Logistics

Announcements, Assignments, and Reminders

- Welcome to the class!
- We’ll meet Tuesday/Thursday from 1:30-3:20pm in our room (Thomson Hall, Room 235).
- Office hours will be: Tuesdays 3:35pm-4:35pm.
Prerequisites

- basic probability, statistics, and random processes (e.g., EE505 or a Stat 5xx class or consent of the instructor).
- Knowledge of and access to matlab (some of the homeworks will require matlab). See https://www.ee.washington.edu/computing/index.html if you are a UW student for access via the EE systems.
- Knowledge of some programming language (C, C++, Java, etc.)
- Some signal processing (e.g., EE518, meaning basic linear systems, FIR/IIR filter theory, low-pass/band-pass/high-pass filters, etc.).
- The course is open to students in all UW departments.

Class web links and infrastructure

- Check in with our web page (http://j.ee.washington.edu/~bilmes/classes/ee516_spring_2013/) for up to date announcements, homeworks, etc.
- All homeworks will be due via our dropbox (https://catalyst.uw.edu/collectit/dropbox/bilmes/26924)
- All questions should be posted to our discussion board (https://catalyst.uw.edu/gopost/board/bilmes/32667/)
- You can contact me anonymously if you wish via anonymous email (https://catalyst.uw.edu/umail/form/bilmes/4208)
Homeworks

- There will be 2-3 homeworks this quarter, due about 1-2 weeks after they are assigned.
- Some will involve programming assignments.
- All will be due electronically via our dropbox (https://catalyst.uw.edu/collectit/dropbox/bilmes/26924), no paper assignments accepted.

Final Project

- There will be a final project for the course,
- The projects must have something to do with speech processing or recognition.
- Ideally, it will be original research related, but it may also be an in-depth paper review.
- What will be due (and what is graded):
  - 15-20 minute conference style presentation
  - presentation slides (electronically)
  - A 4-page writeup summarizing the presentation.
  - Preliminary deadlines leading up to the final project (e.g., project summaries and progress reports).
- All will be due electronically via our dropbox (https://catalyst.uw.edu/collectit/dropbox/bilmes/26924), no dead-tree assignments accepted.
- Final presentations will be Monday, June 10th, 2013. Time/location TBD
- Otherwise, no regular midterm and final exam.
On Final Project

- Project should ideally be on some aspect of the material we have learnt, some aspect of speech processing or recognition. Possible good projects include:
  - an implementation and reporting and experience that you gain in doing this. Application to real data.
  - A paper summary, of papers that we are not going to cover in this class.
  - A new idea of your own, new algorithms and/or theoretical results. (e.g., new approach to speech recognition, new deep method, etc.).
- Ideal project would lead to a conference and/or journal paper.
- Fine to combine it with your own research (assuming you are working on speech).

Grading

Grades will be based on a combination of:
- final project (50%)
- class attendance (10%)
- class participation (20%)
- homework assignments (20%)
Guest lecturers this term

The last three class lectures will be by the following speakers:

- Dr. Geoff Zweig, MSR, on advanced language model methods
- Dr. Alex Acero, MSR, on speech synthesis methods.
- Dr. Li Deng, MSR, on deep models for speech recognition.

Our Main Text

Huang, Acero, Hon, “Spoken Language Processing”.
Other Books & Sources

- Clark & Yallop, “An Intro to Phonetics and phonology”
- Ladefoged “A Course in Phonetics”
- Lieberman & Blumstein “Speech physiology, speech perception, and acoustic phonetics”
- K. Stevens, “Acoustic Phonetics”
- Malmberg, “Manual of phonetics”
- Rossing, “The Science of Sound”
- http://www.ling.upenn.edu/courses/ling001/schedule.html Linguistics 001, University of Pennsylvania

Outline of today

- Overview of course objectives and handout.
- Brief overview of the course
  - human/computer interaction
  - Origins of speech processing
  - Speech synthesis
  - Speech and auditory perception
  - Automatic speech recognition
  - Applications in industry
Interface between humans and computers has evolved over the years. Initially, it involved re-programming, punched cards, programming languages. Since the late 1960s, real “interaction” was possible, with the computer mouse. This evolved into Windows, Icons, Mouse, Pointer (WIMP) interfaces. More recently, other forms of interaction, Writing pad/tablets (touch), scanners, cell-phone, cameras, TVs, wearable computers (watches, glasses), Overall Goal: efficient, easy, and ergonomic human-computer interaction. True human-human contact almost always involves speech. Hence, speech should be at the forefront of the ideal human/computer interaction mechanism.

Speech and Hearing, most natural mode of human-human communication Speech can be used for human-computer interface as well. This class is an introduction to how such systems work. For a speech HCI system, need multiple things:

- Speech production & perception
- Speech processing (what is speech, how does it work, and how to process it).
- Speech synthesis (probably around 1 lecture).
- Speech recognition (mapping the acoustic signal to text, most of the course).
- Dialog systems (how to build a conversational system with a computer and generate appropriate responses to a human).
Origin of Speech Processing: To Dazzle?

- From Wikipedia: The Turk (Mechanical Turk) was a fake chess-playing machine, late 18th century. From 1770-1854, it was exhibited as an automaton, though exposed by 1820s as a hoax. Constructed in 1770 by Wolfgang von Kempelen (1734-1804) to impress an Empress, the “machine” could play chess and other games.

Wolfgang Ritter von Kempelen’s Speaking Machine

- Very early on, there was desire to build a real speaking machine.
- Wolfgang Ritter von Kempelen (late 1700s) actually built a real “speaking machine” that was not a hoax.
- Figure from: Dudley & Tarnoczy, JASA 22, 1950 and shows Wheatstone’s version, 1835.
Wolfgang Ritter von Kempelen's Speaking Machine

The following shows a re-creation
http://www.youtube.com/watch?v=zYRVqrfY3tQ

Speech Synthesis

As we've just seen: mechanical devices to achieve speech synthesis were conceived of in the realm of fiction, and first devised in the late 18th/early 19th century.

The invention of the telephone in the late 19th century, and the subsequent efforts to reduce the bandwidth requirements of transmitting voice, led back to the idea.

In the 1930s, the telephone engineers at Bell Labs developed the famous Voder, a speech synthesizer that was unveiled to the public to great fanfare at the 1939 World’s Fair, but that required a skilled human operator.
Origins of Speech Processing

If I could determine what there is in the very rapidly changing complex speech wave that corresponds to the simple motion of the lips and tongue, if I could then analyze speech for these quantities, I would have a set of speech defining signals that could be handled as low frequency telegraph currents with resulting advantages of secrecy, and more telephone channels in the same frequency space as well as a basic understanding of the carrier nature of speech by which the lip reader interprets speech from simple motions

– Homer Dudley, 1935

Homer Dudley’s Voder

- One of the first to understand the information bearing element of speech (all done at Bell Labs)
- Originally demonstrated 1939 Worlds Fair
Homer Dudley’s Voder

- Finger keyboard
- Voiced/unvoiced excitation
- Resonator filters (BPF)
- 10-key keyboard
- Foot controls pitch

History of Speech Synthesis

- In general, over the years, speech synthesis has increased both in terms of its “flexibility” and its “naturalness”
- In general, these characteristics trade-off with one another.
- Visit web page: http://www.festvox.org/history/klatt.html including examples of the voder.
Speech Synthesis Today

- A real industry (some say it is a solved problem)
- Goal: Overcoming consumer resistance
- Humanizing the machine’s voice for the human for the speaker
  - Augmentative Communication
  - Deaf or disabled Speech (e.g., Steven Hawking)
  - Humanizing the machine’s voice for the listener

Or maybe a bit more ominous: Activision Character Tech Demo in Real Time http://www.youtube.com/watch?v=y2GhOLHzI1c

Speech Synthesis: Broad overview

- Text Processing
- Prosody Prediction
- Waveform Generation

My office was on St. Mary’s St. one block from the coffee shop.

My office was on Saint Mary’s Street, one block from the coffee shop.

*My office \( \text{was on Saint} *\text{Mary’s Street} \) \( | \) \( | \)
*one block from the *coffee shop.
Introduction/Overview

Speech Synth

Speech Recon

Where is Speech Recognition?

Speech Proc

Summary

Scratch

Text Normalization

- Unrestricted text-to-speech synthesis requires: Text normalization: interpreting abbreviations and non-standard words. Example from Sproat et al. 99:

  57 ST E/1st & 2nd Ave Huge drmn 1 BR 750+ sf, lots of sun & clsts. Sundeck & Indy facils. Askg $187K, maint $868 util incl. Call Bkr Peter

  Play file://ads.fest.wav

- Needs linguistic analysis: parsing, word sense disambiguation (bass: fish or instrument?).

- Human-computer dialogues use text generators that can potentially provide this information

- Inherent trade off in synthesis, flexibility vs. naturalness

Origin of Electronic/Computational Speech Processing

- Key original motivation is simple bandwidth reduction - how to describe the intelligibility information on speech as cheaply as possible.

  0-4kHz ⇒ 8kHz sps, 8 bits ⇒ 64k bps vs. 1.2k bps for modern speech coders

- Cell phones - goal: reduce bandwidth

- LPC - linear prediction coefficients

- CELP - code excited linear prediction

- SOLA (synchronous overlap & add)

- Reduce time-span w/o reducing spectral quality or pitch.

- Speech synthesis and speech compression are highly related conceptually

  - Speech communications: encoding and compression, transmission of speech description, decoding and decompression from description

  - Speech synthesis: decoding and decompression from description
Speech/Auditory Perception

- Important for both speech recognition and speech coding
- Human auditory system — We have a working system, so should study it.
- The auditory periphery and critical band analysis — Auditory system does quite a bit before sending signal to brain.
- Analytical results:
  - perceptually inspired spectral analysis
  - autocorrelogram processing
  - speech intelligibility lies within modulation domain
- Moving up the auditory nerve towards brain, knowledge becomes much more sketchy.

Machine Learning & Statistical Pattern Recognition

- Fundamental problem is too difficult to program explicitly (e.g., rule-based system)
- Solution: Write a program that can learn how to solve the problem
- Many possible learning algorithms (beyond this course)
- Need lots of “training data”, so the program can learn.
- This is what statisticians have been doing for years (but not necessarily on speech)
Is speech recognition a solved problem?

- In other words, is there no more research to be done, and we can think of speech recognition as off the shelf technology?
- 2001, the Hal 9000 computer suggests people thought it would be solved by 2001.
- where “it” = Seamless interactive intelligent conversations with a computer using naturally spoken conversational speech
- So, is automatic speech recognition (ASR) a solved problem at this point?
- Yes and no.
- First, lets ask SIRI if speech recognition is a solved problem . . .
- Next, lets look at a relatively recent plot.
Why is the problem so difficult

- Background noise, “cocktail party” effect.
- Channel differences between training and testing: Head-mounted vs. desktop mic: 10% vs. 70% WER for a speaker-trained commercial system
- Read versus spontaneous speech:
  
  yeah yeah I’ve noticed that that’s one of the first things I do when I go home is I either turn on the tv or the radio it’s really weird

  Play file://read2n.wav vs. file://spon2n.wav
- Speaker variability: accent, dialect, situational (motherese), age (child vs. older speaker), and natural variability between humans (idiolect).
Technology Development

We Are About Here

Research & Development needed to move to the right.

Automatic Speech Recognition (ASR)

Block diagram overview of ASR:

```
<table>
<thead>
<tr>
<th>Audio</th>
<th>Speech</th>
<th>Signal</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>Probability Calculation</td>
<td>Select Maximum</td>
</tr>
<tr>
<td>Feature Extraction</td>
<td>Likelihoods p(X</td>
<td>M)</td>
</tr>
<tr>
<td>Model Database M1 = &quot;How to wreck a nice beach&quot; M2 = &quot;How to recognize speech&quot; M3 = ...</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

“How to wreck a nice beach”
**Inspiration: Communication and Information Theory**

Semantic Intent → Linguistic Representation → Word Sequence → Sub-Word Sequence

$S$ → $L$ → $W$ → $U$

glotal & articulatory control sequence → Markov state sequence → acoustic waveform

$A$ → $Q$ → $X$

acoustic waveform at ear

$Y$

---

**“Fundamental” Equation of Automatic Speech Recognition**

\[
W^* = \arg\max_W \Pr(W|X) \\
= \arg\max_W \Pr(X|W)\Pr(W)
\]

- $X$ is a transformation of the acoustic signal into a sequence of vectors.
- $W$ is a sequence or string of words
- $\Pr(X|W)$ is the acoustic model and $\Pr(W)$ is the language model.
- Actually, this is the fundamental equation of minimum Bayes-error statistical pattern recognition. If truth abides by these distributions, this will minimize error.
Five Stages of Speech Recognition Systems

- Stage 1: Signal Processing/Feature Extraction
- Stage 2: Acoustic Modeling
- Stage 3: Pronunciation Modeling
- Stage 4: Language Modeling
- Stage 5: Spoken Dialog and higher level

Stage 1: Signal Processing/Feature Extraction

- Spectral Processing, Downsampling, and orthogonalization: Mel-frequency cepstral coefficients (MFCCs)

Speech signals are “framed” (short-time analysis)
Stage 2: Acoustic Modeling

- Mixtures of Gaussian Densities

\[ \Pr(q | q) = \sum_j c_j N(q | \mu_j, \Sigma_j) \]

- These are the most widely used still, even in the context of deep systems since Gaussian mixture hidden Markov model systems (that we will talk extensively about in this course) are needed to build deep systems.

- Other “densities” are possible, such as neural networks, deep models, conditional exponential models, etc.

Stage 3: Pronunciation Modeling

- Markov Chains determine the set of possible sequences of phones, syllables, or other constituent units.

- Example: The pronunciation Markov transition matrix for the word “and”

[Diagram of a Markov chain with states representing sounds]
Stage 3: Pronunciation Modeling

- Many possible units may be used to specify a pronunciation
- Most common: phonemes and their realizations, phones, ex using ARPAbet. Example: chocolate pudding ⇒ CaKxlIt pUdG
- phones can often be characterized acoustically (using formants, and their characteristic frequencies)
- syllables are another unit, not so widely used although researched.
- individual articulatory gestures: semi-asynchronous and independent parallel controls in the human vocal tract. Very promising research area for pronunciation modeling.

Stage 4: Language Modeling

- Goal: describe the probabilities of sequences of words, $p(w|h)$
- Most common language models: Trigrams $p(w_n|w_{n-1}, w_{n-2})$
- Difficult parameter estimation problem (e.g., 60k types, $2.16 \times 10^{14}$ entries, and this a small vocabulary size, many production systems have vocabulary sizes of 500k!!)
- Estimation is difficult (thus smoothing, backoff, and many other methods are used)
Stage 2+3+4: Hidden Markov Models

- Different utterances will have different lengths
- E.g., stop consonants ('k', 'g', 'p') are almost always short, but vowels are typically long(er), and need a model for these variable length units.
- Need a way of comparing and scoring variable length features.
- Dynamic Time Warping (earlier solution)
- Modern systems use statistical approach: Hidden Markov Models (HMMs)

What is a model?

Model Airplane

- mimicry of a real process or system
- Accuracy is task-specific - model good for a 14-year old hobbyist vs. a 1 year old baby
Markov Chains

- One of the simplest forms of time-series models
- The “present” is fairly local, only the current state.
- Can exist in various orders. Example, 1st, 2nd, and 3rd order Markov chain.

Corresponds to

Left: \[ p(x_5|x_4)p(x_4|x_3)p(x_3|x_2)p(x_2|x_1)p(x_1) \] (1.3)
Center: \[ p(x_5|x_4, x_3)p(x_4|x_3, x_2)p(x_3|x_2, x_1)p(x_2|x_1)p(x_1) \] (1.4)
Right: \[ p(x_5|x_4, x_3, x_2)p(x_4|x_3, x_2, x_1)p(x_3|x_2, x_1)p(x_2|x_1)p(x_1) \] (1.5)

Stage 2+3+4: Hidden Markov Models
Stage 2+3+4: Hidden Markov Models
overlapping windows of speech

- A Markov chain with other variables hanging off of it.
- Typically, the Markov chain is hidden, and what is observed are the $X$ variables.
- Markov Chain is typically 1st order, but extensions exist.
- Multiple possible training algorithms (e.g., some discriminative and some generative) are available, each with computational implications.
HMM: As factorization

- It is well known that the distribution of an HMM \( p(x_{q:T}, q_{1:T}) \) can be factored thusly:

\[
p(x_{1:T}, q_{1:T}) = p(x_T, q_T | x_{1:T-1}, q_{1:T-1}) p(x_{1:T-1}, q_{1:T-1}) \\
= p(x_T | q_T, x_{1:T-1}, q_{1:T-1}) p(q_T | x_{1:T-1}, q_{1:T-1}) \\
p(x_{1:T-1}, q_{1:T-1}) \\
= p(x_T | q_T) p(q_T | q_{T-1}) p(x_{1:T-1}, q_{1:T-1}) \\
= p(q_1) \prod_{t=2}^{T} p(q_t | q_{t-1}) \prod_{t=1}^{T} p(x_t | q_t)
\]

- This allows us to perform very efficient probabilistic inference by distributing sums inside the factors.
- If we start from the left, we get the \( \alpha \) (left-to-right) recursion.
- If we start from the right, we get the \( \beta \) (right-to-left) recursion.

Dynamic Conditional Random Field

\[
p(y|x) = p(y_{1:T} | x_{1:T}) = \frac{1}{Z(x)} \prod_{t=1}^{T} g(x_t, y_t) \prod_{t=2}^{T} h(y_t, y_{t-1}) \tag{1.6}
\]

\[
= \frac{1}{Z(x)} \prod_{t} h(y_t, y_{t-1}) g(x_t, y_t) \tag{1.7}
\]

- This is a conditional model only, not a joint model of \( x, y \).
- We need only that the hidden variables \( y \) factorize, the input \( \bar{x} \) is always observed, so does not contribute to state space.
- Often convex, so relatively easy to optimize (many simple iterative approaches, e.g., perceptron updates or gradient steps).
Stage 4: Models have large search space, can be expensive

- We really have a model over words $W$, word pronunciations $A$, substates of phones $Q$ and then acoustics $X$, and we desire to compute the most probable word sequence, considering all possible ways those words might be generated. I.e., we wish to compute

$$w^* \in \arg\max_w \sum_{a,q} p(w, a, q | \bar{x})$$

(1.8)

- In practice, we often use Viterbi decoding, an approximation to the above

$$(w^*, a^*, q^*) \in \arg\max_{w,a,q} p(w, a, q | \bar{x})$$

(1.9)

and use the resulting $w$ as our hypothesis.

Stage 5: Spoken Dialog Systems

- Imperfect spoken syntax, and processing thereof
- Language Understanding (artificial intelligence)
- Discourse modeling
- Language Generation, speech synthesis, prosody is important
- Synthetic and Flexible vs. Natural and Constrained.
- Some tradeoff in the above is critical for user acceptance.
Where is speech recognition being used today?
Where is speech recognition being used today?

Branches of Speech Processing

- Speech production & Perception
  - Goal: understand how speech is produced in the human vocal system, and how speech is perceived by the human auditory system & brain.

- Speech Synthesis
  - Goal: from simple description (e.g., text) produce a natural sounding voice that someone will both understand and not mind hearing.

- Speech Coding
  - From a human’s voice, “code” and compress it down to its fundamental information bearing element so that it can be transmitted efficiently (e.g., by radio, where bandwidth limitations are severe)

- Speech Recognition
  - From a human’s voice, determine the string of words and other sounds that were said, and represent that in a form amenable to action by computer (e.g., ASCII, or some computer-based representation)

- Dialog systems
  - Build a computer system that is able to have a “conversation” with a human in as natural a way as possible.
  - all draw from many of the subjects, as shown on next page.
Computer Speech Processing

Speech recognition and synthesis require learning about all of the following topics

- Human Anatomy & Physiology
- Phonetics
- Speech Production & Vocal Mechanism
- Physics
- Acoustics
- Human (Auditory) Perception
- Computational Auditory Scene Analysis
- Psychoacoustics
- Signal processing
- Information Theory
- Communications Theory
- Statistical Pattern recognition
- Probability & Statistics
- Multiple Languages and Language Universals
- Machine Learning
- Linguistics
- Text Processing
- Word Pronunciations & Dictionaries
- Human Language Learning
- Natural Language Processing
- Spoken Dialog Understanding
- Cognitive Science
- Artificial Intelligence
- Computer Science & Algorithm Design
- Human Computer Interface Design
- Software Engineering & Computer Architecture
- High-performance & Parallel computing
- Fixed-point DSP micro-processors

Applications of speech processing

- Speech Modification
- Speech Coding
- Speech Enhancement
- Speech Recognition
- Speaker Recognition/Speaker ID (who is speaking?)
- Language Recognition/ID (what language is it?)
- Speech Synthesis
- Spoken Dialog Management
- Spoken Language Processing (IR)
- Spoken Language Analysis (conversations, group discussions)
- Speech coding and communications
- Speech compression
- Speech training, therapy, and language learning
Read: Chapters 1 and 2 in our book.