Automatic Speech Recognition: Introduction

Peter Bell

Automatic Speech Recognition— ASR Lecture 1
12 January 2026

Automatic Speech Recognition — ASR

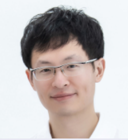
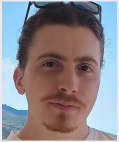
Course details

- **Lectures:** 18 lectures, delivered in person
- **Labs:** Weekly lab sessions – using OpenFst (openfst.org) and Python
  - Lab sessions will start in Week 3
- **Assessment:**
  - First five lab sessions worth **10%**, assessed by the lab demonstrator
  - Coursework, building on the lab sessions, worth **40%**
  - ***Closed*** book exam in April or May worth **50%**

https://opencourse.inf.ed.ac.uk/asr

Course details

- **People:**
  - Course organiser: Peter Bell
  - Assistant lecturer: Hao Tang
  - Guest lecturer: Ondrej Klejch
  - TA: George Karakasidis
  - Demonstrators: Emily Gaughan, Yen Meng

Lectures

18 lectures in total

- 3 lectures delivered by Hao, including: Signal Signal Analysis (lectures 2-3) and Self Supervised Learning for Speech (lecture 17)
- 1 guest lecture delivered by Ondrej on a cutting-edge research topic (lecture date TBC)
- The remaining 14 lectures delivered by me

Labs

- Series of weekly labs using Python, OpenFst and Kaldi
- They count towards 10% of the course credit
- Labs start week 3 – expected to be three lab groups
- You will need to work **in pairs**
- Labs will give you hands-on experience of using HMM algorithms to build your very own ASR system from scratch
    - These labs are an important pre-requisite for the coursework – take advantage of the demonstrator support!

Other teaching support

- Teaching assistant George Karakasidis will help with lab and coursework setup, answering questions online and marking the lab submissions
- We use Piazza, and aim for a quick response time throughout the semester and right up until the exam
- I don't run regular office hours but am happy to meet any students by arrangement at almost any time (individually or in a group)

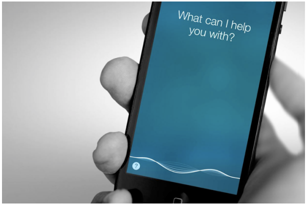
Your background

If you have taken:

- Speech Processing *and* any of (AML, PMR, MLP)
  - Perfect!
- Any of AML, PMR or MLP) *but not* Speech Processing (probably you are from Informatics)
  - You'll require some speech background:
    - A couple of the lectures will cover material that was in Speech Processing, particularly related to signal processing
    - Some additional background study (including material from Speech Processing)
- Speech Processing *but none of* (AML, PMR or MLP) (probably you are from SLP)
  - You'll benefit from gaining some machine learning background (especially neural networks)
    - A couple of introductory lectures on neural networks provided for SLP students
    - Some additional background study might be needed

What is speech recognition?

What is speech recognition?



What is speech recognition?

Speech-to-text transcription

- Transform recorded audio into a sequence of words
- Just the words, no meaning.... But do need to deal with acoustic ambiguity: "Recognise speech?" or "Wreck a nice beach?"

Sometimes also considering...

- Speaker diarization: Who spoke when?
- Speech recognition: what did they say?
- Paralinguistic aspects: how did they say it? (timing, intonation, voice quality)
- Speech understanding: what does it mean?

What we won't cover

What we won't cover

OpenAI

```
pip install git+https://github.com/m-bain/whisperx.git
```

If already installed, update package to most recent commit

```
pip install git+https://github.com/m-bain/whisperx.git --upgrade
```

If wishing to modify this package, clone and install in editable mode:

```
$ git clone https://github.com/m-bain/whisperX.git
$ cd whisperX
$ pip install -e .
```

You may also need to install ffmpeg, rust etc. Follow openAI instructions here https://github.com/openai/whisper#setup.

What we won't cover

OpenAI

Python usage 🐍

```
import whisperx
import gc

device = "cuda"
audio_file = "audio.mp3"
batch_size = 16 # reduce if low on GPU mem
compute_type = "float16" # change to "int8" if low on GPU mem (may reduce accuracy)

# 1. Transcribe with original whisper (batched)
model = whisperx.load_model("large-v2", device, compute_type=compute_type)

# save model to local path (optional)
# model_dir = "/path/"
# model = whisperx.load_model("large-v2", device, compute_type=compute_type, download_root=m

audio = whisperx.load_audio(audio_file)
result = model.transcribe(audio, batch_size=batch_size)
print(result["segments"]) # before alignment
```

Teaching philosophy

- We don't just focus on cutting-edge methods – aim to give you a thorough understanding of how the field developed from the 1980s onwards
- Most lectures focus on the underlying theory, though some are on particular applied topics
- Emphasis on learning by doing, using the labs and coursework
- Course materials are largely self-contained, though the recommended reading will improve your understanding

Why is speech recognition difficult?

From a linguistic perspective

Many sources of variation

Speaker Tuned for a particular speaker, or speaker-independent? Adaptation to speaker characteristics, eg. age, gender, vocal tract length

From a linguistic perspective

Many sources of variation

Speaker Tuned for a particular speaker, or speaker-independent? Adaptation to speaker characteristics, eg. age, gender, vocal tract length

Environment Noise, competing speakers, channel conditions (microphone, phone line, room acoustics)

From a linguistic perspective

Many sources of variation

- **Speaker** Tuned for a particular speaker, or speaker-independent? Adaptation to speaker characteristics, eg. age, gender, vocal tract length
- **Environment** Noise, competing speakers, channel conditions (microphone, phone line, room acoustics)
- **Style** Continuously spoken or isolated? Planned monologue or spontaneous conversation?

From a linguistic perspective

Many sources of variation

- **Speaker** Tuned for a particular speaker, or speaker-independent? Adaptation to speaker characteristics, eg. age, gender, vocal tract length
- **Environment** Noise, competing speakers, channel conditions (microphone, phone line, room acoustics)
- **Style** Continuously spoken or isolated? Planned monologue or spontaneous conversation?
- **Vocabulary** Machine-directed commands, scientific language, colloquial expressions

From a linguistic perspective

Many sources of variation

- **Speaker**: Tuned for a particular speaker, or speaker-independent? Adaptation to speaker characteristics, eg. age, gender, vocal tract length
- **Environment**: Noise, competing speakers, channel conditions (microphone, phone line, room acoustics)
- **Style**: Continuously spoken or isolated? Planned monologue or spontaneous conversation?
- **Vocabulary**: Machine-directed commands, scientific language, colloquial expressions
- **Accent/dialect**: Within-language accent variation, non-native speakers

From a linguistic perspective

Many sources of variation

- **Speaker** Tuned for a particular speaker, or speaker-independent? Adaptation to speaker characteristics, eg. age, gender, vocal tract length
- **Environment** Noise, competing speakers, channel conditions (microphone, phone line, room acoustics)
- **Style** Continuously spoken or isolated? Planned monologue or spontaneous conversation?
- **Vocabulary** Machine-directed commands, scientific language, colloquial expressions
- **Accent/dialect** Within-language accent variation, non-native speakers
- **Other paralinguistic** Emotion, socio-economic background, . . .

From a linguistic perspective

Many sources of variation

- **Speaker** Tuned for a particular speaker, or speaker-independent? Adaptation to speaker characteristics, eg. age, gender, vocal tract length
- **Environment** Noise, competing speakers, channel conditions (microphone, phone line, room acoustics)
- **Style** Continuously spoken or isolated? Planned monologue or spontaneous conversation?
- **Vocabulary** Machine-directed commands, scientific language, colloquial expressions
- **Accent/dialect** Within-language accent variation, non-native speakers
- **Other paralinguistic** Emotion, socio-economic background, . . .
- **Language spoken** Estimated 7,000 languages, most with limited training resources; code-switching; language change

From a machine learning perspective

- As a classification problem: very high dimensional output space

From a machine learning perspective

- As a classification problem: very high dimensional output space
- As a sequence-to-sequence problem: very long input sequence (although limited re-ordering between acoustic and word sequences)

From a machine learning perspective

- As a classification problem: very high dimensional output space
- As a sequence-to-sequence problem: very long input sequence (although limited re-ordering between acoustic and word sequences)
- Data is often noisy, with many "nuisance" factors of variation in the data

From a machine learning perspective

- As a classification problem: very high dimensional output space
- As a sequence-to-sequence problem: very long input sequence (although limited re-ordering between acoustic and word sequences)
- Data is often noisy, with many “nuisance” factors of variation in the data
- Very limited quantities of training data available (in terms of words) compared to text-based NLP
    - Manual speech transcription is very expensive (10x real time)

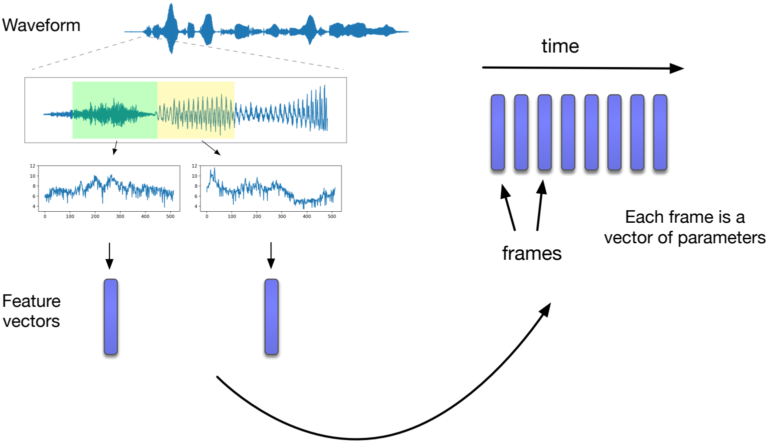
From a machine learning perspective

- As a classification problem: very high dimensional output space
- As a sequence-to-sequence problem: very long input sequence (although limited re-ordering between acoustic and word sequences)
- Data is often noisy, with many "nuisance" factors of variation in the data
- Very limited quantities of training data available (in terms of words) compared to text-based NLP
    - Manual speech transcription is very expensive (10x real time)
- Hierachical and compositional nature of speech production and comprehension makes it difficult to handle with a single model

The speech recognition problem

- We generally represent recorded speech as a sequence of acoustic feature vectors (observations), X and the output word sequence as W
- At recognition time, our aim is to find the most likely W, given X
- To achieve this, statistical models are trained using a corpus of labelled training utterances (X^n, W^n)

Representing recorded speech (X)



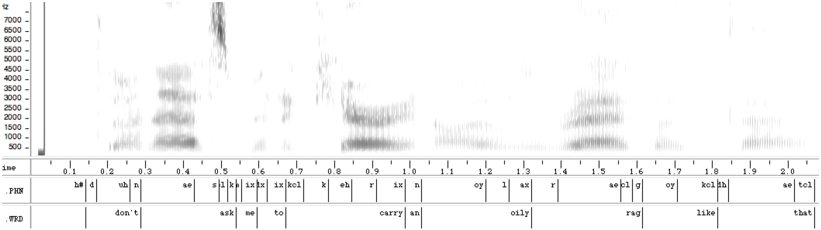
Represent a recorded utterance as a sequence of *feature vectors*

Reading: Jurafsky & Martin section 9.3

Acoustic units

- **Phonemes**
  - abstract unit defined by linguists based on contrastive role in word meanings (eg "pat" vs "bat")
  - 40–50 phonemes in English
- **Phones**
  - speech sounds defined by the acoustics
  - phones may be *allophones* of the same phoneme (eg /p/ in "pit" and "spit")
  - limitless in number
- Possible alternatives: syllables, characters ("graphemes"), automatically derived units, ...

(Slide taken from Martin Cooke from long ago)



Labels may be at different levels: words, phones, sentences, etc.

Labels may or may not be *time-aligned* – do we know the start and end times of an acoustic segment corresponding to a label?

Reading: Jurafsky & Martin chapter 7 (especially sections 7.4, 7.5)

Two machine learning challenges

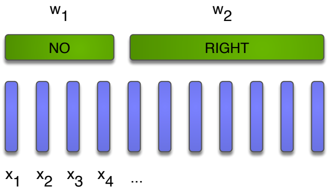
In **training** the model:

Aligning the sequences X^n and W^n for each training utterance

Two machine learning challenges

In **training** the model:

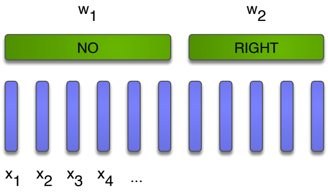
Aligning the sequences X^n and W^n for each training utterance



Two machine learning challenges

In **training** the model:

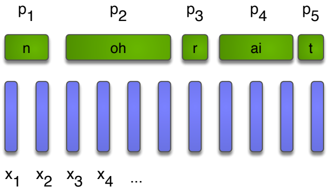
Aligning the sequences X^n and W^n for each training utterance



Two machine learning challenges

In **training** the model:

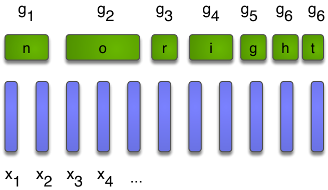
Aligning the sequences X^n and W^n for each training utterance



Two machine learning challenges

In **training** the model:

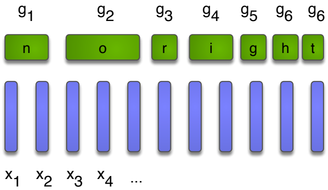
Aligning the sequences X^n and W^n for each training utterance



Two machine learning challenges

In **training** the model:

Aligning the sequences X^n and W^n for each training utterance



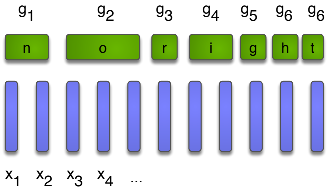
In **performing recognition**:

Searching over all possible output sequences W
to find the most likely one

Two machine learning challenges

In **training** the model:

Aligning the sequences X^n and W^n for each training utterance

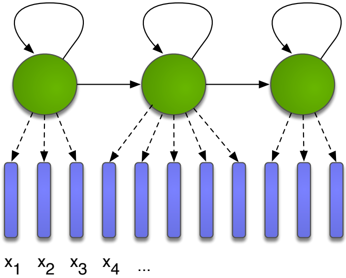


In **performing recognition**:

Searching over all possible output sequences W
to find the most likely one

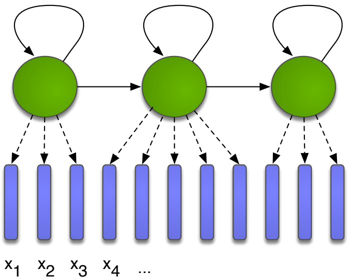
The **hidden Markov model** (HMM) provides a good solution to both problems

The Hidden Markov Model



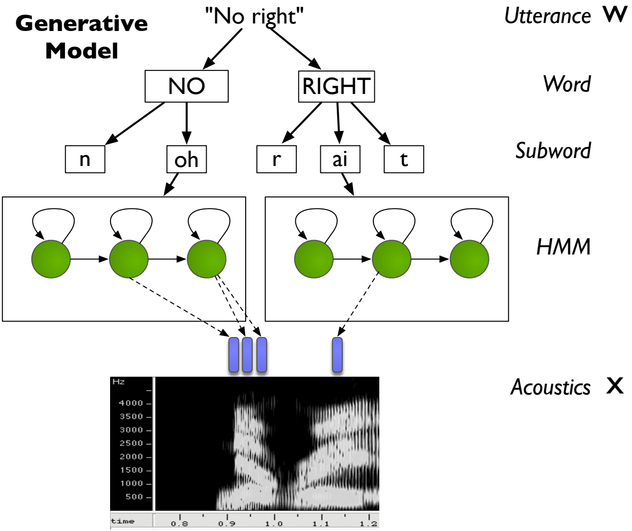
- A simple but powerful model for mapping a sequence of continuous observations to a sequence of discrete outputs
- It is a **generative** model for the observation sequence – also a **noisy channel** model
- Algorithms for training (forward-backward) and recognition-time decoding (Viterbi)

The Hidden Markov Model



- A simple but powerful model for mapping a sequence of continuous observations to a sequence of discrete outputs
- It is a **generative** model for the observation sequence – also a **noisy channel** model
- Algorithms for training (forward-backward) and recognition-time decoding (Viterbi)
- Later in the course we will also look at newer all-neural, fully-differentiable "end-to-end" models

Hierarchical modelling of speech



"Fundamental Equation of Statistical Speech Recognition"

If X is the sequence of acoustic feature vectors (observations) and W denotes a word sequence, the most likely word sequence W^* is given by

$$W^* = \arg\max_{W} P(W \mid X)$$

"Fundamental Equation of Statistical Speech Recognition"

If X is the sequence of acoustic feature vectors (observations) and W denotes a word sequence, the most likely word sequence W^* is given by

$$W^* = \arg\max_{W} P(W \mid X)$$

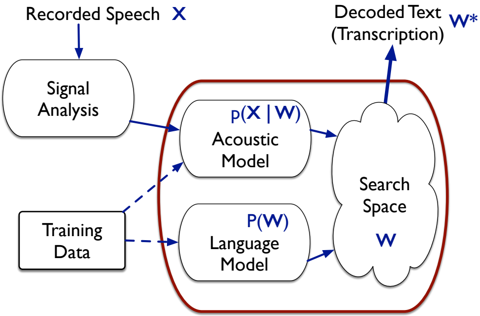
Applying Bayes' Theorem:

$$\begin{aligned} P(W \mid X) &= \frac{p(X \mid W) P(W)}{p(X)} \\ &\propto p(X \mid W) P(W) \\ W^* &= \arg\max_{W} \underbrace{p(X \mid W)}_{\text{Acoustic model}} \quad \underbrace{P(W)}_{\text{Language model}} \end{aligned}$$

Speech Recognition Components

$$W^* = \arg\max_{W} p(X \mid W) P(W)$$

Use an acoustic model, language model, and lexicon to obtain the most probable word sequence W^* given the observed acoustics X



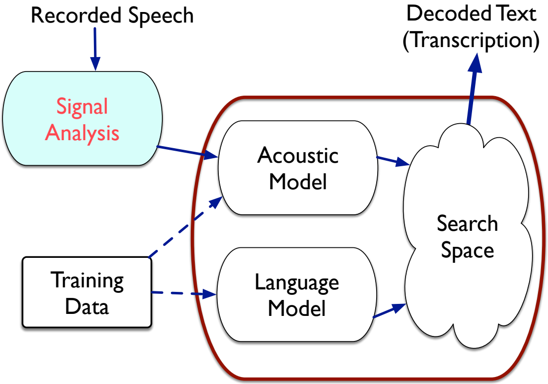
Evaluation

- How accurate is a speech recognizer?
- String edit distance
  - Use dynamic programming to align the ASR output with a reference transcription
  - Three type of error: insertion, deletion, substitutions
- Word error rate (WER) sums the three types of error. If there are N words in the reference transcript, and the ASR output has S substitutions, D deletions and I insertions, then:

$$\text{WER} = 100 \cdot \frac{S + D + I}{N} \% \qquad \text{Accuracy} = 100 - \text{WER}\%$$

- Speech recognition evaluations: common training and development data, release of new test sets on which different systems may be evaluated using word error rate

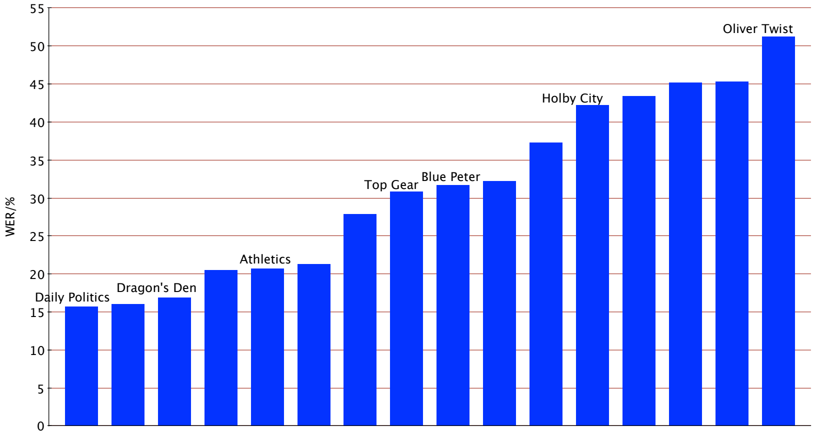
Next Lecture



Example 1: recognising TV broadcasts (2015)

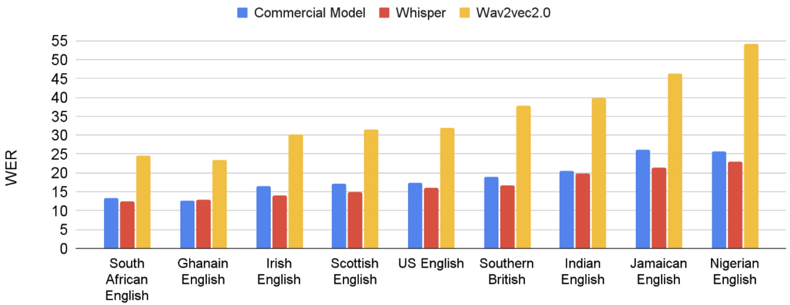
MGB CHALLENGE

Example 1: recognising TV broadcasts (2015)



Example 2: recognising conversations (2023)

EDACC
The Edinburgh International
• Accents of English Corpus •



Reading

- Jurafsky and Martin (2008). *Speech and Language Processing* (2nd ed.): Chapter 7 (esp 7.4, 7.5) and Section 9.3.
- General interest:
  - *The Economist Technology Quarterly*, "Language: Finding a Voice", Jan 2017.
    http://www.economist.com/technology-quarterly/2017-05-01/language
  - *The State of Automatic Speech Recognition: Q&A with Kaldi's Dan Povey*, Jul 2018.
    https://medium.com/descript/the-state-of-automatic-speech-recognition-q-a-with-kaldis-dan-povey-c860aada9b85