td>Can I have the rest of your sandwich?</td>
<td>Question</td>
<td>Request</td>
</tr>
<tr>
<td>I want the rest of your sandwich</td>
<td>Declarative</td>
<td>Request</td>
</tr>
<tr>
<td>Give me your sandwich!</td>
<td>Imperative</td>
<td>Request</td>
</tr>
</tbody>
</table>
Abbott: Well, Costello, I'm going to New York with you. Bucky Harris the Yankee's manager gave me a job as coach for as long as you're on the team.

Costello: Look Abbott, if you're the coach, you must know all the players.

Abbott: I certainly do.

Costello: Well you know I've never met the guys. So you'll have to tell me their names, and then I'll know who's playing on the team.

Abbott: Oh, I'll tell you their names, but you know it seems to me they give these ball players now-a-days very peculiar names.

Costello: You mean funny names? ……………………. 

Abbott: Well, let's see, we have on the bags, Who's on first, What's on second, I Don't Know is on third...

Costello: That's what I want to find out.

Abbott: I say Who's on first, What's on second, I Don't Know's on third.
Dialogue act ambiguity

- INFO-REQUEST: Who’s on first?
- STATEMENT: Who’s on first.
Dialogue Act ambiguity

- Can you give me a list of the flights from Atlanta to Boston?
  - This looks like an INFO-REQUEST.
  - If so, the answer is:
    - YES.
  - But really it’s a DIRECTIVE or REQUEST, a polite form of:
    - Please give me a list of the flights…
- What looks like a QUESTION can be a REQUEST
**Dialogue Act ambiguity**

- Similarly, what looks like a STATEMENT can be a QUESTION:

<table>
<thead>
<tr>
<th></th>
<th>OPEN-OPTION</th>
<th>I was wanting to make some arrangements for a trip that I’m going to be taking uh to LA uh beginning of the week after next</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>HOLD</td>
<td>OK uh let me pull up your profile and I’ll be right with you here. [pause]</td>
</tr>
<tr>
<td>A</td>
<td>CHECK</td>
<td>And you said you wanted to travel next week?</td>
</tr>
<tr>
<td>U</td>
<td>ACCEPT</td>
<td>Uh yes.</td>
</tr>
</tbody>
</table>
Indirect speech acts

- Utterances which use a surface statement to ask a question
- Utterances which use a surface question to issue a request
DA interpretation as statistical classification

- Lots of clues in each sentence that can tell us which DA it is:
  - Words and Collocations:
    - *Please or would you*: good cue for REQUEST
    - *Are you*: good cue for INFO-REQUEST
  - Conversational Structure
    - *Yeah* following a proposal is probably AGREEMENT; *yeah* following an INFORM probably a BACKCHANNEL
Statistical classifier model of dialogue act interpretation

- Our goal is to decide for each sentence what dialogue act it is
- This is a **classification task** (we are making a 1-of-N classification decision for each sentence)
- With **N** classes (= number of dialog acts).
- Two models corresponding to the kinds of cues from the input sentence.
  - Conversational Structure: Probability of one dialogue act following another
    \[ P(\text{Answer}|\text{Question}) \]
  - Words and Syntax: Probability of a sequence of words given a dialogue act:
    \[ P(\text{“do you”} | \text{Question}) \]
Dialogue System Evaluation

- **Key point.**
- Whenever we design a new algorithm or build a new application, need to evaluate it.
- **Two kinds of evaluation**
  - **Extrinsic:** embedded in some external task
  - **Intrinsic:** some sort of more local evaluation.

- How to evaluate a dialogue system?
- What constitutes success or failure for a dialogue system?
Dialogue System Evaluation

- It turns out we’ll need an evaluation metric for two reasons
  - 1) the normal reason: we need a metric to help us compare different implementations
    - can’t improve it if we don’t know where it fails
    - Can’t decide between two algorithms without a goodness metric
  - 2) a new reason: we will need a metric for “how good a dialogue went” as an input to reinforcement learning:
    - automatically improve our conversational agent performance via learning
PARADISE evaluation

- Maximize Task Success
- Minimize Costs
  - Efficiency Measures
  - Quality Measures
Task Success

- % of subtasks completed
- Correctness of each question/answer/error msg
- Correctness of total solution
  - Attribute-Value matrix (AVM)
  - Kappa coefficient
- Users’ perception of whether task was completed
Task Success

• Task goals seen as Attribute-Value Matrix

ELVIS e-mail retrieval task (Walker et al ’97)

“Find the time and place of your meeting with Kim.”

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selection Criterion</td>
<td>Kim or Meeting</td>
</tr>
<tr>
<td>Time</td>
<td>10:30 a.m.</td>
</tr>
<tr>
<td>Place</td>
<td>2D516</td>
</tr>
</tbody>
</table>

• Task success can be defined by match between AVM values at end of task with “true” values for AVM
Efficiency Cost


- Total elapsed time in seconds or turns
- Number of queries
- Turn correction ratio: number of system or user turns used solely to correct errors, divided by total number of turns
Quality Cost

- # of times system failed to understand anything
- # of times system had to provide help
- Inappropriateness (verbose, ambiguous) of system’s questions, answers, error messages
Another key quality cost

- “Concept accuracy” or “Concept error rate”
- % of semantic concepts that the NLU component returns correctly
- I want to arrive in Austin at 5:00
  - DESTCITY: Boston
  - Time: 5:00
- Concept accuracy = 50%
- Average this across entire dialogue
- “How many of the sentences did the system understand correctly”
User Satisfaction:
Sum of Many Measures

Was the system easy to understand? (TTS Performance)
Did the system understand what you said? (ASR Performance)
Was it easy to find the message/plane/train you wanted? (Task Ease)
Was the pace of interaction with the system appropriate? (Interaction Pace)
Did you know what you could say at each point of the dialog? (User Expertise)
How often was the system sluggish and slow to reply to you? (System Response)
Did the system work the way you expected it to in this conversation? (Expected Behavior)
Do you think you'd use the system regularly in the future? (Future Use)
Performance Functions from Three Systems

- ELVIS User Sat. = 0.21 * COMP + 0.47 * MRS - 0.15 * ET
- TOOT User Sat. = 0.35 * COMP + 0.45 * MRS - 0.14 * ET
- ANNIE User Sat. = 0.33 * COMP + 0.25 * MRS + 0.33 * Help

- COMP: User perception of task completion (task success)
- MRS: Mean (concept) recognition accuracy (cost)
- ET: Elapsed time (cost)
- Help: Help requests (cost)
Performance Model

- Perceived task completion and mean recognition score (concept accuracy) are consistently significant predictors of User Satisfaction
- Performance model useful for system development
  - Making predictions about system modifications
  - Distinguishing ‘good’ dialogues from ‘bad’ dialogues
  - As part of a learning model
Now that we have a success metric

- Could we use it to help drive learning?
- In recent work we use this metric to help us learn an optimal **policy** or **strategy** for how the conversational agent should behave.
New Idea: Modeling a dialogue system as a probabilistic agent

A conversational agent can be characterized by:

- The current knowledge of the system
  - A set of states $S$ the agent can be in
- A set of actions $A$ the agent can take
- A goal $G$, which implies
  - A success metric that tells us how well the agent achieved its goal
  - A way of using this metric to create a strategy or policy $\pi$ for what action to take in any particular state.
What do we mean by actions $A$ and policies $\pi$?

- Kinds of decisions a conversational agent needs to make:
  - When should I ground/confirm/reject/ask for clarification on what the user just said?
  - When should I ask a directive prompt, when an open prompt?
  - When should I use user, system, or mixed initiative?
A threshold is a human-designed policy!

- Could we learn what the right action is
  - Ask an open question or a direct one?
  - Present one option or several?
- By learning a policy which,
  - given various information about the current state,
  - dynamically chooses the action which maximizes dialogue success
Utility

- A utility function
  - maps a state or state sequence
  - onto a real number
  - describing the goodness of that state
  - i.e. the resulting “happiness” of the agent

- Principle of Maximum Expected Utility:
  - A rational agent should choose an action that maximizes the agent’s expected utility
Utility Agent
(see Walker et al. 1998 in reading)
Conversational Dialogue Systems: Challenges

- Language understanding: methods are expensive, data intensive
- Dialogue manager: design is currently an art, domain specific, labor intensive
- Language generation: canned utterances or template-based, domain specific, labor intensive
- **Active Research**: Methods for quickly customizing systems to new domains, users, applications