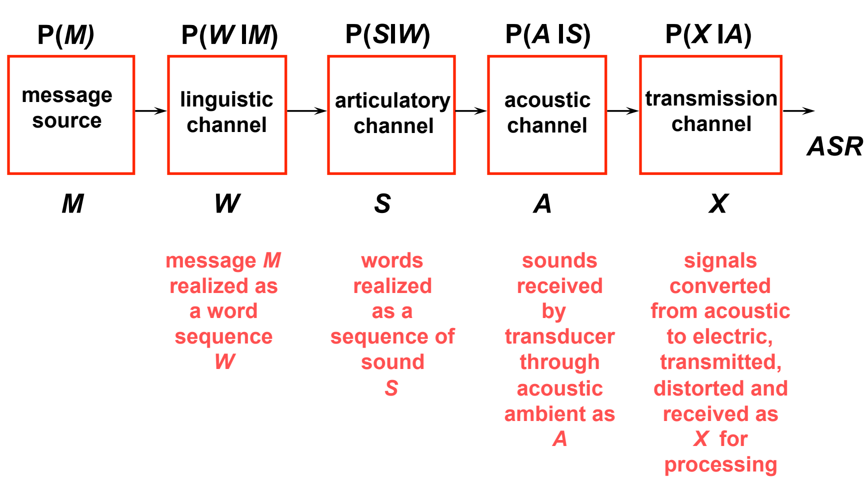
CSE6328.3
Speech & Language Processing

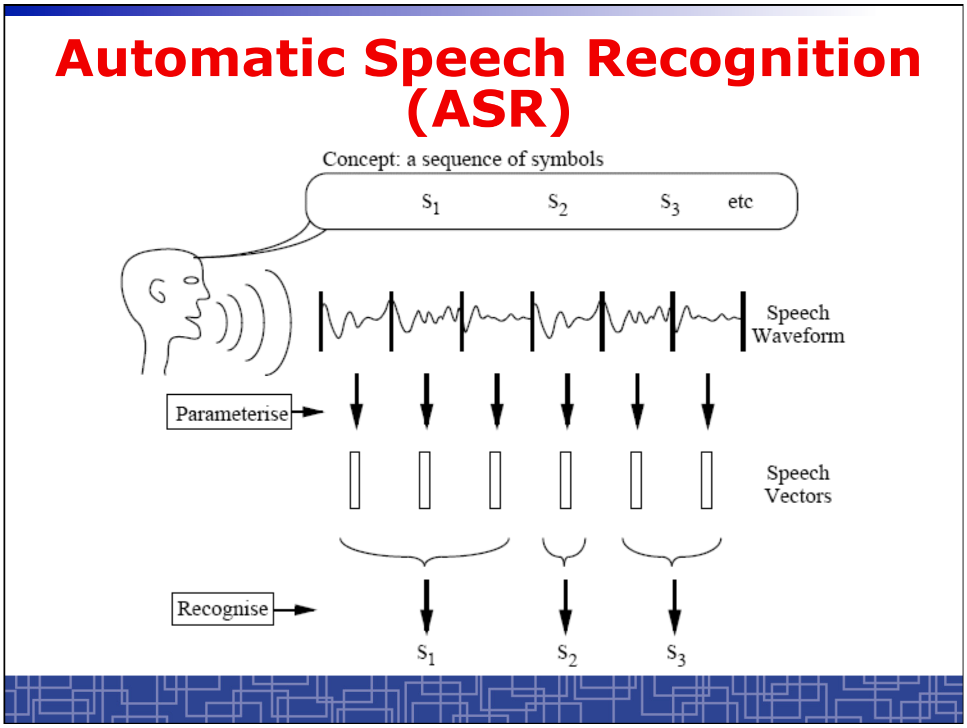
No.7

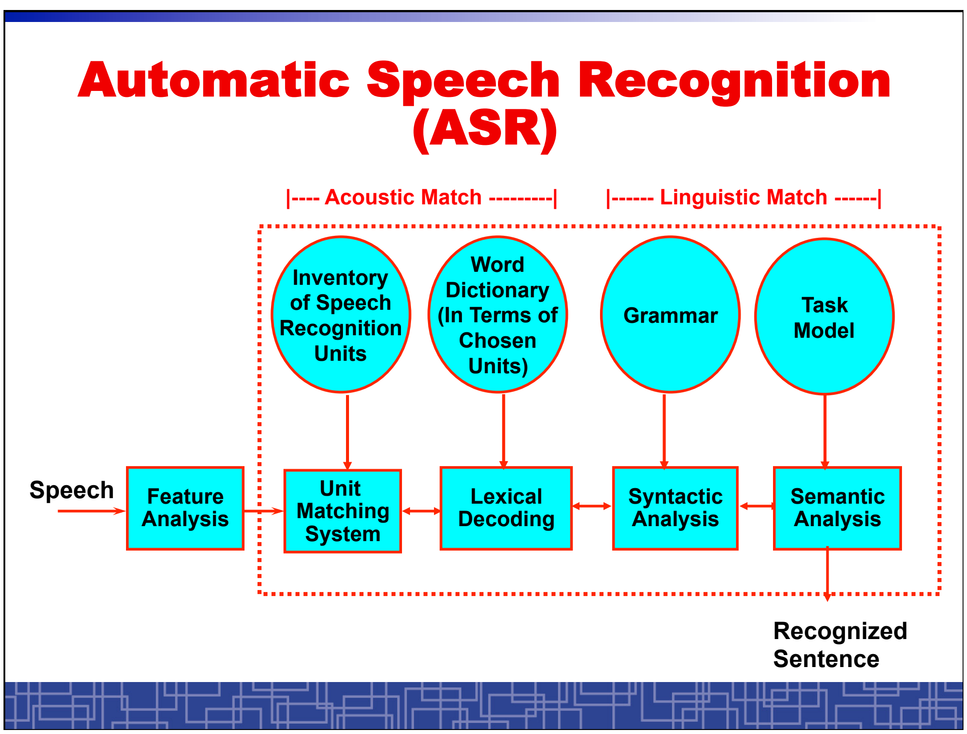
Automatic Speech Recognition(I):
Introduction & Acoustic Modeling

Prof. Hui Jiang
Department of Computer Science and Engineering
York University

Message Generation and ASR - Information Theoretic View




Automatic Speech Recognition (ASR)
Concept: a sequence of symbols
S1 S2 S3 etc
Speech Waveform
Parameterise
Speech Vectors
Recognise
S1 S2 S3


Automatic Speech Recognition (ASR)
|---- Acoustic Match ---------|
|------ Linguistic Match ------|
Inventory of Speech Recognition Units
Word Dictionary (In Terms of Chosen Units)
Grammar
Task Model
Speech
Feature Analysis
Unit Matching System
Lexical Decoding
Syntactic Analysis
Semantic Analysis
Recognized Sentence

ASR System Components

- Feature Extraction
  - framing and short-time spectral/cepstral analysis
- Acoustic Modeling of Speech Units
  - fundamental speech unit selection
  - statistical pattern matching (HMM unit) modeling
- Lexical Modeling
  - pronunciation network
- Syntactic and Semantic Modeling
  - deterministic or stochastic finite state grammar
  - N-gram language model
- Search and Decision Strategies
  - best-first or depth-first, DP-based (or breadth-first) search
  - modular vs. integrated decision strategies

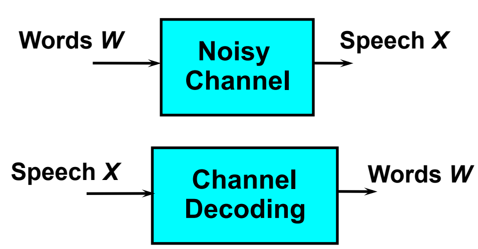
ASR Terminology

- Vocabulary (Lexicon)
  - words that can be recognized in an application
  - More words imply more errors and more computation
- Grammars
  - syntax (word order) that can be used
  - the way words are put together to form phrases & sentences, some are more likely than others
  - can be deterministic or stochastic
- Semantics
  - usually not properly modeled or represented
- Keyword Spotting
  - listening for a few specific words within an utterance
  - Phrase Screening (Rejection): capability to decide whether a candidate keyword is a close enough match to be declared a valid keyword

Types Of ASR Systems (Technology Dimensions)

- Isolated vs. continuous ASR
  - Isolated = pauses required between each word
  - Continuous = no pauses required
- Small vs. medium vs. large vocabulary
- Speech unit selection: whole vs. sub-word (phone, syllable, etc.)
  - Whole word modeling: each HMM → one word
    - requires data collection of all words to be recognized;
    - hard to share data among words; hard to add new words
  - Sub-word modeling: each HMM → phoneme/syllable
    - Solves all the above problems;
    - BUT poor to model coarticulation → use context-dependent sub-word models: e.g., bi-phone, tri-phone, etc.
- Read vs. spontaneous (degree of fluency)
- Multilingual and dialect/accent variations

ASR Formulation



- ASR can be viewed as a (noisy) channel decoding or pattern classification problem.
- The solution to ASR (the plug-in MAP decision rule):

$$\hat{W} = \arg\max_{W\in\Gamma} p(W \mid X) = \arg\max_{W\in\Gamma} P(W)\cdot p(X \mid W)$$

$$= \arg\max_{W\in\Gamma} \overline{P}_{\Gamma}(W)\cdot \overline{p}_{\Lambda}(X \mid W)$$

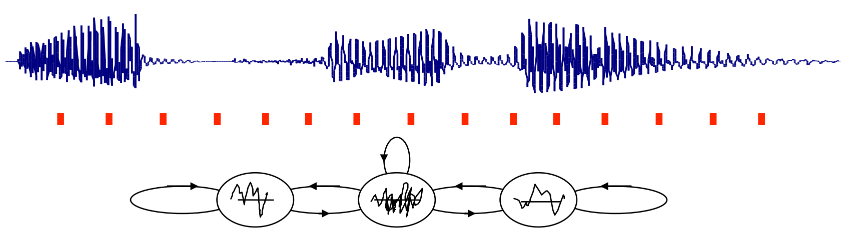
ASR Solution

$$\hat{W} = \arg\max_{W \in \Omega} p(W \mid X) = \arg\max_{W \in \Omega} P(W) \cdot p(X \mid W)$$

$$= \arg\max_{W \in \Omega} \bar{P}_{\Gamma}(W) \cdot \bar{p}_{\Lambda}(X \mid W)$$

- $\bar{p}_{\Lambda}(X \mid W)$ — ***Acoustic Model (AM)***: gives the probability of generating feature ***X*** when ***W*** is uttered.
  - Need a model for every ***W*** to model all speech signals (features) from ***W*** → HMM is an ideal model for speech
  - Speech unit selection: what speech unit is modeled by each HMM? (phoneme, syllable, word, phrase, sentence, etc.)
    - Sub-word unit is more flexible (better)
- $\bar{P}_{\Gamma}(W)$ — ***Language Model (LM)***: gives the probability of ***W*** (word, phrase, sentence) is chosen to say.
  - Need a flexible model to calculate the probability for all kinds of ***W*** → Markov Chain model (n-gram)
- Search space Ω

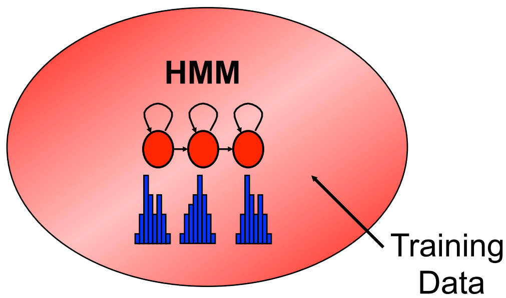
HMM: an ideal speech model

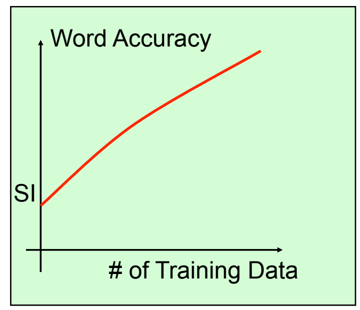


- Variations in speech signals: temporal & spectral
- Each state represents a process of measurable observations.
- Inter-process transition is governed by a finite state Markov chain.
- Processes are stochastic and individual observations do not immediately identify the hidden state.

HMM models spectral and temporal variations simultaneously

Acoustic Modeling of Speech Units and System Performance



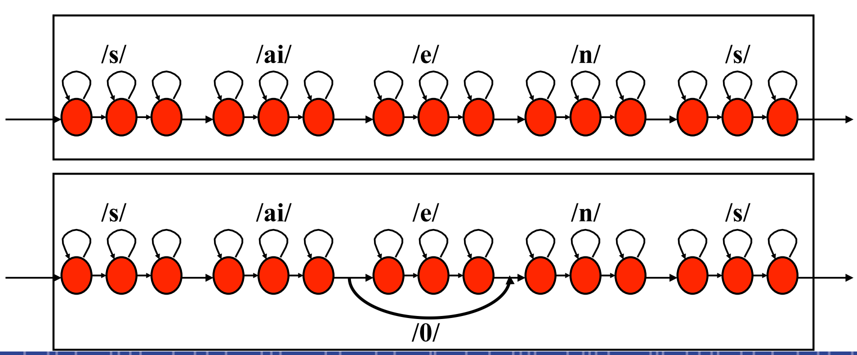


In a typical system, each phoneme in the language is modeled by a 3-state left-to-right continuous density Gaussian mixture HMM (CDHMM), and background noise is modeled by a 1-state CDHMM

Up to thousand of hours of speech data have been used to train HMM's

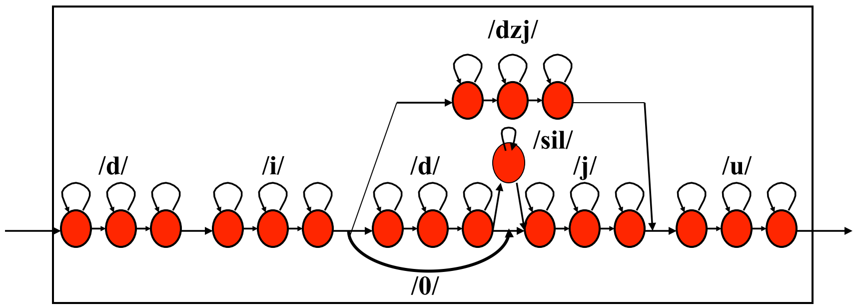
Lexical Modeling

- **Assume each HMM → a monophone model (context-independent)**
  - **American English: 42 monophone → 42 distinct HMMs**
  - **concatenation of phone models (phone HMM's)**
  - **Lexicon: /science/ = /s/+/ai/+/e/+/n/+/s/ or /s/+/ai/+/n/+/s/**
  - **multiple pronunciations and pronunciation network**



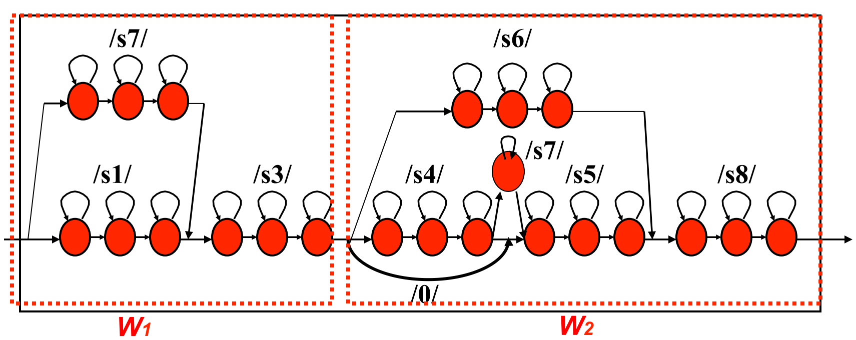
Word-Juncture Modeling

- Co-articulation effect
  - soft change:
    - simple concatenation of word models (word HMM's)
    - possible pronunciation variations
  - hard change: "did you" = /d/+/i/+/dzj/+/u/
  - source of major errors in many ASR systems
  - easier to handle in syllabic languages with open syllables (vowel or nasal endings, e.g. Japanese, Mandarin, Italian)



From Words to Word Sequences

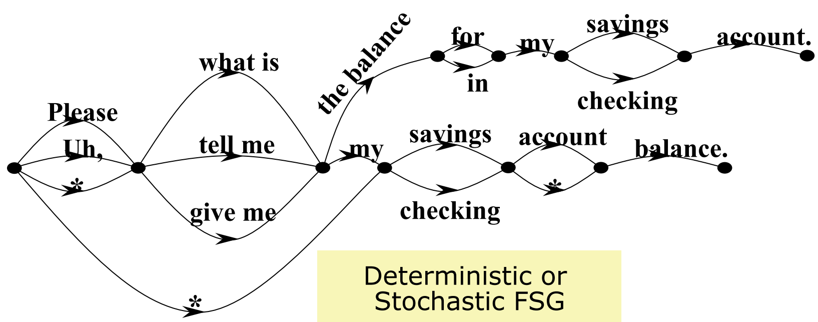
- word → word sequence → beyond



- Syntax Model (Grammar Network): a huge HMM network (a huge composite HMM) to represent all possible and valid word sequences
  - Finite state approximation of word constraints
  - Deterministic or stochastic finite state grammar
  - Large word network for large ASR problems (e.g. |V|=60K)

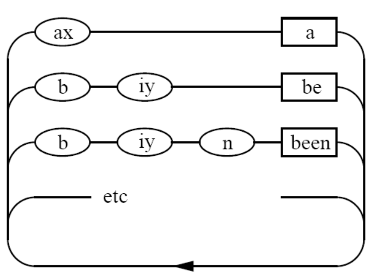
A Finite-State Grammar Example

- Finite-state grammar for a simple account query task:
  - Each arc represents a word or phrase except those marked "*" which allow parts of the phrase to be bypassed.
  - This grammar allows phrases such as "Please tell me my checking account balance."



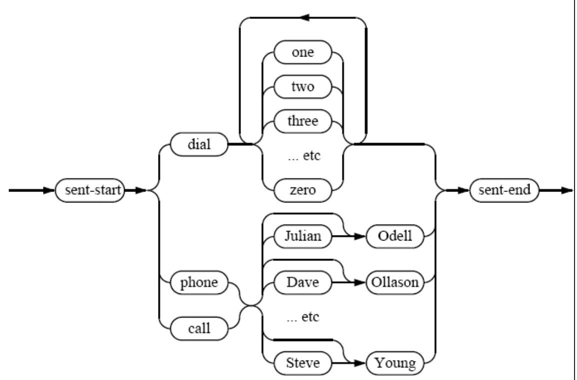
Deterministic or Stochastic FSG

Other examples of Grammar Network



Word-loop grammar:

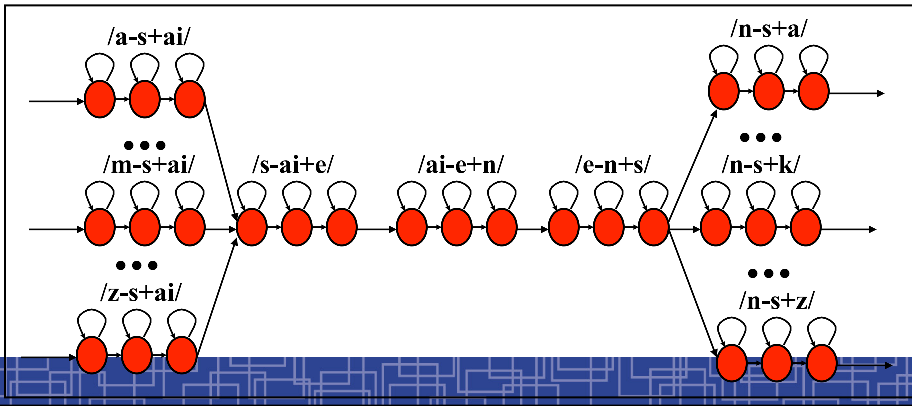
- For all possible sentences.
- Each branch represents a word in vocabulary
- May add transition probabilities from language models



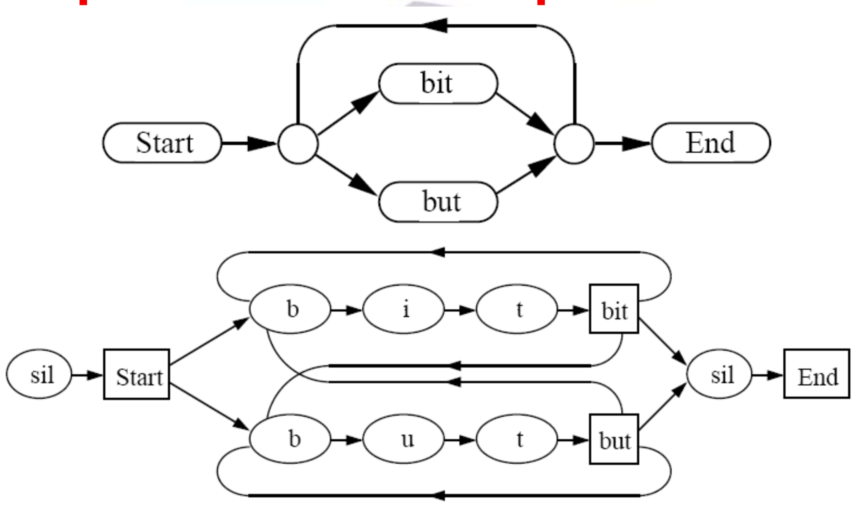
Grammar for Voice Dialing

Modeling Triphone (Biphone)

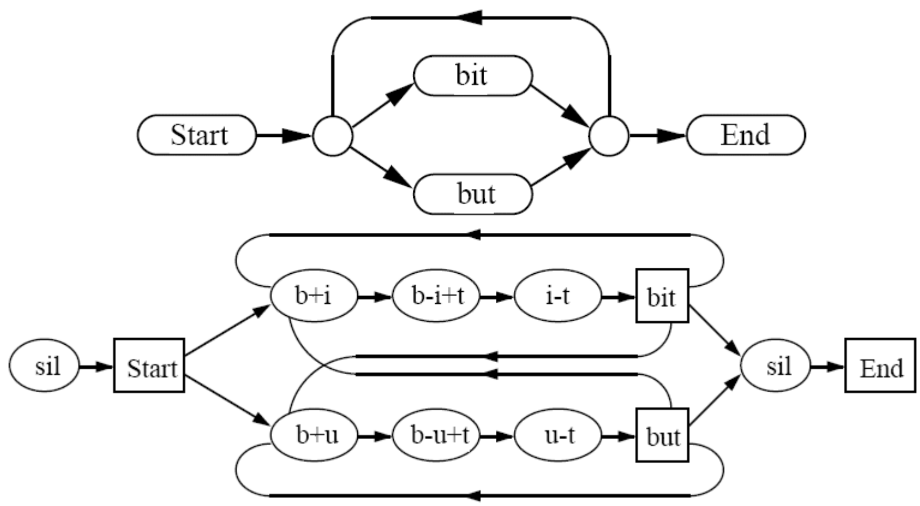
- Monophone modeling is too simple to model coarticulation phenomenon ubiquitous in speech.
- Modeling context-dependent phonemes: biphone, triphone, etc.
  - American English: 42X42X42 triphones → 74,088 HMMs
- The idea of concatenation equally applies to context-dependent HMMs except context agreement between adjacent HMMs, which may complicate network especially in boundary.



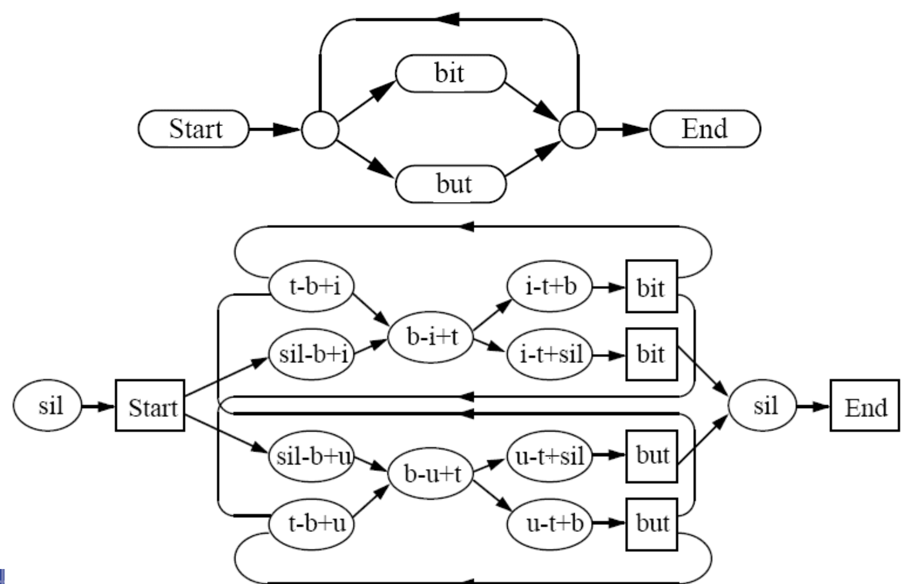
Example (1): grammar network expansion with monophone HMMs



Example (2): grammar network expansion with word-internal triphone HMMs



Example (3): grammar network expansion with cross-word triphone HMMs



ASR: Viterbi search

- Assume we build the grammar network for the task, and all physical HMMs attached in the network have been estimated.
- An unknown speech utterance, → a sequence of feature vectors *Y.*
- Speech recognition is nothing more than a viterbi search:
  - The whole network viewed as a composite HMM *Λ.*
  - *Y* is viewed as input data, find the optimal alignment path (viterbi path, state sequence) *S** traversing the whole network (from START to END).

$$S^* = \arg\max_{S \in \Omega} \Pr(S) \cdot p(Y, S \mid \Lambda)$$
$$= \arg\max_{S \in \Omega} \Pr(W_S) \cdot p(Y, S \mid \Lambda)$$

  - Once *S** is found, the recognition results (word sequence) can be derived by backtracking the Viterbi path.

Equivalent or not ?

- Theoretical solution:

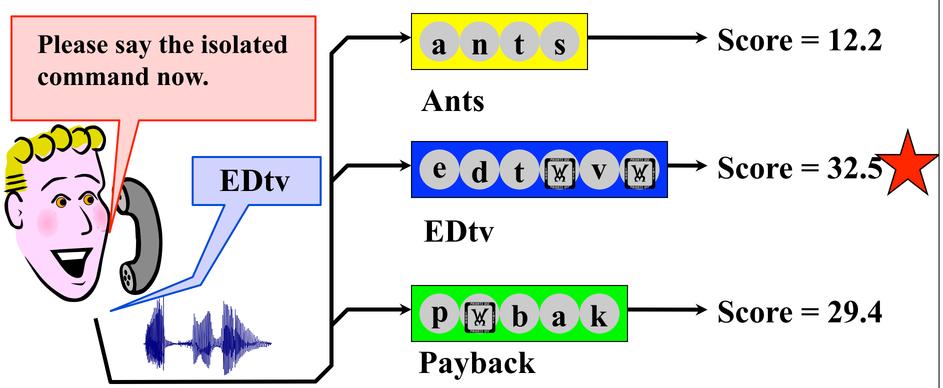
$$\hat{W} = \arg\max_{W \in \Gamma} p(W \mid X) = \arg\max_{W \in \Gamma} P(W) \cdot p(X \mid W)$$
$$= \arg\max_{W \in \Gamma} \overline{P}_{\Gamma}(W) \cdot \overline{p}_{\Lambda}(X \mid W)$$
$$= \arg\max_{W \in \Gamma} \Pr(W) \cdot \sum_{S \in \mathrm{O}_w} p(Y, S \mid \Lambda)$$

- Practical solution:

$$S^* = \arg\max_{S \in \Omega} \Pr(S) \cdot p(Y, S \mid \Lambda)$$
$$= \arg\max_{S \in \Omega} \Pr(W_S) \cdot p(Y, S \mid \Lambda)$$

Isolated-word ASR

- Isolated-word speech recognition is a special case:
  - Solution 1: building a multi-branch FSG network (one word per branch).
  - Solution 2: no overall network; examine all words one by one; each time a word → a small HMM network → Viterbi/Forward-Backward to calculate score.



ASR Problems

$$\hat{W} = \arg\max_{W \in \Gamma} p(W \mid X) = \arg\max_{W \in \Gamma} P(W) \cdot p(X \mid W)$$

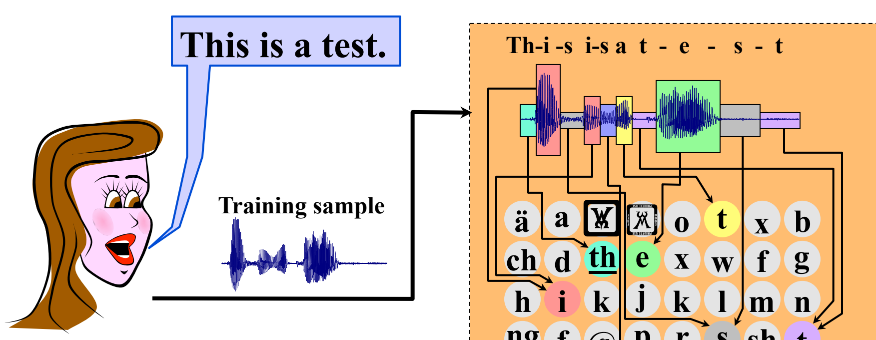
$$= \arg\max_{W \in \Gamma} \overline{P}_{\Gamma}(W) \cdot \overline{p}_{\Lambda}(X \mid W)$$

- Training Stage:
  - ***Acoustic modeling***: how to select speech unit and estimate HMMs reliably and efficiently from available training data.
  - ***Language modeling***: how to estimate n-gram model from text training data; handle data sparseness problem.
- Test Stage:
  - ***Search***: given HMM's and n-gram model, how to efficiently search for the optimal path from a huge grammar network.
    - Search space is extremely large
    - Call for an efficient pruning strategy

Acoustic Modeling

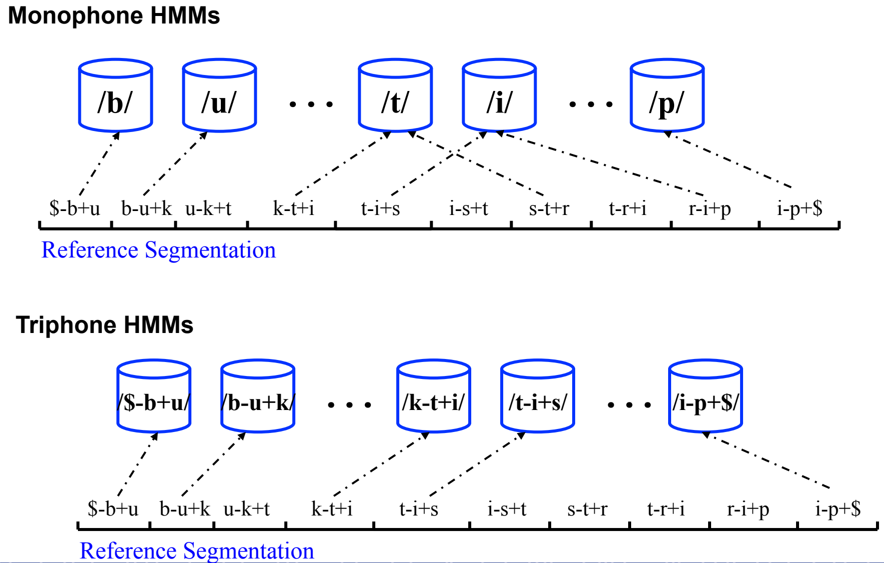
- Selection of speech Units: what speech unit is modeled by an HMM; task-dependent.
  - Digit/digit-string recognition: a digit by a HMM → 10-12 HMMs
  - Large vocabulary: monophone→biphone→triphone→beyond
- HMM topology selection:
  - Phoneme: 3-state left-right without skipping state
  - Silence or pause: 1-state HMM (with skipping transition)
  - Digit/word: 6-12 states left-right no state skipping
- HMM type selection:
  - Top choice: Gaussian mixture CDHMM
  - Number of Gaussian mixtures in each state could vary depending on the amount of training data. (e.g., 1,2,…,20)
- HMM parameters estimation:
  - ML (Baum-Welch algorithm)
  - Bayesian: MAP
  - Discriminative Training: MMI, MCE

Training Speech Recognizer (monophone HMMs)



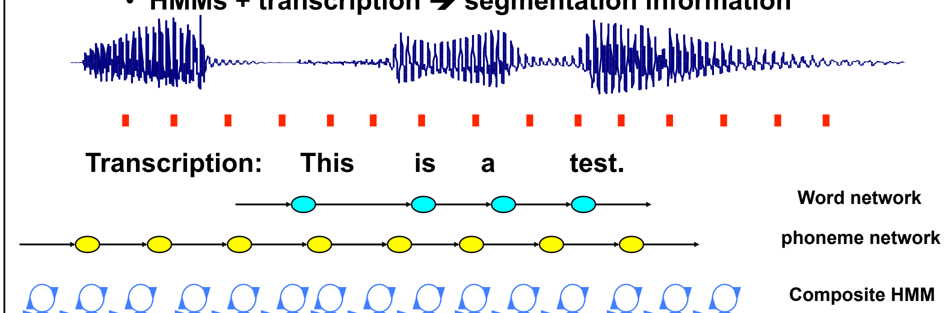
Thousands of training samples are combined to build 42 sub-word models, one for each phoneme.

Segmental Training: find the proper data segment for each HMM



Reference Segmentation

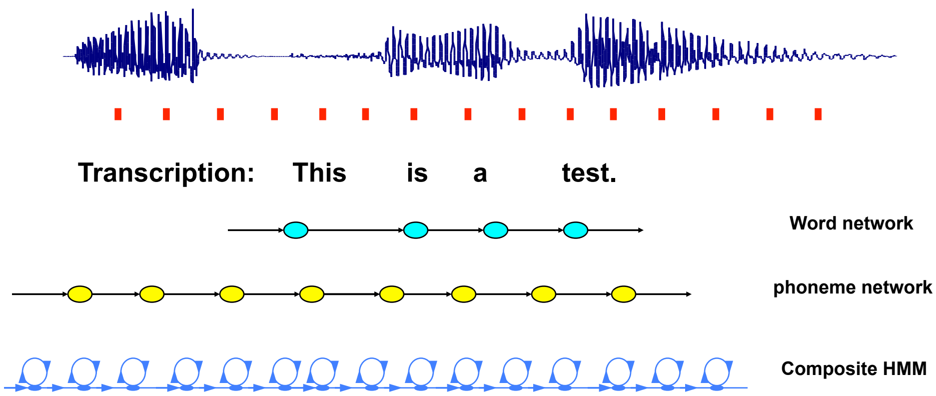
- **Where the segmentation information comes from?**
  - **Human labeling: tedious, time-consuming, expensive;**
    - **Only a small amount is affordable; used for bootstrap.**
  - **Automatic segmentation if an initial HMM set is available.**
    - **Forced-alignment: Viterbi algorithm; Need transcription only**
    - **HMMs + transcription → segmentation information**



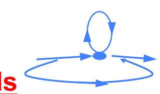
Run the Viterbi algorithm to backtrack segmentation information

Embedded Training

- Only need transcription for each utterance; no segmentation is needed; automatically tune to optimal segmentation during training.

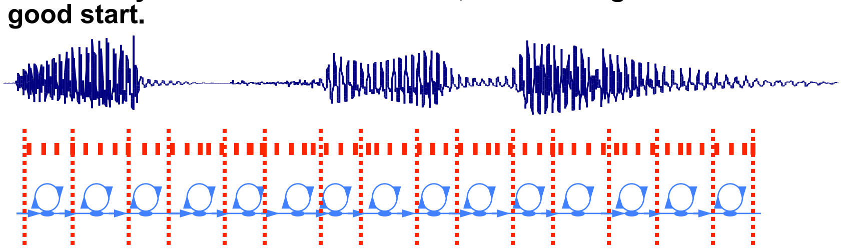


- Run the Baum-Welch Algorithm to estimate all parameters in the composite HMM;
- May add optional 1-state silence models between words



HMM Parameters Initialization

- If boundary information is unknown, uniform segmentation seems a good start.



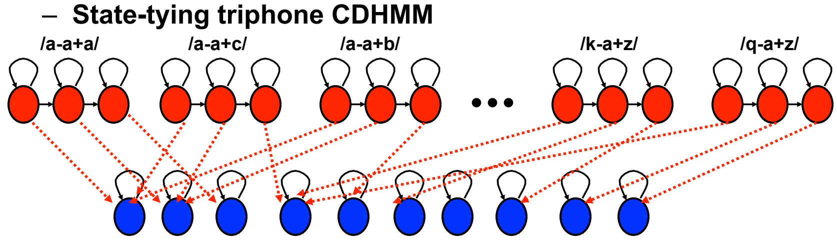
- A good strategy to avoid bad local maximum in training:
  - Progressively increasing complexity of models
  - For Gaussian mixture CDHMM
    - Build a single Gaussian per state; optimize
    - Split the mixture → 2-mixture CDHMM; optimize
    - Gradually increase the number of mixtures
  - Monophone → triphone → …

Parameter Tying

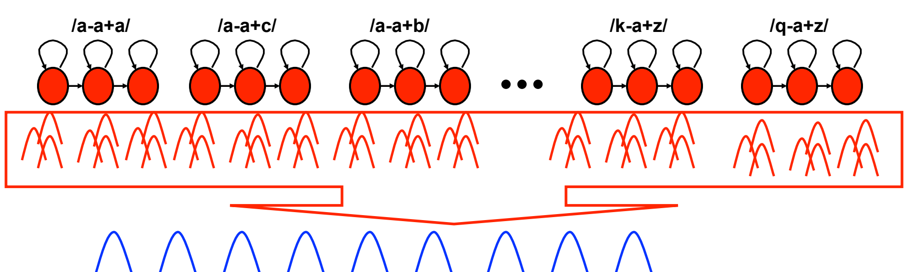
- Parameter tying: some model parameters of different classes are tied to be equivalent to reduce the total number of free parameters.
  - Trade-off between resolution and precision
- Why need parameter tying?
  - In ASR, we always have tremendous amount of parameters to be estimated from limited amount of training data.
  - In triphone system: 42*42*42*3*10*(39+39*39)+more
  - Some triphones seldom occur even in large corpora.
- Manual parameter tying based on prior phonetic knowledge.
- Several automatic methods to tie HMM parameters systematically:
  - State-tied CDHMM
  - Phonetically Tied Mixtures (PTM) CDHMM
  - Semi-Continuous HMM

HMM tying: State-tied vs. PTM

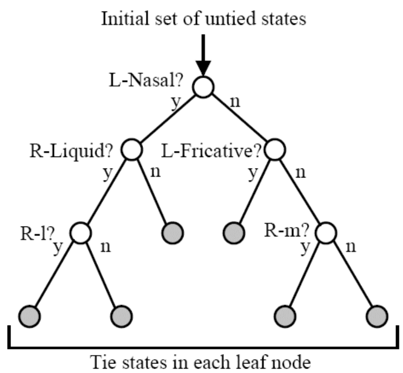
- All allophone models of a phone, to say *a*
  - State-tying triphone CDHMM



  - Phonetically Tied Mixtures (PTM) triphone CDHMM



Phonetic Decision Tree: HMM state-tying

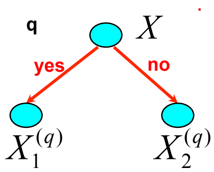


- The questions relate to the phonetic context to the immediate left or right.
- Binary question examples:

(1) Is the left phone is a nasal?
(2) Is the right phone a fricative?
(3) Is the left phone "l"?
(4) …

- A phonetic decision tree is built to tie the same state of a triphone set derived from the same monophone.
- Each phonetic decision tree is a binary tree in which a question is attached to each intermediate node.
- Each terminal (leaf) node represents a distinct state cluster in tying.
- Given a tree, from root → leaf
  - Find the cluster it ties with
  - Even applicable to unseen triphone (which we don't have data at all)
- Data-driven decision tree growing method:
  - Entropy reduction → likelihood increase

Phonetic Decision Tree: HMM state-tying



- X represents all data corresponding to the state of one triphone set. X is a set of feature vectors.
- Modeling the data in each node with a single Gaussian model:
  - estimate common mean μ_X and covariance Σ_X:

$$H(X) = \int N(X \mid \mu_X, \Sigma_X) \cdot \log N(X \mid \mu_X, \Sigma_X)\, dX$$

$$= C + \log |\Sigma_X|$$

- For any question Q, split data and calculate for each child node:

$$H(X_1^{(q)}) = C_1 + \log |\Sigma_{X_1^{(q)}}|$$

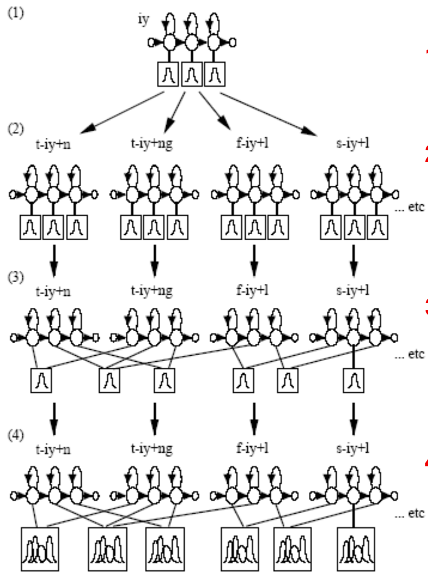
$$H(X_2^{(q)}) = C_2 + \log |\Sigma_{X_2^{(q)}}|$$

- Choose the question which maximizes entropy reduction:

$$q^* = \arg\max_q H(X) - \frac{|X_1^{(q)}|}{|X|} H(X_1^{(q)}) - \frac{|X_2^{(q)}|}{|X|} H(X_2^{(q)})$$

$$= \arg\max_q |X| \log |\Sigma_X| - |X_1^{(q)}| \cdot \log |\Sigma_{X_1^{(q)}}| - |X_2^{(q)}| \cdot \log |\Sigma_{X_2^{(q)}}|$$

HMM state-tying using decision tree



1) Initially train 3-state left-right single Gaussian monophone CDHMM.

2) For tri-phones occurring frequently, clone its corresponding monophone as initial, then re-train from data using Baum-Welch algorithm.

3) For all triphones derived from the same monophone, building 3 phonetic trees for each state to tie these states in certain way.

4) Keeping the state-tying structure, increment the number of Gaussian mixand in each state until the performance is optimal.

Measuring Accuracy (ASR Errors)

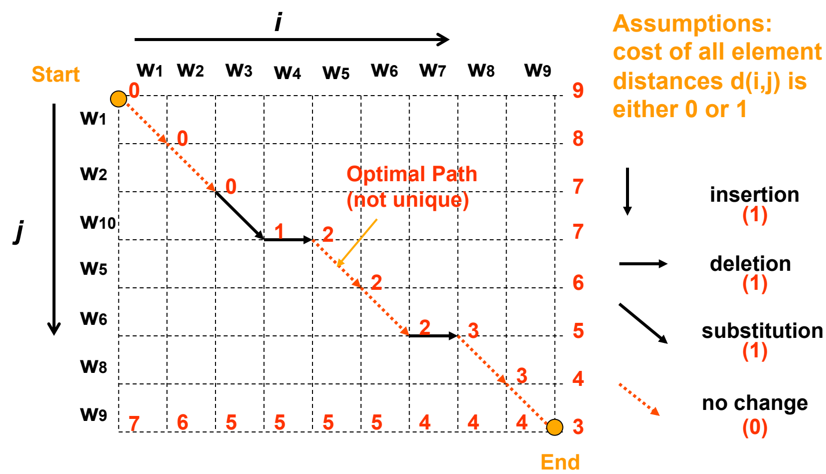
- Word Accuracy
  - In continuous ASR, not easy to count (substitution/deletion/ insertion errors) .
  - Minimum Edit distance → minimum substitution + deletion + insertion errors
  - Word Accuracy:

$$\text{Word Accuracy} = 100\% \times \frac{\text{sub} + \text{del} + \text{ins}}{\#\ \text{words in correct transcriptions}}$$

- String Accuracy
  - correct recognition of all words in an utterance
- Semantic Accuracy
  - correct interpretation of meaning of an utterance; take the correct action based on the utterance; correct recognition of all semantic attributes

String Edit Distance: minimum errors

Correct:	W_1 W_2 W_3 W_4 W_5 W_6 W_7 W_8 W_9
Recognized:	W_1 W_2 W_{10} W_5 W_6 W_8 W_9



Algorithm for Minimum Edit Distance

```
begin initialize u(), r(), I <- length[U], J <- length[R], D[0,0]=0
        i <- 0
        do i <- i+1
            D[i,0] = i
        until i = I
        j <- 0
        do j <- j+1
            D[0,j] <- j
        until j = J
        i <- 0; j <- 0
        do i <- i+1
            do j <- j+1
               D[i,j]=min{D[i-1,j]+1, D[i,j-1]+1, D[i-1,j-1]+q(u(i),r(j))}
                        (insertion) (deletion) (substitution or no change)
            until j = J
        until i = I
   return D[I,J]
end
```

Initialize boundaries with large distances

q(u(i),r(j)) is 1 for substitution and 0 for no change

Minimum Edit Distance

Factors Determining Accuracy

- How Words Are Spoken by a Speaker
  - poor articulation and mispronounced words
  - co-articulation by running words together
    - this supper = this upper
  - speaker characteristics
    - speaking rate, loudness, dialect, etc.
- The Words Themselves..
  - homophones: similar sounding words (blue - blew)
  - Acoustic confusion
  - ambiguity: multiple meanings (checking)

Accuracy (Cont'd)

- The Speaker Population
  - general public, captive audience
  - naïve or frequent users
- The Speaking Environment
  - channel, microphone, ambient noise, etc.
- Rejection Processing
  - important component for building intelligent user interface
  - confidence measure needed for error correction, repair, deciding how much to confirm, partial understanding
- Human Factors
  - ASR solutions are as much an art form as a science (sometime proper prompting is very effective)
  - transaction design to maximize success rate

Speech Recognition Difficulties (Robustness)

- Variability of sounds (e.g. words, phrases)
  - within a single speaker: variable length patterns, no clear boundaries
  - across speakers: accent, style, pronunciation, etc.
- Transducer and channel variability
- Environmental noise and acoustics
- Speaker production errors
  - hesitations, repairs, extraneous speech
  - variability in expressions
  - mismatch in user expectation and system capabilities

DARPA ASR Benchmark