CS 4120: Natural Language Processing
Instructor: Prof. Lu Wang
Northeastern University
Webpage: www.ccs.neu.edu/home/luwang

Logistics
• Arrangement for the rest of the semester!
  • 3/19, 3/24: dialogue systems
  • 3/26: project feedback meetings (see piazza for arrangement)
  • 4/2: chatbots
  • 4/2: discourse analysis
  • We will release lecture recordings in the morning, a quiz covering the same content will be available on blackboard (you can do the quiz anytime that day until midnight, no time limit).
  • All regular lecture time becomes the instructor's office hours.
  • Both instructor and TA's office hours will be online.
  • Details in piazza post @88.

What is Natural Language Dialogue?
• Communication involving
  • Multiple contributions
  • Coherent interaction
  • More than one participant
• Interaction modalities
  • Input: Speech, typing, writing, gesture
  • Output: Speech, text, graphical display, animated face/body (embodied virtual agent)

What is involved in NL dialogue
• Understanding
  • What does a person say?
    • Identify words from speech signal
      • "Please close the window"
  • What does the speech mean?
    • Identify semantic content
      • "request (subject: close (object: window))"
  • What were the speaker's intentions?
    • Speaker requests an action in a physical world

What is involved in NL dialogue
• Managing interaction
  • Internal representation of the domain
  • Identify new information
  • Identifying which action to perform given new information
    • "close the window", "set a thermostat" -> physical action
    • "what is the weather like outside?" -> call the weather API
  • Determining a response
    • "OK", "I can't do it"
    • Provide an answer
    • Ask a clarification question

What is involved in NL dialogue
• Access to information (Can you deliver this action?)
• To process a request "Please close the window" you (or the system) needs to know:
  • There is a window
  • Window is currently opened
  • Window can/cannot be closed
What is involved in NL dialogue

- Producing language
- Deciding when to speak
- Deciding what to say
- Choosing the appropriate meaning
- Deciding how to present information
  - So partner understands it
  - So expression seems natural

When is automatic dialogue system useful?

- When hands-free interaction is needed
  - In-car interface
  - In-field assistant system
- Command-and-control interface
- Language tutoring
- Immersive training
- When speaking is easier than typing
  - Voice search interface
  - Virtual assistant (Siri, Google Now)
- Replacing human agents (cutting cost for companies)
  - Call routing
  - Menu-based customer help
  - Voice interface for customer assistance

Visions of dialogue from science fiction

- Hal “2001: A Space Odyssey” (1968)
  - Naturally conversing computer
- Star Trek (original 1966)
  - Natural language command and control
- Her (2013)
  - A virtual partner with natural dialogue capabilities

Conversational Agents aka Dialogue Agents

- Phone-based Personal Assistants
  - Siri, Cortana, Google Now
- Talking to your car
- Communicating with robots
- Clinical uses for mental health
- Chatting for fun

Two classes of systems

1. (Goal-based) Dialogue agents
   - Siri, interfaces to cars, robots
   - Booking flights or restaurants
2. Chatbots

Two classes of systems

1. (Goal-based) Dialogue agents (this lecture)
   - Siri, interfaces to cars, robots
   - Booking flights or restaurants
2. Chatbots
Examples of modern Virtual Assistant dialogue systems

• Apple Siri
  • Supports questions in a set of domains
  • Answers open-end questions
  • Cute “Easter egg” responses

Examples of modern Virtual Assistant dialogue systems

• Android Google Now (2013)
  • Predictive search assistant
  • Works across different Windows devices
  • Aims to be able to “talk about anything”

Embedded devices with dialogue capabilities

• Amazon Echo (2014) – home assistant device
  • Plays music
  • With voice commands
  • Question answering
  • Get weather, news
  • More complex questions, like
    • “How many spoons are in a cup?”
  • Setting timer
  • Manages TODO lists
Architectures for Practical Dialogue Systems

• Finite-State
  *Simple information: e.g., passwords or credit cards*
• Frame-Based
  *All commercial and academic system (SIRI etc.)*

Finite-State Dialog Management

Consider a trivial airline travel system:
  Ask the user for a departure city
  Ask for a destination city
  Ask for a time
  Ask whether the trip is round-trip or not

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 called single initiative.
• Initiative: who has control of conversation
• In normal human-human dialogue, initiative shifts back and forth between participants.
Instead, the state of the art:
Frame-based dialogue

- A kind of mixed initiative
  - The conversational initiative shifts between system and user
  - The structure of the frame guides 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 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 much flexibility

Architectures for Practical Dialogue Systems

- Finite-State
  - Simple information: e.g., passwords or credit cards
- Frame-Based
  - All commercial and academic system (SIRI etc.)

System Initiative

System completely controls the conversation

+ Simple to build
- User always knows what they can say next
- System always knows what user can say next
- Known words: Better performance from ASR
- Known topic: Better performance from NLU (NL understanding)
- OK for VERY simple tasks (entering a credit card, or login name and password)
- Too limited
The Frame
- A set of slots, to be filled with specific information
- Each associated with a question to the user

<table>
<thead>
<tr>
<th>Slot</th>
<th>Question</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORIGIN</td>
<td>What city are you leaving from?</td>
</tr>
<tr>
<td>DEST</td>
<td>Where are you going?</td>
</tr>
<tr>
<td>DEPT DATE</td>
<td>What day would you like to leave?</td>
</tr>
<tr>
<td>DEPT TIME</td>
<td>What time would you like to leave?</td>
</tr>
<tr>
<td>AIRLINE</td>
<td>What is your preferred airline?</td>
</tr>
</tbody>
</table>

Frames are mixed-initiative
- 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 can fill 3 slots and not ask these questions again!

The Natural Language Understanding Component

Show me morning flights from Boston to SF on Tuesday.

SHOW:

FLIGHTS:
- ORIGIN:
  - CITY: Boston
  - DATE: Tuesday
  - TIME: morning
- DEST:
  - CITY: San Francisco

Often called "dialogue state" detection

Dialogue state: representation of what the user wants at any point in a dialogue
- Which slots got filled in the last sentence?
- What is the current state of the frame?
  - All the values of the filled slots
- What is the user's last "dialogue act":
  - Did they ask me a question?
  - Inform me of something?

How to do Frame-based Natural Language Understanding?
- Rule-based models
- Statistical models

Siri uses GUS architecture:
Condition-Action Rules
- Active Ontology: relational network of concepts
- data structures: a meeting has
  - a date and time,
  - a location,
  - a topic
  - a list of attendees
- rule sets that perform actions for concepts
  - the date concept turns string
    - Monday at 2pm into
    - date object date(DAY,MONTH,YEAR,HOURS,MINUTES)
Part of ontology for meeting task

Statistical Natural Language Understanding

- Statistical classifiers to map words to semantic frame-fillers
- Given a set of labeled sentences
  “I want to fly to San Francisco on Tuesday”
  Destination: San Francisco
  Depart-date: Tuesday
- Requirements: Lots of labeled data

Statistical Slot filling

- Given a sentence:
  I want to go from Boston to SF
- Classifier predicts which slot the user wants to fill
  Output: (ORIGIN, DESTINATION, DEPARTURE-DATE, AIRLINE)

Statistical Slot filling

- Given a sentence:
  I want to go from Boston to SF
- Classifier predicts which slot the user wants to fill
  Output: (ORIGIN, DESTINATION, DEPARTURE-DATE, AIRLINE)
  Features: Words, Named Entities
- Classifier or sequence model predicts the filler:

Evaluation

1. Slot Error Rate for a Sentence
   # of inserted/deleted/substituted slots
   # of total reference slots for sentence

2. End-to-end evaluation (Task Success)

Evaluation

“Make an appointment with David at 11:30 in Rm 101”

<table>
<thead>
<tr>
<th>Slot</th>
<th>Filler</th>
</tr>
</thead>
<tbody>
<tr>
<td>PERSON</td>
<td>David</td>
</tr>
<tr>
<td>TIME</td>
<td>11:30 p.m.</td>
</tr>
<tr>
<td>ROOM</td>
<td>Rm 101</td>
</tr>
</tbody>
</table>
Evaluation

“Make an appointment with David at 11:30 in Rm 101”

<table>
<thead>
<tr>
<th>Slot</th>
<th>Filler</th>
</tr>
</thead>
<tbody>
<tr>
<td>PERSON</td>
<td>David</td>
</tr>
<tr>
<td>TIME</td>
<td>11:30 p.m.</td>
</tr>
<tr>
<td>ROOM</td>
<td>Rm 101</td>
</tr>
</tbody>
</table>

Slot error rate: 1/3
Task success: At end, was the correct meeting added to the calendar?

Intentions

• After understanding “what is said”...
  • Dialogue Act
  • Grounding and Confirmation
  • Rejection

Dialogue Act Detection

• Dialogue Act: The dialogue function of the sentence
  • Question
  • Command
  • Suggestion
• Given a user’s sentence:
  • Context: David wants to eat Italian food.
  • He asked: How many Italian restaurants are in walking distance?
  • Was that a question?

Dialogue Act detection is hard

Can you give me a list of the flights from Atlanta to Boston?
  • This looks like a QUESTION.
  • It has a question-mark, starts with “can you”
  • If so, the answer is:
    • YES.
  • But really it’s a COMMAND, a polite form of:
    Please give me a list of the flights...
  • What looks like a QUESTION can be a COMMAND

Dialogue Act detection

• Rules-based models
• Statistical models/Machine learning models
• Your tasks: think about how to design rules and features to detect dialogue acts

Another example of dialogue act detection: Correction Detection

• If system misrecognizes an utterance, and either
  • Rejects
  • Via confirmation, displays its misunderstanding
• Then user has a chance to make a correction
  • Repeat themselves
  • Rephrasing
  • Saying “no” to the confirmation question.
Corrections

• Unfortunately, corrections are harder to recognize than normal sentences!
• Swerts et al (2000): corrections misrecognized twice as often as non-corrections!!!
• Why?
  • Prosody seems to be largest factor: hyperarticulation
  • Liz Shriberg example:
    • “NO, I am DE-PAR-TING from Jacksonville”

Machine learning to detect user corrections: features

• Lexical information (no, correction, I don’t, swear words)
• Prosodic indicators of hyperarticulation
  • pause duration, word duration
• Length
• LM probability
• Various dialogue features (repetition)

Deciding what to say:

Dialogue act generation/prediction
Should I do a “Confirmation” dialog act?

Natural Language Generation
Given that I’m confirming, what exactly should I say?

Grounding

• Why do elevator buttons light up?
• Clark (1996) (after Norman 1988)
  Principle of closure. Agents performing an action require evidence, sufficient for current purposes, that they have succeeded in performing it
• What is the linguistic correlate of this?

Grounding and Confirmation

• We need to know whether an action succeeded or failed
• Talking is an action!
• I need to know if my action succeeded
  • i.e. the hearer understood my turn!

How do speakers ground?
Clark and Schaefer

• Continued attention:
  • B continues attending to A
• Relevant next contribution:
  • B starts in or next relevant contribution
• Acknowledgement:
  • B nods or says continuer (uh-huh) or assessment (great!)
• Demonstration:
  • B demonstrates understanding A by reformulating A’s contribution, or by collaboratively completing A’s utterance
• Display:
  • B repeats 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, 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 huh, I don't think there are many options for non-stop.
A4: Right. There's three non-stops today.
C5: What are they?
A5: The first one departs PHL at 10:00am arrives Seattle at 12:05 the next day. The second flight departs PHL at 5:15pm, arrives Seattle at 8pm. And the last flight departs PHL at 6:15pm arrives Seattle at 10:20pm.
C6: OK I'll take the 5:15 flight on the night before on the 11th.
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

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?

Confirmation

• Errors: Speech is a pretty errorful channel
• Humans use grounding to confirm that they heard correctly
• Automatic speech recognition (ASR) is way worse than humans!
• Dialogue systems need to do even more grounding and confirmation than humans
• Users are confused when system doesn’t give explicit acknowledgement signal.
  Stitelman et al. (1993), Yankelovich et al. (1995)

Explicit confirmation

S: Which city do you want to leave from?
U: Baltimore
S: Do you want to leave from Baltimore?
U: Yes
Explicit confirmation

U: I'd like to fly from Denver Colorado to New York City on September 21st in the morning on United Airlines
S: Let’s see then. I have you going from Denver Colorado to New York on September 21st. Is that correct?
U: Yes

Implicit confirmation: display

U: I'd like to travel to Berlin
S: When do you want to travel to Berlin?
U: Hi I'd like to fly to Seattle Tuesday morning
S: Traveling to Seattle on Tuesday, August eleventh in the morning. Your name?

Implicit vs. Explicit

• Complementary strengths
• Explicit: easier for users to correct systems’ mistakes (can just say "no")
• But explicit is cumbersome and long
• Implicit: much more natural, quicker, simpler (if system guesses right).

Implicit and Explicit

• Early systems: all-implicit or all-explicit
• Modern systems: adaptive
• How to decide?

Implicit and Explicit

• Early systems: all-implicit or all-explicit
• Modern systems: adaptive
• How to decide?

Rejection

• “I’m sorry, I didn’t understand that.”
• Reject when:
  • ASR confidence is low
  • Best interpretation is semantically ill-formed
• Might have four-tiered level of confidence:
  • Below confidence threshold, reject
  • Above threshold, explicit confirmation
  • If even higher, implicit confirmation
  • Even higher, no confirmation

Should also consider cost of an error: Explicit confirmation before moving money or booking flights