

# Engineering Part IIB: Module 4F11

## Speech and Language Processing

### Lecture 6 : Sub-Word Modelling

Phil Woodland: [pcw@eng.cam.ac.uk](mailto:pcw@eng.cam.ac.uk)

Lent 2016



Cambridge University Engineering Department

## Large Vocabulary Speech Recognition

The next three lectures concentrate on the problems of building acoustic and language models for speech recognition and decoding speech into a word sequence using these models with:

- Large or Very Large Vocabulary ( $> 20000$  words)
- Speaker Independent
- Continuous Speech

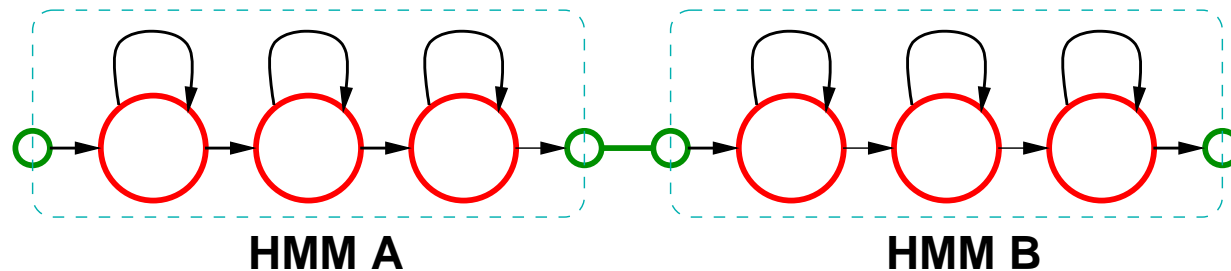
This lecture will discuss

- Training for continuous speech / sub-word models
- Acoustic Model units and in particular context-dependent phone models
- Performance of a complete system



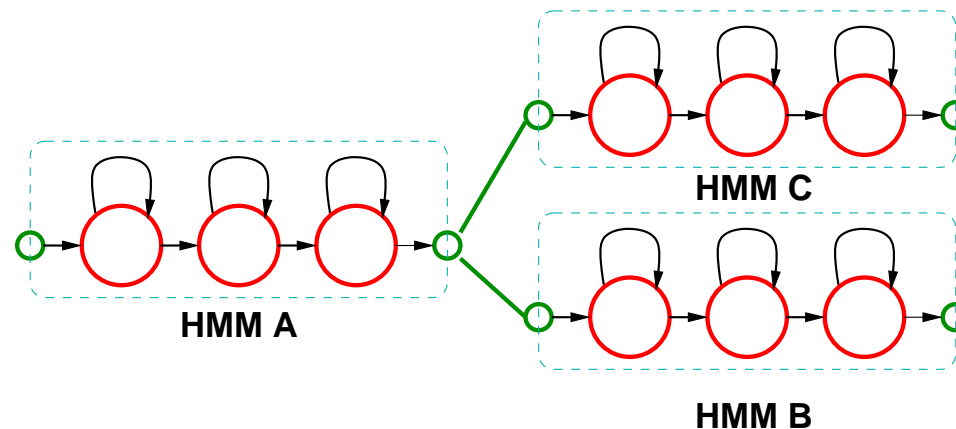
## Composite HMMs

One important advantage of HMMs is that larger HMMs can be constructed by the composition of smaller ones:



The use of unique entry and exit states allows the simple concatenation of HMMs giving

- word models from sub-word units
- sentence models from word models (used in training and recognition ...)



## Training for Continuous Speech Recognition

There are a number of ways to obtain suitable models for continuous speech recognition.

### (i) **Isolated Word Training**

Use HMMs trained on isolated words, does not reflect additional variability

### (ii) **Segmentation of continuous utterances**

Find the segment boundaries using Viterbi alignment, recording the boundary time of each word/model boundary. Train on the resulting segmented data. Note that segmentation and parameter re-estimation can be iterative.

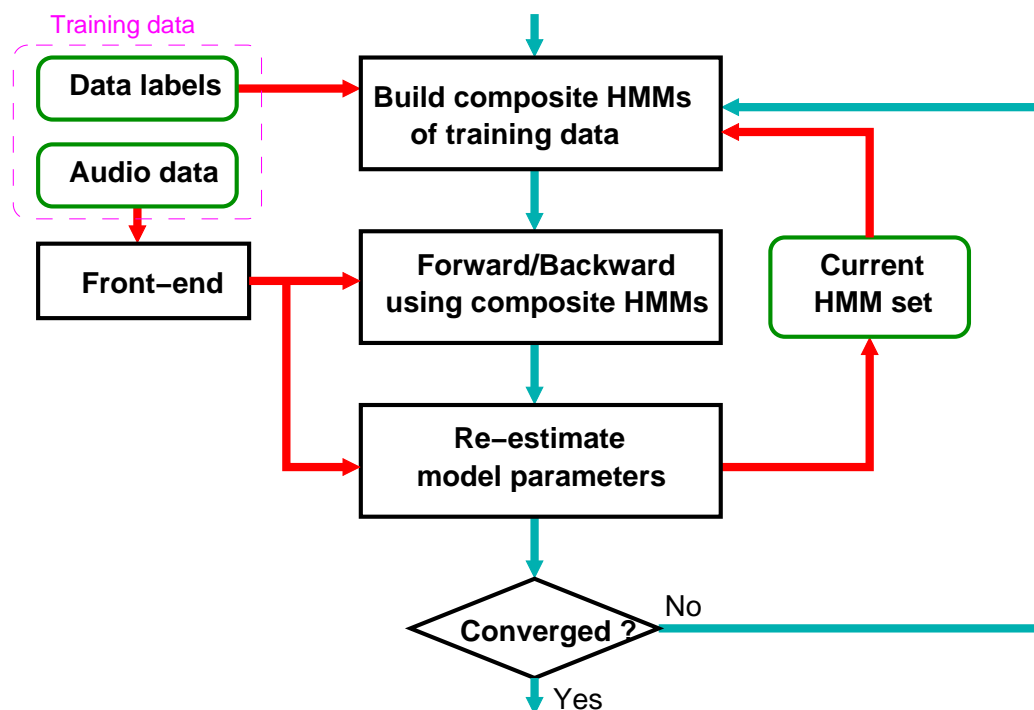
### (iii) **Sentence-level Baum-Welch training**

For each sentence, construct a model for that sentence by concatenating HMMs and then accumulate statistics for all HMMs at once. Uses “probabilistic segmentation”, soft decisions on model boundaries. Initialise models from (i) or (ii) for word HMMs (or use flat-start approach).



## Sentence-Level Baum-Welch Training

In addition to initial HMMs and the audio data, a known model sequence is required. If only word-level training transcripts are available, for sub-word unit training, the model sequence will come from a pronunciation dictionary. Note that a flat start, i.e. all models with equal arbitrary initial model parameters, is possible!

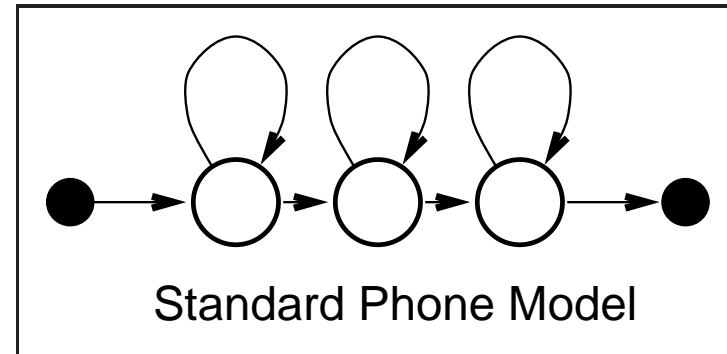


Note that neither the word boundaries nor the sub-word boundaries need to be explicitly known

## Model Structure/Pronunciations

Often the same model structure is used for each HMM

- Three emitting states;
- Left-to-right structure no skips;



Word pronunciation variability is handled by using multiple pronunciation dictionaries. eg

the = / **dh ax** /  
= / **dh iy** /

Models typically trained by selecting one of the possible pronunciations as “correct” given the current model set and training using this single pronunciation. An alternative is to make a network of pronunciations and train on this.

## Acoustic Modelling Units

What should be used as the basic units to model speech?

1. Compact (even for large vocabulary tasks);
2. Simple mapping between recognition units and words (sentences);
3. Account for speech variability (e.g. linguistic structure, **co-articulation** between neighbouring speech sounds);
4. Extensible to handle words not seen in the training data;
5. Well-defined and easily identifiable, so that large training corpora may be constructed.

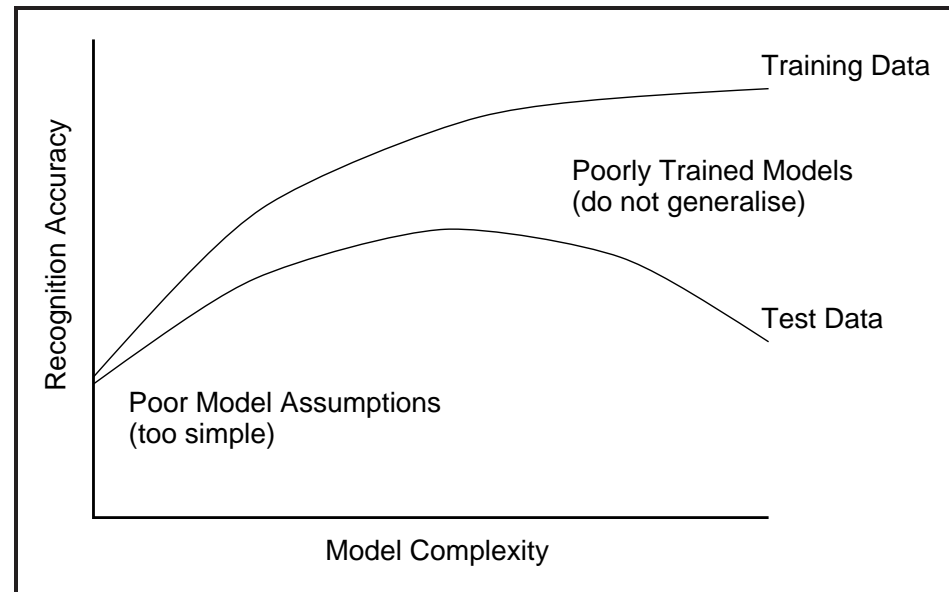
The choice of unit should also be flexible enough so that the “correct” number of models for the available training data can be chosen to avoid **over-training** i.e. too many parameters for the available training data for optimal test performance.



## Limitations of “Word” Speech Units

Words as the unit for speech recognition are impractical as:

1. “Too many” models, memory and computation increases as vocab size;
2. Require “vast” amounts of training data to “correctly” estimate the parameters
3. Cannot construct models for words not seen in training data;
4. Cannot capture inter-word co-articulation effects (though intra-word variability very well modelled).



Increasing the training data will move the “Test Data” peak to the right.



## Possible Speech Units

Possible units that have been proposed are:

<b>Phones</b>	40-50 phones in English Highly context dependent Well defined Non-unique phone-string to word-string e.g. <i>grey twine</i> and <i>great wine</i>
<b>Syllables</b>	10000+ syllables in English Hard to obtain good estimates
<b>Demi-Syllables</b>	2000+ in English, Hard to define

For example the word Segmentation may be written as

<b>Phone</b>	/ s e h g m a x n t e y s h a x n /
<b>Syllable</b>	/ seg men ta tion /

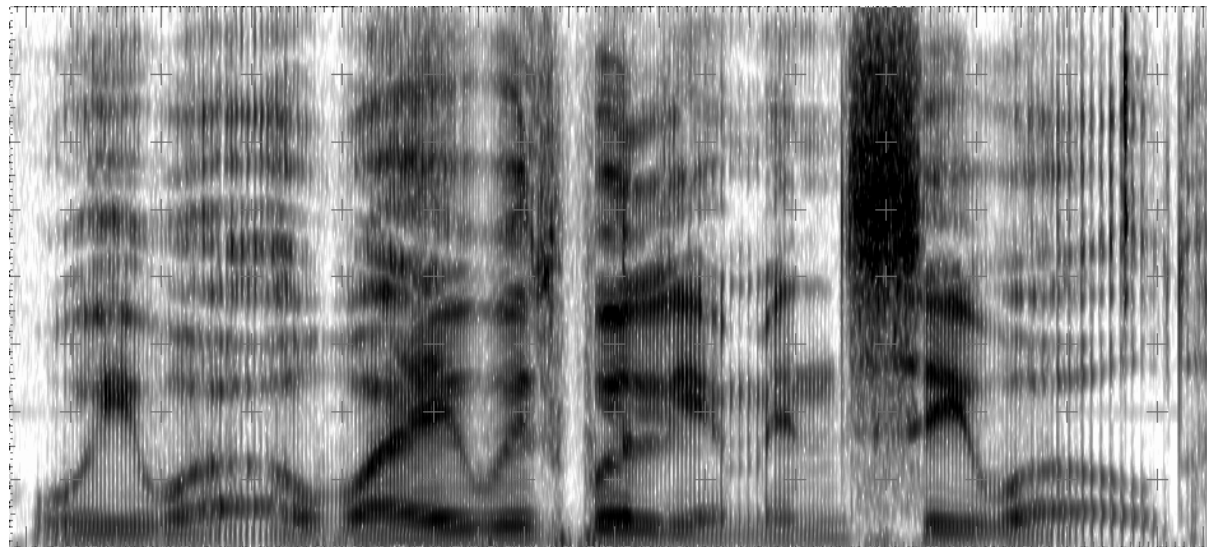
For demi-syllables, split in the vowel in each syllable.

To date, LVR systems for English using syllable based units are rarely used.



## Phone Variation

Co-articulation causes the acoustic realisation in a particular context to be more consistent than the same phone occurring in a wide variety of contexts.



WE WERE AWAY WITH WILLIAM IN SEA WORLD

The diagram shows a spectrogram for the phrase

**We were away with William in Sea World**

Each realisation of the *w* phone varies considerably but the most similar are the two occurrences in the same triphone context (underlined).

To handle this problem **context dependent** phone models may be used

## Context Dependent Phones

*Context Independent* phone models (**Monophones**) can be made more “specific” by taking into account phonetic context to form *Context Dependent* models.

The phonetic transcription for the word Speech can be written as:

<b>Monophone</b>	/ s p iy ch /
<b>Biphones (L)</b>	/ sil-s s-p p-iy iy-ch /
<b>Biphones (R)</b>	/ s+p p+iy iy+ch ch+sil /
<b>Triphones</b>	/ sil-s+p s-p+iy p-iy+ch iy-ch+sil /

Main problem with Context Dependent phones is trainability. For  $N$  phones,  $N^2$  biphones,  $N^3$  triphones etc. (only a smaller set of these will occur).

Here we are using the HTK notation for context dependency in which **l-c+r** denotes a phone **c** with a left context of **l** and a right context of **r**.

Note that HTK is a widely used toolkit for research into HMM-based speech recognition systems. For more information on HTK (documentation, source code etc etc) see <http://htk.eng.cam.ac.uk>.



## Context Dependent Phones (2)

*Word boundary* information may be made use of.

- **Word-internal:** Word boundaries represent a distinct context.

speech task = / sil s+p s-p+iy p-iy+ch iy-ch  
t+ae t-ae+s ae-s+k s-k sil /

- **Cross-word.** Word boundaries ignored (or used as additional context).

speech task = / sil-s+p s-p+iy p-iy+ch iy-ch+t  
ch-t+ae t-ae+s ae-s+k s-k+sil /

For a 26000 word dictionary, designed for transcription of the Wall Street Journal corpus:

<b>Word Internal</b>	14300 distinct contexts
<b>Cross Word</b>	54400 distinct contexts

**Problem:** only 22804 cross-word triphones appear in the training data (WSJ SI-284).

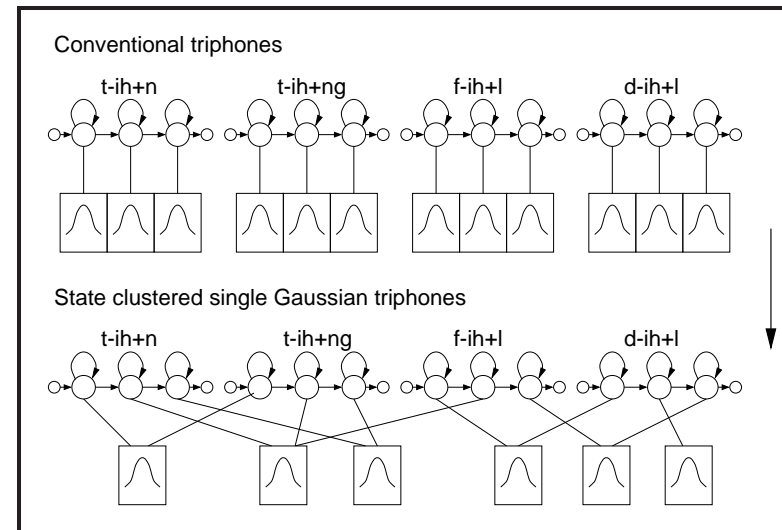


## System Trainability

Need to balance between level of acoustic detail and robust parameter estimation.

- **Backing off** - When insufficient data to train a specific context *back-off* and use a less specific model. *Triphone*  $\rightarrow$  *Biphone*  $\rightarrow$  *Monophone*  
Inflexible, involves large “jumps”.

- **Sharing (Tying)** - *Share* parameters between models that are acoustically similar. Flexible as models of the same complexity can share data, as well as with less specific models.



## Bottom-Up Parameter Tying

For reliable estimates need parameter sharing across contexts. Basic procedure::

1. Models are built for all observed contexts.
2. Merge “models” that are acoustically similar.
3. If sufficient data available **stop** else goto (2).

Two standard forms

- **Generalised Triphones** - The model comparisons and merging may be done at the model level to form *Generalised Triphones*.
- **State-Clustered Triphones** - Comparison and clustering may be performed on the state level to form *State-Clustered Triphones*. Allows left state to be clustered independently of the right state.

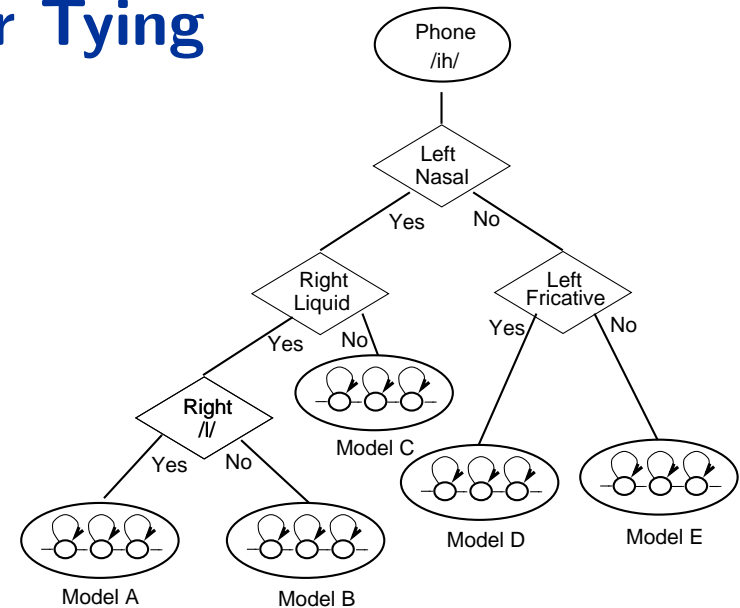
### Limitations -

- Unreliable for contexts that occur rarely in training data.
- Unable (without using back-off) to cluster contexts not seen in training data.



# Top-Down Parameter Tying

- Binary decision tree, at each node yes/no decision to form context equivalence classes
- Can be used at model or state level



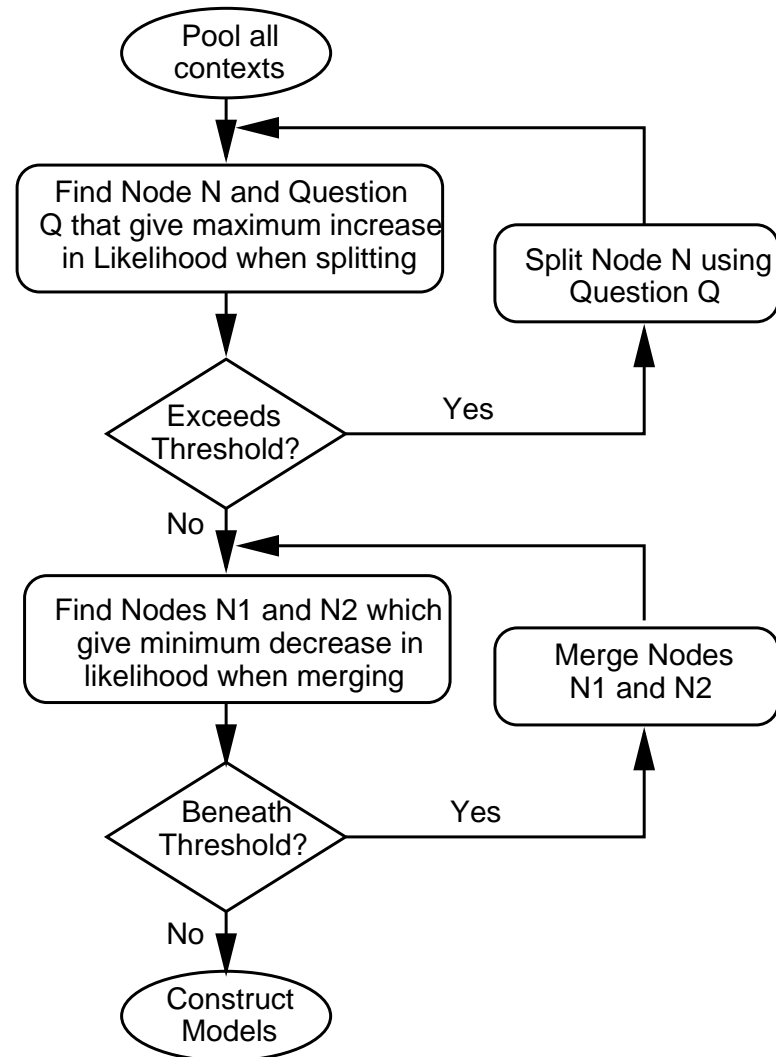
## Advantages

1. No need to back-off, handles unseen contexts elegantly.
2. Allows *expert* knowledge to be incorporated
3. Allows any degree of context dependency to be simply incorporated.

## Disadvantages

1. Only locally optimal splits are selected.
2. Not all question combinations normally asked

## Constructing Decision Trees



To make decision tree generation computationally efficient:

1. The frame/state alignment is not altered (hence the contribution of the transition probabilities may be ignored).
2. Single component Gaussians are accurate enough to decide on contexts.

In practice an additional splitting threshold is introduced that ensures there is sufficient data at a terminal node for robust estimation of a Gaussian mixture model.



## Change in Log Likelihood

The change in the data log likelihood in splitting contexts between two (single Gaussian) models must be calculated to select the best question to ask at each stage of tree construction.

If we hypothesise a split in the contexts at a particular node,  $p$ , into two descendants,  $r$  and  $s$ , then the increase in log likelihood is simply

$$-\frac{1}{2} \log(|\Sigma_s|) N_s - \frac{1}{2} \log(|\Sigma_r|) N_r + \frac{1}{2} \log(|\Sigma_p|) N_p$$

where  $\Sigma_p$ ,  $\Sigma_r$ ,  $\Sigma_s$ , are the covariance matrices associated with the data from node  $p$  and the splits  $s$  and  $r$ , and  $N_p$ ,  $N_r$ , and  $N_s$  are the total number of frames associated with each of the nodes  $p$ ,  $s$  and  $r$  so that  $N_p = N_r + N_s$ .

Note that if the mean vector and covariance matrix is stored for each context along with an occupation count the required covariance for any combination of states (i.e. any point in the tree) can be very simply computed.

To compute the change in likelihood for a merge (after splitting is complete) then then a similar likelihood change formula can be used.



## Top Down Decision Trees: Summary

- Aim is to group contexts into **equivalence classes**
- Normally have a separate tree for each emitting state position of each phone
- Manually define sensible questions that can be asked
  - Actual questions asked **automatically chosen** to increase approx log likelihood
  - Normally more general questions asked near top of tree (more occurrences)
  - Aim to have questions that will **generalise**
- Use single Gaussian (diagonal) statistics at each stage of tree construction
  - Simple approximation but seems good enough: simplifies computation and stats needed
- Tree splitting is a **greedy** procedure: best split at each step but end result only locally optimal
- Depth of tree will depend on the amount of data available for each state/phone
- When trees generated, tied-state HMMs can be generated for **all contexts**



## Building an HMM System

A (fairly simple) large vocab cross-word triphone system may be built as follows:

1. Using best previous models (eg TIMIT models) to obtain phone alignments.
2. Build single Gaussian monophone models (typically 4 re-estimation iterations).
3. “Clone” monophones for every cross-word triphone context seen in training data.
4. Build single Gaussian unclustered triphone models
5. Perform state-level decision-tree clustering: generates initial single-component state-clustered triphones
6. Train single-component models (typically 4 re-estimation iterations)
7. Increase num mix components (iterative component splitting) & retrain at each level  
(1→2, 2→3, 3→5, 5→7, 7→10, 10→12)

Final system is a **12-component state-clustered cross-word triphone** system.



## A Wall Street Journal System (from 1990's)

System details:

1. **Training Data:** Wall Street Journal training data (284 speakers, with 50–150 sentences from each, 36k sentences total, about 66 hours).
2. **Parameterisation:** 12 MFCCs, normalised log-energy, delta and delta-delta parameters.
3. **Acoustic Models:** 12-component state-clustered cross-word triphones (6.4k distinct states). Both *Gender-Independent* and *Gender-Dependent* forms.
4. **Vocabulary:** 65k word vocabulary. Multiple pronunciations.
5. **Language Model:** Trigram language model.
6. **Test Set:** Unlimited vocabulary, “clean” acoustic environment.

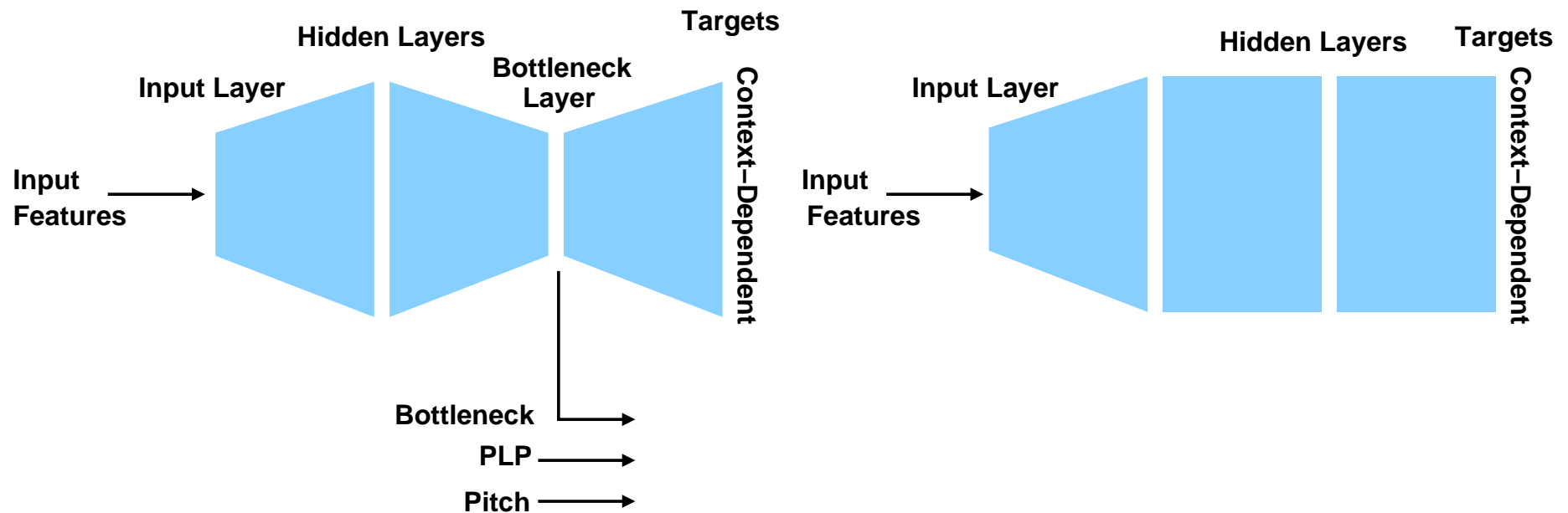
System	WER (%)	
	Dev.	Eval
Gender-Independent	9.43	9.94
Gender-Dependent	9.06	9.39

- Since this is continuous speech recognition errors are either substitutions, deletions or insertions and need to align output and reference to determine error rate



## Using DNNs with Context-Dependent Targets

- Train context-dependent GMM-HMMs and use for state-level alignment
- Context-dependent states of HMMs become DNN targets
- Use of Tandem (left) or Hybrid (right) configuration



- both approaches can yield large reductions in WER (e.g. 25-30%) over GMM-HMMs with only PLP features

## Summary

- Training for continuous speech recognition can use sentence-level Baum-Welch
- Phone-based sub-word units are used as acoustic modelling units
- Context-dependent units are needed to reduce variability due to co-articulation
- Control parameter numbers by backing-off or by parameter sharing
- Many systems use decision-tree based top-down state tying which is efficient due to simple single Gaussian assumptions
- A speaker independent large vocabulary speech recognition system using these principles (ML trained GMM-HMMs on about 60hours of read speech in low noise environment) gives WER of  $<10\%$  (lower if more acoustic training data, adaptation, more advanced training techniques, better language model etc.).
- DNNs used with context dependent state targets defined by GMM-HMMs

