

Dialogue Systems and Chatbots

COSC 6336: Natural Language Processing
Spring 2018

Some content in these slides has been adapted from Svetlana Stoyanchev

Class Announcements

- ★ My office hours in the coming two weeks moved to Friday 11:15am
- ★ Exam is on May 9th, 5:00-8:00pm, topics:
 - Formal Grammars and Syntactic Parsing
 - Statistical Parsing and Dependency Parsing
 - Information Extraction
 - Dialogue Systems

Today's Lecture (Ch. 29 J&M 3rd edition)

★ Dialogue Systems

- Chatbots
- Conversational agents

★ Spoken dialogue systems

- Components
- Evaluation
- Open research areas



I'm trained to use Cognitive Behavior Therapy (CBT)

What's CBT?

CBT is a scientifically validated approach to mental health



it's short and practical

ok how does it work?

No! You have to go to a proper therapist for that 😊

CBT is based on the idea that it's not events themselves that affect us, it's how we think about those events



and how we think is often revealed in what we say

ok



especially what we say to ourselves 🤔 ...

so what do I do now?



so when I check in with you, I'll ask you 3 things: what you're doing, how you're feeling, and your energy level

Conversational Agents

Task oriented:

★ Digital Assistants (Cortana, Alexa, Siri, etc.)

Chatbots:

★ Woebot

★ MS Xiaoice

Conversational Agents

★ Even more complex systems

Tutor: What will the velocity of the object be a second after that (where the initial velocity is 9.8 m/s and the acceleration is 9.8 m/s²)?

Student: Is it 19.6 m/s?

Tutor: Good. A second later, its velocity will be 9.8m/s + 9/8 m/s = 19.6 m/s This is because its acceleration tells us that every second, its velocity increases by 9.8 m/s. So what'll its velocity be, a second after that?

Student: Um. I'm thinking. 29.4 m/s

Tutor: Right. Now back to the man and his keys. We've shown that they both have an acceleration of 9.8 m/s². So, how do their velocities compare with each other, at every second during the fall?

Figure 29.3 The tutoring domain: An excerpt from a tutorial interaction between a physics student and the ITSPOKE system of [Forbes-Riley and Litman \(2011\)](#) based on the Why2-Atlas text-based tutoring system ([VanLehn et al., 2002](#)).

Why do we want to talk to computer programs?

- ★ This goes back to the beginning of the course ...
- ★ Conversations are the most natural way for us to communicate
- ★ But also, it's hands-free interaction, much needed:
 - In-field assistants
 - In-car interface
 - Command-and-control interface
 - Language tutoring
- ★ Cheaper than human agents:
 - Call routing
 - Menu-based customer help
 - Voice interface for customer assistant

Chatbot systems

Two dominant architectures

1. Rule based:

- a. ELIZA (Weizenbaum, 1966)
- b. Parry (Colby et al., 1971) passes Turing test in 1972

2. Corpus-based systems

- a. Information retrieval
- b. Machine learning

ELIZA

```
function ELIZA GENERATOR(user sentence) returns response

Find the word w in sentence that has the highest keyword rank
if w exists
    Choose the highest ranked rule r for w that matches sentence
    response  $\leftarrow$  Apply the transform in r to sentence
    if w = 'my'
        future  $\leftarrow$  Apply a transformation from the 'memory' rule list to sentence
        Push future onto memory stack
    else (no keyword applies)
        either
            response  $\leftarrow$  Apply the transform for the NONE keyword to sentence
        or
            response  $\leftarrow$  Pop the top response from the memory stack
return(response)
```

Figure 29.5 A simplified sketch of the ELIZA algorithm. The power of the algorithm come from the particular transforms associated with each keyword.

Information Retrieval-based Chatbots

- ★ **Key Idea:** use existing dialogues from human-human or human-computer interactions
- ★ Examples of retrieval algorithms:

$$r = response \left(\operatorname{argmax}_{t \in C} \frac{q^T t}{||q|| ||t||} \right)$$

where the system returns the next turn to t

- ★ But another valid choice is to return t directly
- ★ Commercial versions of IR chatbots: MS Xiaolce and Cleverbot

Machine Learning-based Chatbots

Sequence to sequence models (Shang et al, 2015; Vinyals and Lee, 2015; Sordoni et al., 2015)

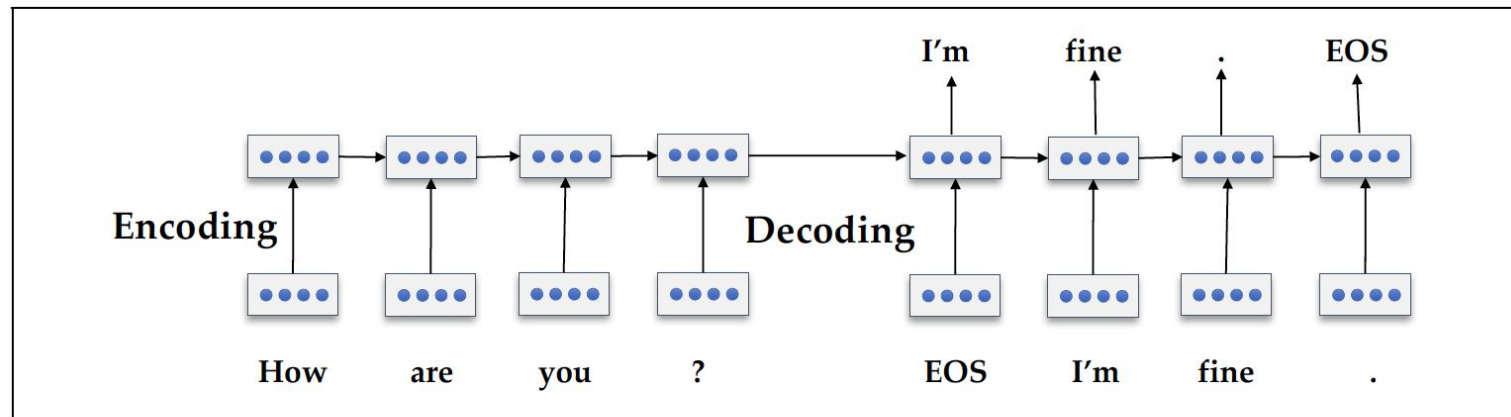


Figure 29.6 A sequence to sequence model for neural response generation in dialog.

Sequence to sequence models

Some drawbacks:

- ★ Generating responses is not really an MT task
- ★ Seq2seq models lack information about the context
- ★ These models may prefer repetitive responses

Sequence to sequence models

Input	tell me ... how long have you had this falling sickness ?
Vanilla-SEQ2SEQ	i'm not a doctor.
Adversarial	a few months, i guess .
Input	so i had the doctors test sammy 's response to conditioning .
Vanilla-SEQ2SEQ	sammy wrote the test sammy wrote the test .
Adversarial	so he took the pills .
Input	they didn 't have much success with this problem commander .
Vanilla-SEQ2SEQ	they 're not the only ones who have been in the system .
Adversarial	can we find someone else ?

Figure 29.7 Sample responses generated by a SEQ2SEQ model trained either with a vanilla maximum likelihood objective, or adversarially trained to produce sentences that are hard for an adversary to distinguish from human sentences (Li et al., 2017).

Evaluation of Chatbots

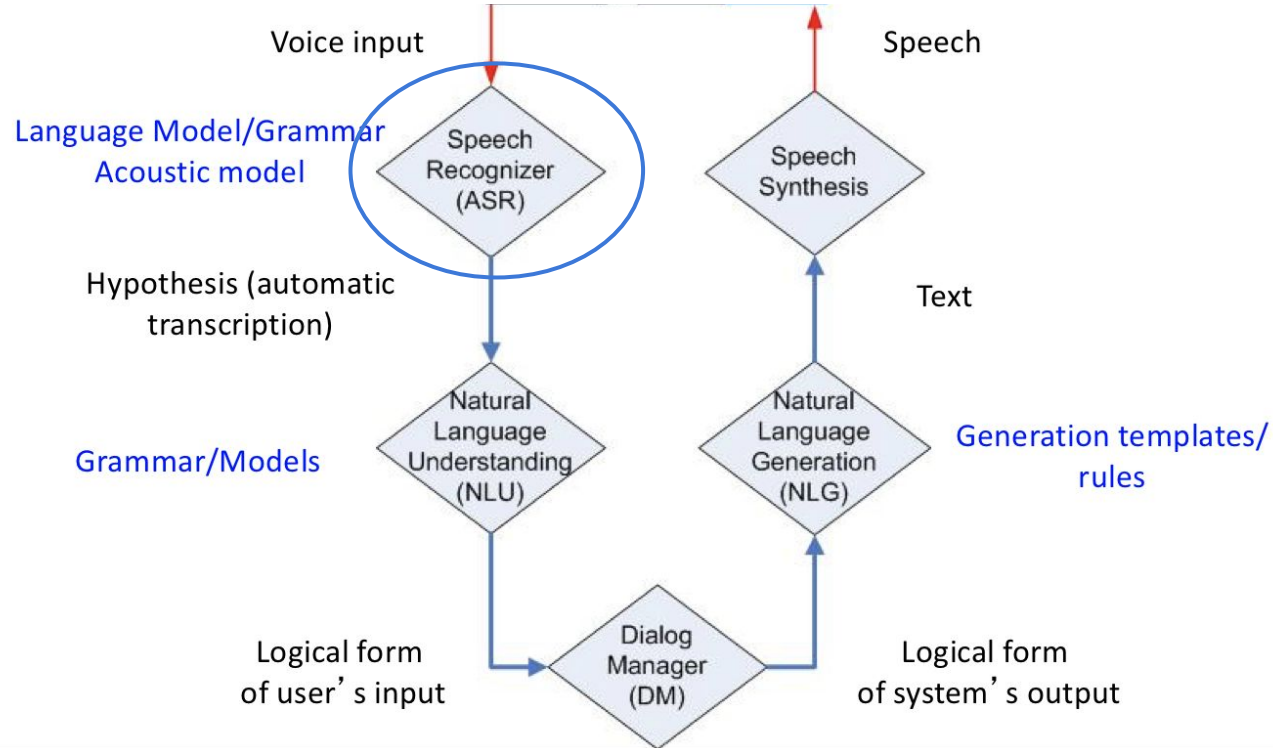
- ★ The best evaluation is humans
- ★ Recent research is proposing automated evaluators (Lowe et al., 2017)
- ★ Adversarial evaluation where we train a Turing-like evaluator (Bowman et al., 2016; Kannan and Vinyals 2016, Li et al., 2017).

Spoken Dialogue Systems (SDS)

Aspects of DS

- ★ Which modalities does the system use
- ★ Back-end
- ★ How much world knowledge does the system have
- ★ How much personal knowledge does it have and use

SDS Components



Speech Recognition

- ★ Convert speech signal into text
- ★ Most SDS use off-the-shelf speech recognizers
 - Research systems are highly configurable:
 - Kaldi – most used research recognizer
 - Sphinx/pocket sphinx (java API)
 - Industry (free cloud version), not configurable
 - Google
 - Nuance
 - AT&T Watson

Speech Recognition

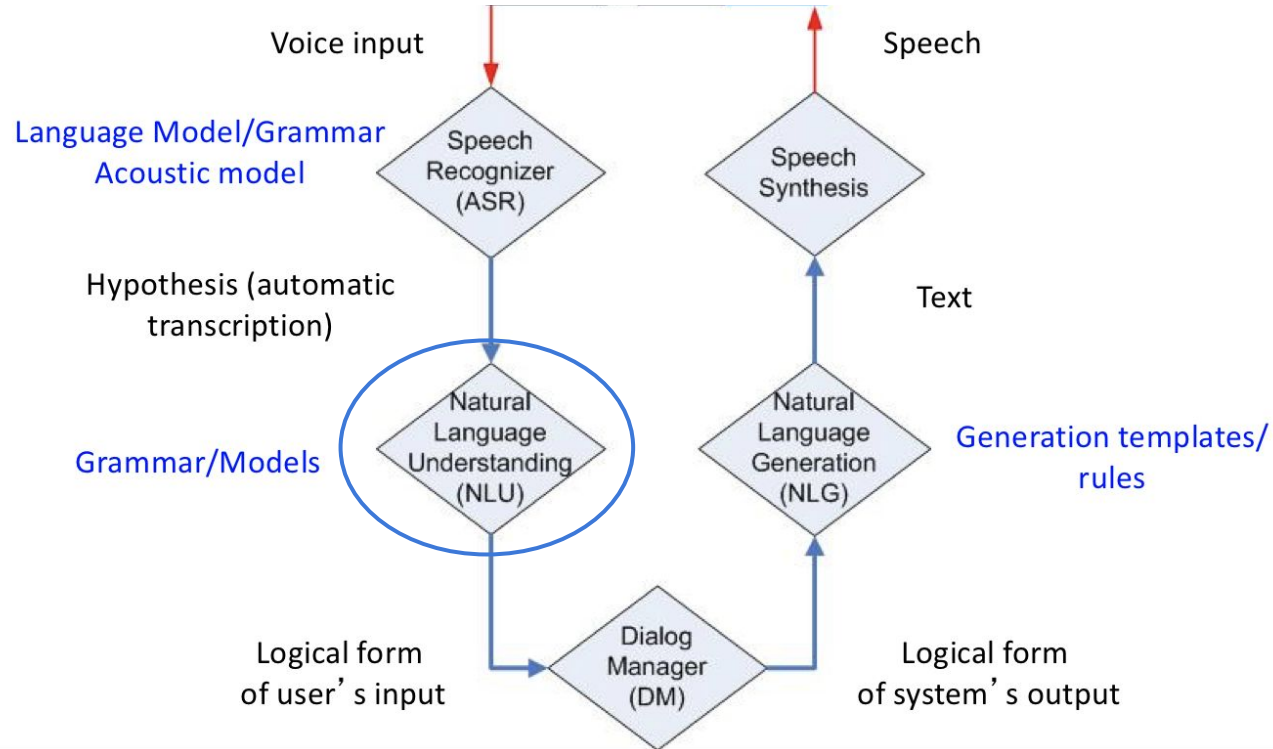
- ★ Statistical process
- ★ Use acoustic models that map signal to phonemes
- ★ Use language models (LM)/grammars that describe the expected language
- ★ Open-domain speech recognition use LM built on large corpora

Speech Recognition

- ★ Challenges: recognition errors due to
- ★ Noisy environment
- ★ Speaker accent
- ★ Speaker interruption, self correction, etc.

SYS:	Welcome to the CMU Let's Go bus information. What can I do for you?
USR:	I need to go from Oakland
ASR:	<i>I need to go .from. can't</i>
SYS:	Where do you wanna leave from?
USR:	to Waterfront
ASR:	<i>told. .me. Waterfront</i>
SYS:	Leaving from Waterfront. Did I get that right?
USR:	Oakland
ASR:	<i>Oakland</i>
	...

SDS Components



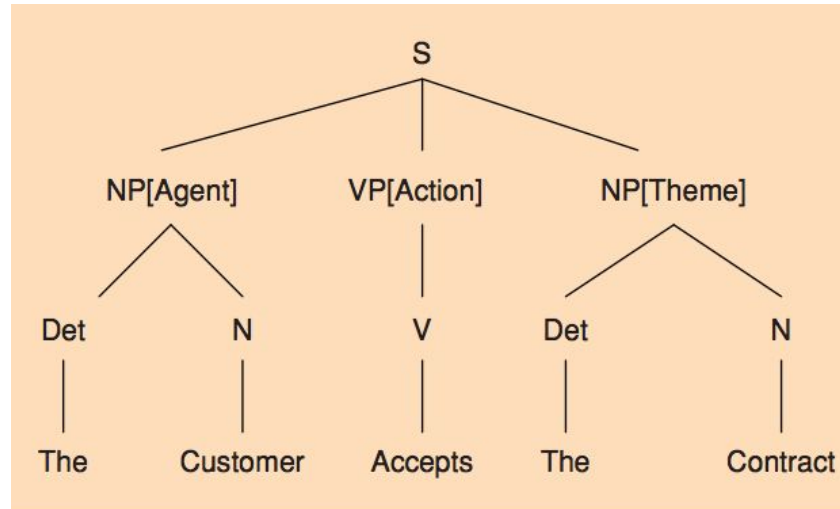
NLU

Convert input text into internal representation. Example internal representation in wit.ai:

```
{  
  "msg_body": "what is playing at Lincoln Center",  
  "outcome": {  
    "intent": "get_shows",  
    "entities": {  
      "Venue": {  
        "value": "Lincoln Center",  
      }  
    },  
  },  
  "confidence": 0.545  
},  
"msg_id": "c942ad0f-0b63-415f-b1ef-84fbfa6268f2"  
}
```

NLU Approaches

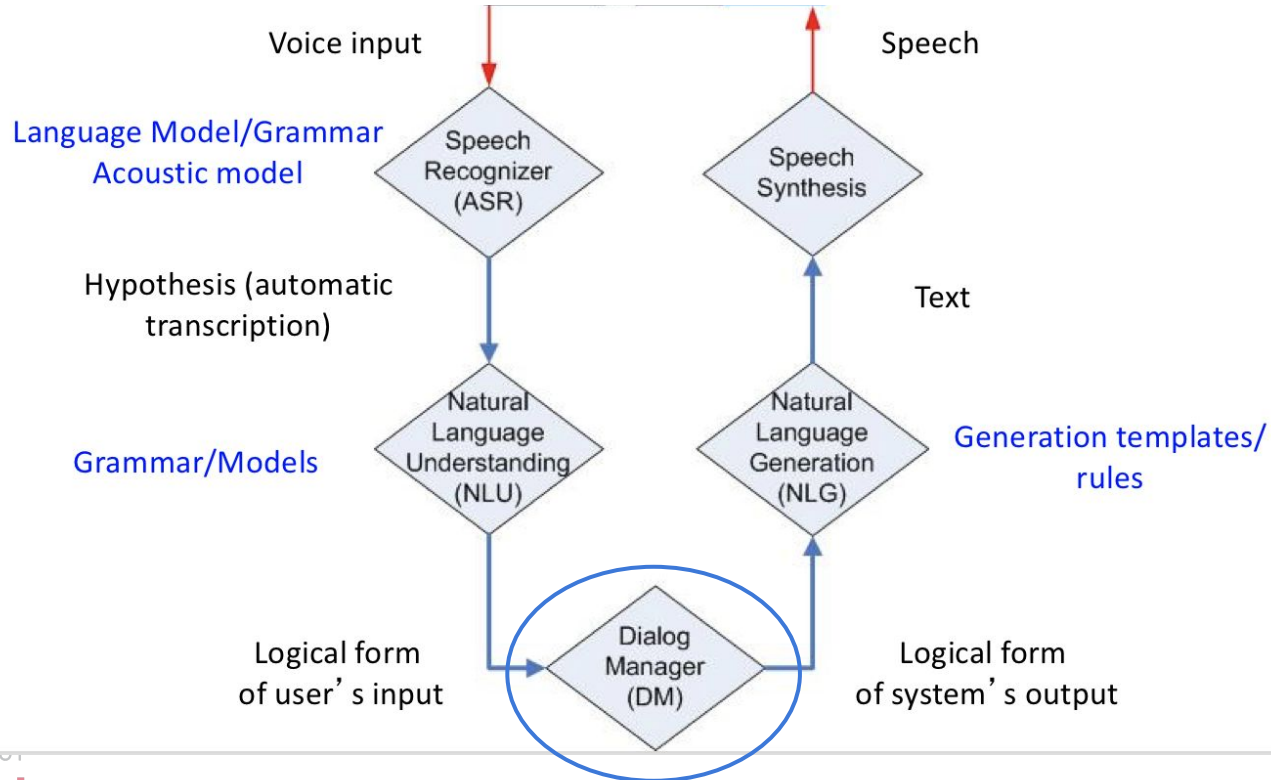
- ★ Can be based on simple phrase matching
 - “leaving from PLACE”
 - “arriving at TIME”
- ★ Can use deep or shallow syntactic parsing



NLU Approaches

- ★ Can be rule-based
 - Rules define how to extract semantics from a string/syntactic tree
- ★ Or Statistical
 - Train statistical models on annotated data
 - Classify intent
 - Tag named entities

SDS Components



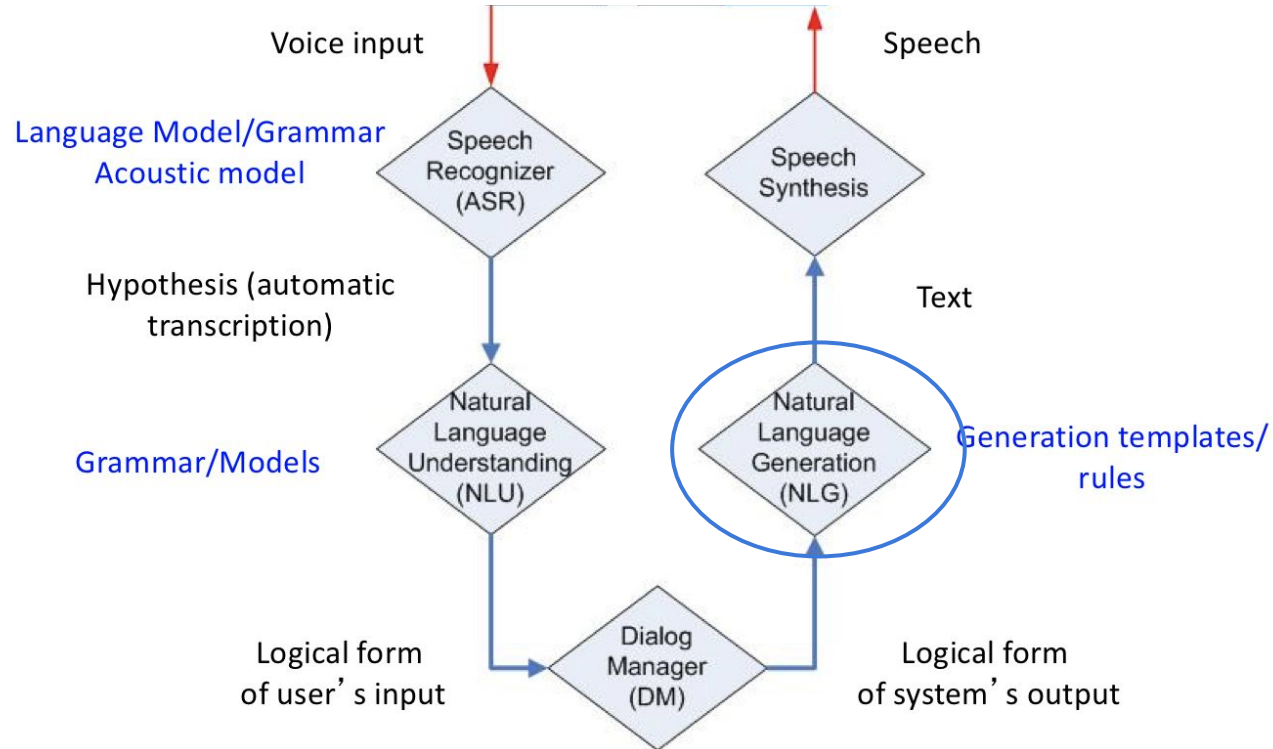
Dialogue Manager

- ★ Is the “brain” of an SDS
- ★ Decides on the next system action/dialogue contribution
- ★ SDS module concerned with dialogue modeling
- ★ Dialogue modeling: formal characterization of dialogue, evolving context, and possible/likely continuations

Dialogue Manager Approaches

- ★ Rule-based
 - Key phrase reactive
 - Finite state/Tree based
 - model the dialogue as a path through a tree or finite state graph structure
 - Information-state Update
- ★ Statistical (learn state transition rules from data or on-line)
- ★ Hybrid (a combination of rules and statistical method)

SDS Components



NLG Approaches

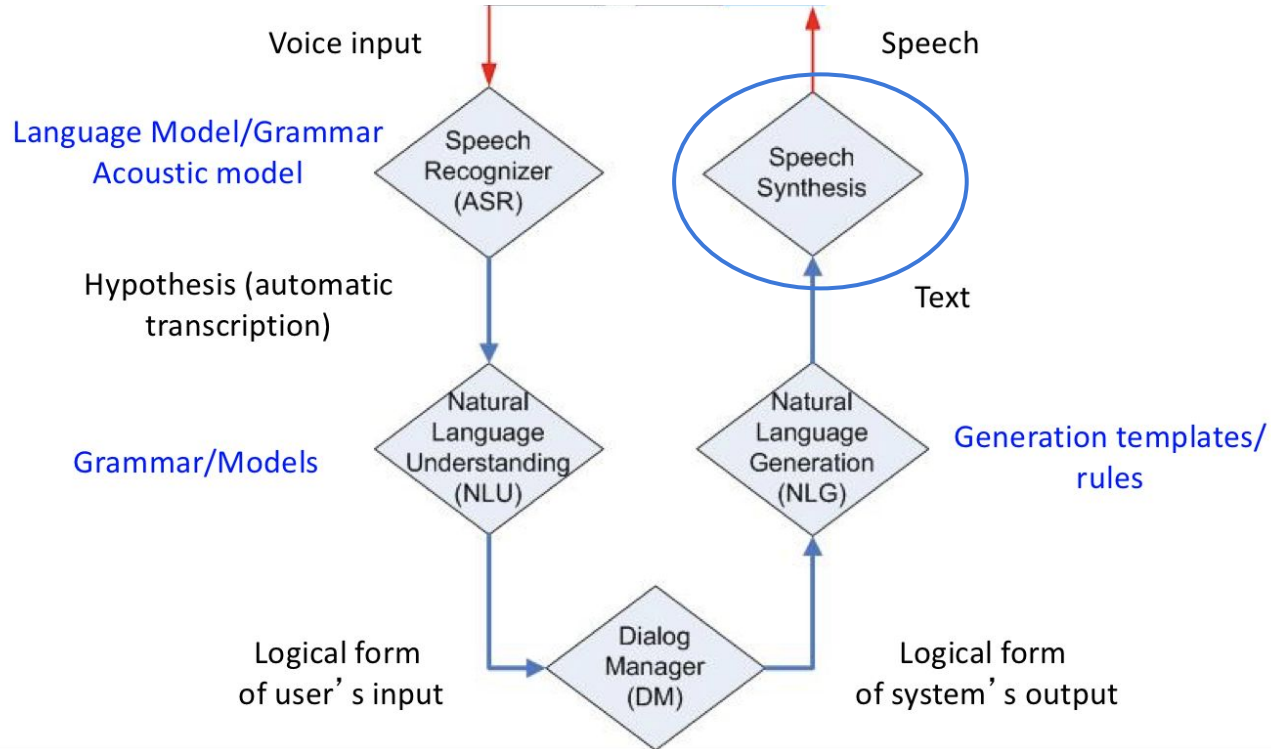
- ★ Presenting semantic content to the user
- ★ Template-based, in a airline reservation system:
 - User: “Find me a ticket from **New York** to **London**”
 - System: “What date do you want to travel?”
 - User: “**March 10**”
 - System: “There is a **United** flight from **Newark airport** to **London Heathrow** on **March 10** leaving at **9:15 AM**”
 - Template: There is a **AIRLINE** flight from **AIRPORT** to **AIRPORT** on **DATE** leaving at **TIME**

NLG

Content selection

- ★ User asks “Find me restaurants in Chelsea”
- ★ System finds 100 restaurants
- ★ NLG decides how to present a response and which information to present
 - “I found 100 restaurants, the restaurant with highest rating is ...”
 - “I found 100 restaurants, the closest to you is ...”
 - “I found 100 restaurants, I think you would like ...”

SDS Components

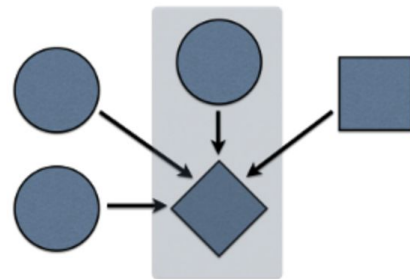


Tools for Developing DS

- ★ OpenDial – DM framework; Pier Lison (2014)
- ★ Wit.ai – A tool for building ASR/NLU for a system
- ★ Amazon Alexa
- ★ Google Assistant
- ★ VoiceXML

OpenDial

- ★ Pier Lison's PhD thesis 2014
- ★ DM components can run either synchronously or asynchronously
- ★ ASR/TTS: OpenDial comes with support for commercial off-the-shelf ASR (Nuance & AT&T Watson)
- ★ NLU: based on probabilistic rules
 - XML NLU rules
- ★ DM: rule-based. Dialogue states triggered with rules
 - XML DM rules
- ★ NLG: template-based
 - XML NLG rules



OpenDial

Wit.ai

- ★ 1.5 year start up recently bought by Facebook
- ★ Web-based GUI to build a hand-annotated training corpus of utterances
- ★ Developer types utterances corresponding to expected user requests
- ★ Builds a model to tag utterances with intents
- ★ Developer can use API using python, javascript, ruby, and more
- ★ Given speech input, output intent and entity tags in the output



Evaluation of DS

- ★ User satisfaction after interacting with a DS
- ★ But user evals are expensive, other proxy metrics:
 - Task completion success (% of correctly filled slots)
 - Efficiency costs: total time to complete task, # turns, or # queries
 - Quality cost: # the system failed to return an answer, or # of time outs

Current Research Topics in SDS

- ★ turn-taking
- ★ mixed-initiative
- ★ referring in dialogue
- ★ grounding and repair
- ★ dialogue act modeling
- ★ dialogue act recognition
- ★ error recovery in dialogue
- ★ prosody and information structure
- ★ Argumentation & persuasion
- ★ incremental speech processing
- ★ multi-modal dialogue
- ★ multi-party dialogue (3 or more participants)
- ★ tutorial dialogue
- ★ multi-task dialogue
- ★ embodied conversational agents
- ★ human---robot dialogue interaction

Current Research Topics in SDS

- ★ dialogue tracking in other language
 - -processing systems (machine translation, summarization)
- ★ non-cooperative dialogue systems (negotiation, deception)
- ★ affective dialogue systems
- ★ dialogue with different user populations (children, elderly, differently abled)
- ★ dialogue “in the wild”
- ★ long-term Dialogue Companions
- ★ user behavior, including entrainment in dialogue