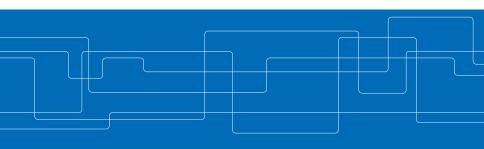


# DT2119 Speech and Speaker Recognition Lab 2

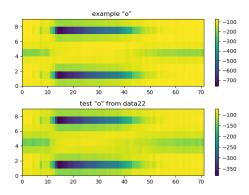
Zijian Fan, Lingxi Xiong

May 2, 2019





#### 5.1 Gaussian emission probabilities



isolated['o'] = ['sil'] + prondict['o'] + ['sil']



#### 5.2 Forward Algorithm

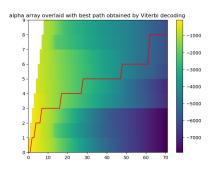
Maximum likelihood accuracy trained on all speakers: 43/44 trained on one speaker: 34/44

Viterbi algorithm accuracy trained on all speakers: 44/44 trained on one speaker: 34/44 (same mistakes as ML)

► Computational complexity Viterbi scoring:  $\mathcal{O}(n^2T)$ Forward scoring:  $\mathcal{O}(n^2T)$ 



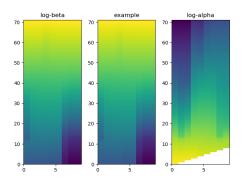
#### 5.3 Viterbi Approximation



- ►  $logV_n(j) = max_{i=0}^{M-1}(logV_{n-1}(i) + loga_{ij}) + log\phi_j(x_n)$
- $B_n(j) = argmax_{i=0}^{M-1}(logV_{n-1}(i) + loga_{ij})$



#### 5.4 Backward Algorithm



$$\triangleright \beta_n(i) = P(x_{n+1}, ..., x_{N-1} \mid z_n = s_i, \theta)$$

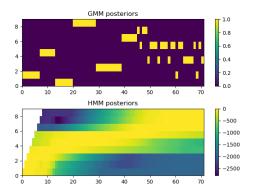


#### 6.1 State posterior(Gamma) probabilities

HMM posteriors: γ<sub>n</sub>(i) = P(z<sub>n</sub> = s<sub>i</sub> | X, θ) Summing along state: 1 (in linear domain) Summing along the time axis: expectation visiting time for each state Summing over both states and time steps = length of the observation(71)



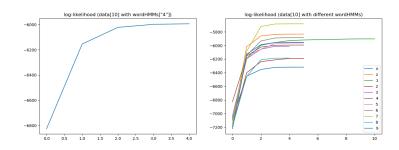
#### 6.1 State posterior(Gamma) probabilities



- HMM posteriors: γ<sub>n</sub>(i) = P(z<sub>n</sub> = s<sub>i</sub> | X, θ)
  GMM posteriors: γ<sub>n</sub><sup>GMM</sup>(i) = P(z<sub>n</sub> = s<sub>i</sub> | x<sub>n</sub>, φ)



## 6.2 Retraining the emission probabilities distributions





### Thank you for your Attention!