Speech Processing 15-492/18-492

Spoken Dialog Systems
Deployment of real systems
From the lab to real world

- **Developing a real usable system**
  - Strategy to deployment
  - Engineering issues
  - Stability issues

- **Call analysis**
  - Finding issues in calls
From the lab to the real world

- **Build system that works for you**
- **Have locals (developers) use it**
  - Analyze their usage and build a new system
- **Have subset of target users use it**
  - Analyze their usage and build a new system
- **Have target users use it**
  - Analyze their usage and build a new system
- **Over time after successful deployment**
  - Analyze usage and build a new system
Improving a system

Improving components

- Acoustic models (channel-appropriate data)
- Language models (what people actually say)
- Grammar models (how to interpret it)
- Dialog strategies (how people get what they want)
- New functionality (asking for things you could give)
Improving Acoustic models

- **Collect date of target audience using system**
  - It’s from the right channel
  - It’s the right distribution of vocabulary
  - It’s the right distribution of dialects
  - It’s the right distribution of style
- **Collect over time**
  - You may have different users and different times of the week.
Collecting real data:

- What words and phrases are used
  - (Are people polite, rude, include greetings …)
- How do people actually ask for things you offer

Grammar additions

- Modify your grammar to deal with what real users actually say
Dialog Strategies and New Functions

- **What are people doing**
  - Analyze dialog states and look for novel events
    - (e.g. adding “next” and “previous” bus)

- **What are they saying when you tell them you can’t help.**
  - What functionality are they asking for
What is happening in your 100K calls?
- Estimate success
- Count number of turns
- Count number of known errors
- Trace different dialog state sequences
  - Are some more likely to fail.
- What should you do next to improve task success.
Call Analysis

- **Cannot listen to the all**
  - Need optimal way to sub-sample
- **Find the “interesting” calls**
- **Who are your users**
  - First time callers, repeat callers
  - What classes succeed
  - What classes fail
Dialog System Customer

- **USAir, Amtrak, AT&T ....**
- They wish to minimize operator calls
  - % of calls dealt with automatically
  - (successfully or not ?)
Dialog Users

- They want the service to work
- They want it to work in an obvious way
- It must be better than waiting for a human
- It must be able to deal with their task
- It must get better for them
Must be easy to maintain
  * Not require re-design every day

Must be fast
  * The more calls you serve the money you make
  * The faster calls are the less equipment you need

Must adapt to the needs
  * Customer and users
Engineering

- Must be reliable hardware/software
- Can it deal with (near) simultaneous calls
- Can it deal with very long calls
- POTS is not a very stable system
  - Hard to detect hang up
- It has to run 24/7
- You must detect hangs automatically
  - Not easy on some operating systems
Cost per call
- What is the average length of a call
- How many simultaneous calls per machine

Can it scale to 10, 100, 1,000 …

Can you deal with call volume
- What are the peaks
- What are the down times

Can your (Amazon) Cloud deal with that?
Spoken Dialog Systems

- **Types of systems**
  - Task oriented, question/answering systems
  - Mixed initiative systems
  - HMIHY: classification tasks

- **Dialog Components**
  - ASR, Parsing
  - Dialog Manager
  - Generation, TTS
Spoken Dialog Systems

- Development Systems
  - VoiceXML
  - Olympus

- Deploying systems
  - Iterative development
  - Call analysis
  - Adapting to improve usage
SDS Architecture

- Recognizer
- Audio Server
- Synthesizer
- Parser
- Dialog Mgr
- Language Generation
- Domain Agents
- Internet