Introduction to
Statistical Speech Recognition
Lecture 1

Instructor: Preethi Jyothi
Course Plan (I)

• Cascaded ASR System
  - Acoustic Model (AM)
  - Pronunciation Model (PM)
  - Language Model (LM)
• Weighted Finite State Transducers for ASR
  - **AM**: HMMs, DNN and RNN-based models
  - **PM**: Phoneme and Grapheme-based models
  - **LM**: Ngram models (+smoothing), RNNLMs
• Decoding Algorithms, Lattices
Course Plan (II)

- End-to-end Neural Models for ASR
  - CTC loss function
  - Encoder-decoder Architectures with Attention
- Speaker Adaptation
- Speech Synthesis
- Recent Generative Models (GANs, VAEs) for Speech Processing

Check www.cse.iitb.ac.in/~pjyothi/cs753 for latest updates

Moodle will be used for assignment/project-related submissions and all announcements

Image from: Chan et al., Listen, Attend and Spell: A NN for LVCSR, ICASSP 2016
Other Course Info

• Teaching Assistants (TAs):
  - Vinit Unni (vinit AT cse)
  - Saiteja Nalla (saitejan AT cse)
  - Naman Jain (namanjain AT cse)

• TA office hours: Wednesdays, 10 am to 12 pm (tentative)
  Instructor 1-1: Email me to schedule a time

• Readings:
  - No fixed textbook. “Speech and Language Processing” by Jurafsky and Martin serves as a good starting point.
  - All further readings will be posted online.

• Audit requirements: Complete all assignments/quizzes and score ≥ 40%
Course Evaluation

- 3 Assignments OR 2 Assignments + 1 Quiz  35%

  - At least one programming assignment
    - Set up ASR system based on a recipe & improve said recipe

- Midsem Exam + Final Exam  15% + 25%

- Final Project  20%

- Participation  5%

Attendance Policy? Strongly advised to attend lectures. Also, participation points hinges on it.
Academic Integrity Policy
Assignments/Exams

• Always cite your sources (be it images, papers or existing code repos). Follow proper citation guidelines.

• Unless specifically permitted, collaborations are not allowed.

• Do not copy or plagiarise. Will incur significant penalties.
Always cite your sources (be it images, papers or existing code repos). Follow proper citation guidelines.

Unless specifically permitted, collaborations are not allowed.

Do not copy or plagiarise. Will incur significant penalties.
Final Project

• Projects can be on any topic related to speech/audio processing. Check website for abstracts from a previous offering.

• No individual projects and no more than 3 members in a team.

• Preliminary Project Evaluation: Short report detailing project statement, goals, specific tasks and preliminary experiments

• Final Evaluation:
  - Presentation (Oral or poster session, depending on final class strength)
  - Report (Use ML conference style files & provide details about the project)

• Excellent Projects:
  - Will earn extra credit that counts towards the final grade
  - Can be turned into a research paper

SEP 1-7
NOV 7-14
#1: Speech-driven Facial Animation


Videos from: https://sites.google.com/view/facial-animation
#2: Speech2Gesture


Image from: http://people.eecs.berkeley.edu/~shiry/projects/speech2gesture/
#3: Decoding Brain Signals Into Speech
Introduction to ASR
Automatic Speech Recognition

- Problem statement: Transform a spoken utterance into a sequence of tokens (words, syllables, phonemes, characters)

- Many downstream applications of ASR. Examples:
  - Speech understanding
  - Spoken translation
  - Audio information retrieval

- Speech demonstrates variabilities at multiple levels: Speaker style, accents, room acoustics, microphone properties, etc.
History of ASR

RADIO REX (1922)
History of ASR

SHOEBOX (IBM, 1962)

1 word

Freq. detector
History of ASR

- 1922: 1 word Freq. detector
- 1932: 16 words Isolated word recognition
- 1942: 1000 words Connected speech
- 1952: 1000 words Connected speech
- 1962: 1000 words Connected speech
- 1972: 1000 words Connected speech
- 1982: 1000 words Connected speech
- 1992: 1000 words Connected speech
- 2002: 1000 words Connected speech
- 2012: 1000 words Connected speech

Hidden Markov Models (1980s)
History of ASR

DEEP NEURAL NETWORK BASED SYSTEMS (>2010)

1 word
Freq. detector

16 words
Isolated word recognition

1000 words
Connected speech

10K+ words
LVCSR systems

How are ASR systems evaluated?

• Error rates computed on an unseen test set by comparing $W^*$ (decoded sentence) against $W_{ref}$ (reference sentence) for each test utterance
  - Sentence/Utterance error rate (trivial to compute!)
  - Word/Phone error rate

• Word/Phone error rate (ER) uses the Levenshtein distance measure: What are the minimum number of edits (insertions/deletions/substitutions) required to convert $W^*$ to $W_{ref}$?

On a test set with $N$ instances:

$$ER = \frac{\sum_{j=1}^{N} \text{Ins}_j + \text{Del}_j + \text{Sub}_j}{\sum_{j=1}^{N} \ell_j}$$

$\text{Ins}_j, \text{Del}_j, \text{Sub}_j$ are number of insertions/deletions/substitutions in the $j^{th}$ ASR output

$\ell_j$ is the total number of words/phones in the $j^{th}$ reference
Remarkable progress in ASR in the last decade

NIST STT Benchmark Test History – May. ’09

100%
100% WER (in %)

10% WER (in %)

1% WER (in %)

1% Range of Human Error In Transcription


Image from: http://www.itl.nist.gov/iad/mig/publications/ASRhistory/2018
Statistical Speech Recognition

Pioneer of ASR technology, Fred Jelinek (1932 - 2010): Cast ASR as a channel coding problem.

Let $\mathbf{O}$ be a sequence of acoustic features corresponding to a speech signal. That is, $\mathbf{O} = \{O_1, \ldots, O_T\}$, where $O_i \in \mathbb{R}^d$ refers to a d-dimensional acoustic feature vector and $T$ is the length of the sequence.

Let $\mathbf{W}$ denote a word sequence. An ASR decoder solves the foll. problem:

$$W^* = \arg \max_{W} \Pr(W | \mathbf{O})$$

$$= \arg \max_{W} \Pr(\mathbf{O} | W) \Pr(W)$$
Simple example of isolated word ASR

• Task: Recognize utterances which consist of speakers saying either “up" or “down" or “left” or “right” per recording.

• Vocabulary: Four words, “up”, “down”, “left”, “right”

• Data splits
  - Training data: 30 utterances
  - Test data: 20 utterances

• Acoustic model: Let’s parameterize $\Pr_{\theta}(O \mid W)$ using a Markov model with parameters $\theta$. 
Word-based acoustic model

Transition probabilities going from state $i$ to state $j$

Probability of generating $O_i$ from state $j$

Compute $\Pr(O \mid \text{"up"}) = \sum Q \Pr(O, Q \mid \text{"up"})$

Figure 2.1: Standard topology used to represent a phone HMM.

Acoustic model: The most commonly used acoustic models in ASR systems today are Hidden Markov Models (HMMs). Please refer to Rabiner (1989) for a comprehensive tutorial of HMMs and their applicability to ASR in the 1980’s (with ideas that are largely applicable to systems today). HMMs are used to build probabilistic models for linear sequence labeling problems. Since speech is represented in the form of a sequence of acoustic vectors $O$, it lends itself to be naturally modeled using HMMs.

Efficient algorithm exists. Will appear in a later class.
Isolated word recognition

- Pr(O | "up")
- Pr(O | "down")
- Pr(O | "left")
- Pr(O | "right")

Compute \( \arg \max_w \Pr(O | w) \) for a composite acoustic features \( O \).
Small tweak

- Task: Recognize utterances which consist of speakers saying either “up" or “down" multiple times per recording.
Small tweak

- Task: Recognize utterances which consist of speakers saying either “up" or “down" \textbf{multiple times} per recording.
Small vocabulary ASR

- Task: Recognize utterances which consist of speakers saying one of 1000 words **multiple times** per recording.

- Not scalable anymore to use words as speech units

- Model using phones instead of words as individual speech units
  - Phonemes are abstract, subword units that distinguish one word from another (minimal pair; e.g. “pan” vs. “can”)
  - Phones are actually sounds that are realized and not language-specific units

- What's an obvious advantage of using phones over entire words?
  Hint: Think of words with zero coverage in the training data.
Architecture of an ASR system

speech signal

Acoustic Feature Generator

Acoustic Model (phones)

Pronunciation Model

SEARCH

word sequence \( W^* \)

Language Model
Cascaded ASR $\Rightarrow$ End-to-end ASR

Single end-to-end model that directly learns a mapping from speech to text
ASR Progress contd.

Voice Recognition Software Finally Beats Humans At Typing, Study Finds

Microsoft researchers achieve new conversational speech recognition milestone

Amazon’s AI system could cut Alexa speech recognition errors by 15%

What are some unsolved problems related to ASR?

- State-of-the-art ASR systems do not work well on regional accents, dialects.
- Code-switching is hard for ASR systems to deal with.
- How do we rapidly build competitive ASR systems for a new language? Low-resource ASR and keyword spotting.
- How do we recognize speech from meetings where a primary speaker is speaking amidst other speakers?
Next class: HMMs for Acoustic Modeling