Speech Processing 15-492/18-492

Spoken Dialog Systems
Beyond basic dialogs
Building your own dialogs
Back-channelizing

- **Human response to speech**
  - *Robots don’t really do this*
  - *Uhms, errs filler works*
  - *Yeah, uh-huh, hm, right, okay*
  - *Typically words *not* in the lexicon*
  - *Prosody delivery is important*
  - *Timing is important*
H It is like a party, like, “rave” type party or like
C well, it’s someone’s house
H yeah
C there’s going to be, I mean there’s like, they’re going to be spinning. So, in that sense, maybe, but it’s just at someone’s house, like
H yah-yeah
C It’s in the middle of the night, that, too, but

(from Nigel Ward UTEP)
Timing

- Replies happen before question ends
- Humans can guess when turn is ending
  - Combination of semantics, prosody (and arrogance)
- Human-machine dialogs more restricted
Gesture and Gaze

- **What you look at when talking**
- **What the machine should look at**
- **Talking to the machine vs talking to your friend**
Laughter

◆ **Most common non-verbal vocal production**

◆ **Should machines laugh?**
  - Yes to fit in with the other participants

◆ **Laughing takes different forms**
  - Near verbal (ha ha ha)
  - Vocal but unlike speech
  - Subvocal
  - Overlayed on speech
Participant in Meeting

- **Machine participants in meetings**
  - At least follow the speaker
  - Know when to agree/laugh etc
  - Know when it can speak
    - Needs to watch how people interact
Needs to watch what you do

- When are you busy
- When are you interruptable
- What is the importance of the information
- (Cell phone just rings, no matter where you are)

Look at human brain state

- Find when you are thinking
- Busy, thinking, dreaming
How do humans interact with machines

- **Look at human-human calls**
- **“Pretend” they are talking to a machine**
  - “Wizard of Oz” (WOZ)
  - Have a human play a machine
  - Need to constrain the human
    - Give them “robotic” voice
    - Constrain their options
Building a New Dialog Systems

- **What will it do?**
  - Write down a typical dialog
  - No *really* write down a typical dialog
  - Write a second (simpler) one

- **Look at human-human dialogs**
  - What information is being passed
  - Can you avoid the hard ASR parts
    - (Avoid large numbers of names)
Breaking down the task

**What is the ontology**

- *What entity types must you deal with*
  - e.g. Busses, times, bus stops
- *How will people say them*
  - List *many* yourself and ask others
- *How should your system say them*
  - Consistently, and in a way that’s easy to recognize
What is the flow of the dialog

- How should you order the questions
- Should you allow multiple orders
- Is this ordering reasonable for your users
  - Ask others, you are too close to the task
- Test with your written down dialogs
  - (You did write them down didn’t you?)
Write grammars for what response

- Test them with multiple examples
- (Get others too if you can)

Test it with text.

- ASR will have errors
- Test by typing first, easier to debug
Testing the dialog

- **Check for one dialog you know works**
- **Test it in the system**
  - Modify your grammar/dialog accordingly
- **Then try the variations**
- **Get others to test it**
- **Does it do the task you expect**
Help

- **Try to be consistent and concise**
  - Give good examples of what to say
  - Give multiple levels of help
  - Nobody will listen …. 

- **Test your help advice**
  - Is it really useful?
SDS Architecture