Lecture 5 Chatbot and Languange Fundamental

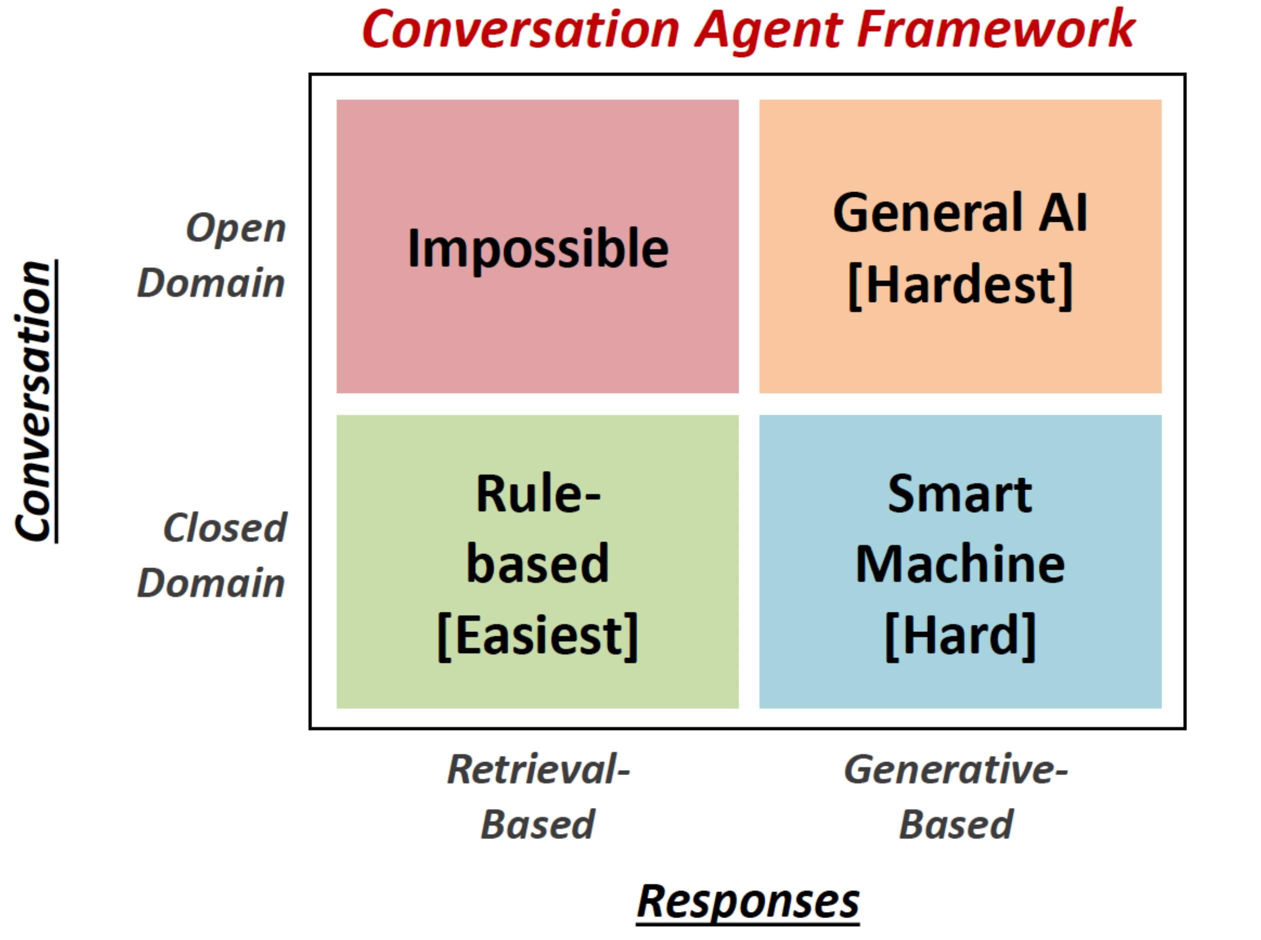
Coversational Agent

Concepts

— a software program

— interprets and respond the statement made by user in the natural language

— integrates computational linguistics techniques with communication over the internet



Conversation Agent Framework

Conversation / Responses	Retrieval-Based	Generative-Based
Open Domain	Impossible	General AI [Hardest]
Closed Domain	Rule-based [Easiest]	Smart Machine [Hard]

Goal-oriented Coversational Agent

For a particular task, utilizing short conversations to get information from the user to help complete

Frame-bsed Approch

1. Based on a "domain ontology"

   — a knowledge structure representing user intentions

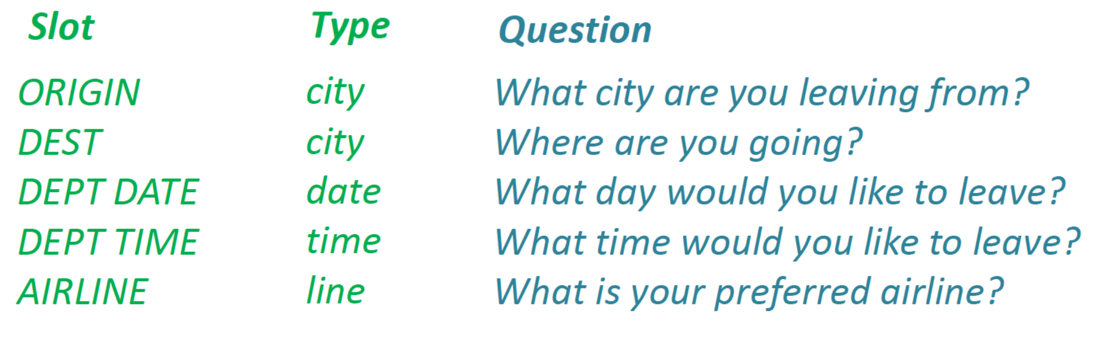
2. One or more frame

   — Each collection has a set of slots

   — Each slot having a value

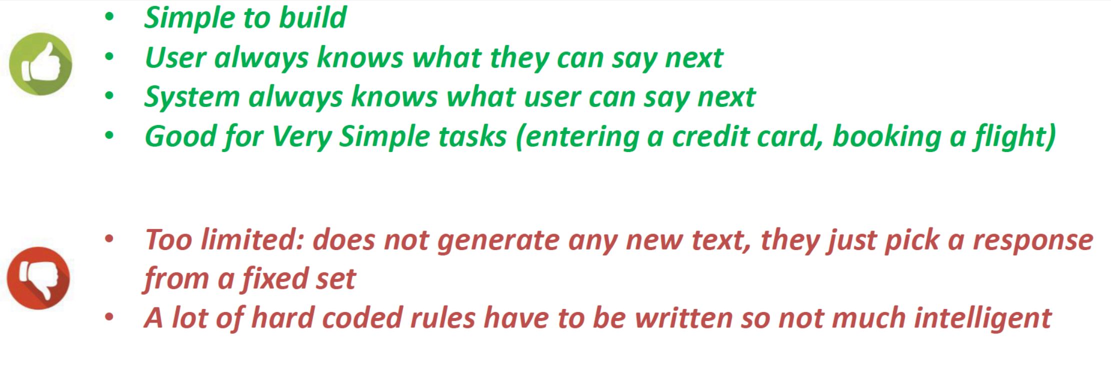
— Each set of slots, to be filled with information of a given typer

— Each type is associated with a question to the user



Slot	*Type*	*Question*
ORIGIN	*city*	*What city are you leaving from?*
DEST	*city*	*Where are you going?*
DEPT DATE	*date*	*What day would you like to leave?*
DEPT TIME	*time*	*What time would you like to leave?*
AIRLINE	*line*	*What is your preferred airline?*

1. Dialoge is structured in a sequence of predetermined utterance
2. Attributes
    a. system completely controls the conversation with the user
    b. It asks the user a series of questions
    c. ingnore (misinterpreting) anything the user says that is not a direct answer to the system's questions
3. Dialogue Intiative: System/single initiative



👍

- ***Simple to build***
- ***User always knows what they can say next***
- ***System always knows what user can say next***
- ***Good for Very Simple tasks (entering a credit card, booking a flight)***

👎

- ***Too limited: does not generate any new text, they just pick a response from a fixed set***
- ***A lot of hard coded rules have to be written so not much intelligent***

4. Initiative issue: handlling mutilple answers in one sentence to all questions
5. Solution: Mixed initiative

- Use the structure of the frame to guide dialogue
- system ask 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
- Approach:"**Frame and slot sematics**", to represent meaning of sentences

Condition-action rules Approach

1. Based on" active ontology"

   — relational network of concepts

2. Data structures: concepts with relation

   e.g. a meeting—>(a date,a loction,a topic,a list of attendees)

3. Rule(condition +action): sets that perform actions for concepts

   e.g. convert date to string

4. Improvement: ML (require lots of labelled data)

   given a set of *labelled* sentences, build a classifier to map from one to the author(words —> semantics frame-fillers)

Chatbots(Chat-oriented conversational agent)

For handlling full conversations, mimicking the unstructured flow of a human-to-human conversation

Rule-based Approach

1. Pattern-Action Rules(Eliza)

— pattern mathcing

— very basic reconstruction rules

— some programmed responses to special keywords

— randomisation to avoid getting stuck in a rut

— when all fails, some stock responses

2. Pattern-Action Rules + A mental model(Parry)

— same pattern-rule structrue as Eliza

— analysis the personal attributes with hand-written rules

Corpus-based (/w large chat corpus)

1. Information retrieval (IR) based

— Mine conversation of human or human-machine chats

— with large corpus (Twitter, movie dialogue etc.)

1. Returnthe response to the most similar turn
   - Take user's turn (q) find a similar (TF-idf) turn (t) in the corpus
   - Grab whatever the responses was to t:

     $r = response(argmax_{t \in C} \frac{q^T t}{||q||t||})$

2. Return the most similar turn:

   r=$argmax_{t \in C} \frac{q^T t}{||q||t||}$

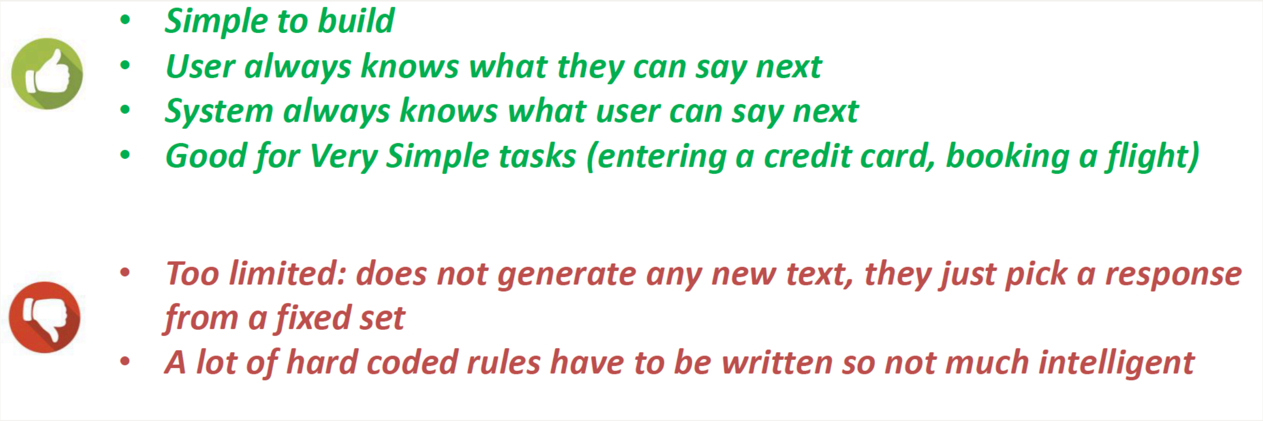
3. fine to user other features, e.g user features, prior turns, non-dialogue text

2. DNN

— Think of response generation as a task of transducing from the user's prior turn to the system's turn

— Train on: Movie Dialogues, Twitter Conversation

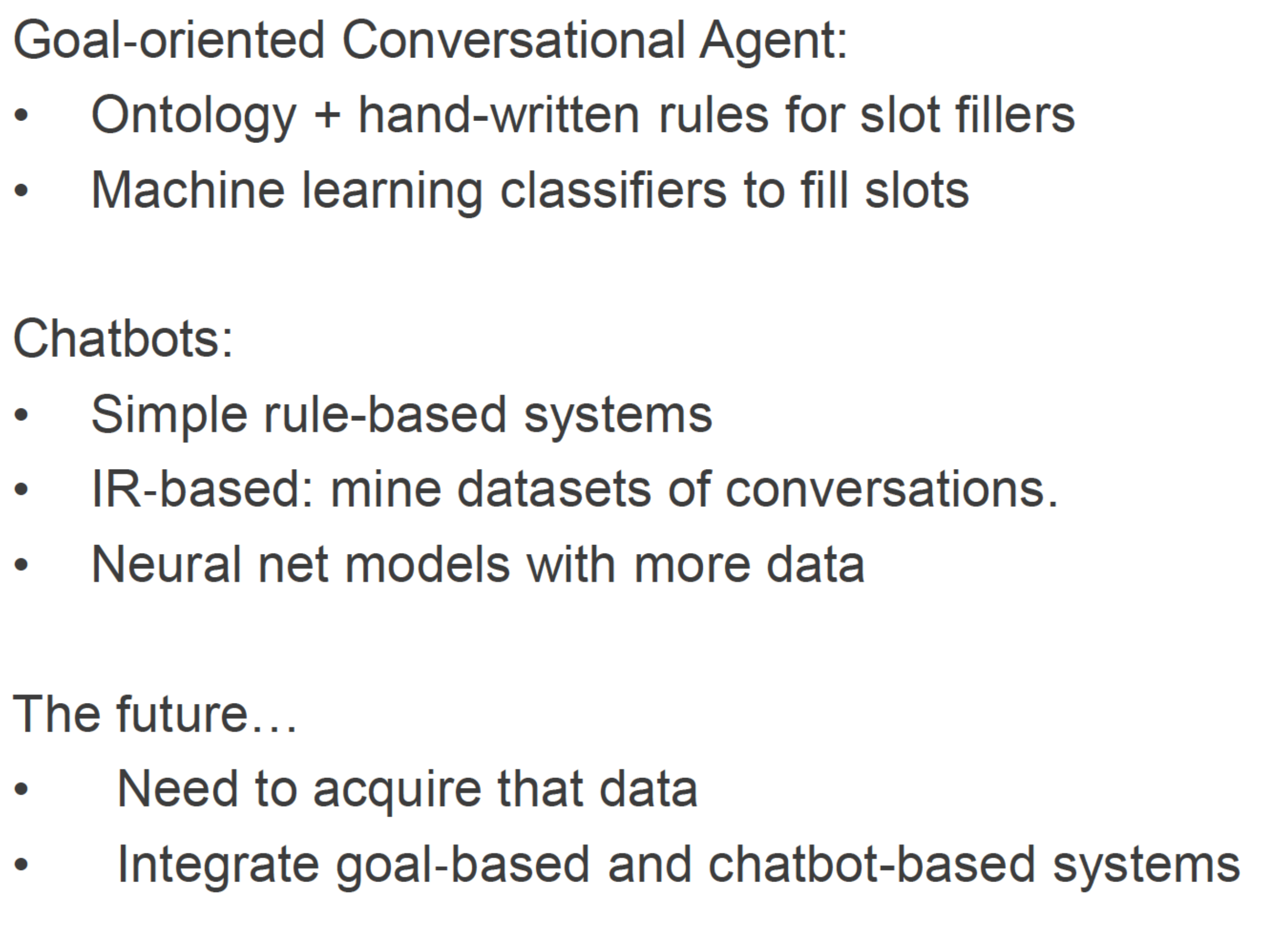
— Train DNN: map from user 1 turn to user 2 response



👍
- *Simple to build*
- *User always knows what they can say next*
- *System always knows what user can say next*
- *Good for Very Simple tasks (entering a credit card, booking a flight)*

👎
- *Too limited: does not generate any new text, they just pick a response from a fixed set*
- *A lot of hard coded rules have to be written so not much intelligent*

Summary



Goal-oriented Conversational Agent:

- Ontology + hand-written rules for slot fillers
- Machine learning classifiers to fill slots

Chatbots:

- Simple rule-based systems
- IR-based: mine datasets of conversations.
- Neural net models with more data

The future…

- Need to acquire that data
- Integrate goal-based and chatbot-based systems

Languange Fundamental

1. Phonology/Morphology

- Composed of a prefix , an affix
- The structure of words

2. Sytax

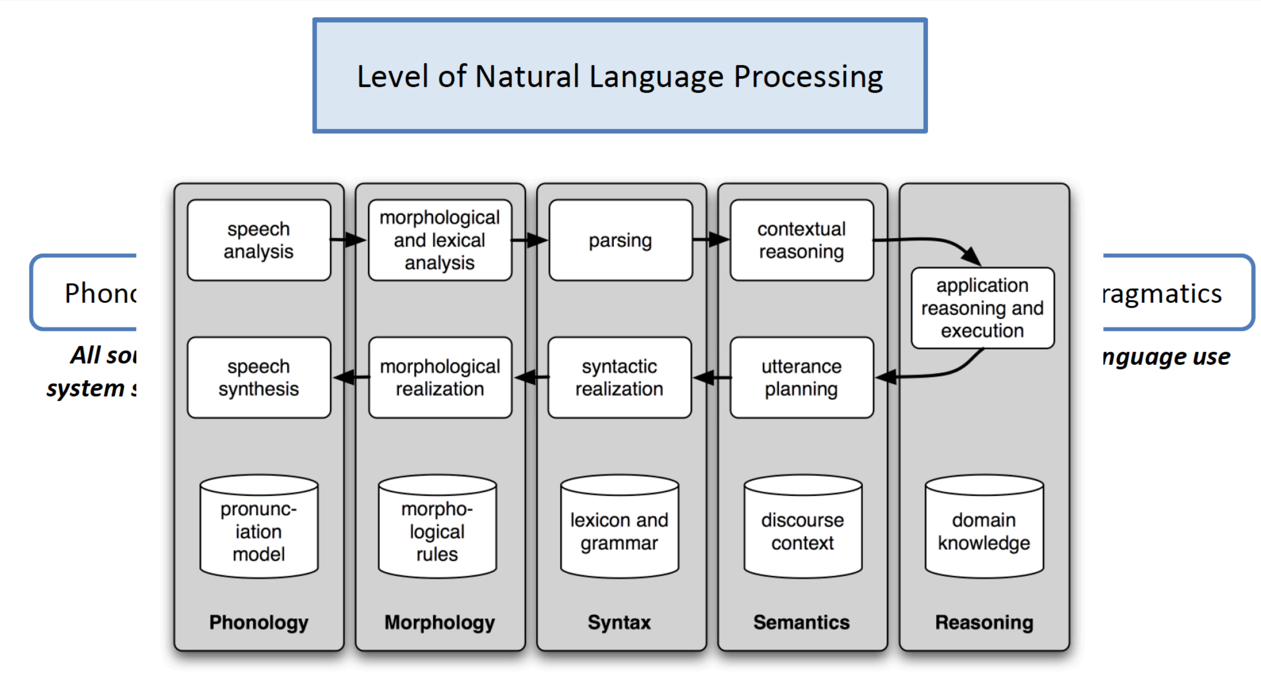
- The way words are used to form phrases

3. Sematics

- Compositional semantics: the construction of meaning based on syntax
- Lexical semantics: the meaning of individual words

4. Pragmatics

- Meaning in contex



- Pieces of sounds: in or not in the language? How sounds can combine?

  —> ***Phonology***

- Meaning: the context of the utterance

  —> ***Sematics and Pragmatics***

- Pieces of words : bases, roots and affixes. how words are formed or marked via other processes?

  —> ***Morphology***

- The order of words in the sentence: How words comibine? How words go in relation to another

  —> ***Syntax***

- Words and morphemes: mental dictionary

  —> ***Lexcicon***

Text Preprocessing

Normalization

- Need to 'Normalize' terms

  — IR: indexed convex & query terms must have same form e.g. U.S.A=USA

- Implicitly define equivalence classes of terms

  — e.g. deleting periods in aterm

- Alternative: asymmetric expansion
- powerful but less efficient

Case Folding

- Application like IR: lower case all letters
- for sentiment analysis, machine translation and informarion extraction

  — case is helpful (US v.s us)

Lemmatization

- Reduce inflections or variant forms to base form

  — e.g that's ->that is; is, are,am ->be

- Have to find correct dictionary headword form

- ML

Morphology

- Morphemes : small meaningful units that make up words

  — Stems: core meaning-bearing units

  — Affixes: bits and pieces that adhere to stems

  — often with grammatical functions

Stemming

- Reduce terms to their stems in IR
- Stemming is crude chopping of affixes

Sentence Segmentation

- Indentifying relatively unambiguous

  — e.g !,?

- Identifying ambiguous

  — e.g "." for abbreviation,numbers,sentence boundary

- Build a binary classifier (Decision Tree)

  — looks at a sentence boundary "."

  — decides EndOfSentence/BegOfSentence

  — classifiers: hand-written rules, regular expressions, ML

Regular Expression

- Fixing two types of error

  — Type I (False Positives): matching cases that we should not have matched

  — Type II (False Negatives): not matching cases that we should have matched

- Reducing error

  — Increasing accuracy and precision (minimizing FP)

  — Increasing coverage and recall (minimising FN)

Summary

1. RE

   — Main tool for text preprocession

   — sophisticated sequences

2. ML (hard task)

   — RE could be used as features

   — better for generalisations