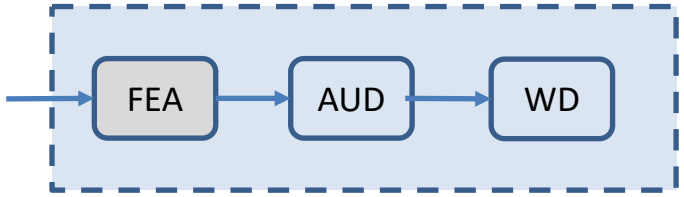
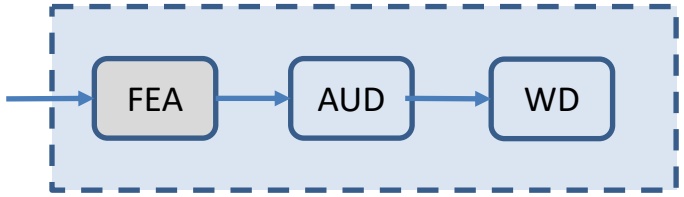


Directly Learning Words from Acoustic Observations

Daichi Mochihashi
With
Takahiro Shinozaki, Shinji Watanabe, and Graham Neubig

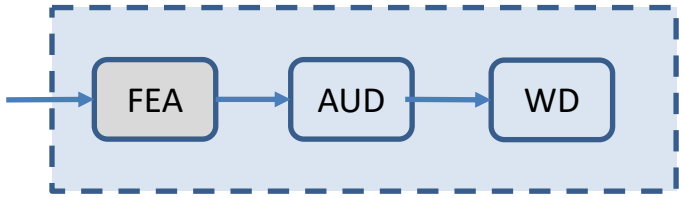


NPYLM: string->words

first,shedreamedoflittlealiceherself,andonceagainthetinyhandswereclaspedupo
nherknee,andthebrighteagereyeswerelookingupintohersshecouldheartheveryto
nesofhervoice,andseethatqueerlittletossofherheadtokeepbackthewanderinghai
rthatwouldalwaysgetintohereyesandstillasshelistened,orseemedtolisten,thewho
leplacearoundherbecamealivethestrangecreaturesofherlittlesister'sdream.thelo
Nggrassrustledatherfeetasthewhiterabbithurriedbythefrightenedmousesplashed
Hiswaythroughtheneighbouringpoolshecouldheartherattleoftheteacups…

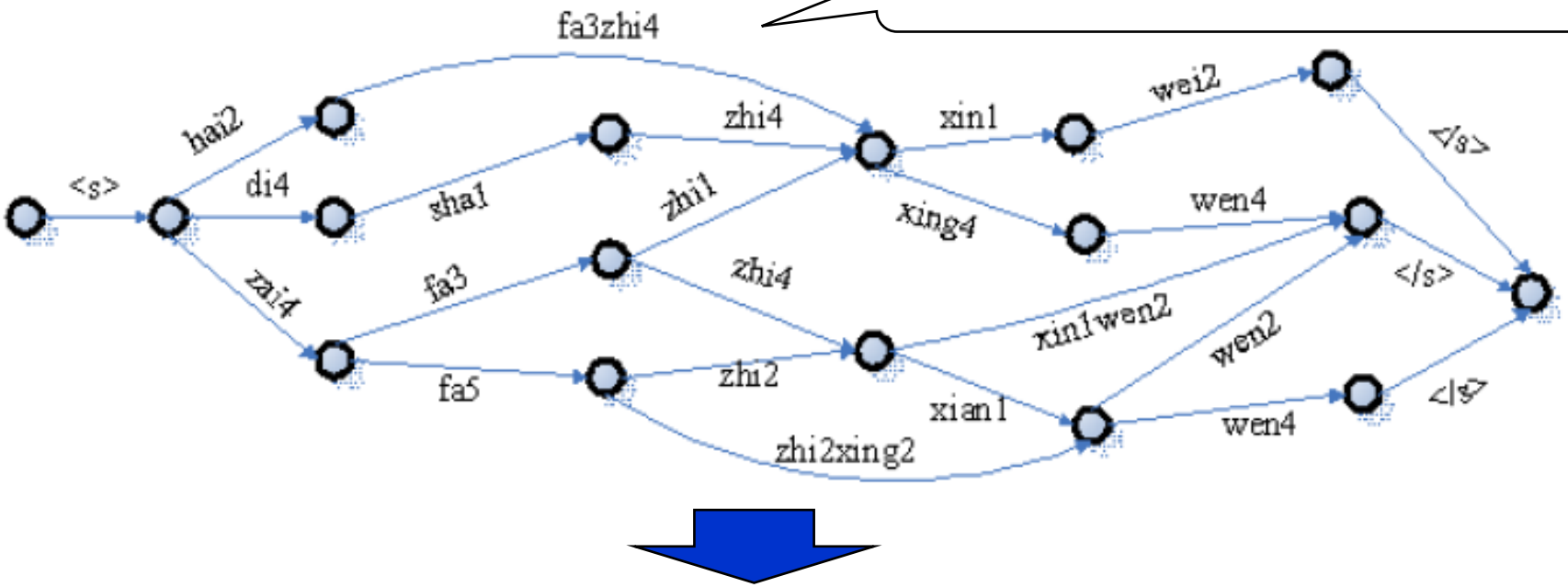
NPYLM

first, she dream ed of little alice herself ,and once again the tiny hand s were clasped upon her knee ,and the bright eager eyes were looking up into hers -- shecould hearthe very tone s of her voice , and see that queer little toss of herhead to keep back the wandering hair that would always get into hereyes -- and still as she listened , or seemed to listen , thewhole place a round her became alive the strange creatures of her little sister 'sdream. thelong grass rustled ather feet as thewhitera bbit hurried by -- the frightened mouse splashed his way through the neighbour ing pool -- shecould hearthe rattle ofthe tea cups…



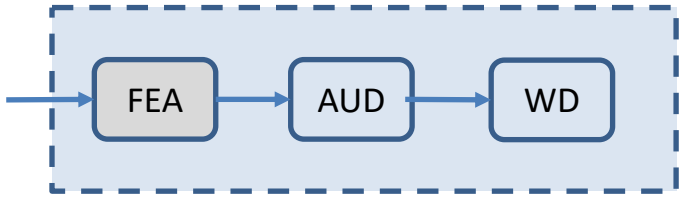
Latticelm: lattice->words

Obtained from acoustic HMM



"zai4fa3 zhi4 xin1wen2"

- Even if the input is ambiguous, WFST allows word segmentation over a phonetic lattice



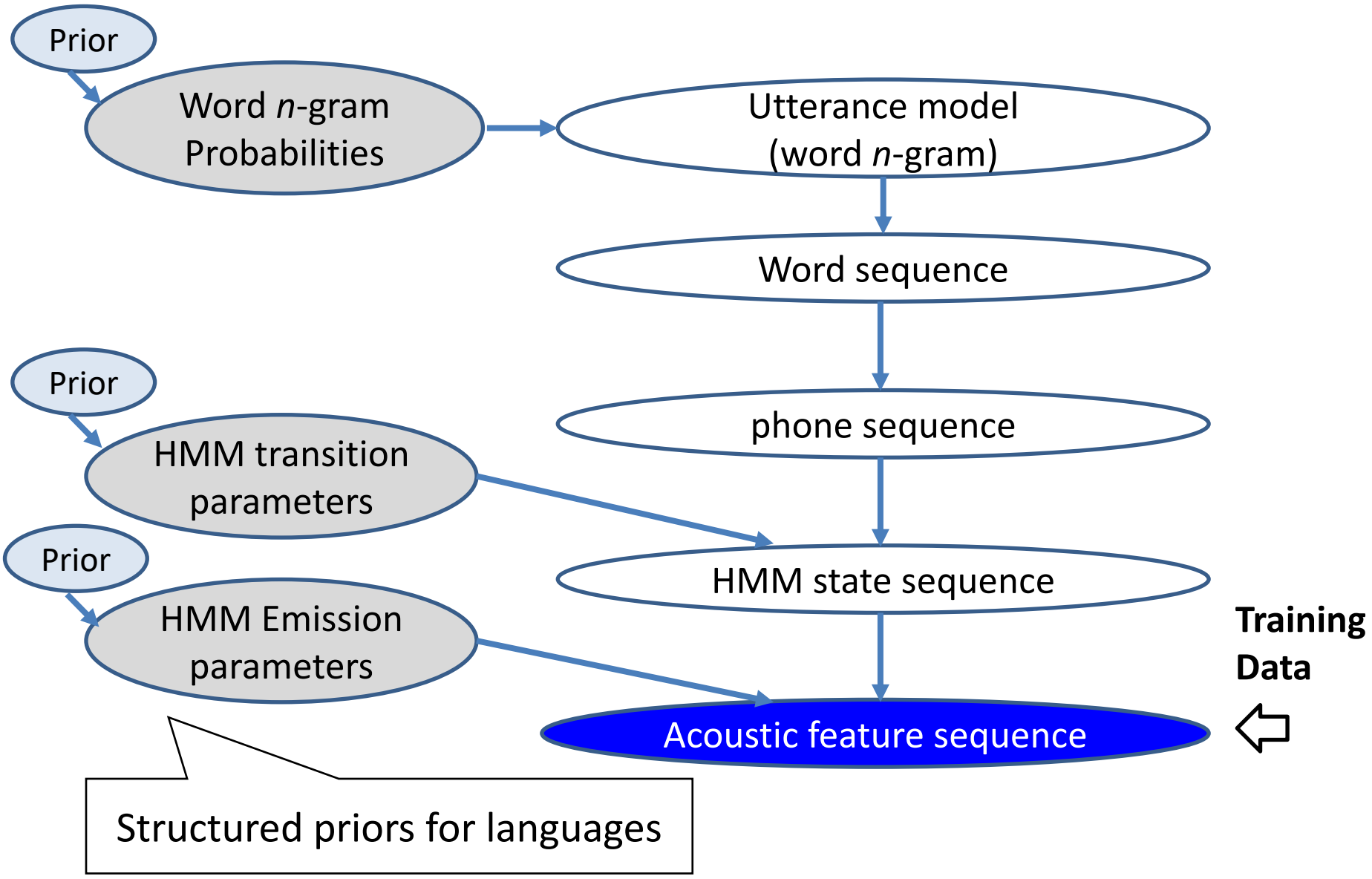
This work: Audio->words

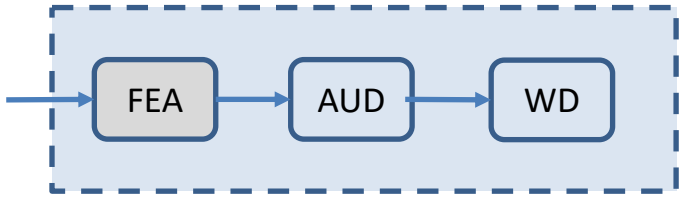
771 223 741 950 496 30 30 779 402 402 93 93 507 305 689 804 76 76 711 711
145 145 145 145 145 145 255 809 809 115 119 171 872 944 311 311 444 444
997 168 325 325 197 404 336 74 710 112 973 367 942 589 884 463 463 140
427 427
425 458 651 458 513 513 340 173 173 173 210 353 609 609 182 182 182 182
126 252 862 261 472 472 140 140 846 65 284 1011 329 392 361 393 540 916
916 295 740 553 553 553 553 969 969 488 488 174 174 880 880 581 196 196
386 900 900 900 562

“a cf b htz” (= “I am a boy”)
“jxwa mqu rdz?” (= “How are you guys?”)

- Induce “phone” and possible “word” (=sequence of phones) only from the acoustic data

Learning From Acoustic Data





The General Strategy

Compose many WFSTs iteratively!

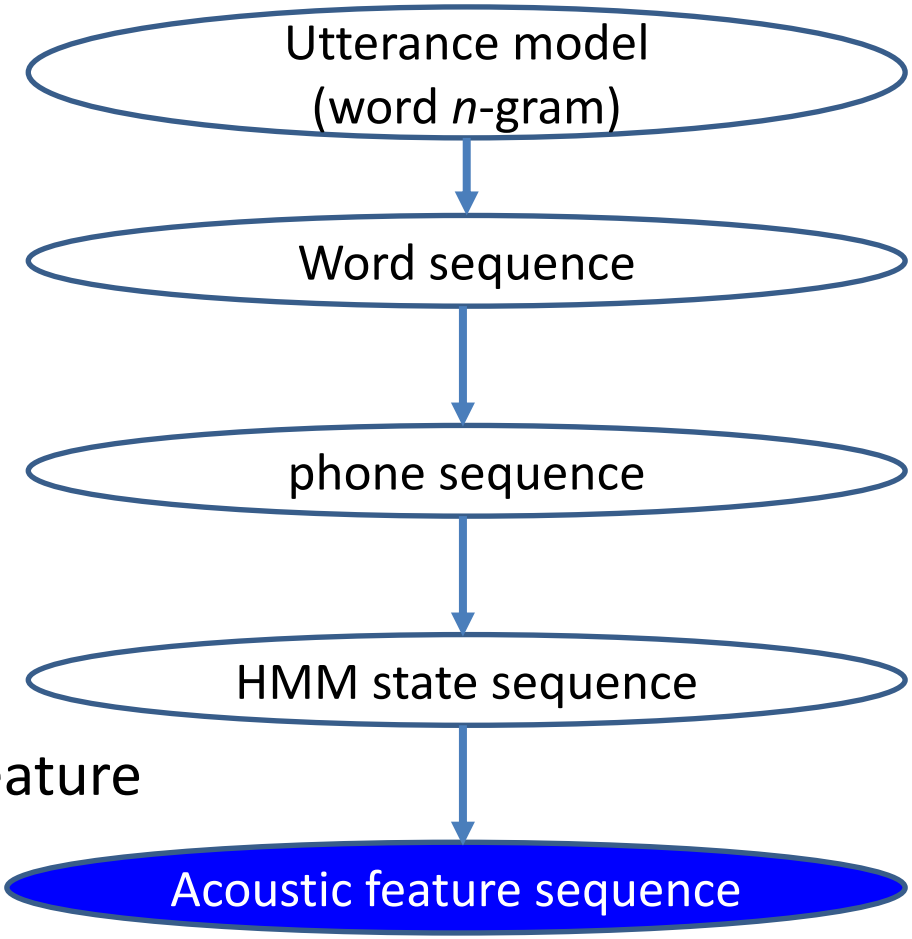
WFST: word->phone
(NPYLM on phones)

○

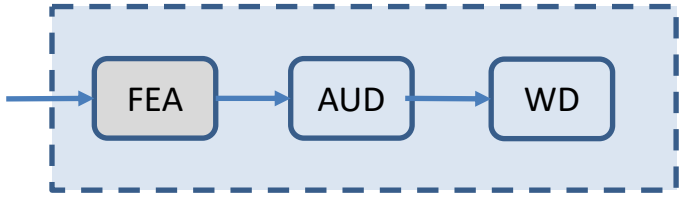
WFST: phone->state
(monophone transition)

○

WFST: state->acoustic feature



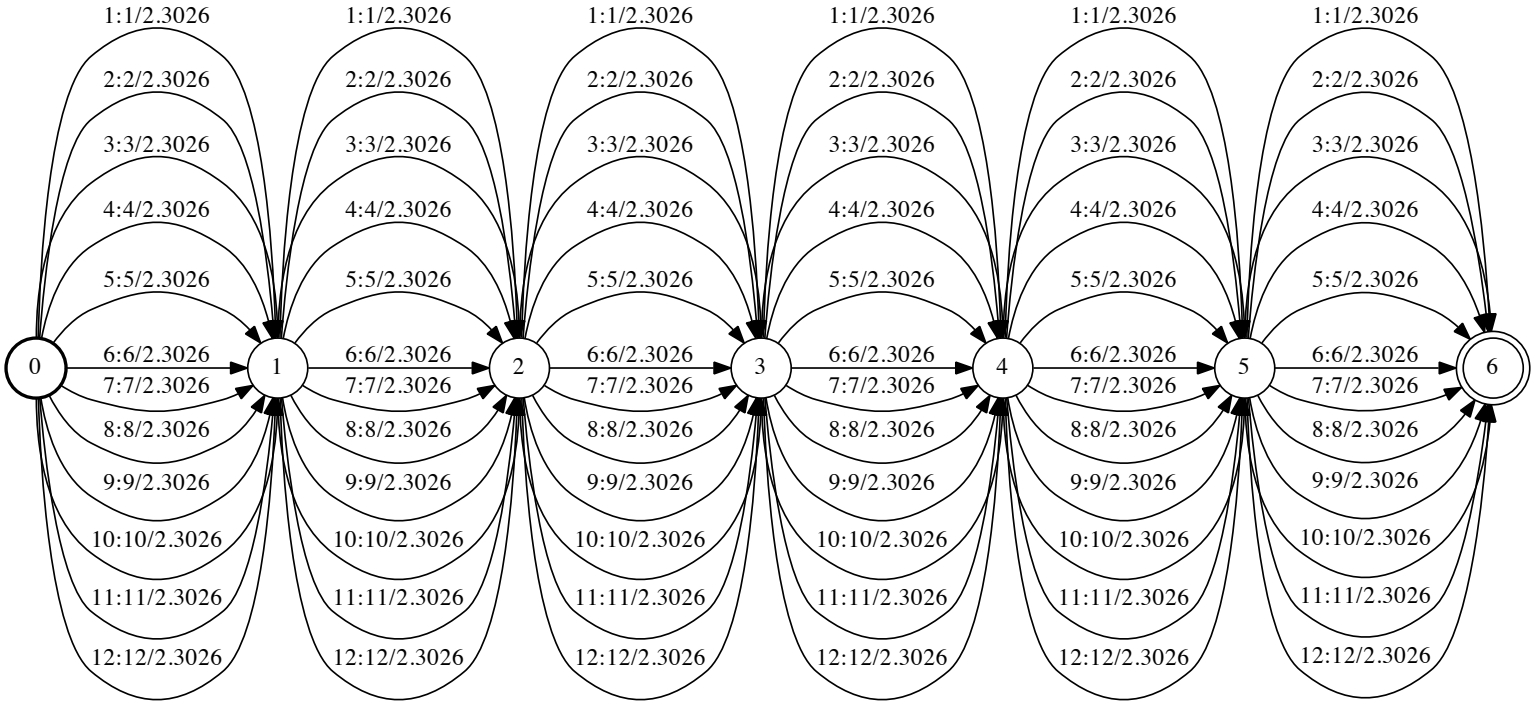
Training Data ⇦



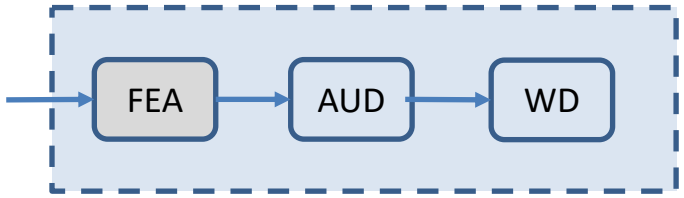
Observation FST

Observation (discretized MFCC):

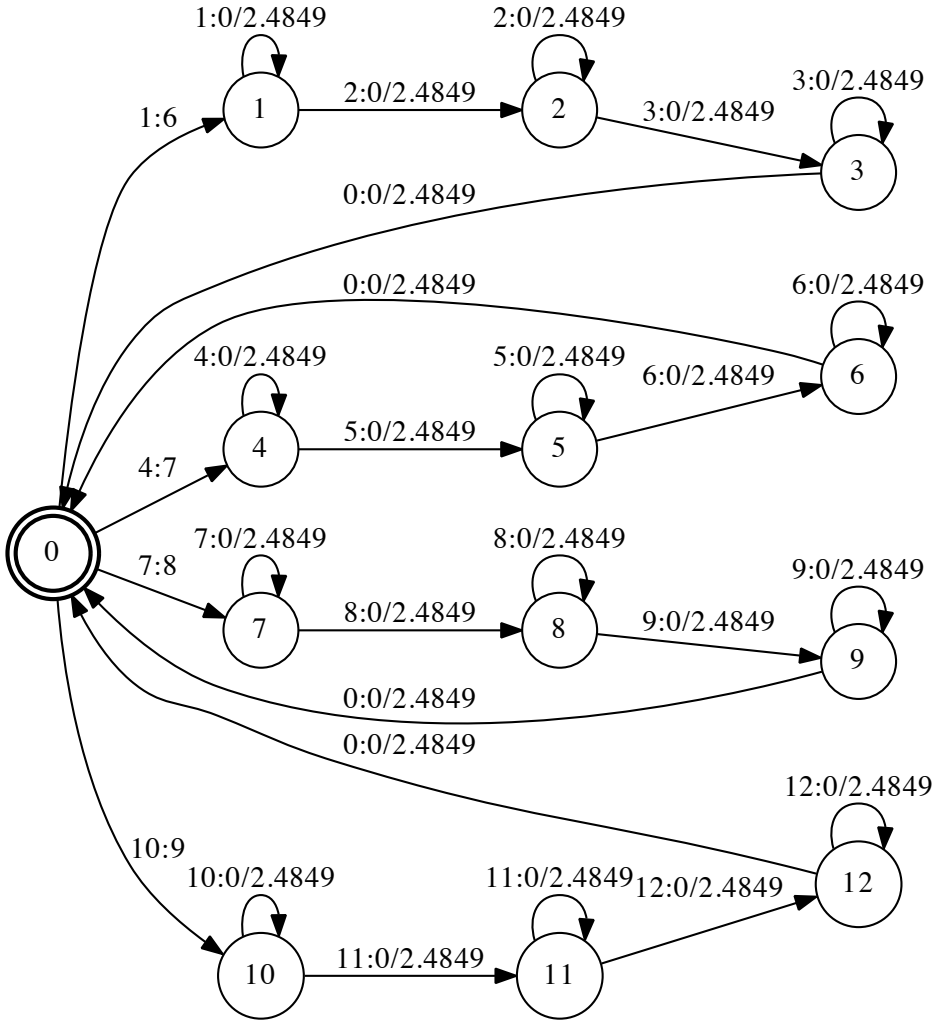
771 223 741 950 496 30



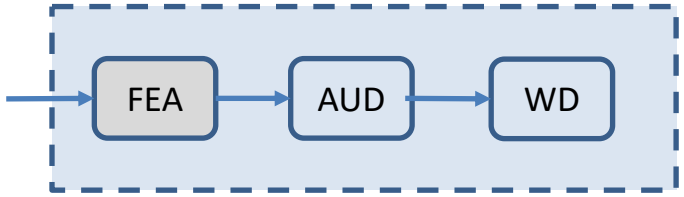
- For simplicity, assume only 4 phones
  - Left-to-right 3 states x 4 = 12 states in total



Monophone FST

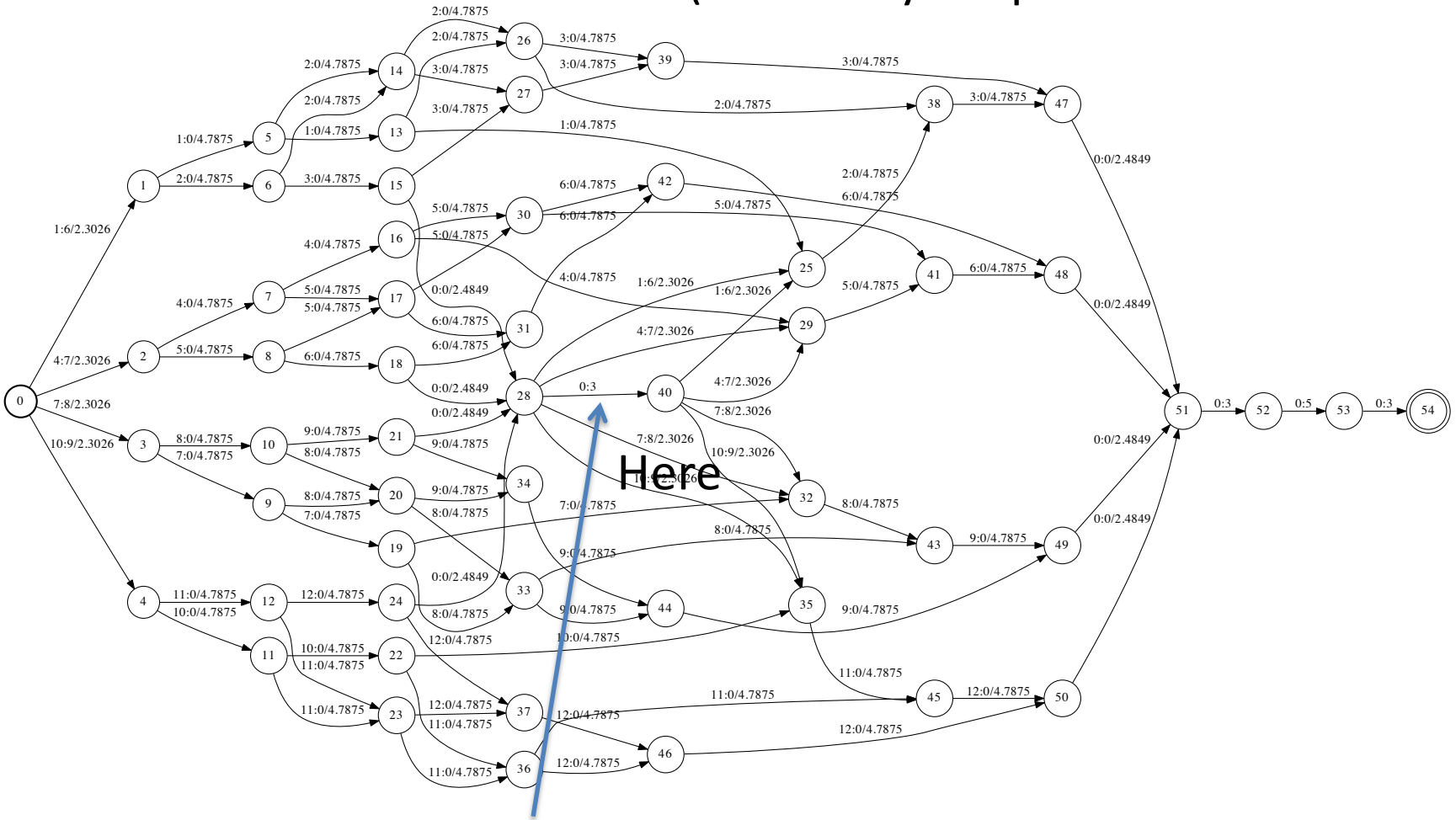


- Each phone is modeled as a left-to-right HMM and transition between them

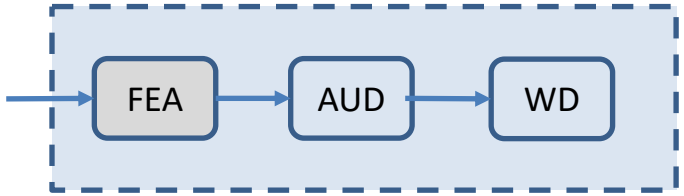


Composed HMM FST

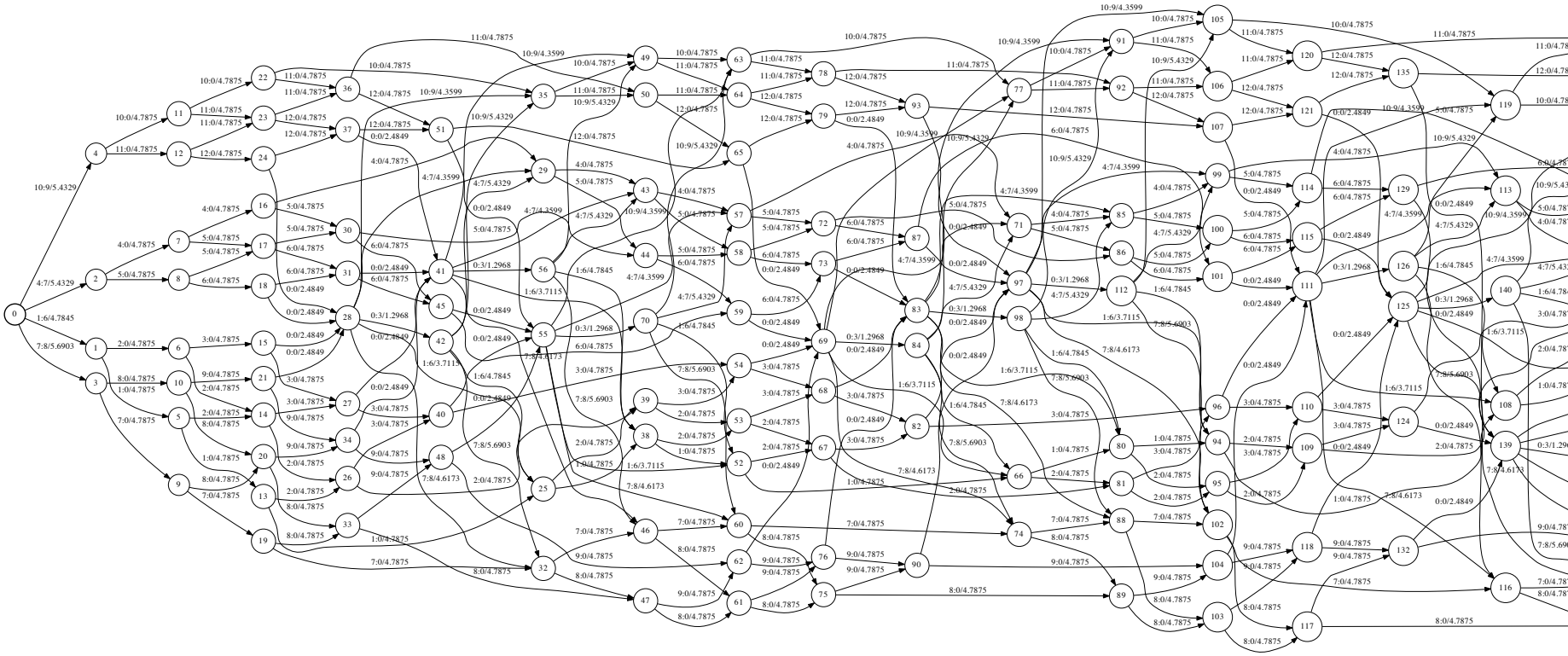
(Extremely simple case for 6 frames)



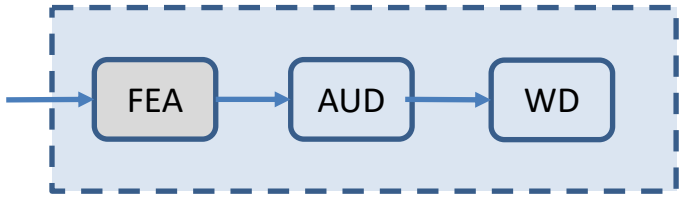
- Possible boundary is added for 6 frames here



Compose with LM



- Many “bypasses” (=existing words) are added by a language model to finally sample from

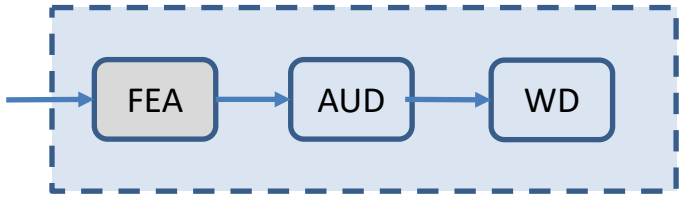


Implementation

- Extending Prof. Umbach‘s *LatticeWordSegmentation* program

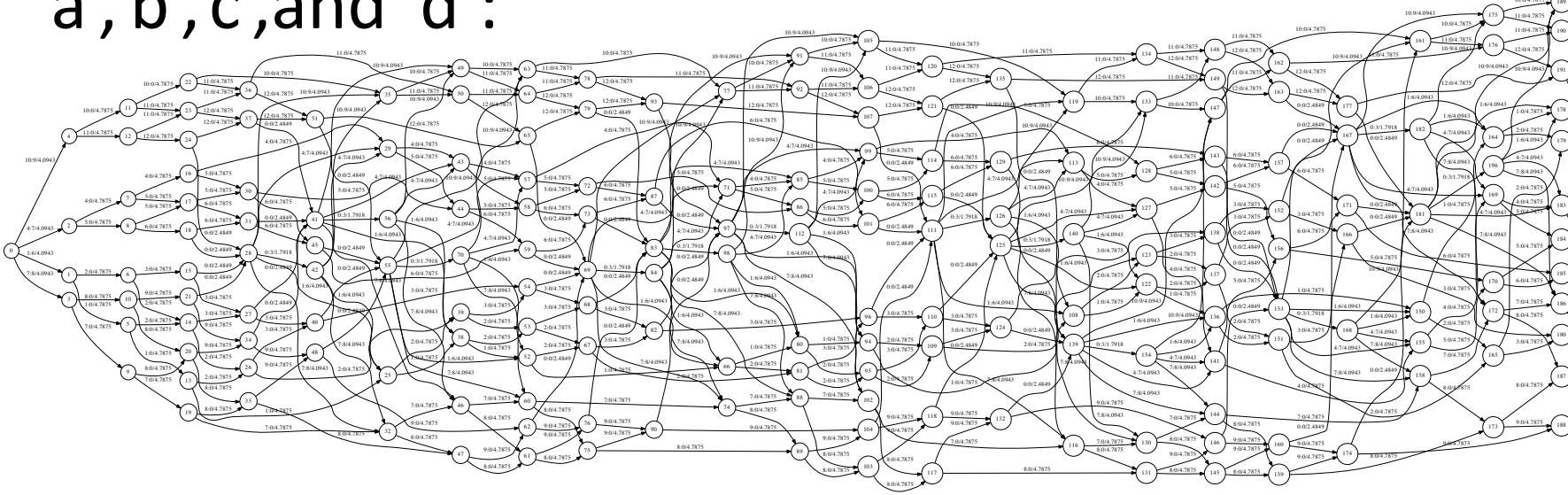
```
LogVectorFst InputFst = (
      Params.InputType == INPUT_ACFEAT ?
      acHMM->createInputFst (InputSeq(CurrentIndex),
                             Params.AmScale) :
      InputFileData.GetInputFst (CurrentIndex)
    );
```

  – And many, many fixes are required to run!

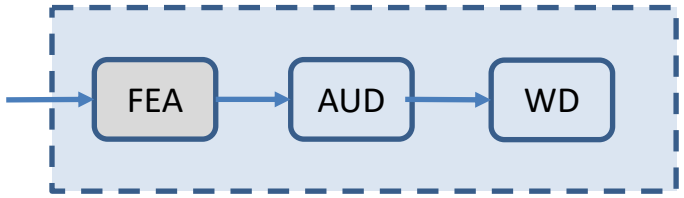


It worked!

- Let’s name the 4 unsupervised phones as ‘a’,‘b’,‘c’,and ‘d’:



Example of sampling from WFST (4 unsupervised phones)
Data: 349 349 66 359 359 769 769 837 726 726 726 123 621 263 49 49 49 79 79 329 410 -> [“bd” “d” “b”]
Low-level HMM states: 4 4 5 6 6 0 4 4 5 5 5 5 6 6 0 1 2 3 0 0 7 7 8 9 0 0 0 0

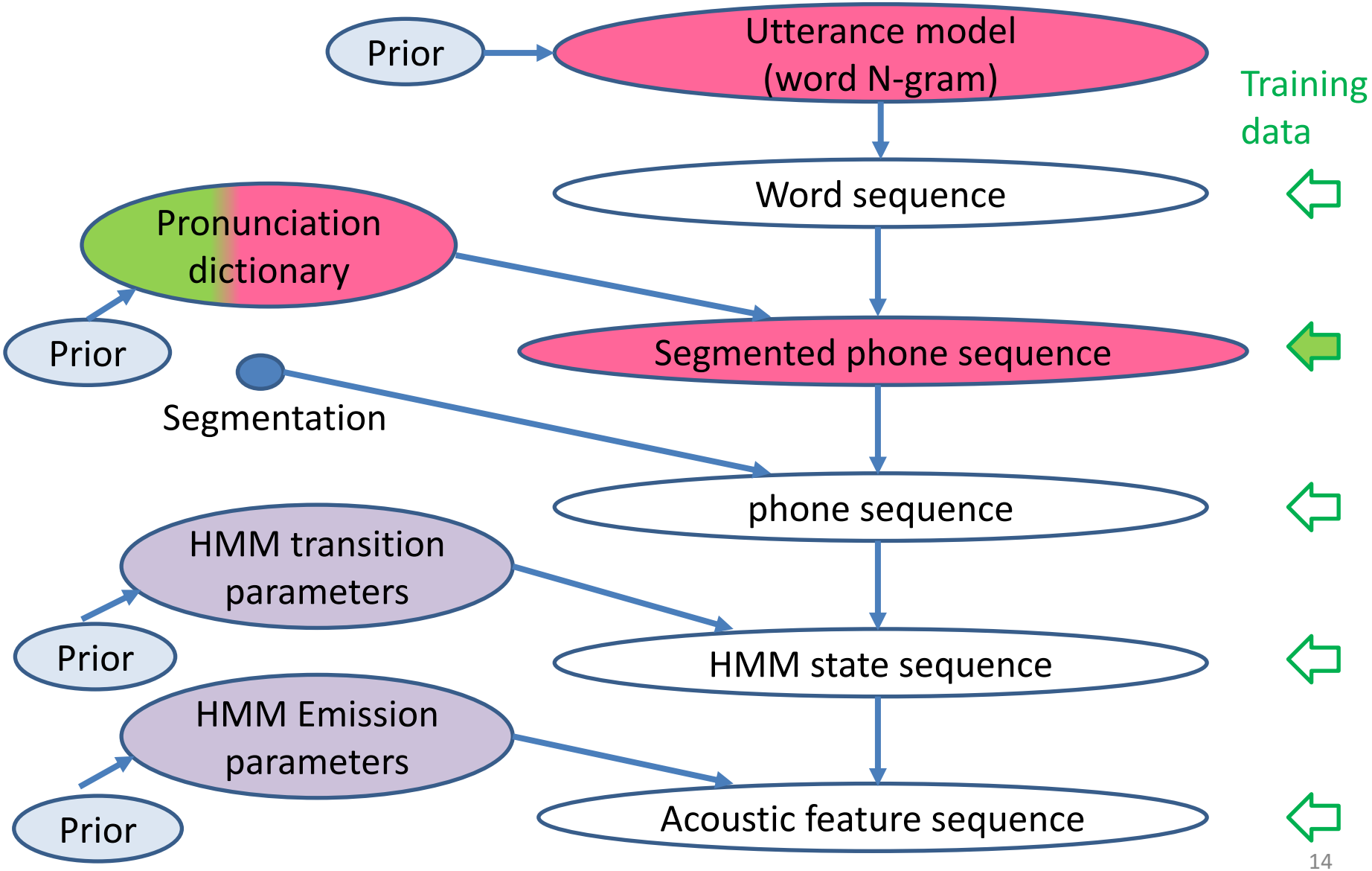


Caveats

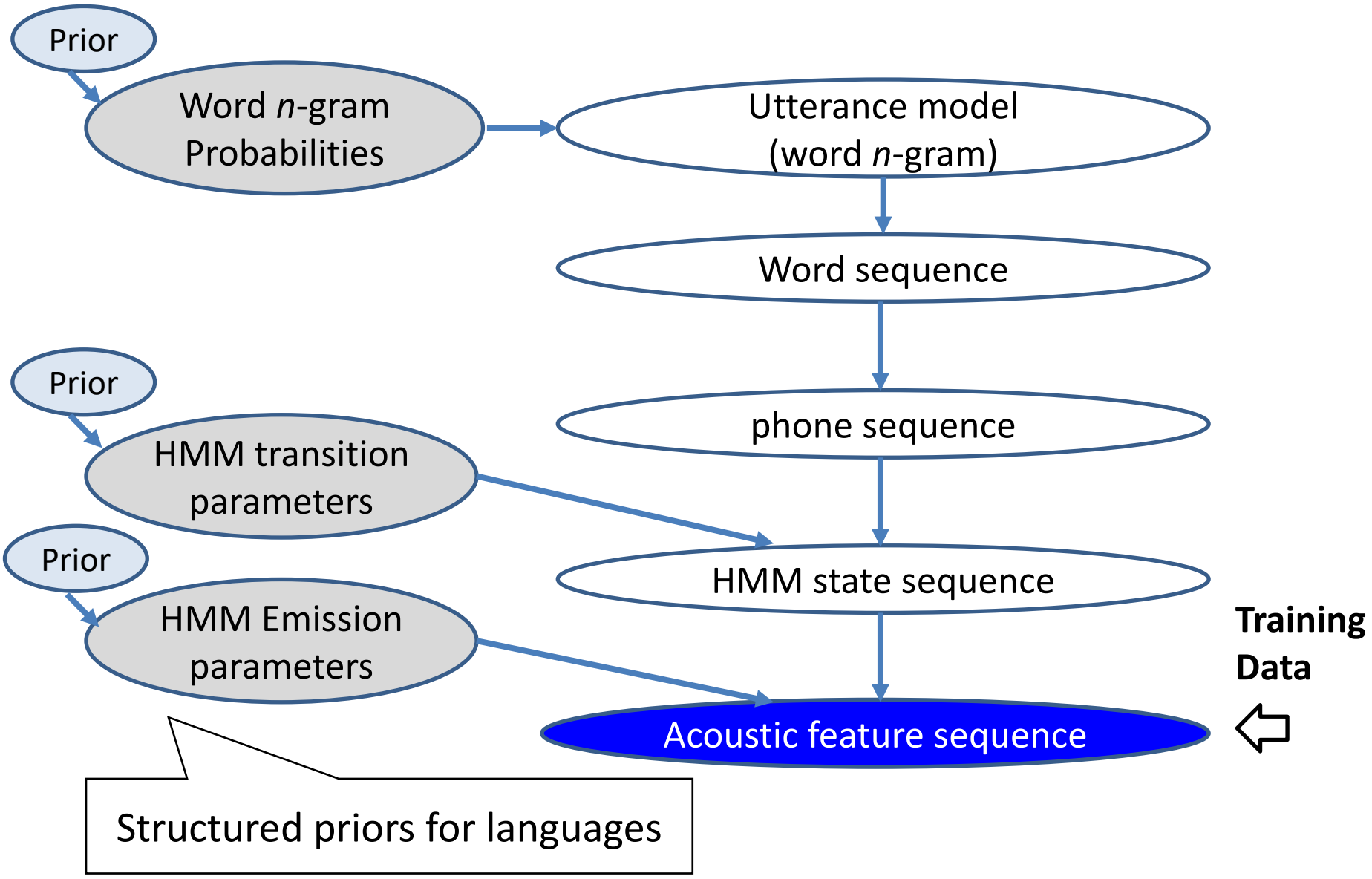
- Words like “aab” “bc” “d” using unsupervised phone have no relationship to existing letters
- Takahiro’s work will help!
  - Knowing some relationship between sound and spelling will connect the unsupervised results with our knowledge.

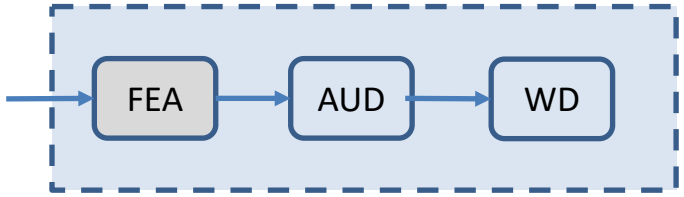
  “hello”→/h e l o:/
  “morning”→/m o: n i ng/

Takahiro's approach



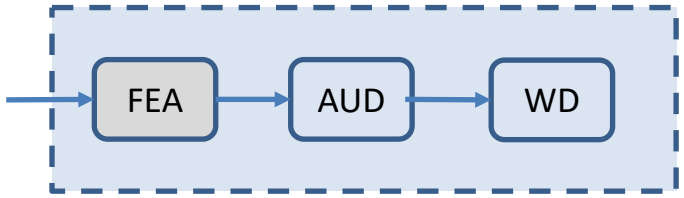
Completely unsupervised approach





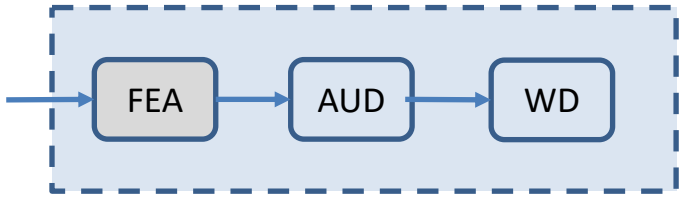
Future work

- Model updates (complicated, still trying)
- Assume some known language model priors
- Assume some known G2P relationship
- Experiments with large WSJ data (Shinji prepared for us)



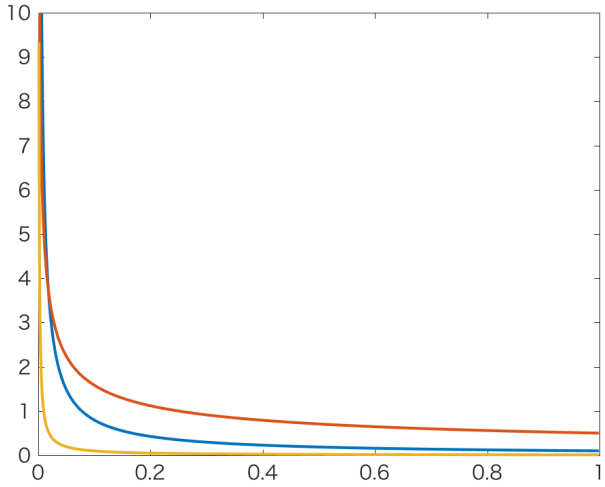
Stochastic pruning

- Heuristic pruning has been long employed in speech and language processing
- But it samples from incorrect posterior, harmful for unsupervised learning..
- Solution: Use Slice sampling (Blunsom&Cohn 2010) to sample from true posterior while pruning!

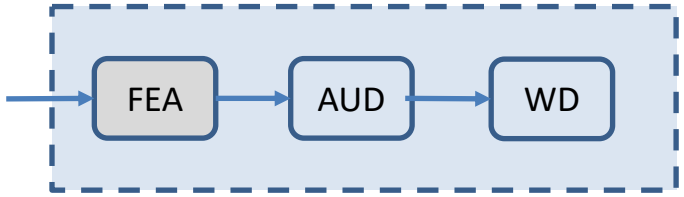


Stochastic pruning

- Recipe: use Beta distribution to dynamically sample each threshold u_a for arc a
- Propagate message of $\log \mathrm{Be}(u_a|\alpha,\beta)$ for unpruned arc instead of -log probability itself
- Why this is correct? → Gave a tutorial on WS

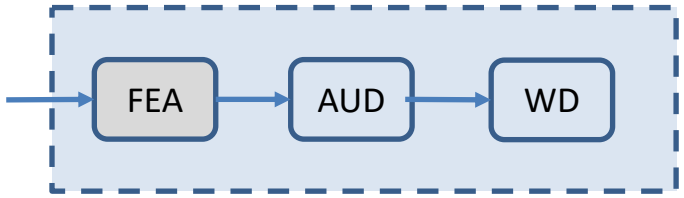


Beta distribution on [0,1)



Stochastic pruning

- Depth-first search to find the previous path used in MCMC (complicated for segment FST)
- Currently: performance comparative to non-pruned version.
  - Why? → Word segmentation WFST is quite sparse, having only a few arcs per node
- Sparsity means we might not have to visit every node, algorithmic techniques will work



Conclusion

Data: 349 349 66 359 359 769 769 837 726 726 726 123 621 263 49 49 49 79 79 329 410 -> [“bd” “d” “b”]

- Now “words” can be directly induced from acoustic features .. YES, WE CAN!
- Connection to other knowledge (dictionary, G2P, ...) is necessary and useful
- More complex acoustic model (eg. triphones and neural HMM) will help much
- Thanks for the great opportunity!