Tutorial Overview: Outline

Part I

- Signal processing
- Speech recognition
  - acoustic modeling
  - language modeling
  - decoding
- Semantic interpretation
- Speech synthesis

Part II

- Discourse and dialogue
  - Discourse interpretation
  - Dialogue management
  - Response generation
- Dialogue evaluation
- Data collection

Discourse & Dialogue Processing

- Discourse interpretation:
  - Understand what the user really intends by interpreting utterances in context
- Dialogue management:
  - Determine system goals in response to user utterances based on user intention
- Response generation:
  - Generate natural language utterances to achieve the selected goals

Discourse Interpretation

- Goal: understand what the user really intends
- Example: Can you move it?
  - What does “it” refer to?
  - Is the utterance intended as a **simple yes-no query** or a **request to perform an action**?
- Issues addressed:
  - Reference resolution
  - Intention recognition
- Interpret user utterances in context

Reference Resolution

U: Where is A Bug’s Life playing in Summit?
S: A Bug’s Life is playing at the Summit theater.
U: When is it playing there?
S: It’s playing at 2pm, 5pm, and 8pm.
U: I’d like 1 adult and 2 children for the first show.
How much would that cost?

- Knowledge sources:
  - Domain knowledge
  - Discourse knowledge
  - World knowledge

Reference Resolution: In Theory

- Focus stacks:
  - Maintain recent objects in stack
  - Select objects that satisfy semantic/pragmatic constraints starting from top of stack
  - May take into account discourse structure
- Centering:
  - Backward-looking center (Cb): object connecting the current sentence with the previous sentence
  - Forward-looking centers (Cf): potential Cb of the next sentence
  - Rule-based filtering & ranking of objects for pronoun resolution

Reference Resolution: In Practice

- Non-existent: does not allow the use of anaphoric references
- Allows only simple references:
  - utilizes the focus stack reference resolution mechanism
  - does not take into account discourse structure information
- Example:

U: Where is A Bug's Life playing in Summit?

Summit
A Bug's Life

Summit
theater
A Bug's Life

S: A Bug's Life is playing at the Summit theater.

U: When is **it** playing **there**?

Intention Recognition

B: I have to wash my hair.

Lucent Technologies

A: Would you like to go to the hairdresser?

- B’s utterance should be interpreted as an **acceptance** of A’s proposal.

Lucent Technologies

A: What’s that smell around here?

- B’s utterance should be interpreted as an **answer** to A’s question.

A: Would you be interested in going out to dinner tonight?

- B’s utterance should be interpreted as a **rejection** of A’s proposal.

Intention Recognition (Cont'd)

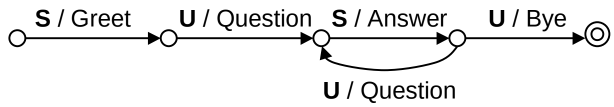
- Goal: to recognize the intent of each user utterance as one (or more) of a set of dialogue acts based on context
- Sample dialogue actions:
  - Switchboard DAMSL
    - Conventional-closing
    - Statement-(non-)opinion
    - Agree/Accept
    - Acknowledgment
    - Yes-No-Question/Yes-Answer
    - Non-verbal
    - Abandoned
  - Verbmobil
    - Greet/Thank/Bye
    - Suggest
    - Accept/Reject
    - Confirm
    - Clarify-Query/Answer
    - Give-Reason
    - Deliberate
- On-going standardization efforts (Discourse Resource Initiative)

Intention Recognition: In Theory

- Knowledge sources:
  - Overall dialogue goals
  - Orthographic features, e.g.:
    - punctuation
    - cue words/phrases: "but", "furthermore", "so"
    - transcribed words: "would you please", "I want to"
  - Dialogue history, i.e., previous dialogue act types
  - Dialogue structure, e.g.:
    - subdialogue boundaries
    - dialogue topic changes
  - Prosodic features of utterance: duration, pause, F0, speaking rate

Intention Recognition: In Theory (Cont'd)

- Finite-state dialogue grammar:
  - e.g.



- Plan-based discourse understanding:
  - Recipes: templates for performing actions
  - Inference rules: to construct plausible plans
- Empirical methods:
  - Probabilistic dialogue act classifiers: HMMs
  - Rule-based dialogue act recognition: CART, Transformation-based learning

Intention Recognition: In Practice

- Makes assumptions about (high-level) task-specific intentions: e.g.,
  - Call routing: *giving destination information*
  - ATIS: *requesting flight information*
  - Movie information system: *movie showtime or theater playlist information*
- Does not allow user-initiated complex dialogue acts, e.g. confirmation, clarification, or indirect responses

S1: What's your account number?
U1: Is that the number on my ATM card?

S2: Would you like to transfer $1,500 from savings to checking?
U2: If I have enough in savings.

Intention Recognition: In Practice (Cont'd)

- User utterances can play one of two roles:
  - Identify one of a set of possible task intentions
  - Provide necessary information for performing a task
- Based on either keywords in an utterance or its syntactic/semantic representation
- Maps keywords or representations to intentions using:
  - Template matching
  - Probabilistic model
  - Vector-based similarity measures

Intention Recognition: Example

U: What time is A Bug's Life playing at the Summit theater?

- Using keyword extraction and vector-based similarity measures:
  - Intention: Ask-Reference: _time
  - Movie: A Bug's Life
  - Theater: the Summit quadplex

Tutorial Overview: Outline

Part I

- Signal processing
- Speech recognition
  - acoustic modeling
  - language modeling
  - decoding
- Semantic interpretation
- Speech synthesis

Part II

- Discourse and dialogue
  - Discourse interpretation
  - Dialogue management
  - Response generation
- Dialogue evaluation
- Data collection

Dialogue Management: Motivating Examples

- Dialogue 1:

S: Would you like movie showtime or theater playlist information?
U: Movie showtime.
S: What movie do you want showtime information about?
U: Saving Private Ryan.
S: At what theater do you want to see Saving Private Ryan?
U: Paramount theater.
S: Saving Private Ryan is not playing at the Paramount theater.

DM: Motivating Examples (Cont'd)

- Dialogue 2:

  S: How may I help you?
  U: When is Saving Private Ryan playing?
  S: For what theater?
  U: The Paramount theater.
  S: Saving Private Ryan is not playing at the Paramount theater, but it's playing at the Madison theater at 3:00, 5:30, 8:00, and 10:30.

DM: Motivating Examples (Cont'd)

- Dialogue 3:

  S: How may I help you?
  U: When is Saving Private Ryan playing?
  S: For what theater?
  U: The Paramount theater.
  S: Saving Private Ryan is playing at the Fairmont theater at 6:00 and 8:30.
  U: I wanted to know about the Paramount theater, not the Fairmont theater.
  S: Saving Private Ryan is not playing at the Paramount theater, but it's playing at the Madison theater at 3:00, 5:30, 8:00, and 10:30.

Comparison of Sample Dialogues

- Dialogue 1:
  - System-initiative
  - Implicit confirmation
  - Merely informs user of failed query
  - Mechanical
  - Least efficient
- Dialogue 2:
  - Mixed-initiative
  - No confirmation
  - Suggests alternative when query fails
  - More natural
  - Most efficient
- Dialogue 3:
  - Mixed-initiative
  - No confirmation
  - Suggests alternative when query fails
  - More natural
  - Moderately efficient

Dialogue Management

- Goal: determine what to accomplish in response to user utterances, e.g.:
  - Answer user question
  - Solicit further information
  - Confirm/Clarify user utterance
  - Notify invalid query
  - Notify invalid query and suggest alternative
- Interface between user/language processing components and system knowledge base

Dialogue Management (Cont'd)

- Main design issues:
  - Functionality: how much should the system do?
  - Process: how should the system do them?
- Affected by:
  - Task complexity: how hard the task is
  - Dialogue complexity: what dialogue phenomena are allowed
- Affects:
  - robustness
  - naturalness
  - perceived intelligence

Task Complexity

- Application dependent
- Examples:

Call Routing — Weather Information / ATIS — Automatic Banking — Travel Planning — University Course Advisement

Simple → Complex

- Directly affects:
  - Types and quantity of system knowledge
  - Complexity of system's reasoning abilities

Dialogue Complexity

- Determines what can be talked about:
  - The task only
  - Subdialogues: e.g., clarification, confirmation
  - The dialogue itself: meta-dialogues
    - Could you hold on for a minute?
    - What was that click? Did you hear it?
- Determines who can talk about them:
  - System only
  - User only
  - Both participants

Dialogue Management: Functionality

- Determines the set of possible goals that the system may select at each turn
- At the task level, dictated by task complexity
- At the dialogue level, determined by system designer in terms of dialogue complexity:
  - Are subdialogues allowed?
  - Are meta-dialogues allowed?
  - Only by the system, by the user, or by both agents?

DM Functionality: In Theory

- Task complexity: moderate to complex
  - Travel planning
  - University course advisement
- Dialogue complexity:
  - System/user-initiated complex subdialogues
    - Embedded negotiation subdialogues
    - Expressions of doubt
  - Meta-dialogues
  - Multiple dialogue threads

DM Functionality: In Practice

- Task complexity: simple to moderate
  - Call routing
  - Weather information query
  - Train schedule inquiry
- Dialogue complexity:
  - About task only
  - Limited system-initiated subdialogues

Dialogue Management: Process

- Determines how the system will go about selecting among the possible goals
- At the dialogue level, determined by system designer in terms of initiative strategies:
  - System-initiative: system always has control, user only responds to system questions
  - User-initiative: user always has control, system passively answers user questions
  - Mixed-initiative: control switches between system and user using fixed rules
  - Variable-initiative: control switches between system and user dynamically based on participant roles, dialogue history, etc.

DM Process: In Theory

- Initiative strategies:
  - Mixed-initiative
  - Variable-initiative
- Mechanisms for modeling initiative:
  - Planning and reasoning
  - Theorem proving
  - Belief modeling
- Knowledge sources for modeling initiative:
  - System beliefs, user beliefs, and mutual beliefs
  - System domain knowledge
  - Dialogue history
  - User preferences

DM Process: In Practice

- Initiative strategies:
  - User-initiative
  - System-initiative
  - Mixed-initiative
  - Variable-initiative
- Mechanisms for modeling initiative:
  - System and mixed-initiative: finite-state machines
  - Variable-initiative: evidential model for computing initiative
    - Knowledge sources:
      - Dialogue history: e.g. user fails to make progress in task
      - Participant roles: advisor/advisee vs. collaborators
      - Features of current utterance: e.g.: ambiguous utterance, underspecified utterance

Dialogue Management: Summary

- In theory:
  - Focuses on developing algorithms for addressing difficult dialogue phenomena in complex tasks
- In practice:
  - Focuses on developing robust, real-time dialogue strategies for handling common dialogue phenomena in simpler tasks

Dialogue Management: Example

- System for providing movie showtime and theater playlist information
- Functionality:
  - Task: provide simple movie information
  - Dialogue: allow system-initiated subdialogues
- Process:
  - Variable-initiative

Dialogue Management Example (Cont'd)

<U has task & dialogue initiative>
U: When is Antz playing at the Fairmont theater?
<S has dialogue initiative (low score for theater)>
<goal: confirm Paramount theater>
S: Did you say the Paramount theater?
U: No, the Fairmont theater.
<S has no initiative>
<goal: answer user question>
S: Antz is playing at the Fairmont theater at 7pm.
U: When is it playing at the Summit theater?
<S has task & dialogue initiative>
<goal: notify failure & suggest alternative>
S: Antz is not playing at the Summit theater, but it's playing at the AMC theater in Summit at 8pm.

Tutorial Overview: Outline

Part I

- Signal processing
- Speech recognition
  - acoustic modeling
  - language modeling
  - decoding
- Semantic interpretation
- Speech synthesis

Part II

- Discourse and dialogue
  - Discourse interpretation
  - Dialogue management
  - Response generation
- Dialogue evaluation
- Data collection

Response Generation

U: Is Saving Private Ryan playing at the Chatham cinema?

S: No, it's not.

- S provides elliptical response

S: No, Saving Private Ryan is not playing at the Chatham cinema.

- S provides full response (which provides grounding information)

S: No, Saving Private Ryan is not playing at the Chatham cinema; the theater's under renovation.

- S provides full response and supporting evidence

Response Generation (Cont'd)

- Goal: generate natural language utterances to achieve goal(s) selected by the dialogue manager
- Issues:
  - Content selection: determining what to say
  - Surface realization: determining how to say it
- **Generation gap**: discrepancy between the actual output of the content selection process and the expected input of the surface realization process

Content Selection

- Goal: determine the propositional content of utterances to achieve goal(s)
- Examples:
  - Antz is not playing at the Maplewood theater; [Nucleus]
  - Would you like the suite? [Nucleus]
  - Can you get the groceries from the car? [Nucleus]

- the theater's under renovation. (evidence) [Satellite]
- It's the same price as the regular room. (motivation) [Satellite]
- The key is on the dryer. (enablement) [Satellite]

Content Selection: In Theory

- Knowledge sources:
  - Domain knowledge base
  - User beliefs
  - User model: user characteristics, preferences, etc.
  - Dialogue history
- Content selection mechanisms:
  - Schemas: patterns of predicates
  - Rule-based generation
  - Plan-based generation:
    - Recipes: templates for performing actions
    - Planner: to construct plans for given goal
  - Case-based reasoning

June 1999 106 Lucent Technologies Bell Labs Innovations

Content Selection: In Practice

- Knowledge sources:
  - Domain knowledge base
  - Dialogue history
- Pre-determined content selection strategies:
  - Nucleus only, no satellite information
  - Nucleus + fixed satellite

June 1999 107 Lucent Technologies Bell Labs Innovations

Surface Realization

- Goal: determine how the selected content will be conveyed by natural language utterances
- Examples:
  - Antz is showing (shown) at the Maplewood theater.
  - The Maplewood theater is showing Antz.
  - It is at the Maplewood theater that Antz is shown.
  - Antz, that's what's being shown at the Maplewood theater.
- Issues:
  - Clausal structure construction
  - Lexical selection

Surface Realization: In Theory

- Typical surface generator requires as input:
  - Semantic representation to be realized
  - Clausal structure for generated utterance
- Surface realization component utilizes a grammar to generate utterance that conveys the given semantic representation

Surface Realization: In Practice

- Canned utterances:
  - Pre-determined utterances for goals; e.g.:
    - Greetings: Hello, this is the ABC bank's operator.
    - Repeat: Could you please repeat your request?
  - Facilitates pre-recorded prompts for speech output
- Template-based generation:
  - Templates for goals; e.g.:
    - Notification: Your call is being transferred to X.
    - Inform: A,B,C,D, and E are playing at the F theater.
    - Clarify: Did you say X or Y?
  - Needs cut-and-paste of pre-recorded segments or full TTS system

Tutorial Overview: Outline

Part I

- Signal processing
- Speech recognition
  - acoustic modeling
  - language modeling
  - decoding
- Semantic interpretation
- Speech synthesis

Part II

- Discourse and dialogue
  - Discourse interpretation
  - Dialogue management
  - Response generation
- Dialogue evaluation
- Data collection

Dialogue Evaluation

- Goal: determine how “well” a dialogue system performs
- Main difficulties:
  - No strict right or wrong answers
  - Difficult to determine what features make a dialogue system better than another
  - Difficult to select metrics that contribute to the overall “goodness” of the system
  - Difficult to determine how the metrics compensate for one another
  - Expensive to collect new data for evaluating incremental improvement of systems

Dialogue Evaluation (Cont’d)

- System-initiative, explicit confirmation
  - better task success rate
  - lower WER
  - longer dialogues
  - fewer recovery subdialogues
  - less natural
- Mixed-initiative, no confirmation
  - lower task success rate
  - higher WER
  - shorter dialogues
  - more recovery subdialogues
  - more natural

Dialogue Evaluation Paradigms

- Evaluating the end result only:
  - Reference answers
- Evaluating both the end result and the process toward it:
  - Evaluation metrics
  - Performance functions

Evaluation Paradigms: Reference Answers

- Evaluates the task success rate only
- Suitable for query-answering systems for which a correct answer can be defined for each query
- For each query:
  - Obtain answer from dialogue system
  - Compare with reference answer
  - Score system performance
- Advantage: simple
- Disadvantage: ignores many other important factors that contribute to quality of dialogue systems

Evaluation Paradigms: Evaluation Metrics

- Different metrics for evaluating different components of a dialogue system:
  - Speech recognizer: word error rate / word accuracy
  - Understanding component: attribute value matrix
  - Dialogue manager:
    - appropriateness of system responses
    - error recovery abilities
  - Overall system:
    - task success
    - average number of turns
    - elapsed time
    - turn correction ratio

Paradigms: Evaluation Metrics (Cont'd)

- Advantage:
  - Takes into account the process toward completing the task
- Limitations:
  - Difficult to determine how different metrics compensate for one another
  - Metrics may not be independent of one another
  - Does not generalize across domains and tasks

Paradigms: Performance Functions

- PARADISE [Walker et al.]: derives performance functions using both task-based and dialogue-based metrics
- User satisfaction:
  - Maximize task success
  - Minimize costs:
    - Efficiency measures: e.g., number of utterances, elapsed time
    - Qualitative measures: e.g., repair ratio, inappropriate utt. ratio
- Performance function derivation:
  - Obtain user satisfaction ratings (questionnaire)
  - Obtain values for various metrics (automatic or manual)
  - Apply multiple linear regression to derive a function relating user satisfaction and various cost factors, e.g.,

$$Perf = .21 * TSR + .47 * MR - .15 * ET$$

Paradigms: Performance Functions (Cont’d)

- Advantages:
  - Allows for comparison of dialogue systems performing different tasks
  - Specifies relative contributions of cost factors to overall performance
  - Can be used to make predictions about future versions of the dialogue system
- Disadvantages:
  - Data collection cost for deriving performance function is high
  - Cost for deriving performance function for multiple systems to draw general conclusions is high

Tutorial Overview: Outline

Part I

- Signal processing
- Speech recognition
  - acoustic modeling
  - language modeling
  - decoding
- Semantic interpretation
- Speech synthesis

Part II

- Discourse and dialogue
  - Discourse interpretation
  - Dialogue management
  - Response generation
- Dialogue evaluation
- Data collection

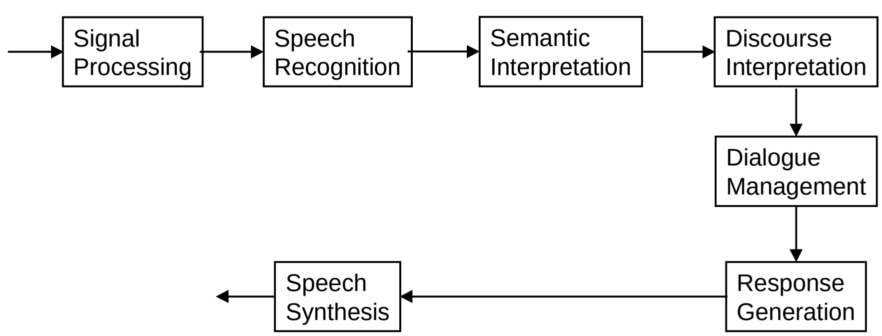
Data Collection: Wizard of Oz Paradigm

- Setup for initial data collection:
  - User communicates with "system" through telephone (speech) or keyboard (text)
  - "System" is actually a human, typically given instructions on how to behave like a system
  - Users are typically given tasks to perform in the target domain
  - Subjects are the users and the "system" can be played by one person
  - Dialogues between "system" and user are recorded and transcribed
- Setup for intermediate system evaluation:
  - Use actual running system, with wizard supervision
  - Wizard can override undesirable system behavior, e.g., correct ASR errors

Data Collection: Wizard of Oz (Cont'd)

- Features of collected data:
  - Typically much less complex than actual human-human dialogues performing the same tasks
  - Captures how humans behave when they talk to computers
  - Captures variations among different subjects in both language and approach when performing the same tasks
- Use of collected data:
  - Particularly useful for designing the interpretation component of the dialogue system
  - Useful for training purposes for ASR systems
  - May also be helpful for designing the dialogue management and response generation components

Summary



Publicly Available Telephone Demos

- Nuance http://www.nuance.com/demo/index.html
  - Banking: 1-650-847-7438
  - Travel Planning: 1-650-847-7427
  - Stock Quotes: 1-650-847-7427
- SpeechWorks http://www.speechworks.com/demos/demos.htm
  - Banking: 1-888-729-3366
  - Stock Trading: 1-800-786-2571
- MIT Spoken Language Systems Laboratory http://www.sls.lcs.mit.edu/sls/whatwedo/applications.html
  - Travel Plans (Pegasus): 1-877-648-8255
  - Weather (Jupiter): 1-888-573-8255
- IBM http://www.software.ibm.com/speech/overview/business/demo.html
  - Mutual Funds, Name Dialing: 1-877-VIA-VOICE