HMM Training

$\equiv$ estimating model parameters

$\lambda \equiv [\pi_i], [a_{ij}], [b_j] \quad 1 \le i,j \le N$

from labeled utterances ← Baum-Welch FB algo

$$\pi_i = \frac{\#\text{ occurences in state "}i\text{" at } t=1}{\#\text{ training utterances}}$$

$$a_{ij} = P(q_j | q_i), \quad \sum a_{ij} = 1 \quad \forall i$$

$$= \frac{\#\text{ transitions from state } i \text{ to state } j}{\#\text{ transitions out of state } i \text{ across utterances}}$$

$b_j(O_t)$ = probab. of O_t computed from state j GMM $\{c_{jk}, \mu_{jk}, \Sigma_{jk}\}$

$$W^* = \arg\max_{W} P(W|\bar{O})$$

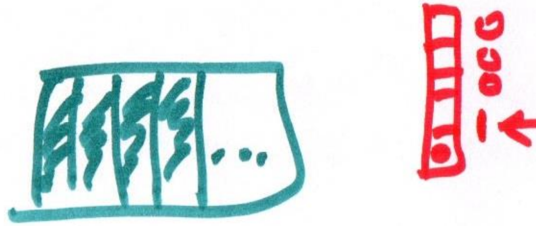
$$= \arg\max_{W} \underbrace{P(\bar{O}|W)}_{AM} \; \underbrace{P(W)}_{LM}$$

$$\sum_{Q} P(\bar{O}|Q, W)\, P(Q|W)$$

$$\approx P(\bar{O}|Q^*, W) \cdot P(Q^*|W)$$

obtained by GMM ← replaced by a DNN

obtained by HMM



$$P(\bar{O}|Q) = \prod_{t} p(o_t|q_t)$$

DNN provides $p(q_t|o_t)$

CDEEP
IIT Bombay

EE 679 L 24 / Slide 3

Language modeling:

set of constraints imposed on word sequences.

~ syntax, semantics (trained on data)

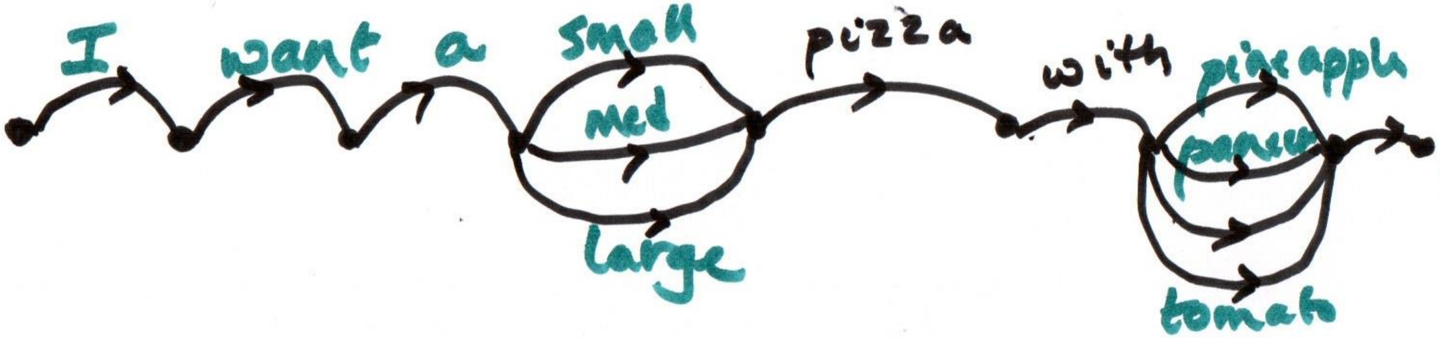
2 types:

FSG ← finite-state grammar

N-gram

CDEEP
IIT Bombay

FSG example : Order a pizza

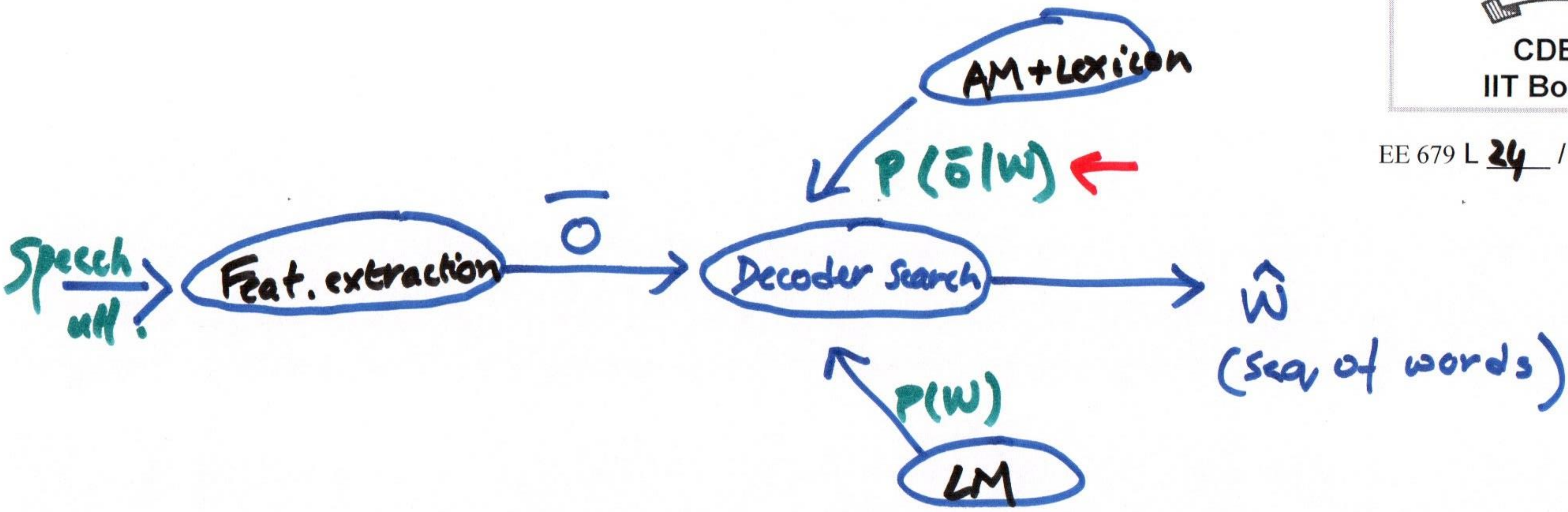


N-gram ← Trigram

$$P(w_q | W_{q-1}) = P(w_q \mid w_{q-N+1}, \ldots w_{q-1})$$

N-gram probabities
are stored in a look-up table

$$P(w_3 | w_1, w_2) = \frac{N(w_1, w_2, w_3)}{N(w_1, w_2)}$$


Speech utt.
Feat. extraction
$\bar{o}$
Decoder Search
$\hat{W}$
(Seq of words)
AM + Lexicon
$P(\bar{o}|W)$
$P(W)$
LM