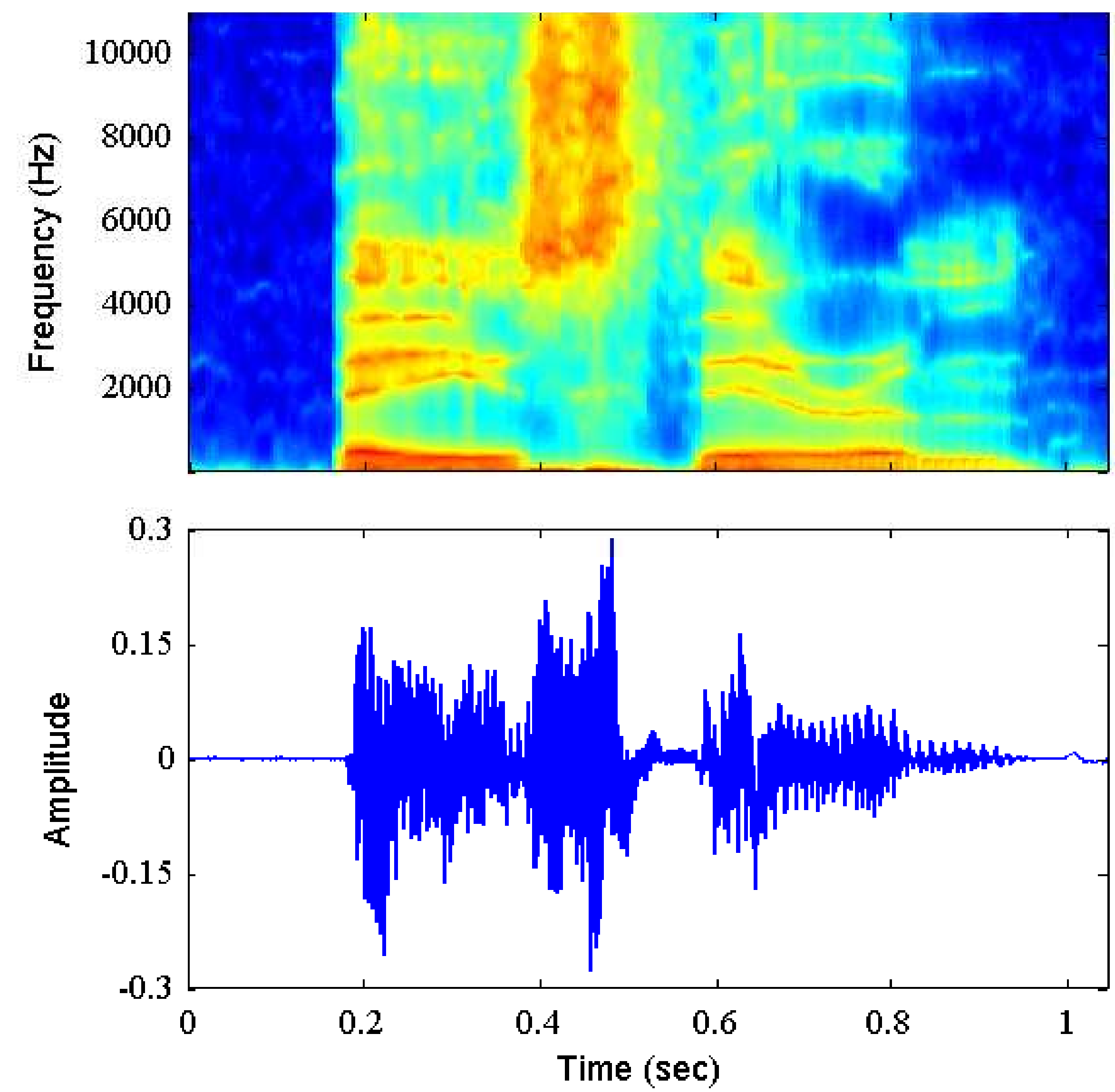
# EECS 391
# Intro to AI

## Models for Sequential Data

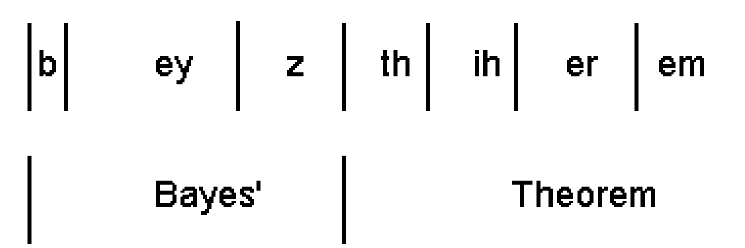
L22 Thu Nov 30

# Examples of sequential data

- speech spectrogram
  - sequences of power spectra
  - typically 30 ms with 10 ms overlap
- audio waveform: raw sample values




- sequences of syllables and words



# Speech Sounds

# Speech sounds: Phones

## Phone Models

Frame features in $P(features|phone)$ summarized by

– an integer in $[0 \ldots 255]$ (using vector quantization); or

– the parameters of a mixture of Gaussians

Three-state phones: each phone has three phases (Onset, Mid, End)

E.g., [t] has silent Onset, explosive Mid, hissing End

$\Rightarrow P(features|phone, phase)$

Triphone context: each phone becomes $n^2$ distinct phones, depending on the phones to its left and right

E.g., [t] in "star" is written [t(s,aa)] (different from "tar"!)

Triphones useful for handling coarticulation effects: the articulators have inertia and cannot switch instantaneously between positions

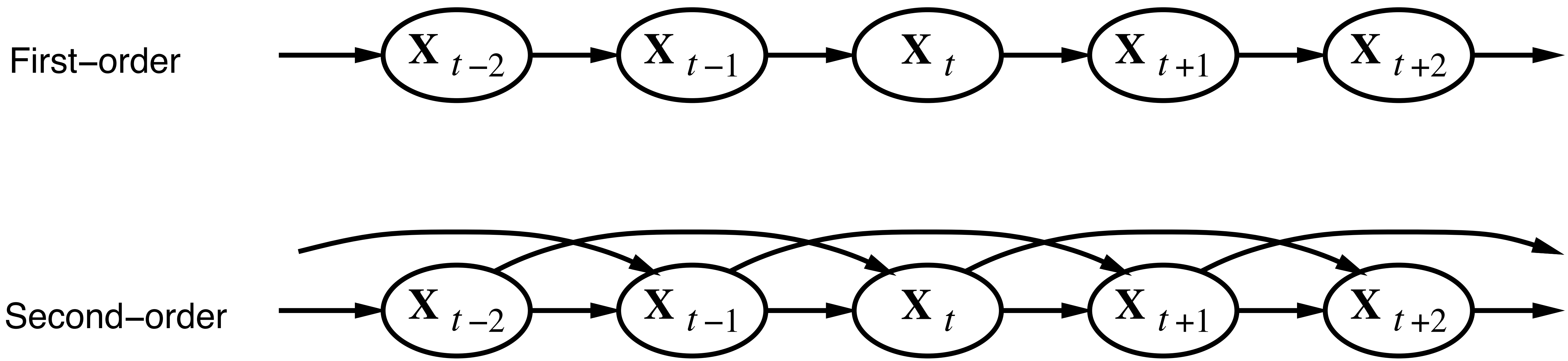
E.g., [t] in "eighth" has tongue against front teeth

# Markov processes (Markov chains)

Construct a Bayes net from these variables: parents?

Markov assumption: $\mathbf{X}_t$ depends on **bounded** subset of $\mathbf{X}_{0:t-1}$

First-order Markov process: $\mathbf{P}(\mathbf{X}_t|\mathbf{X}_{0:t-1}) = \mathbf{P}(\mathbf{X}_t|\mathbf{X}_{t-1})$
Second-order Markov process: $\mathbf{P}(\mathbf{X}_t|\mathbf{X}_{0:t-1}) = \mathbf{P}(\mathbf{X}_t|\mathbf{X}_{t-2}, \mathbf{X}_{t-1})$



*(notes on board)*

# Markov model examples

# Markov model examples

# Markov model examples

# Markov model examples

# Markov model examples

# Markov model examples: word sequences

6. *First-order word model.* (The words are chosen independently but with frequencies as in English.)

REPRESENTING AND SPEEDILY IS AN GOOD APT OR COME CAN DIFFERENT NATURAL HERE HE THE A IN CAME THE TO OF TO EXPERT GRAY COME TO FURNISHES THE LINE MESSAGE HAD BE THESE.

# Markov model examples: word sequences

6. *First-order word model.* (The words are chosen independently but with frequencies as in English.)

   REPRESENTING AND SPEEDILY IS AN GOOD APT OR COME

   CAN DIFFERENT NATURAL HERE HE THE A IN CAME THE TO

   OF TO EXPERT GRAY COME TO FURNISHES THE LINE

   MESSAGE HAD BE THESE.

7. *Second-order word model.* (The word transition probabilities match English text.)

   THE HEAD AND IN FRONTAL ATTACK ON AN ENGLISH

   WRITER THAT THE CHARACTER OF THIS POINT IS

   THEREFORE ANOTHER METHOD FOR THE LETTERS THAT THE

   TIME OF WHO EVER TOLD THE PROBLEM FOR AN

   UNEXPECTED

# Simple language model

Prior probability of a word sequence is given by chain rule:

$$P(w_1 \cdots w_n) = \prod_{i=1}^{n} P(w_i | w_1 \cdots w_{i-1})$$

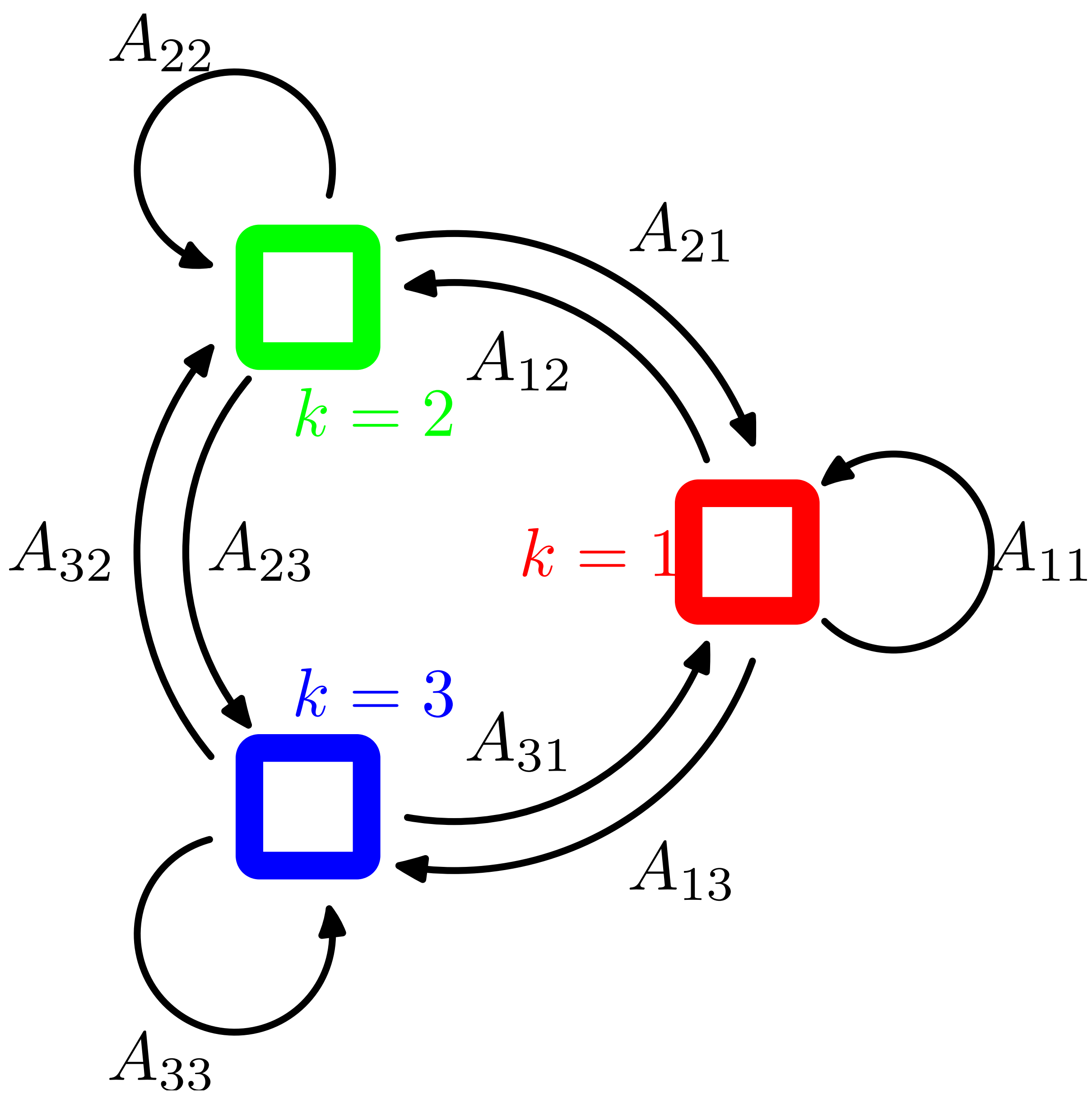
Bigram model:

$$P(w_i | w_1 \cdots w_{i-1}) \approx P(w_i | w_{i-1})$$

Train by counting all word pairs in a large text corpus

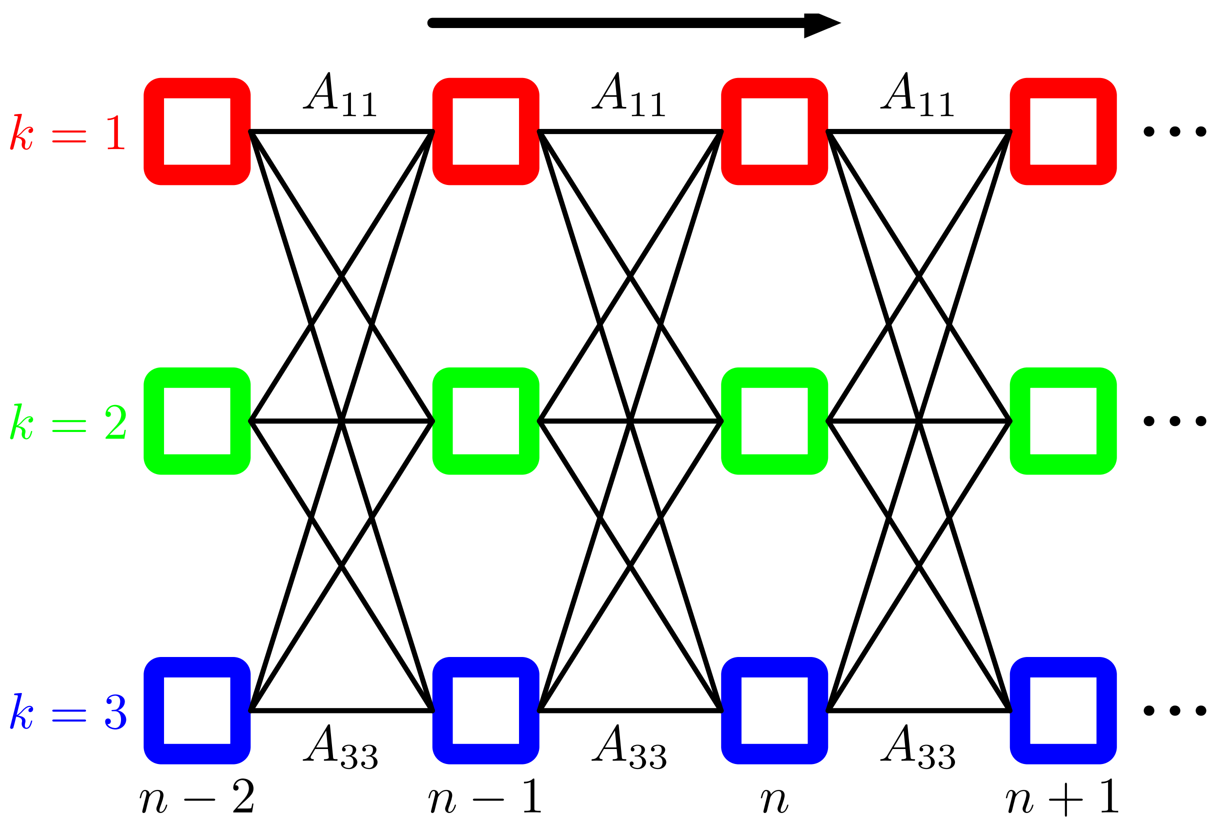
More sophisticated models (trigrams, grammars, etc.) help a little bit

# Transition diagram



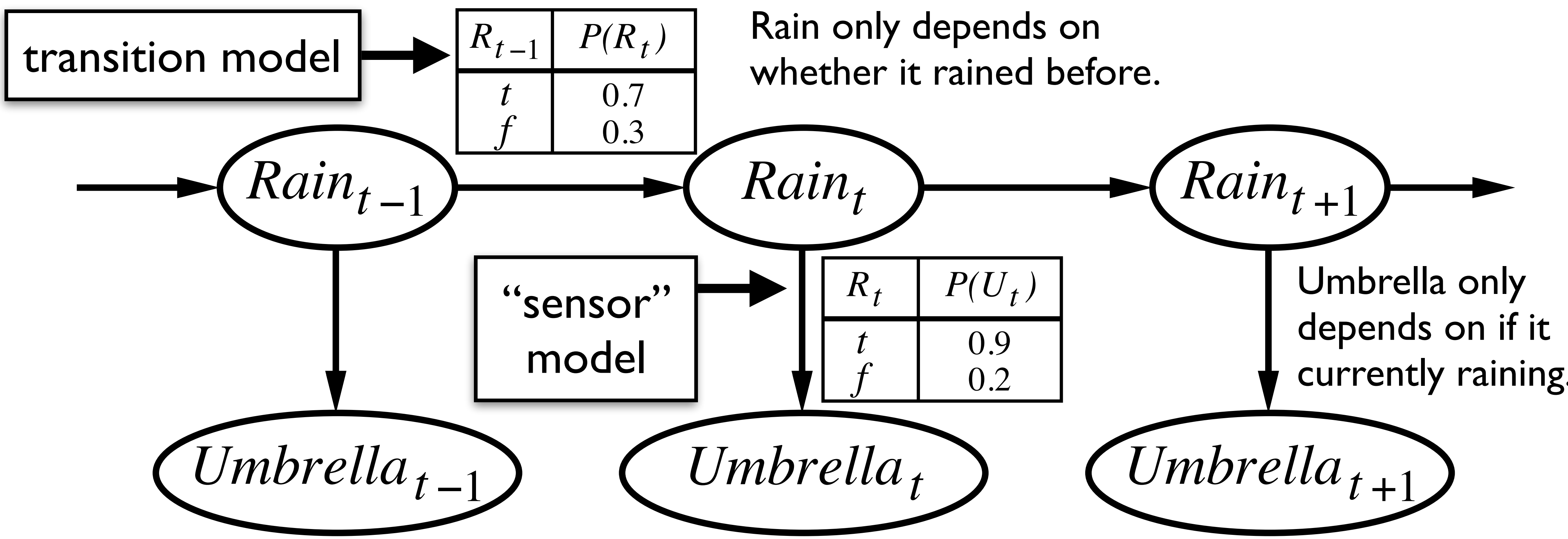

- 3-state model
- $A_{ij}$ are transition probabilities
- Note: this is not a graphical model
  - nodes are not separate variables
  - they're states of a single variable

# Alternative transition diagram




- Unfold the 3-state model overtime
- $A_{ij}$ are transition probabilities
- Still not a graphical model.
- Nodes are states of a single variable, $\mathbf{z}_n$.

# Example




First-order Markov assumption not exactly true in real world!

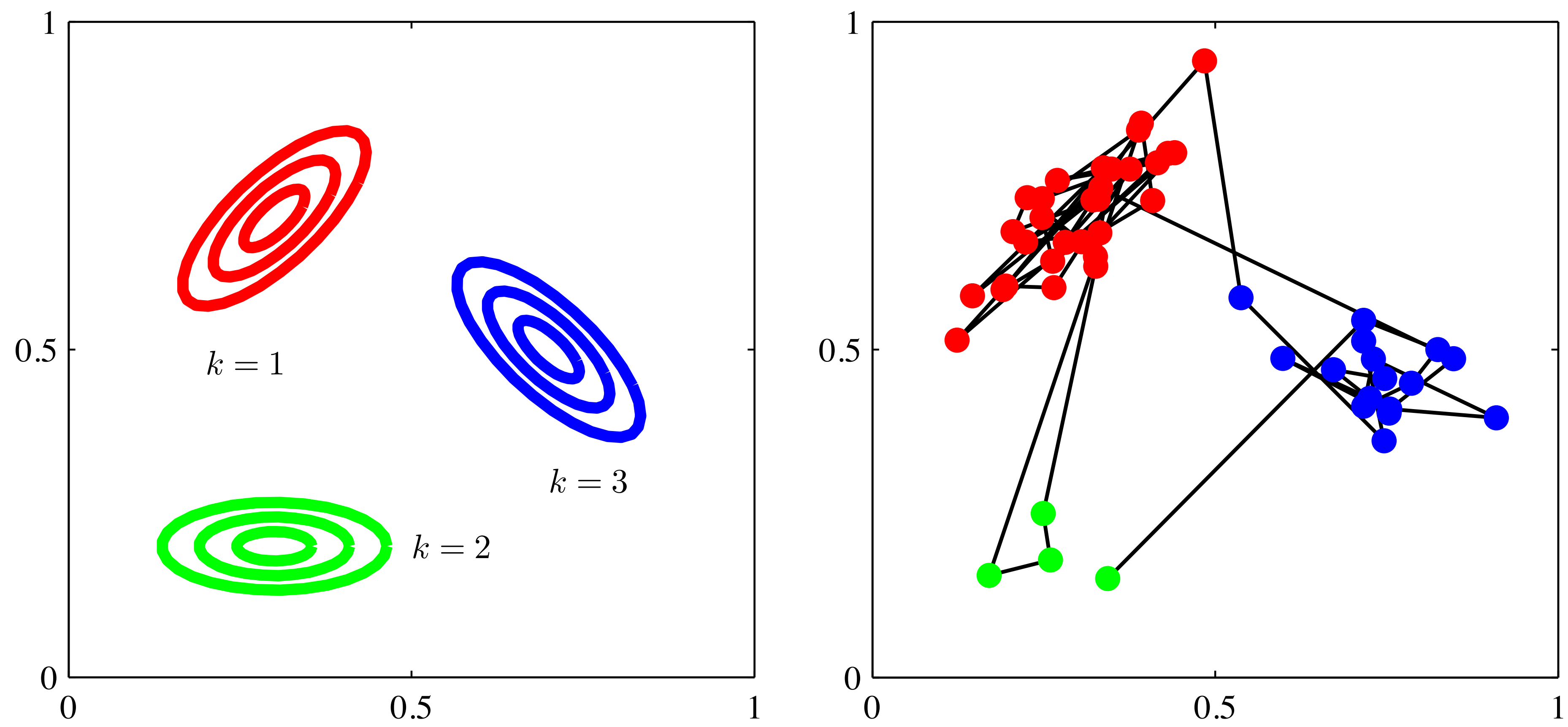
Possible fixes:

1. **Increase order** of Markov process
2. **Augment state**, e.g., add $Temp_t$, $Pressure_t$

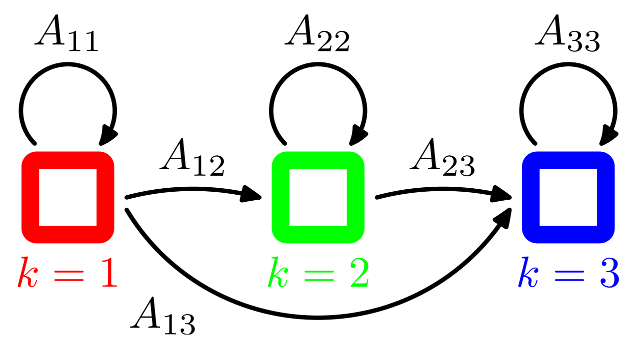
Example: robot motion.

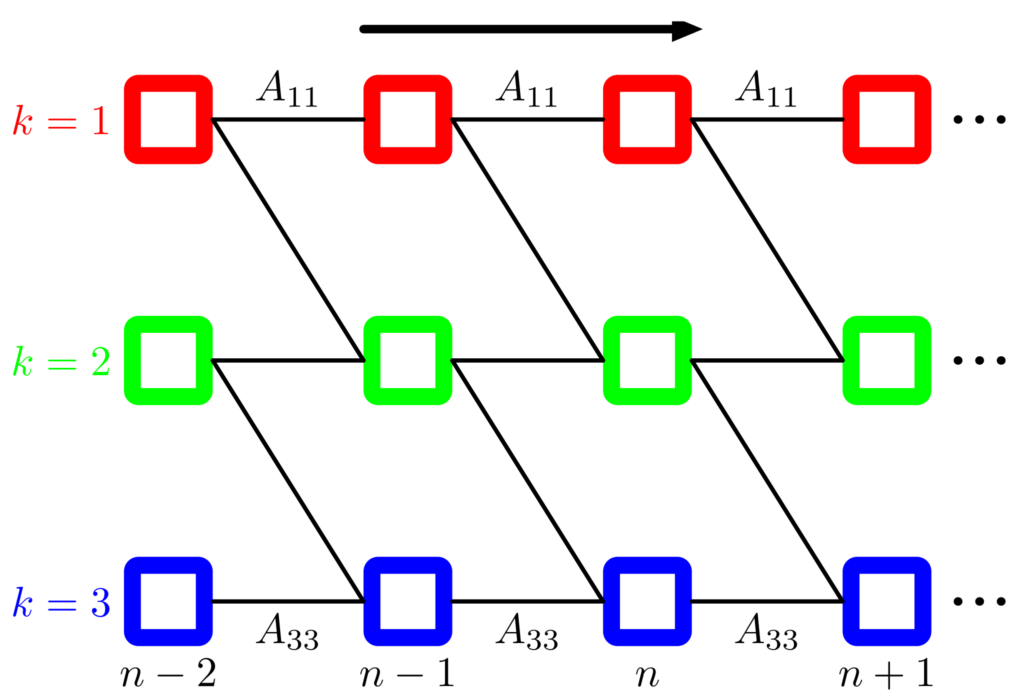
Augment position and velocity with $Battery_t$

# A 3-state Gaussian mixture *emission* model



# A left-to-right HMM




- can't go back to previous state
- $A_{jk} = 0$ if $k < j$

# A phone HMM model

- Frame features in p(features|phone), eg:
  - integer in [0...255] using vector quant.
  - parameters of a mixture of Gaussians
- Three-state-phones: Onset, Mid, End
  - eg: [t] silent onset, explosive mid, hissing end: p(features|phone,phase)
- Triphone context: each phone becomes $n^2$ distinct phones, depending on phones before and after
  - eg: [t] in "star" is [t(s,aa)], different from [t] in "tar"
- Triphones handle *coarticulation* effects

# Word pronunciation models

# Monaural speech recognition challenge

# Monaural speech recognition challenge

# Monaural speech recognition challenge

# Human subjects compared to automated speech recognition

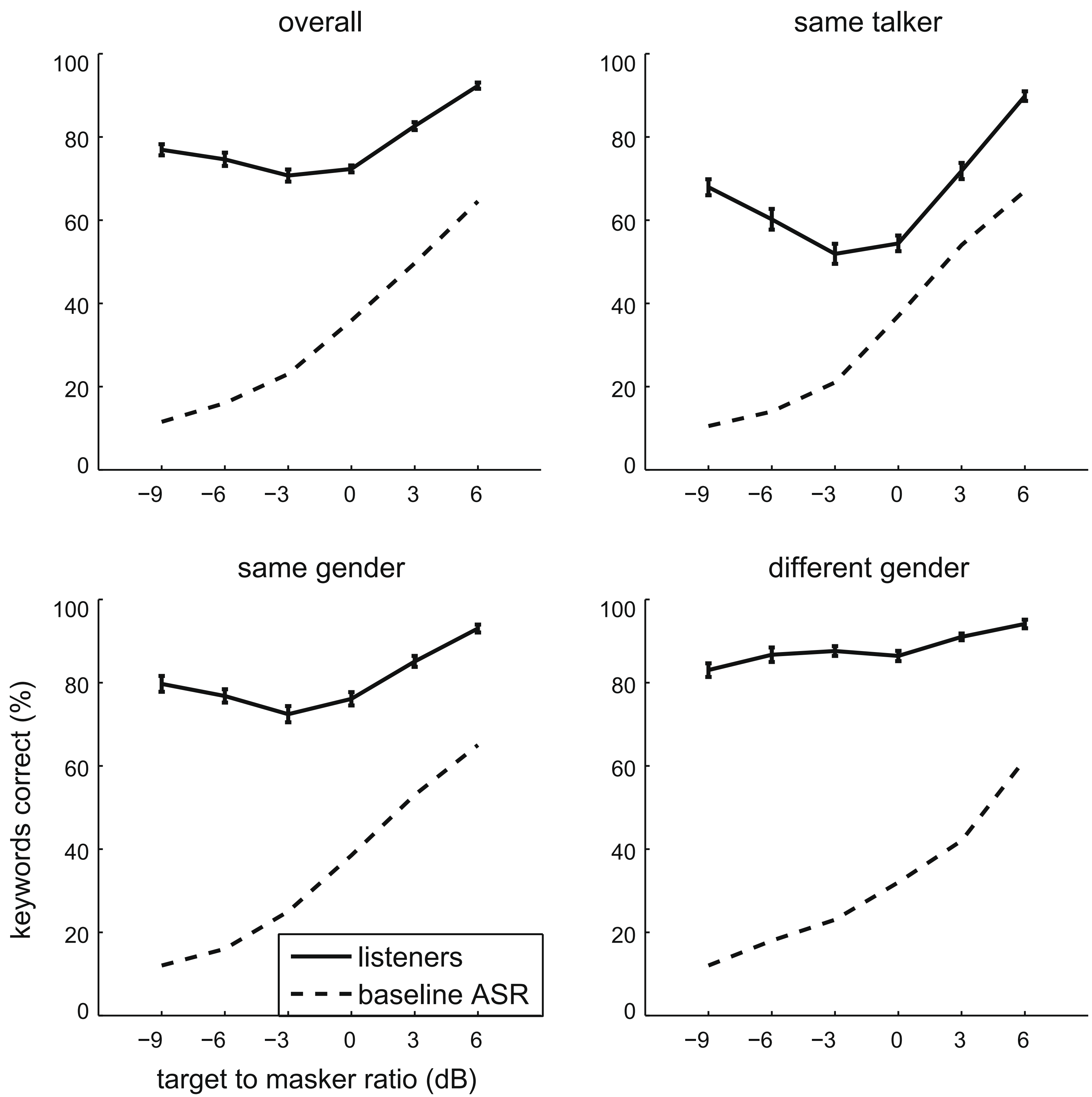


Table 1
Overall word error rate (%) of the ASR systems that were entered into the Pascal 2006 speech separation challenge.

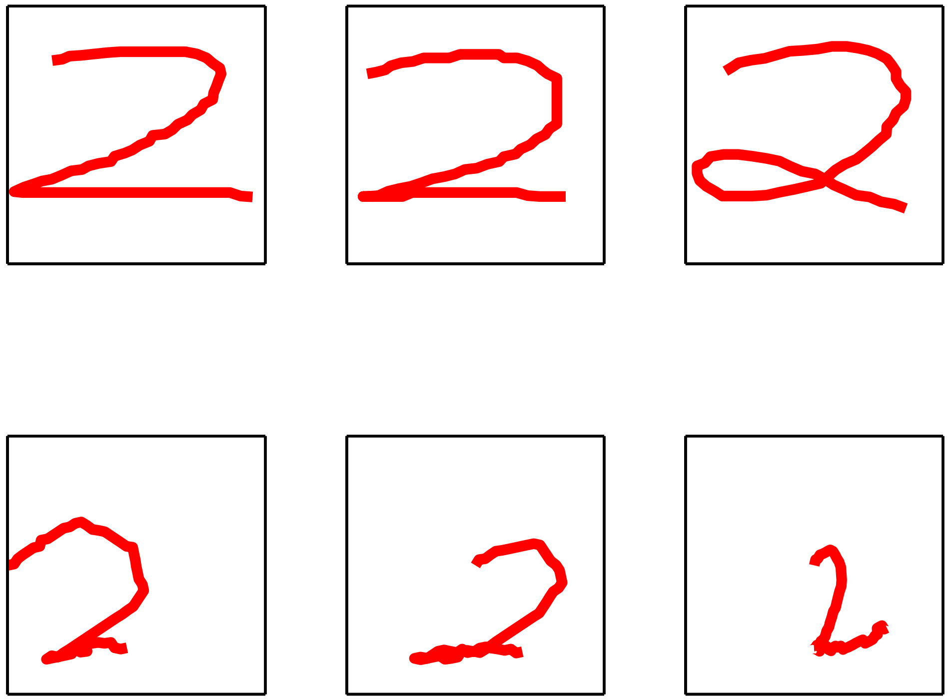
| System | Approach | Accuracy (%) |
|---|---|---|
| Hershey et al. (2010) | Model-based, joint decoding | 78.4 |
| Human listeners | Listening | 77.7 |
| Virtanen (2006)[a] | Model-based, alternating decoding | 65.8 |
| Barker et al. (2010) | CASA, missing features | 63.8 |
| Ming et al. (2010) | Model-based, missing features | 58.4 |
| Schmidt and Olsson (2006)[a] | Non-neg. matrix factorization | 50.2 |
| Weiss and Ellis (2010) | Model-based, joint decoding | 48.0 |
| Li et al. (2010) | Model-based, reconstruction | 47.7 |
| Shao et al. (2010) | CASA, missing features | 45.5 |
| Baseline recognizer | HTK recognizer, no enhancement | 33.4 |
| Deshmukh and Espy-Wilson (2006)[a] | Phase opponency enhancement | 31.6 |
| Every and Jackson (2006)[a] | Pitch-based enhancement | 23.3 |
| Chance | Guessing | 7.0 |

# Hershey et al (2010) from IBM

- estimate speaker identities and gains of both talkers
- separation system combines task grammar with speaker-dependent acoustic models, and acoustic interaction model
- gives two sets of sources, speaker A is target or speaker B is target
- recognize each signal using speaker-dependent labeling

# Task grammar model and acoustic source models

# Hand written digit model



- real examples
- synthetic examples generated from left-to-right HMM trained on 45 examples

# HMM Inference and Learning problems