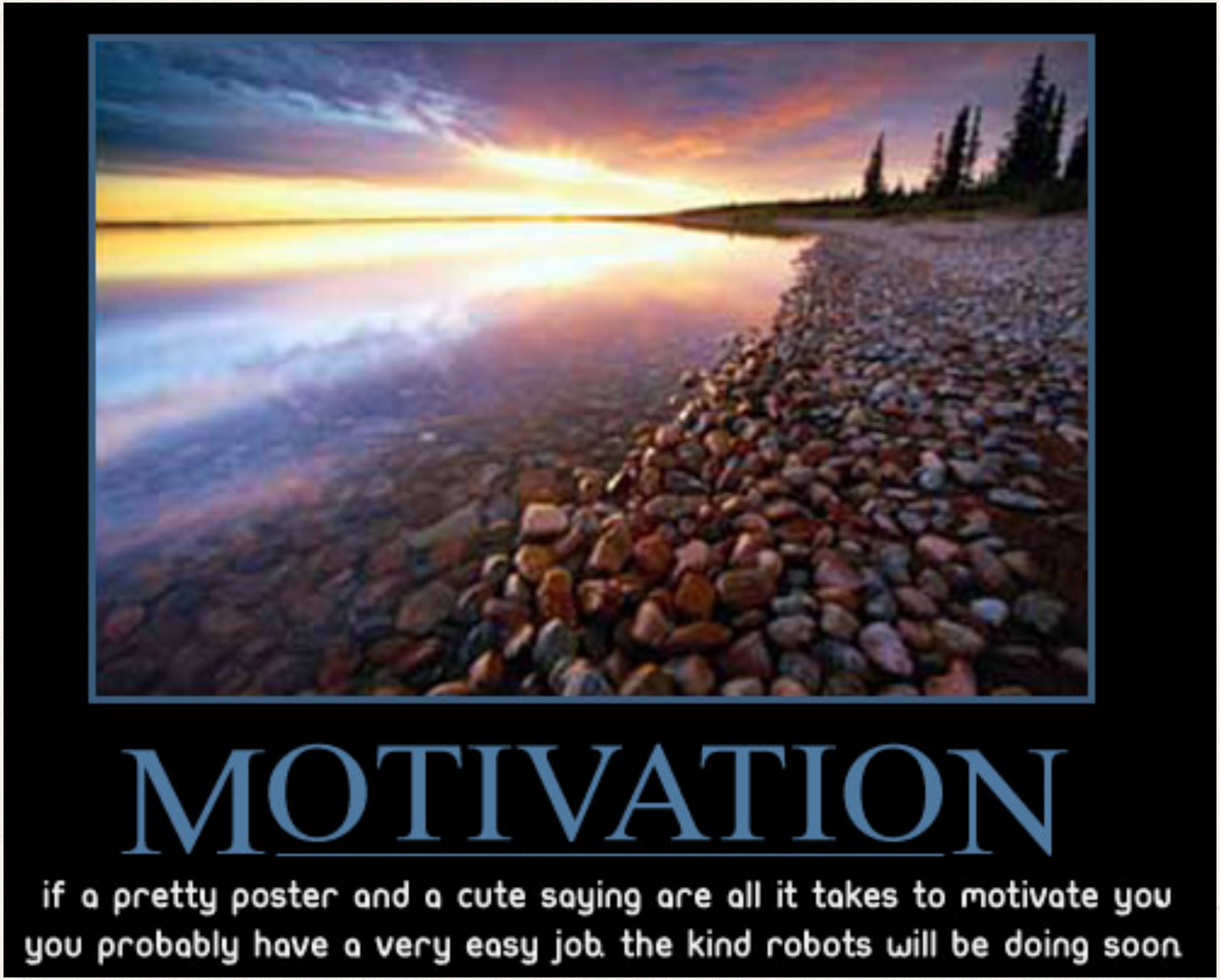
Sequence To Sequence Learning

Logan Mitchell

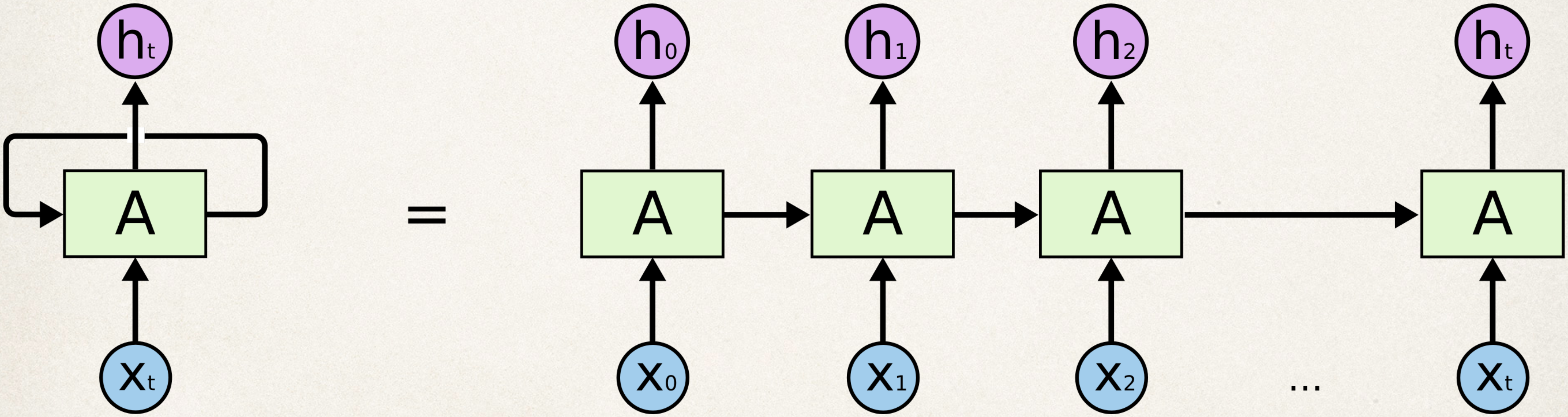
March 29 2016

Motivation


MOTIVATION
if a pretty poster and a cute saying are all it takes to motivate you
you probably have a very easy job. the kind robots will be doing soon.

Sequences

One-One Models



Sequences

Word/Character Prediction

- Predict the next word in a sentence
  - The woman took out _________ purse

Sequences

Part of Speech Tagging

Logan woke up this morning and ate a big bowl of Fruit Loops. On his way to school, a small dog chased after him. Fortunately, Logan's leg had healed and he outran the dog.

Sequences

Part of Speech Tagging

Logan woke up this morning and ate a big bowl of Fruit Loops. On his way to school, a ferocious dog chased after him. Fortunately, Logan's leg had healed and he outran the dog.

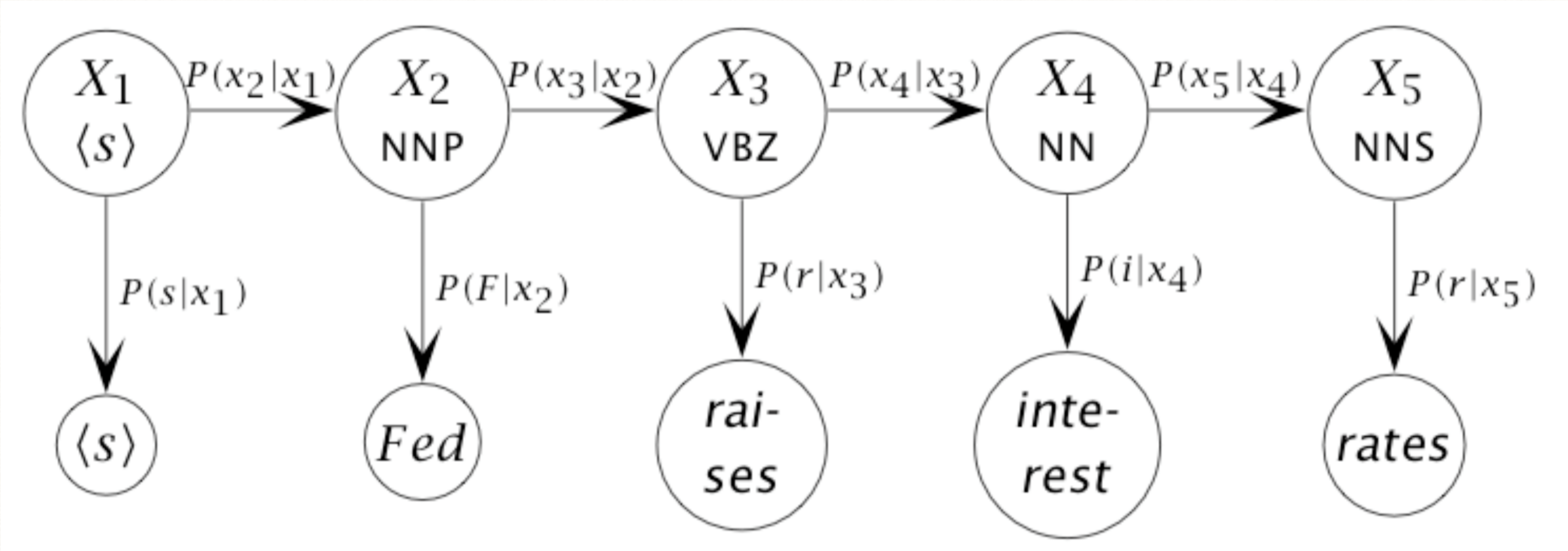
Sequences

Part of Speech Tagging

Logan/NNP woke/VBD up/RP this/DT morning/NN and/CC ate/VB a/DT big/JJ bowl/NN of/IN Fruit/NNP Loops/NNP ./. On/IN his/PRP$ way/NN to/TO school/VB ,/, a/DT ferocious /JJ dog/NN chased/VBN after/IN him/PRP ./. Fortunately/RB ,/, Logan/NNP 's/POS leg/NN had/VBD healed/VBN and/CC he/PRP outran/VB the/DT dog/NN ./.

Sequences

HMM

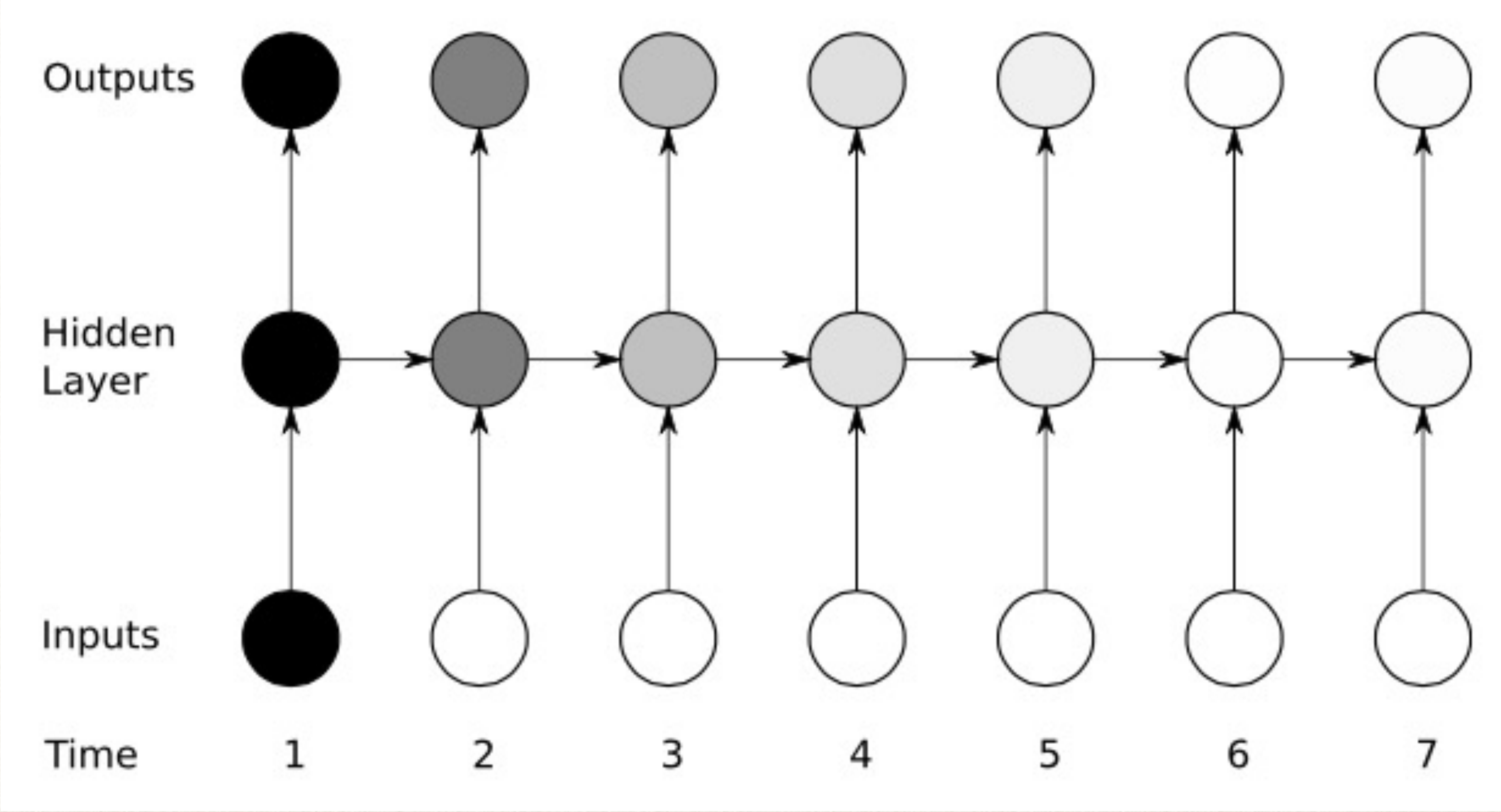


Sequences

CRF

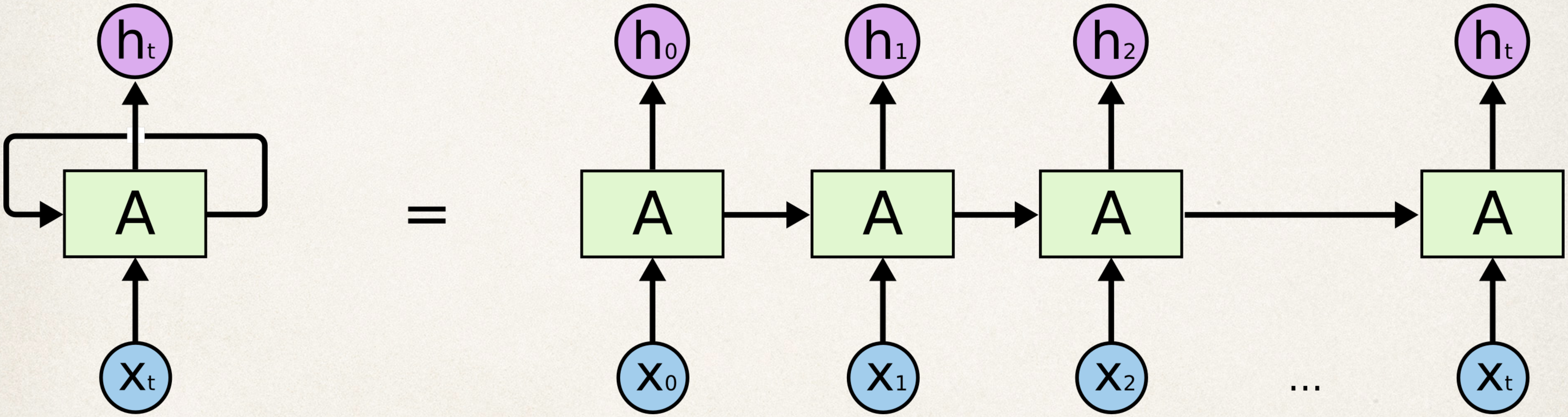
Sequences

Simple RNN



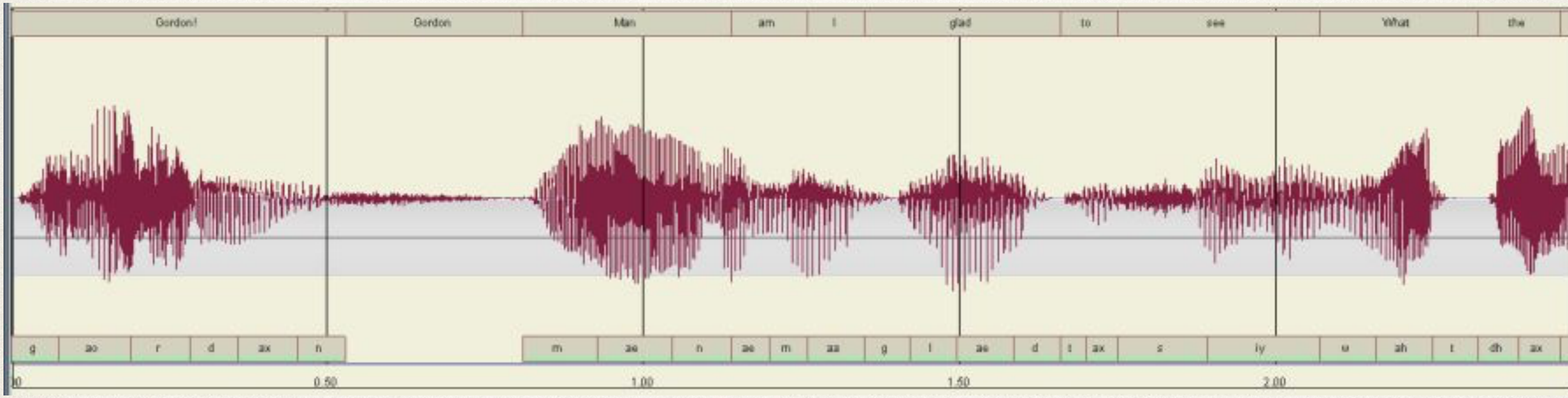
Sequences

One-One Models

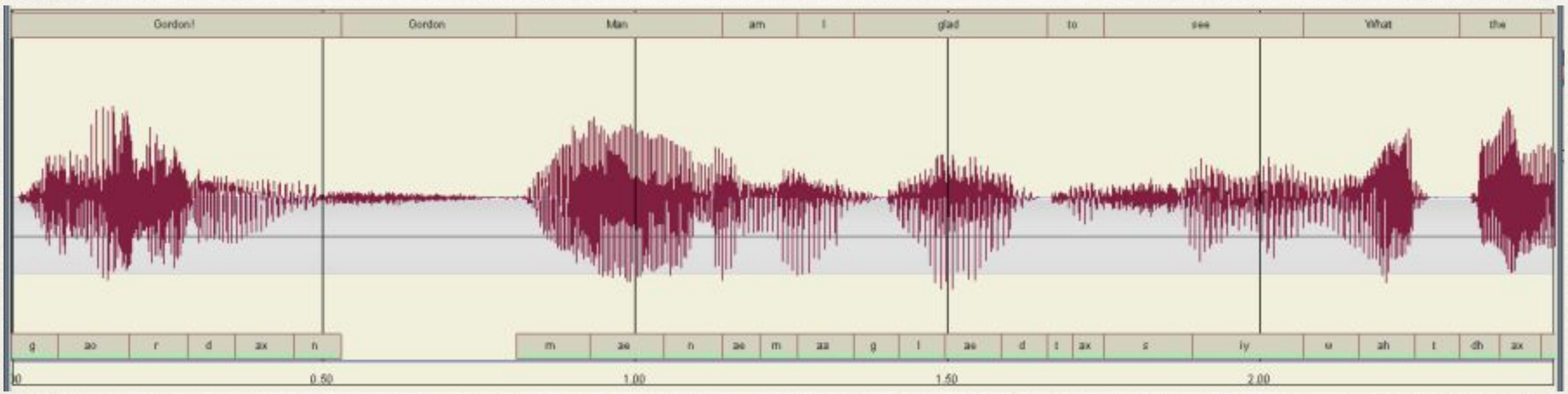


Unsegmented Data

Speech Recognition



Unsegmented Data



- No clear place to segment features
- Label applies to multiple time steps
- Time step where label changes is ambiguous

Speech Recognition - Preprocessing

- Split audio into 10 ms clips (overlapping)
- Create Cepstral Coefficients
  - Convert audio into Frequency Domain
  - Do it again
  - Compute 1st and 2nd derivatives
- Yields 39 features

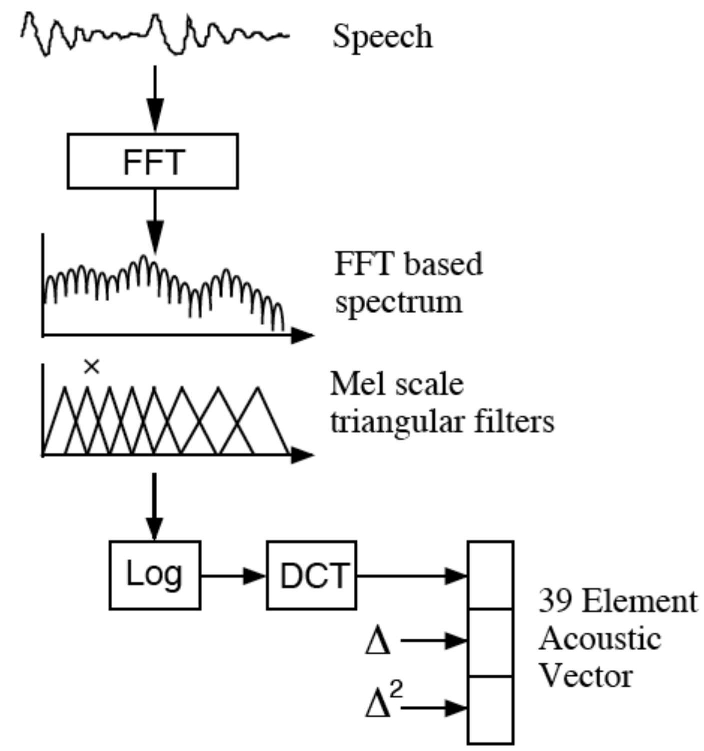
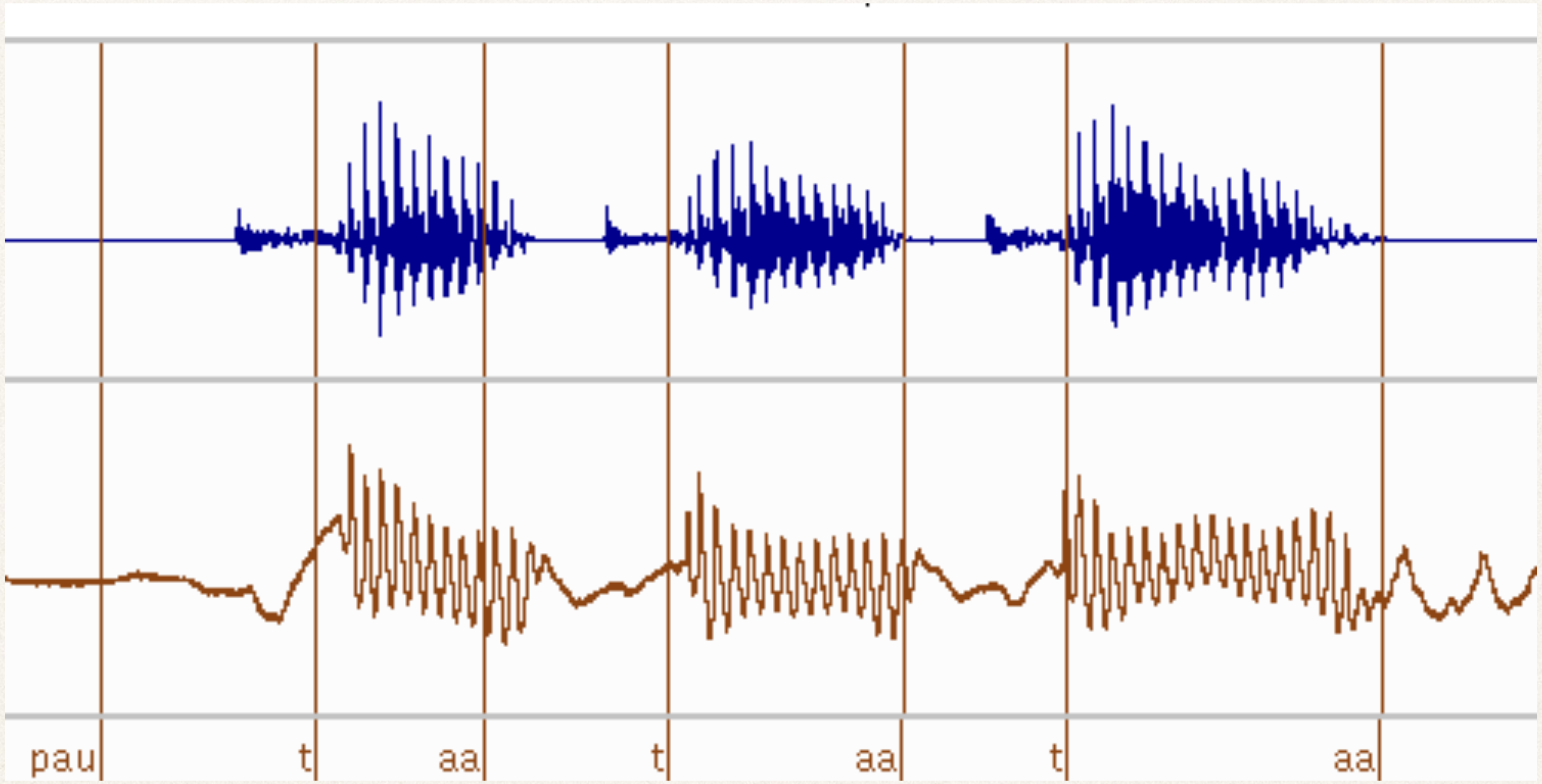


Figure 2: **MFCC-based Front-End Processor.** To perform pattern-matching, the speech waveform must be converted to a sequence of acoustic vectors representing a smoothed log spectrum computed every 10 msecs. Performance is improved by using a non-linear Mel-frequency scale followed by a Discrete Cosine Transform (DCT). The latter has the effect of decorrelating the signal thereby improving assumptions of statistical independence. Finally, first and second differentials are appended to incorporate dynamical information about the signal.

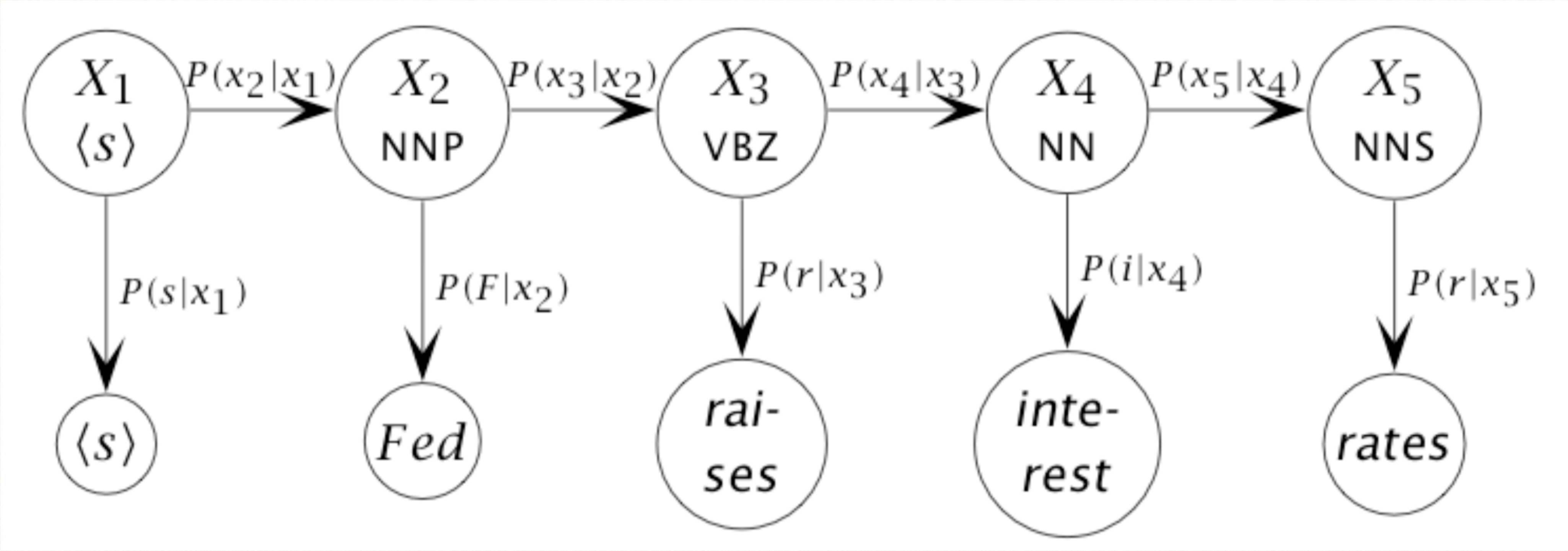
Speech Recognition - Preprocessing



Apply one label per frame

Speech Recognition

HMM

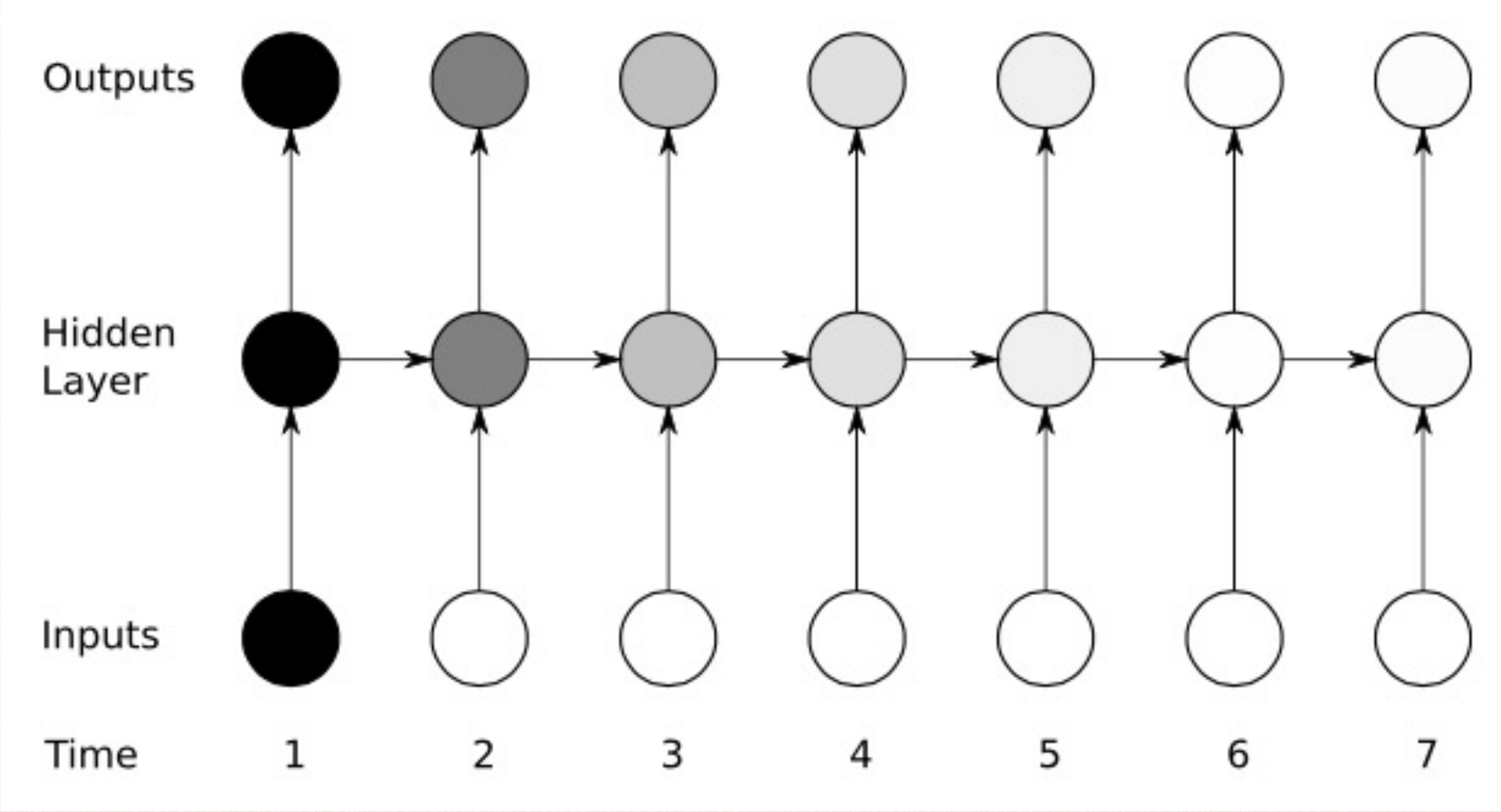


Speech Recognition

CRF

Speech Recognition

Simple RNN



Improvements

Hidden Markov Models

- Three Common Questions
  - What is the probability of a given observation?
  - What is the most probable state sequence of length T given the observation?
  - How can we learn the model parameters?

Forward Algorithm

Forward variable $\alpha_t(i)$ = probability of sub-observation $O_1 \ldots O_t$ and being in S_i at step t

1) Initialization:

$$\alpha_1(i) = \pi_i b_i(O_1), \qquad 1 \leq i \leq N.$$

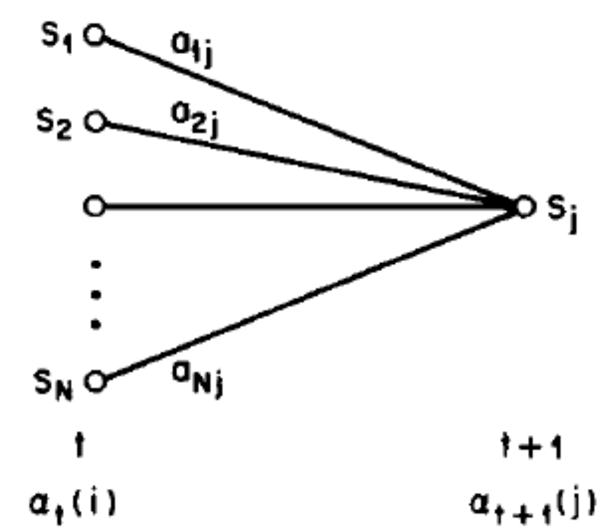
2) Induction:

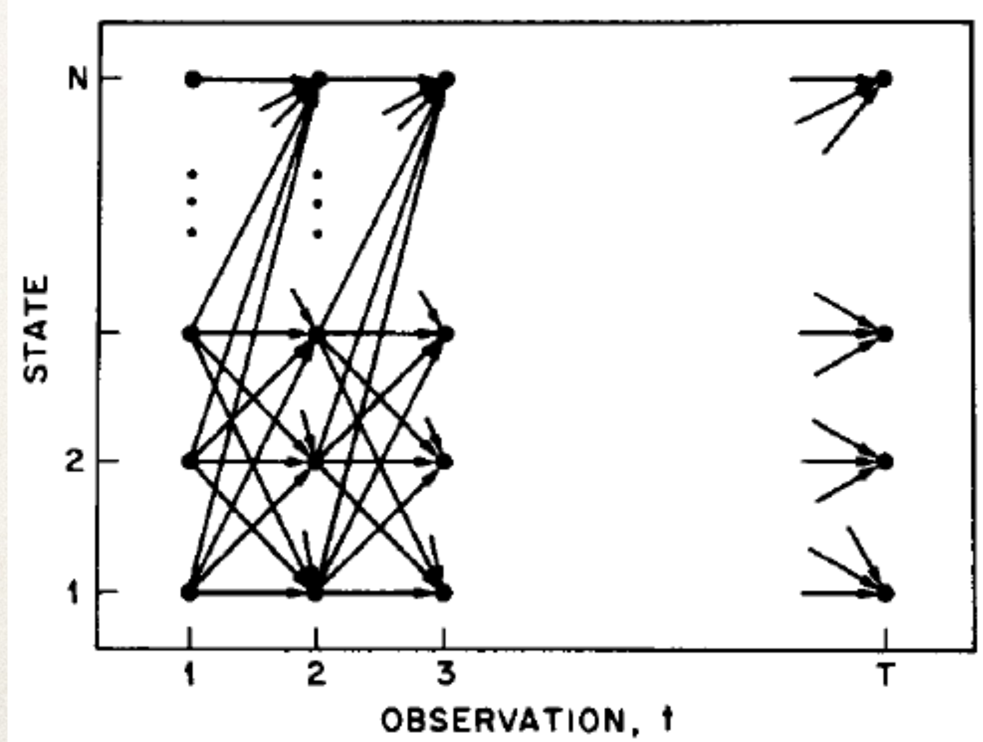
$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij} \right] b_j(O_{t+1}), \qquad 1 \leq t \leq T-1$$

$$1 \leq j \leq N.$$

3) Termination:

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i).$$





Forward Algorithm

$\pi = \{.3, .3, .4\}$

A Transition

0.2	0.5	0.3
0.4	0.4	0.2
0.1	0.4	0.5

B Emission

0.5	0.2	0.3
0.2	0.3	0.5
0.1	0.1	0.8

1) Initialization:

$$\alpha_1(i) = \pi_i b_i(O_1), \quad 1 \leq i \leq N.$$

2) Induction:

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij} \right] b_j(O_{t+1}), \quad 1 \leq t \leq T-1$$

$$1 \leq j \leq N.$$

3) Termination:

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i).$$

What is $P(\text{"F1 F3 F3"} \mid \lambda)$?

	t =1, O_t = F1	t =2, O_t = F3	t =3, O_t = F3
C_1	$.3 \cdot .5 = .15$		
C_2			
C_3			

Forward Algorithm

$$\pi = \{.3, .3, .4\}$$

A Transition

0.2	0.5	0.3
0.4	0.4	0.2
0.1	0.4	0.5

B Emission

0.5	0.2	0.3
0.2	0.3	0.5
0.1	0.1	0.8

1) Initialization:

$$\alpha_1(i) = \pi_i b_i(O_1), \qquad 1 \le i \le N.$$

2) Induction:

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij} \right] b_j(O_{t+1}), \qquad 1 \le t \le T-1$$

$$1 \le j \le N.$$

3) Termination:

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i).$$

What is P("F1 F3 F3" | λ)?

	t =1, O_t = F1	t =2, O_t = F3	t =3, O_t = F3
C_1	$.3 \cdot .5 = .15$		
C_2	$.3 \cdot .2 = .06$		
C_3	$.4 \cdot .1 = .04$		

Forward Algorithm

A Transition

0.2	0.5	0.3
0.4	0.4	0.2
0.1	0.4	0.5

B Emission

0.5	0.2	0.3
0.2	0.3	0.5
0.1	0.1	0.8

$\pi = \{.3, .3, .4\}$

What is P("F1 F3 F3" | λ)?

1) Initialization:

$$\alpha_1(i) = \pi_i b_i(O_1), \qquad 1 \le i \le N.$$

2) Induction:

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j(O_{t+1}), \qquad 1 \le t \le T-1$$

$$1 \le j \le N.$$

3) Termination:

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i).$$

	t =1, O_t = F1	t =2, O_t = F3	t =3, O_t = F3
C_1	.3 · .5 = .15	(.15·.2 + .06·.4 + .04·.1)·.3 = .017	
C_2	.3 · .2 = .06	(.15·.5 + .06·.4 + .04·.4)·.5 = .058	
C_3	.4 · .1 = .04	(.15·.3 + .06·.2 + .04·.5)·.8 = .062	

Forward Algorithm

A Transition

0.2	0.5	0.3
0.4	0.4	0.2
0.1	0.4	0.5

B Emission

0.5	0.2	0.3
0.2	0.3	0.5
0.1	0.1	0.8

$\pi = \{.3, .3, .4\}$

1) Initialization:

$$\alpha_1(i) = \pi_i b_i(O_1), \qquad 1 \le i \le N.$$

2) Induction:

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j(O_{t+1}), \qquad 1 \le t \le T-1$$

$$1 \le j \le N.$$

3) Termination:

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i).$$

What is $P(\text{"F1 F3 F3"} \mid \lambda)$?

$.010 + .028 + .038 = .076$

	$t=1, O_t = \text{F1}$	$t=2, O_t = \text{F3}$	$t=3, O_t = \text{F3}$
C_1	$.3 \cdot .5 = .15$	$(.15{\cdot}.2 + .06{\cdot}.4 + .04{\cdot}.1){\cdot}.3 = .017$	$(.017{\cdot}.2 + .058{\cdot}.4 + .062{\cdot}.1){\cdot}.3 = .010$
C_2	$.3 \cdot .2 = .06$	$(.15{\cdot}.5 + .06{\cdot}.4 + .04{\cdot}.4){\cdot}.5 = .058$	$(.017{\cdot}.5 + .058{\cdot}.4 + .062{\cdot}.4){\cdot}.5 = .028$
C_3	$.4 \cdot .1 = .04$	$(.15{\cdot}.3 + .06{\cdot}.2 + .04{\cdot}.5){\cdot}.8 = .062$	$(.017{\cdot}.3 + .058{\cdot}.2 + .062{\cdot}.5){\cdot}.8 = .038$

Backward Algorithm

Backward variable is the counterpart to forward variable $\alpha_t(i)$

$\beta_t(i)$ = probability of sub-observation $O_{t+1} \ldots O_T$ when starting from S_i at step t

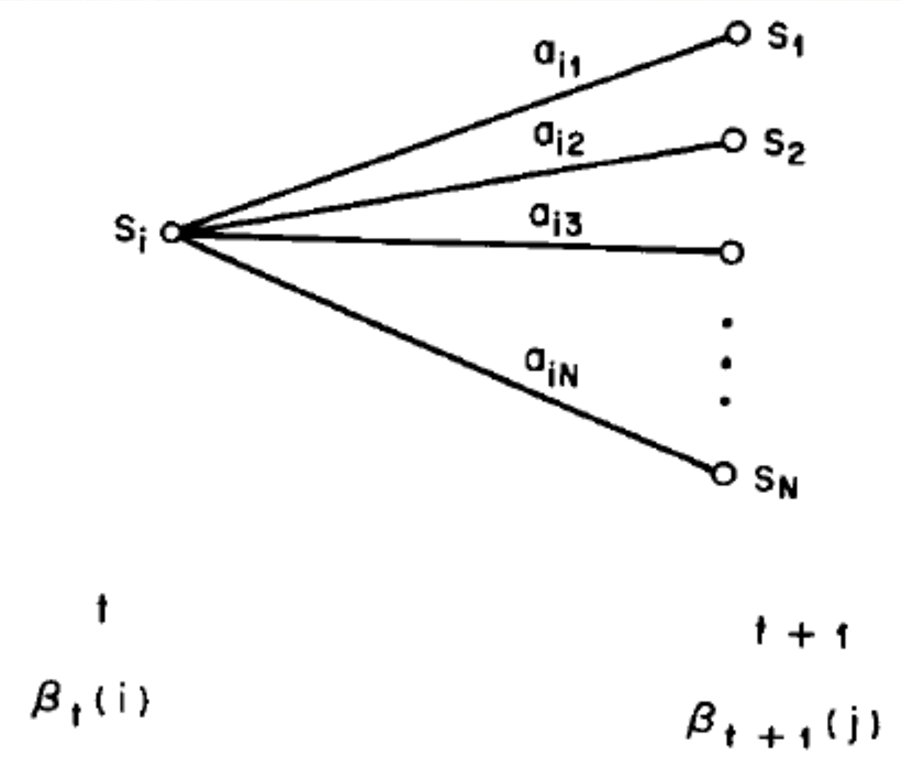
1) Initialization:

$$\beta_T(i) = 1, \qquad 1 \leq i \leq N.$$

2) Induction:

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1})\, \beta_{t+1}(j),$$

$$t = T-1, T-2, \cdots, 1, \; 1 \leq i \leq N.$$



Backward Algorithm

$\pi = \{.3, .3, .4\}$

A Transition

0.2	0.5	0.3
0.4	0.4	0.2
0.1	0.4	0.5

B Emission

0.5	0.2	0.3
0.2	0.3	0.5
0.1	0.1	0.8

1) Initialization:

$$\beta_T(i) = 1, \qquad 1 \le i \le N.$$

2) Induction:

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1})\, \beta_{t+1}(j),$$

$$t = T-1, T-2, \cdots, 1, \; 1 \le i \le N.$$

What is $P(\text{"F1 F3 F3"} \mid \lambda)$?

	t =1, O_{t+1} = F3	t =2, O_{t+1} = F3	T=3
C_1		$.2 \cdot .3 \cdot 1 + .5 \cdot .5 \cdot 1 + .3 \cdot .8 \cdot 1 = .55$	1
C_2			1
C_3			1

Backward Algorithm

$\pi = \{.3, .3, .4\}$

A Transition

0.2	0.5	0.3
0.4	0.4	0.2
0.1	0.4	0.5

B Emission

0.5	0.2	0.3
0.2	0.3	0.5
0.1	0.1	0.8

1) Initialization:

$$\beta_T(i) = 1, \quad 1 \leq i \leq N.$$

2) Induction:

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1})\, \beta_{t+1}(j),$$

$$t = T-1, T-2, \cdots, 1, \; 1 \leq i \leq N.$$

What is $P(\text{"F1 F3 F3"} \mid \lambda)$?

	$t = 1$, O_{t+1} = F3	$t = 2$, O_{t+1} = F3	T=3
C_1		$.2 \cdot .3 \cdot 1 + .5 \cdot .5 \cdot 1 + .3 \cdot .8 \cdot 1 = .55$	1
C_2		$.4 \cdot .3 \cdot 1 + .4 \cdot .5 \cdot 1 + .2 \cdot .8 \cdot 1 = .48$	1
C_3		$.1 \cdot .3 \cdot 1 + .4 \cdot .5 \cdot 1 + .5 \cdot .8 \cdot 1 = .63$	1

Backward Algorithm

$\pi = \{.3, .3, .4\}$

A Transition

0.2	0.5	0.3
0.4	0.4	0.2
0.1	0.4	0.5

B Emission

0.5	0.2	0.3
0.2	0.3	0.5
0.1	0.1	0.8

1) Initialization:

$$\beta_T(i) = 1, \quad 1 \le i \le N.$$

2) Induction:

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1})\, \beta_{t+1}(j),$$

$$t = T-1, T-2, \cdots, 1, 1 \le i \le N.$$

What is $P(\text{"F1 F3 F3"} \mid \lambda)$?

$\Sigma \pi_i b_i(O_1)\, \beta_t(i) =$

$.3 \cdot .30 \cdot .5 + .3 \cdot .26 \cdot .2 + .4 \cdot .36 \cdot .1 =$

$.045 + .016 + .014 = .076$

	t =1, O_{t+1} = F3	t =2, O_{t+1} = F3	T=3
C_1	$.2 \cdot .3 \cdot .55 + .5 \cdot .5 \cdot .48 + .3 \cdot .8 \cdot .63 = .30$	$.2 \cdot .3 \cdot 1 + .5 \cdot .5 \cdot 1 + .3 \cdot .8 \cdot 1 = .55$	1
C_2	$.4 \cdot .3 \cdot .55 + .4 \cdot .5 \cdot .48 + .2 \cdot .8 \cdot .63 = .26$	$.4 \cdot .3 \cdot 1 + .4 \cdot .5 \cdot 1 + .2 \cdot .8 \cdot 1 = .48$	1
C_3	$.1 \cdot .3 \cdot .55 + .4 \cdot .5 \cdot .48 + .5 \cdot .8 \cdot .63 = .36$	$.1 \cdot .3 \cdot 1 + .4 \cdot .5 \cdot 1 + .5 \cdot .8 \cdot 1 = .63$	1

Viterbi Algorithm

Find the most probable sequence given the observation

It is the exact same as the forward algorithm except that we take the max at each time step rather than the sum.

1) Initialization:

$$\delta_1(i) = \pi_i b_i(O_1), \qquad 1 \le i \le N$$

$$\psi_1(i) = 0.$$

2) Recursion:

$$\delta_t(j) = \max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}] b_j(O_t), \qquad 2 \le t \le T, \quad 1 \le j \le N$$

$$\psi_t(j) = \operatorname*{argmax}_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}], \qquad 2 \le t \le T, \quad 1 \le j \le N.$$

3) Termination:

$$P^* = \max_{1 \le i \le N} [\delta_T(i)]$$

$$q_T^* = \operatorname*{argmax}_{1 \le i \le N} [\delta_T(i)].$$

4) Path (state sequence) backtracking:

$$q_t^* = \psi_{t+1}(q_{t+1}^*), \qquad t = T-1, T-2, \cdots, 1.$$

Viterbi Algorithm

$\pi = \{.3, .3, .4\}$

A Transition

0.2	0.5	0.3
0.4	0.4	0.2
0.1	0.4	0.5

B Emission

0.5	0.2	0.3
0.2	0.3	0.5
0.1	0.1	0.8

1) Initialization:

$$\delta_1(i) = \pi_i b_i(O_1), \qquad 1 \le i \le N$$

$$\psi_1(i) = 0.$$

2) Recursion:

$$\delta_t(j) = \max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}] b_j(O_t), \qquad 2 \le t \le T, \quad 1 \le j \le N$$

$$\psi_t(j) = \operatorname*{argmax}_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}], \qquad 2 \le t \le T, \quad 1 \le j \le N.$$

3) Termination:

$$P^* = \max_{1 \le i \le N} [\delta_T(i)]$$

$$q_T^* = \operatorname*{argmax}_{1 \le i \le N} [\delta_T(i)].$$

4) Path (state sequence) backtracking:

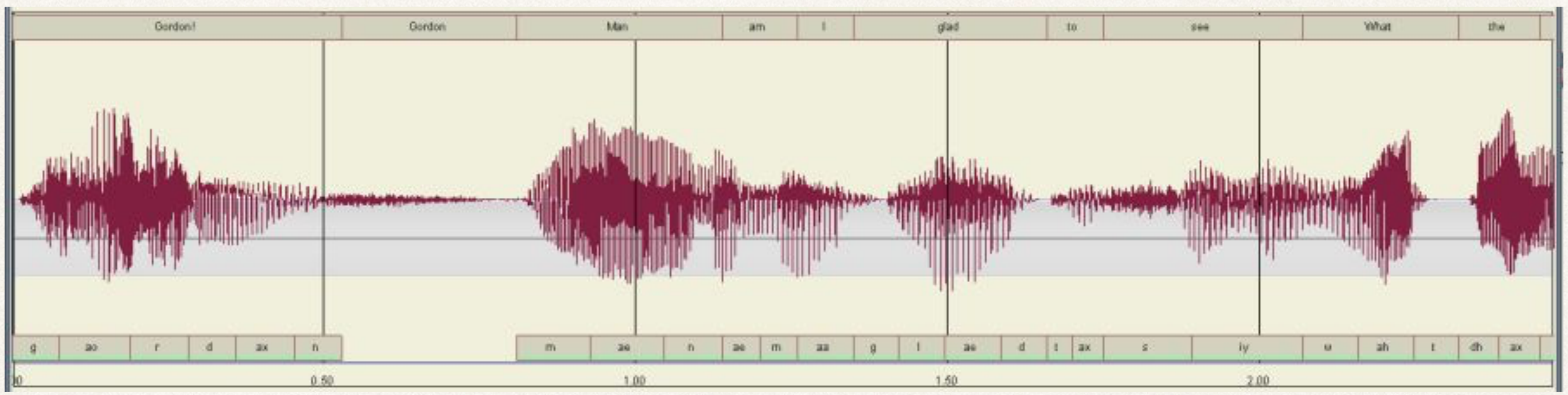
$$q_t^* = \psi_{t+1}(q_{t+1}^*), \qquad t = T-1, T-2, \cdots, 1.$$

What is most probable state sequence given "F1 F3 F3" and λ?

C1, C3, C3

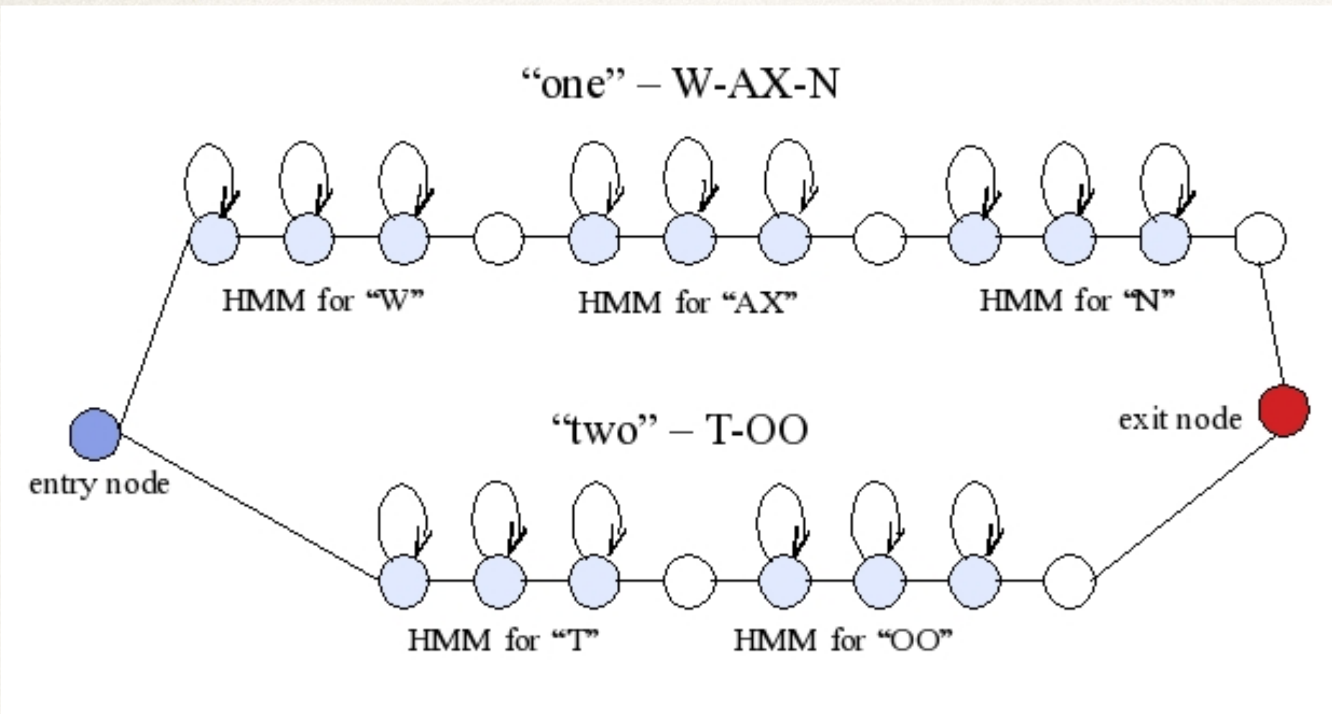
	$t=1, O_t = \text{F1}$	$t=2, O_t = \text{F3}$	$t=3, O_t = \text{F3}$
C_1	$.3 \cdot .5 = .15$	$\max(\underline{.15 \cdot .2}, .06 \cdot .4, .04 \cdot .1) \cdot .3 = .009$	$\max(.009 \cdot .2, \underline{.038 \cdot .4}, .036 \cdot .1) \cdot .3 = .0046$
C_2	$.3 \cdot .2 = .06$	$\max(\underline{.15 \cdot .5}, .06 \cdot .4, .04 \cdot .4) \cdot .5 = .038$	$\max(.009 \cdot .5, \underline{.038 \cdot .4}, .036 \cdot .4) \cdot .5 = .0076$
C_3	$.4 \cdot .1 = .04$	$\max(\underline{.15 \cdot .3}, .06 \cdot .2, .04 \cdot .5) \cdot .8 = .036$	$\max(.009 \cdot .3, .038 \cdot .2, \underline{.036 \cdot .5}) \cdot .8 = \underline{.014}$

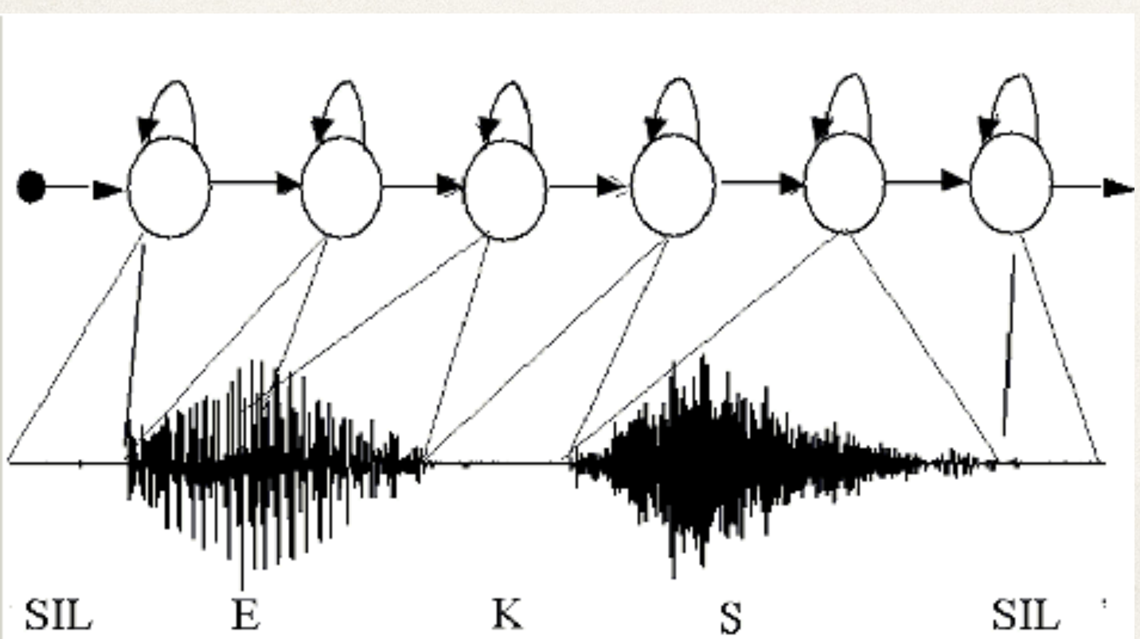
Hidden Markov Models



- How does this help with speech recognition?
- Why don't we model each phoneme with a separate HMM?
- Use beam search with Viterbi to select appropriate phoneme sequence.

Hidden Markov Models





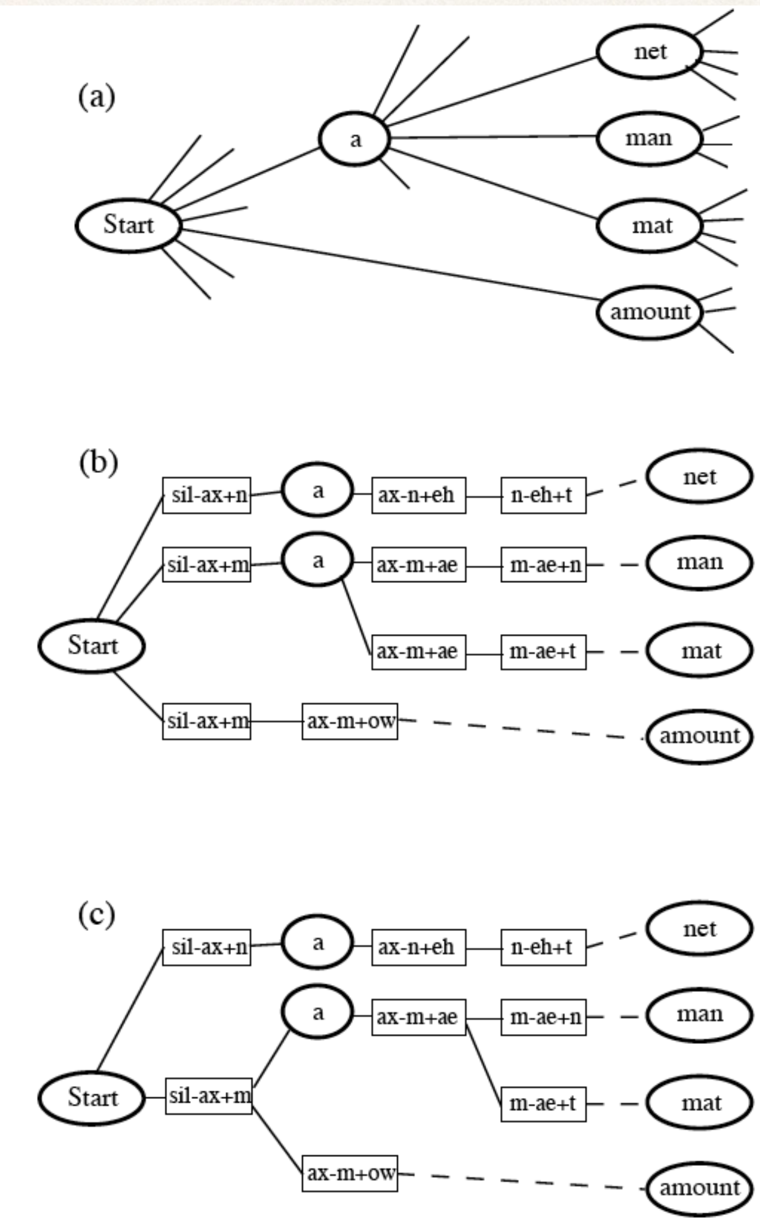
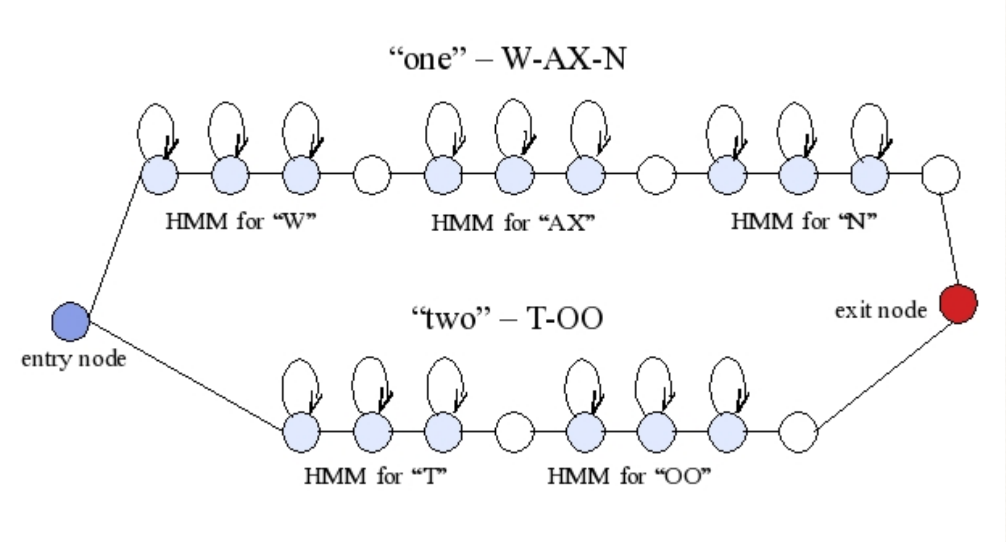
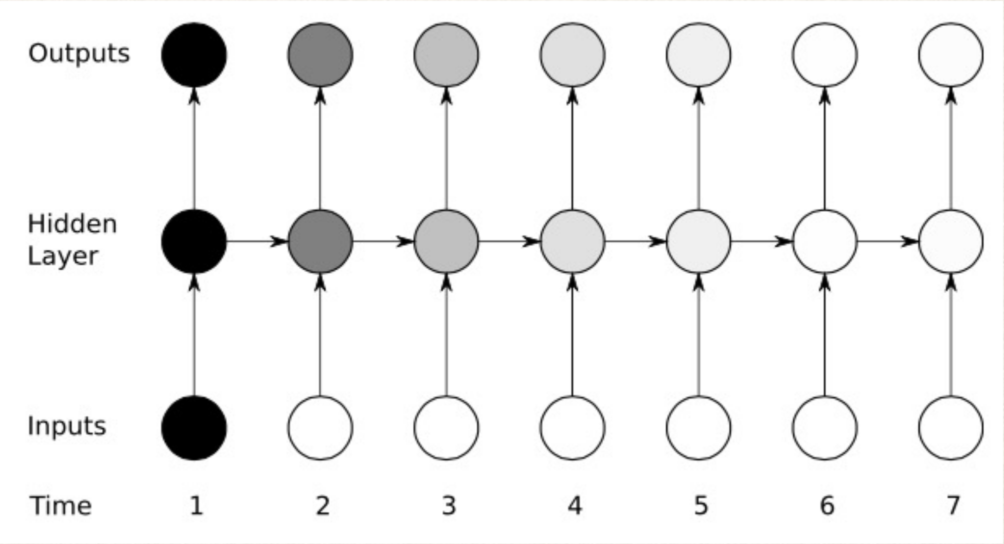


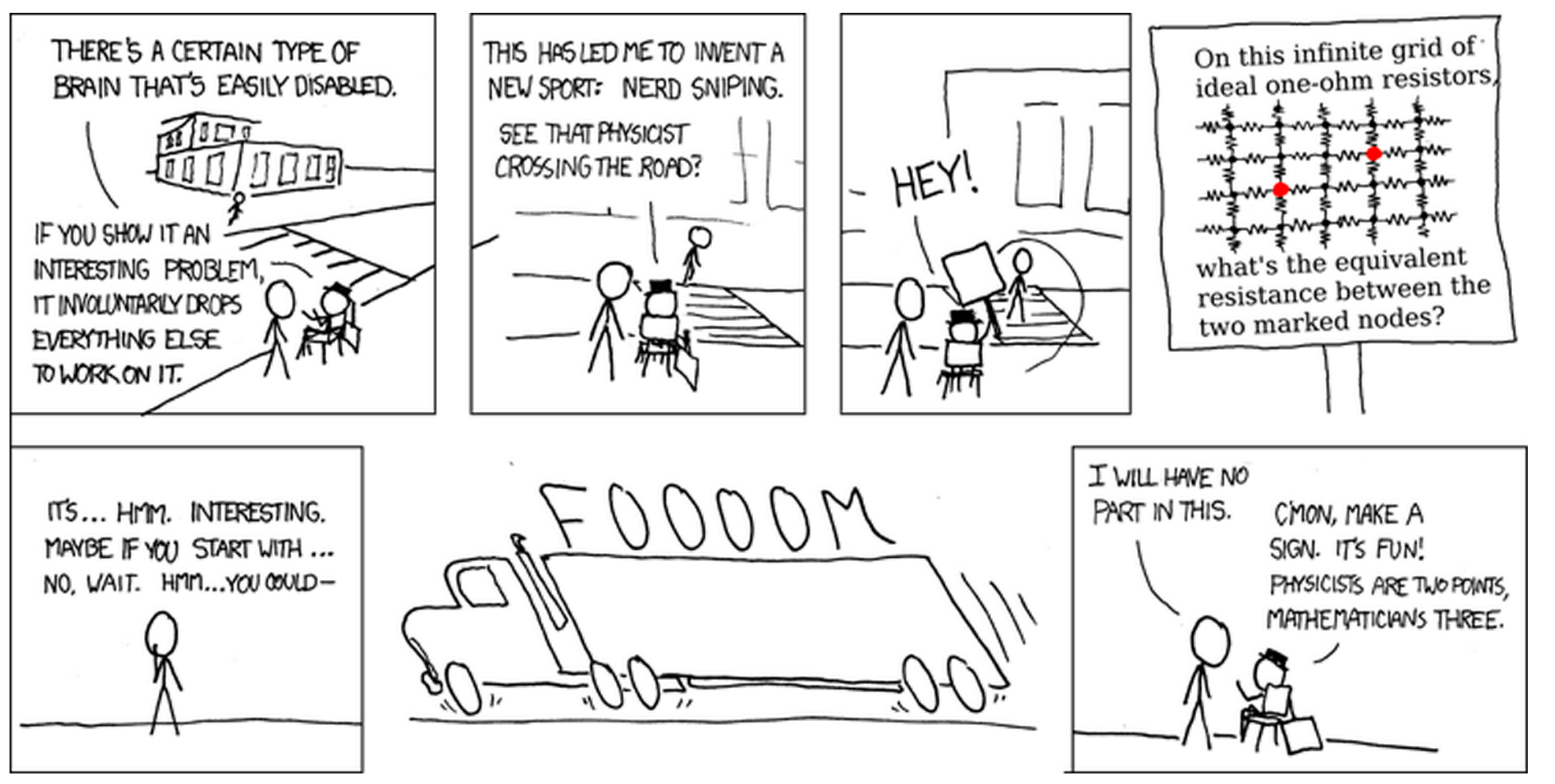
Figure 6: **Fragment of Decoder Network.** In principle, the decoder searches through a network representing all possible word sequences. In practice, only paths corresponding to the most likely word sequences are constructed. Part (a) shows the directed network of words which the recogniser is considering initially. Part (b) shows the same network decomposed into triphones. Note that in order to take account of cross-word context, the first /ax/ sound has to be replicated and the word a duplicated. Part (c) shows that tree-structuring the network can reduce the size of the network.

Hybrid RNNs

- RNNs typically output at each time step
- Could use HMMs on top to generate labelling
- Pros/Cons?





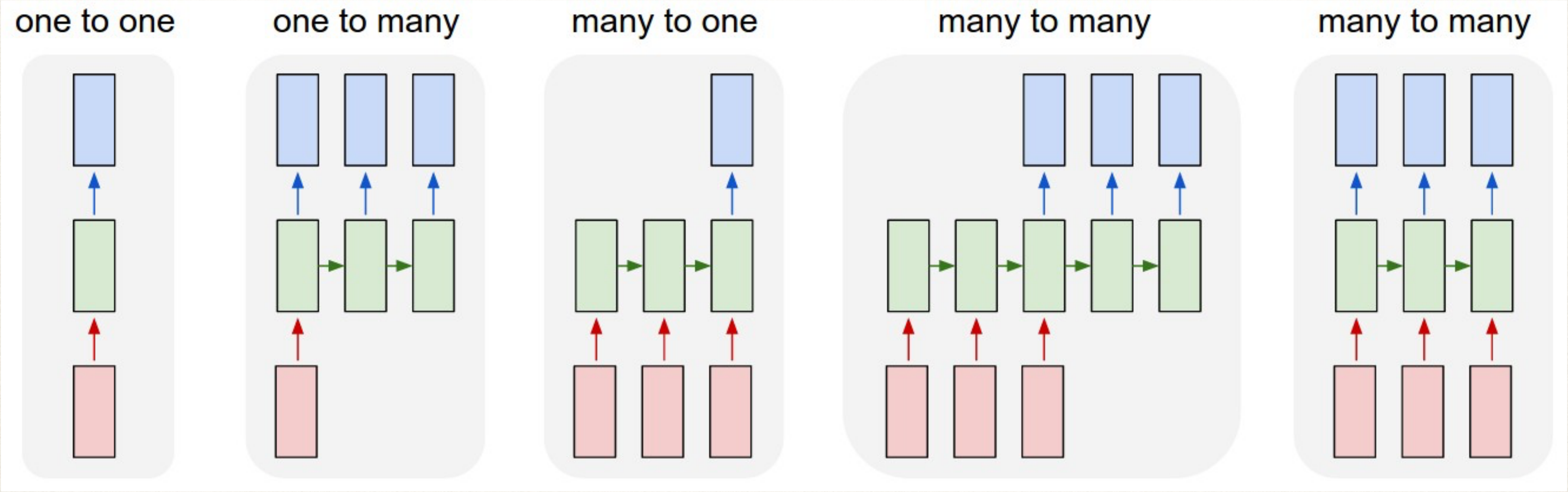


We Begin Again…

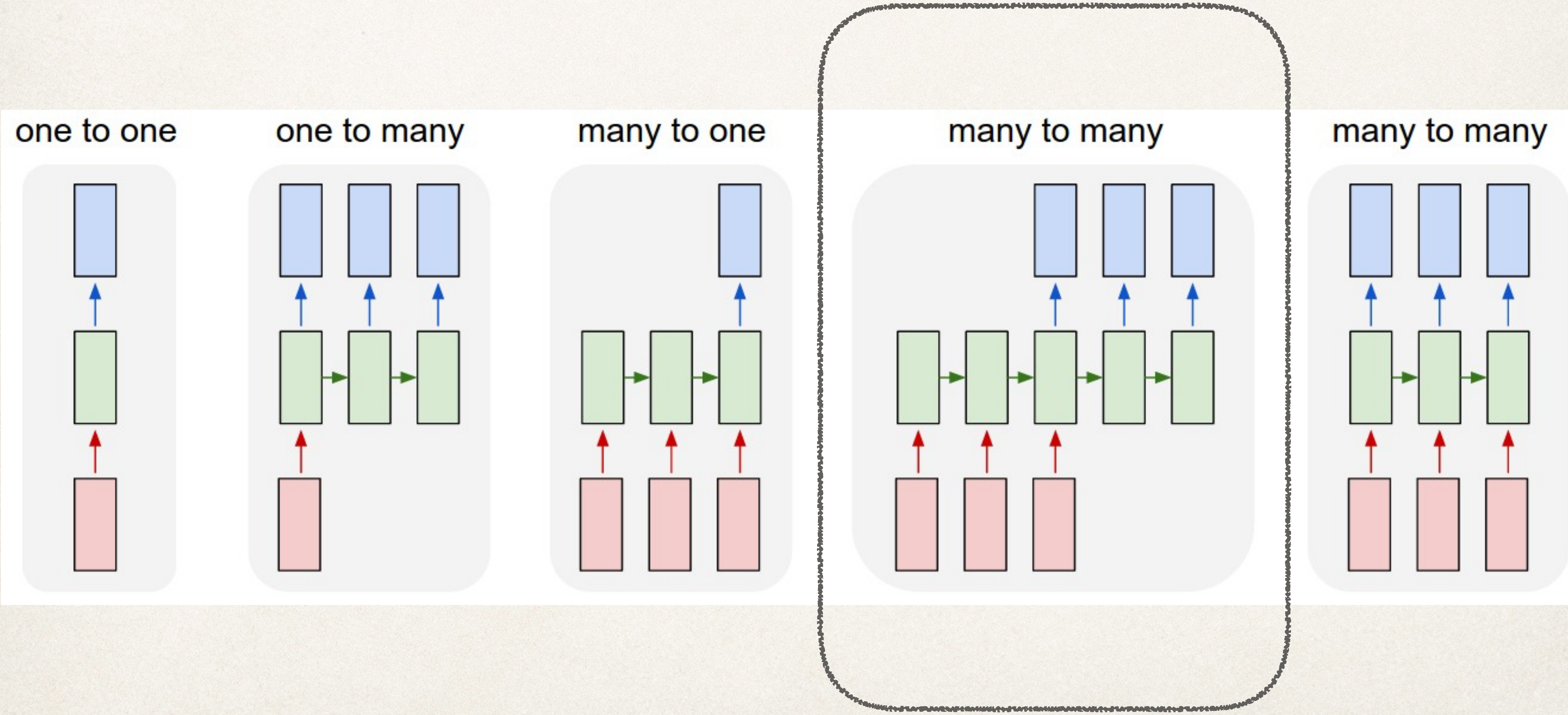
Connectionist Temporal Classification

4/5/16

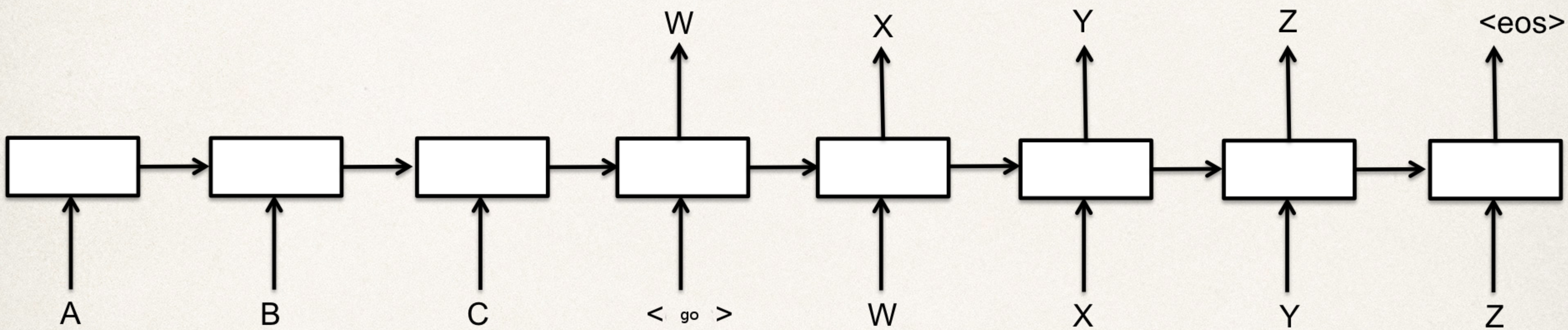
Model Types



Model Types



Translation Models



- Uses 2 RNNs
- State of first RNN initializes second
- Inputs fed into first RNN
- Can generate arbitrary length sequences

Translation Models

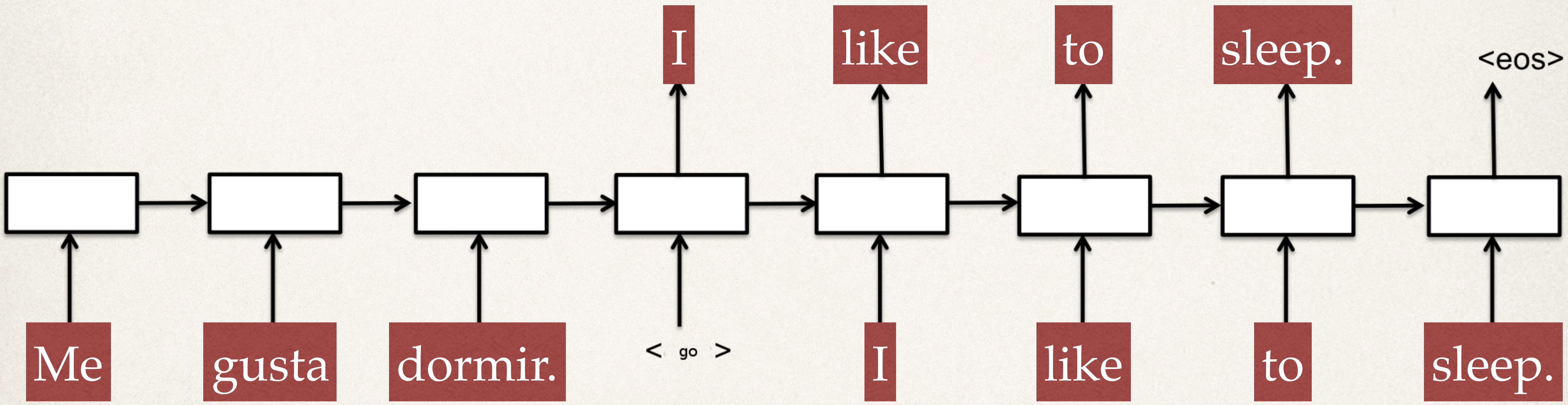
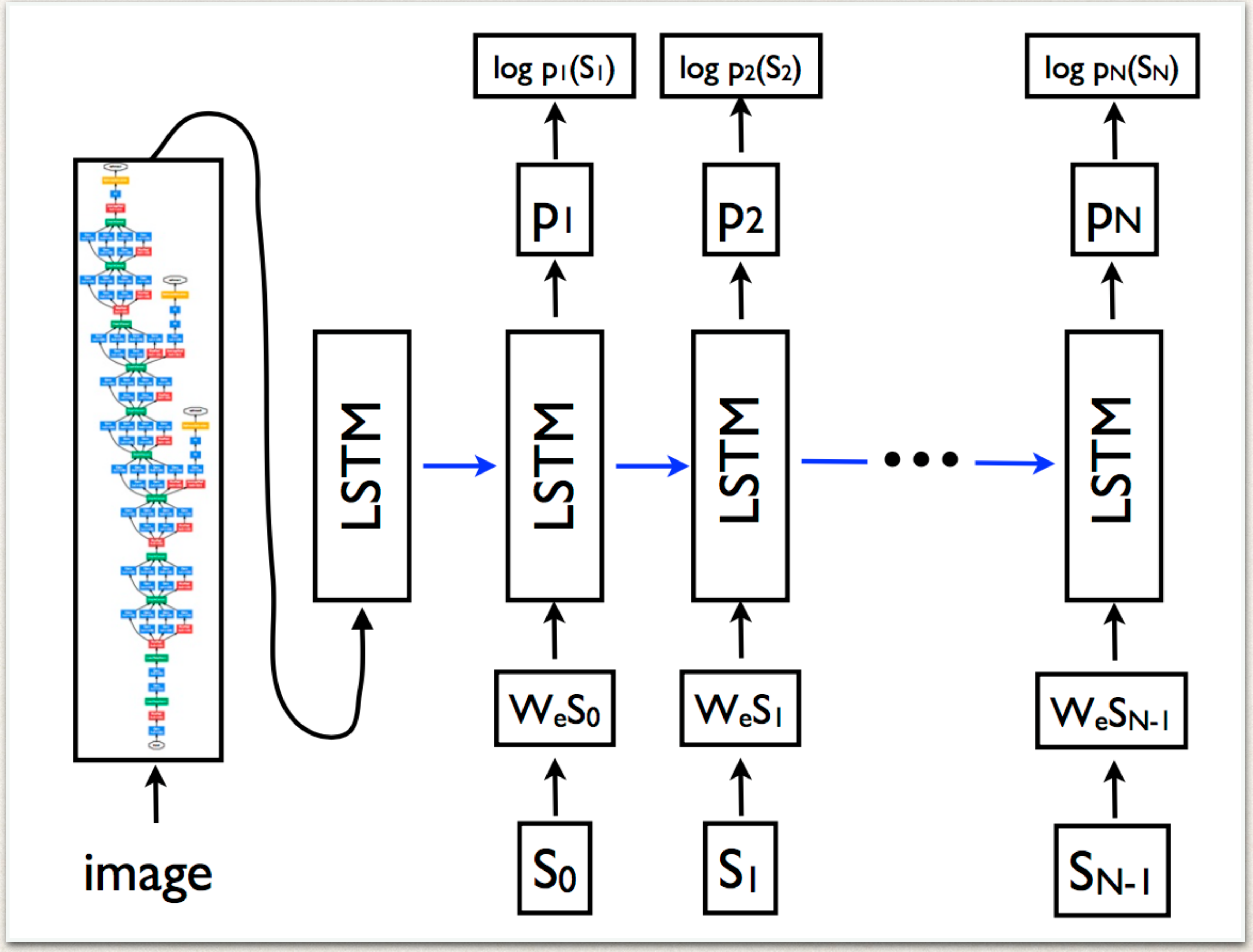
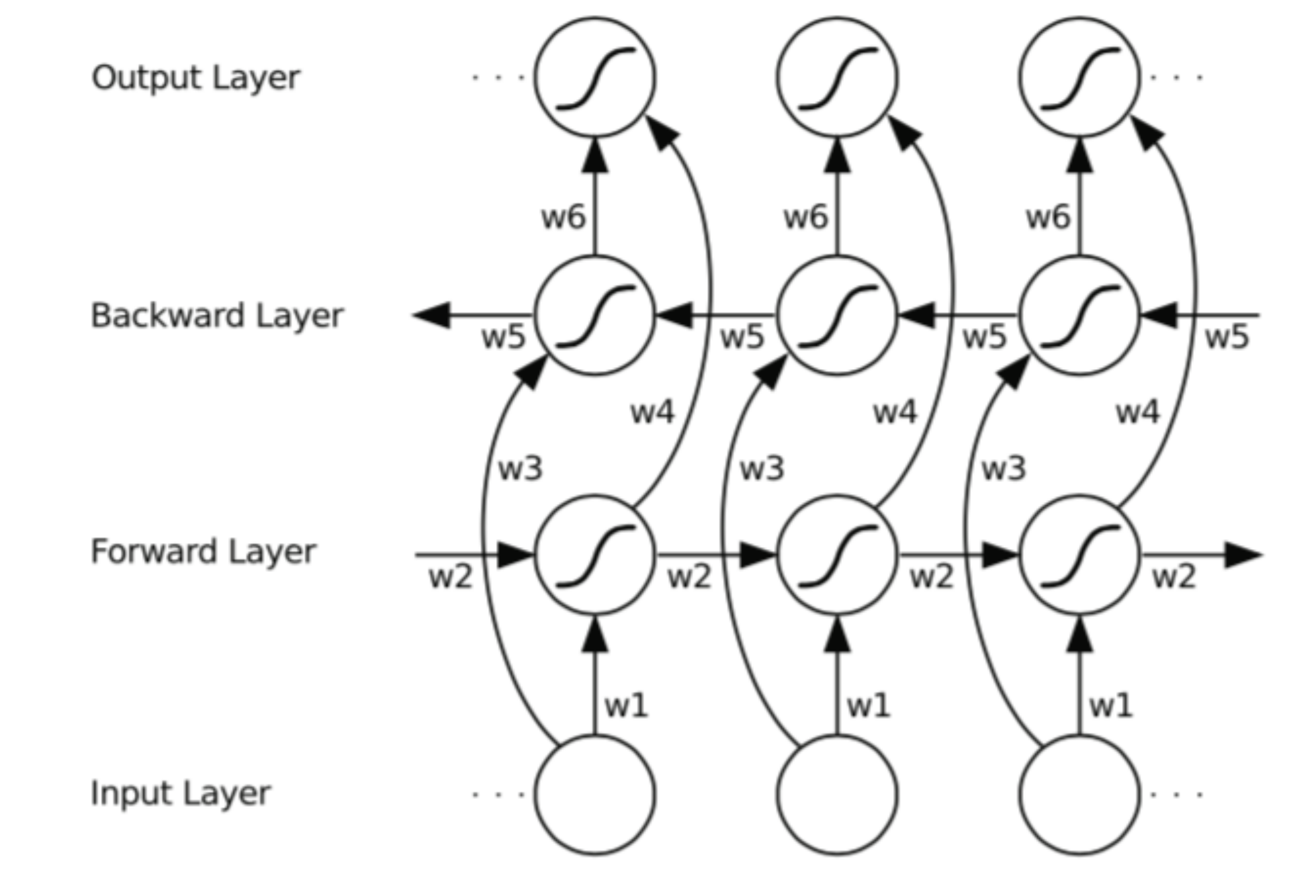


Image Captioning



Connectionist Temporal Classification

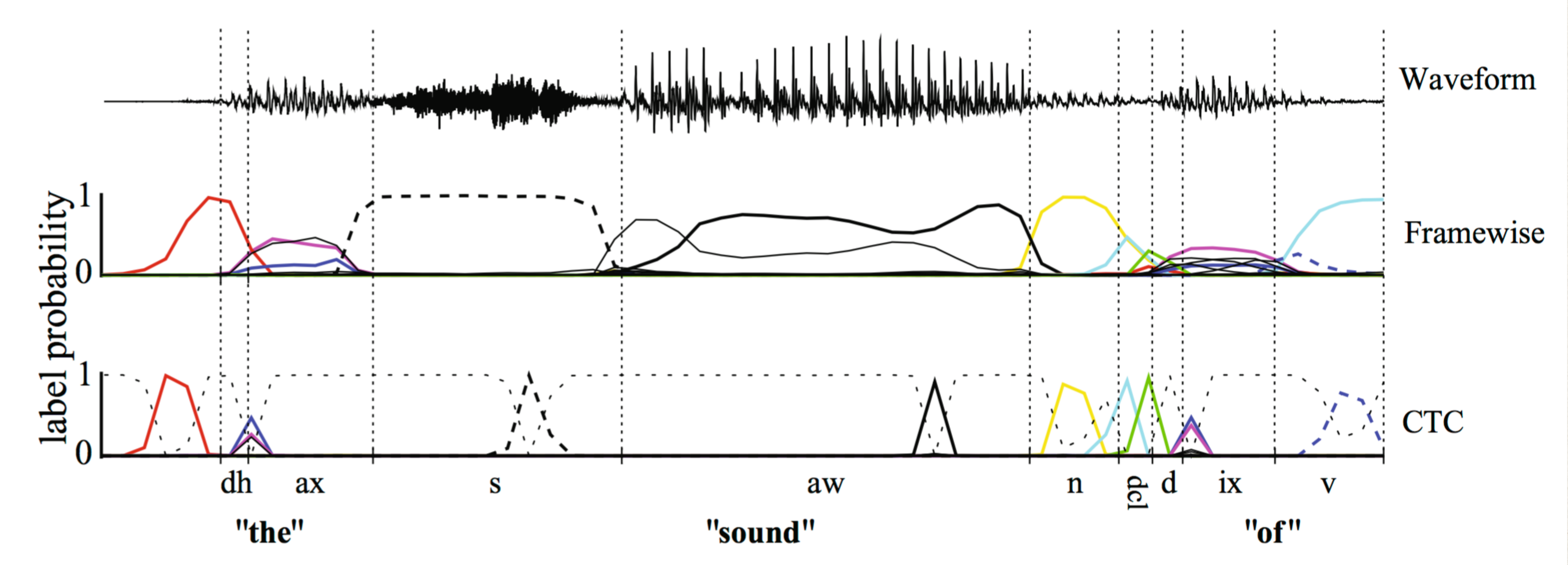
CTC



- Network Structure
    - Any RNN structure
    - Top layer is a softmax
        - No recurrent connection on output layer
- CTC Loss enables training the network end to end.

CTC

Classification Output

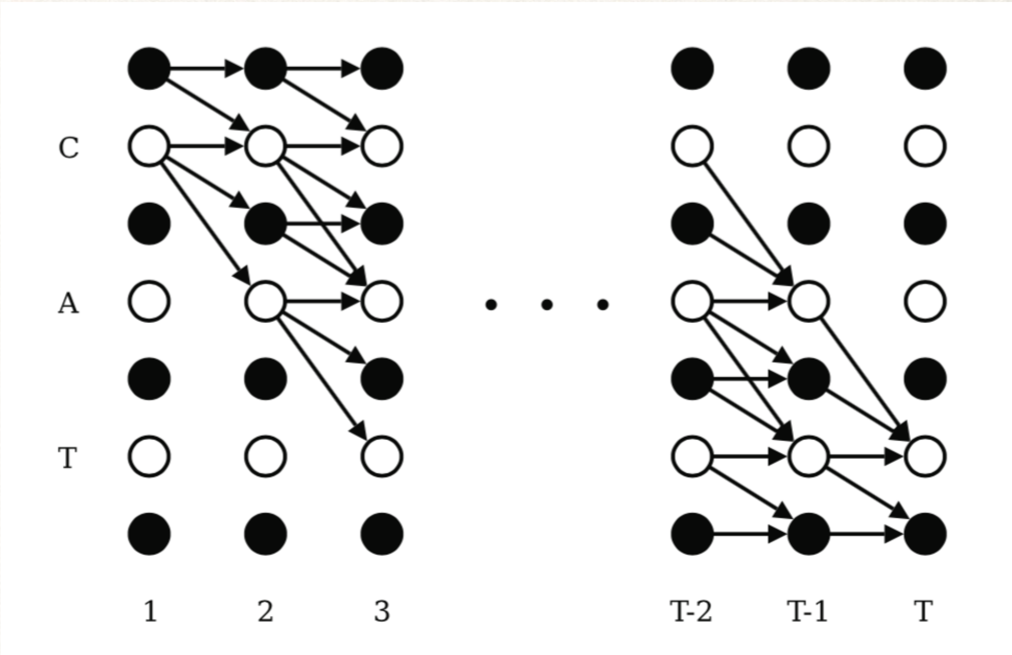


CTC - Alphabet

- Let us assume an Alphabet A
- We define an alphabet $A' = \{A \cup \_\}$
- The output layer will contain $|A'|$ nodes.
- Lets have $A' = \{\text{a}, c, t, \_\}$

CTC - Paths

- For a sequence of length T time
- A path $\pi \in A'^T$
- For $T = 3$:
  - $\pi = (c,a,\_), (a,c,t), (a,t,c), (\_,\_,\_)$, etc



CTC - Path Probabilities

- For $T = 1$
- $A'^T = \{(c),(a),(t),(\_)\}$
- Given some network output $\boldsymbol{y} = N(\boldsymbol{x})$ for some input $\boldsymbol{x}$, what is the probability of some π?

CTC - Path Probabilities

- For $T = 1$

- $A'^T = \{(c),(a),(t),(\_)\}$

- Given some network output $\boldsymbol{y} = N(\boldsymbol{x})$ for some input $\boldsymbol{x}$, what is the probability of some π?

$$p(\pi \mid \boldsymbol{x}) = \boldsymbol{y}_\pi$$

CTC - Path Probabilities

- For any T
- $\pi = \{(a,a,a), (a,a,b), (a,a,c), (a,a,\_), (a,b,a), \text{etc}\}$
- Given the network output $\boldsymbol{y}$, what is the probability of some π?

$$p(\pi|\mathbf{x}) = \prod_{t=1}^{T} y^t_{\pi_t}$$

CTC - Labelings

- Paths π are as long as the input sequence
- Labels l are the actual predictions and could be shorter than π
- A label $l \in A^{\leq T}$
- Need a mapping F: $A'^{T} \longrightarrow A^{\leq T}$

CTC - Labelings

- Define F to simply remove repeated symbols and all blanks

$$(c,c,c,a,a,a,t,t,t)$$
$$(c,c,\_,a,a,t,\_)$$
$$(c,a,t)$$
$$(c,\_,\_,a,a,\_,t,t)$$
$$(c,a,a,a,a,a,a,a,t,t)$$

$$\xrightarrow{F}$$

(c,a,t)

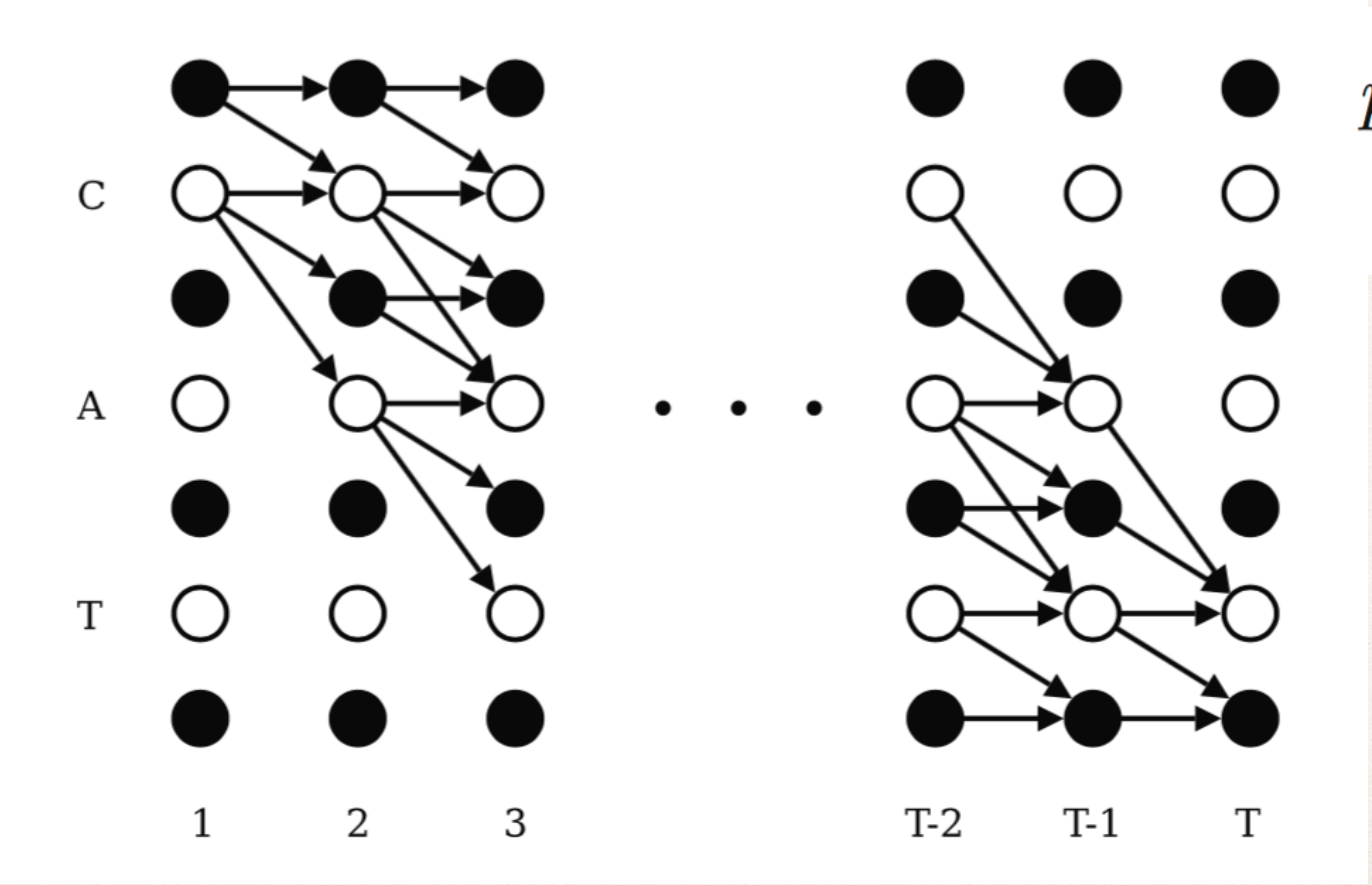
CTC - Label Probability

- To compute the probability of l we need to sum over the probability of all possible π which could produce l

$$p(\mathbf{l}|\mathbf{x}) = \sum_{\pi \in \mathcal{F}^{-1}(\mathbf{l})} p(\pi|\mathbf{x})$$

CTC - Label Probability

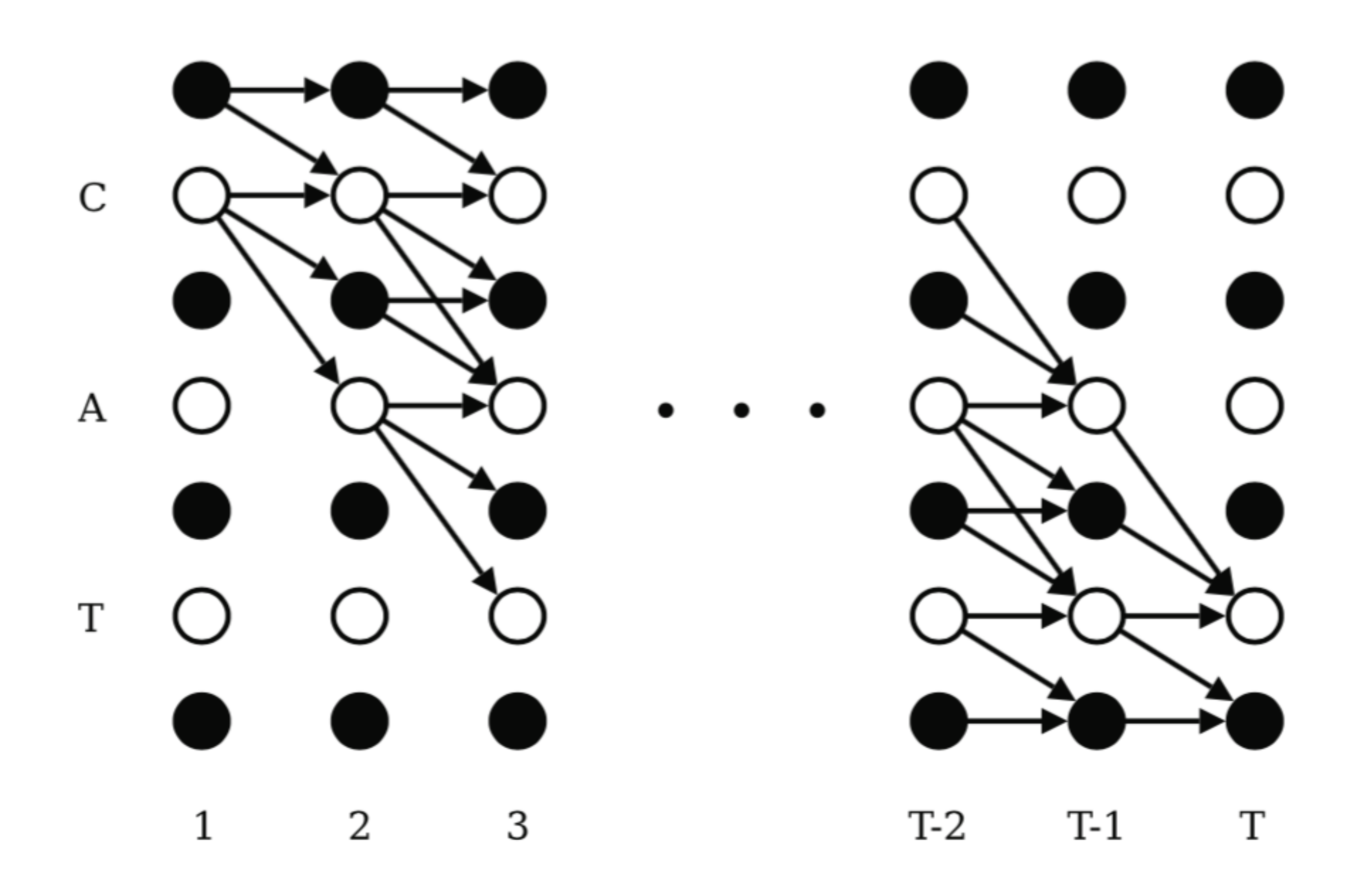
Exponential Number of Paths



$$p(\mathbf{l}|\mathbf{x}) = \sum_{\pi \in \mathcal{F}^{-1}(\mathbf{l})} p(\pi|\mathbf{x})$$

CTC - Forward-Backward

- Let l' be l with added blanks
  - (c,a,t) -> (_,c,_,a,_,t,_)
- Forward variable:
  - $\alpha(t,u)$
  - Summed probabilities of all paths of time t with the first u elements of l'



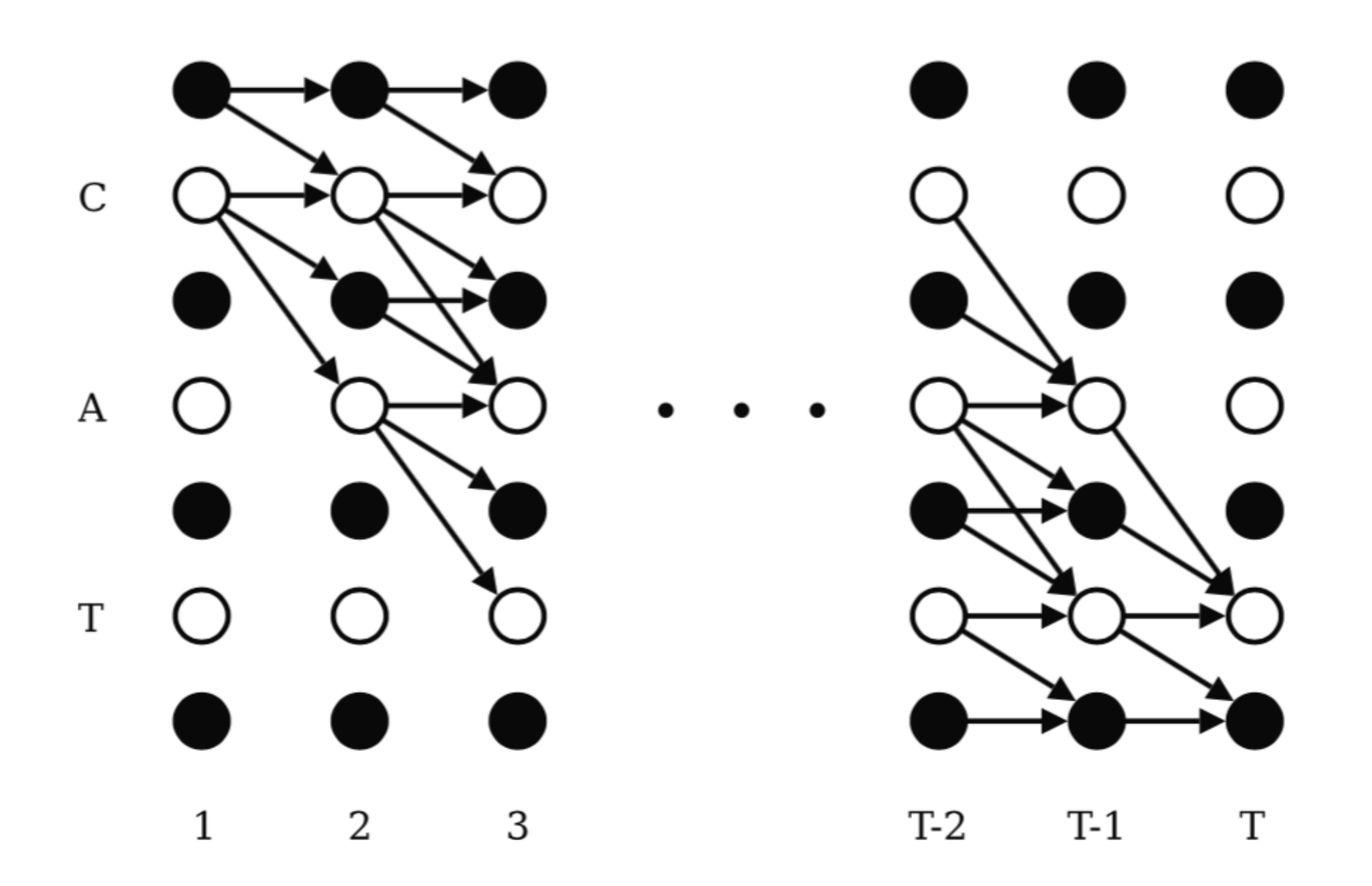
CTC - Forward-Backward

- Initialization:

$$\alpha(1,1) = y_b^1$$
$$\alpha(1,2) = y_{l_1}^1$$
$$\alpha(1,u) = 0,\ \forall u > 2$$

- Recurrence:

$$\alpha(t,u) = y_{l'_u}^t \sum_{i=f(u)}^{u} \alpha(t-1,i)$$



$$f(u) = \begin{cases} u-1 & \text{if } l'_u = \text{blank or } l'_{u-2} = l'_u \\ u-2 & \text{otherwise} \end{cases}$$

Example

$$\alpha(1,1) = y_b^1$$
$$\alpha(1,2) = y_{l_1}^1$$
$$\alpha(1,u) = 0, \ \forall u > 2$$

$$\alpha(t,u) = y_{l'_u}^t \sum_{i=f(u)}^{u} \alpha(t-1,i)$$

$$f(u) = \begin{cases} u-1 & \text{if } l'_u = \text{blank or } l'_{u-2} = l'_u \\ u-2 & \text{otherwise} \end{cases}$$

$$T = 4$$
$$l = (c,a)$$
$$l' = (\_,c,\_,a,\_)$$

	A	C	T	_
1	0.2	0.1	0.1	0.6
2	0.0	0.7	0.2	0.1
3	0.2	0.0	0.0	0.8
4	0.6	0.1	0.1	0.2

	_	c	_	a	_
1	0.6	0.7	0	0	0
2					
3					
4					

Example

$$\alpha(1,1) = y_b^1$$
$$\alpha(1,2) = y_{l_1}^1$$
$$\alpha(1,u) = 0, \ \forall u > 2$$

$$\alpha(t,u) = y_{l'_u}^t \sum_{i=f(u)}^{u} \alpha(t-1,i)$$

$$f(u) = \begin{cases} u-1 & \text{if } l'_u = \text{blank or } l'_{u-2} = l'_u \\ u-2 & \text{otherwise} \end{cases}$$

$T = 4$

$l = (c,a)$

$l' = (\_,c,\_,a,\_)$

	A	C	T	_
1	0.2	0.1	0.1	0.6
2	0.0	0.7	0.2	0.1
3	0.2	0.0	0.0	0.8
4	0.6	0.1	0.1	0.2

	_	c	_	a	_
1	0.6	0.7	0	0	0
2	0.1*(0+0.6) = 0.06	0.7*(0+0.6+0.7) = 0.91			
3					
4					

Example

$$\alpha(1,1) = y_b^1$$
$$\alpha(1,2) = y_{l_1}^1$$
$$\alpha(1,u) = 0, \ \forall u > 2$$

$$\alpha(t,u) = y_{l'_u}^t \sum_{i=f(u)}^{u} \alpha(t-1,i)$$

$$f(u) = \begin{cases} u-1 & \text{if } l'_u = \text{blank or } l'_{u-2} = l'_u \\ u-2 & \text{otherwise} \end{cases}$$

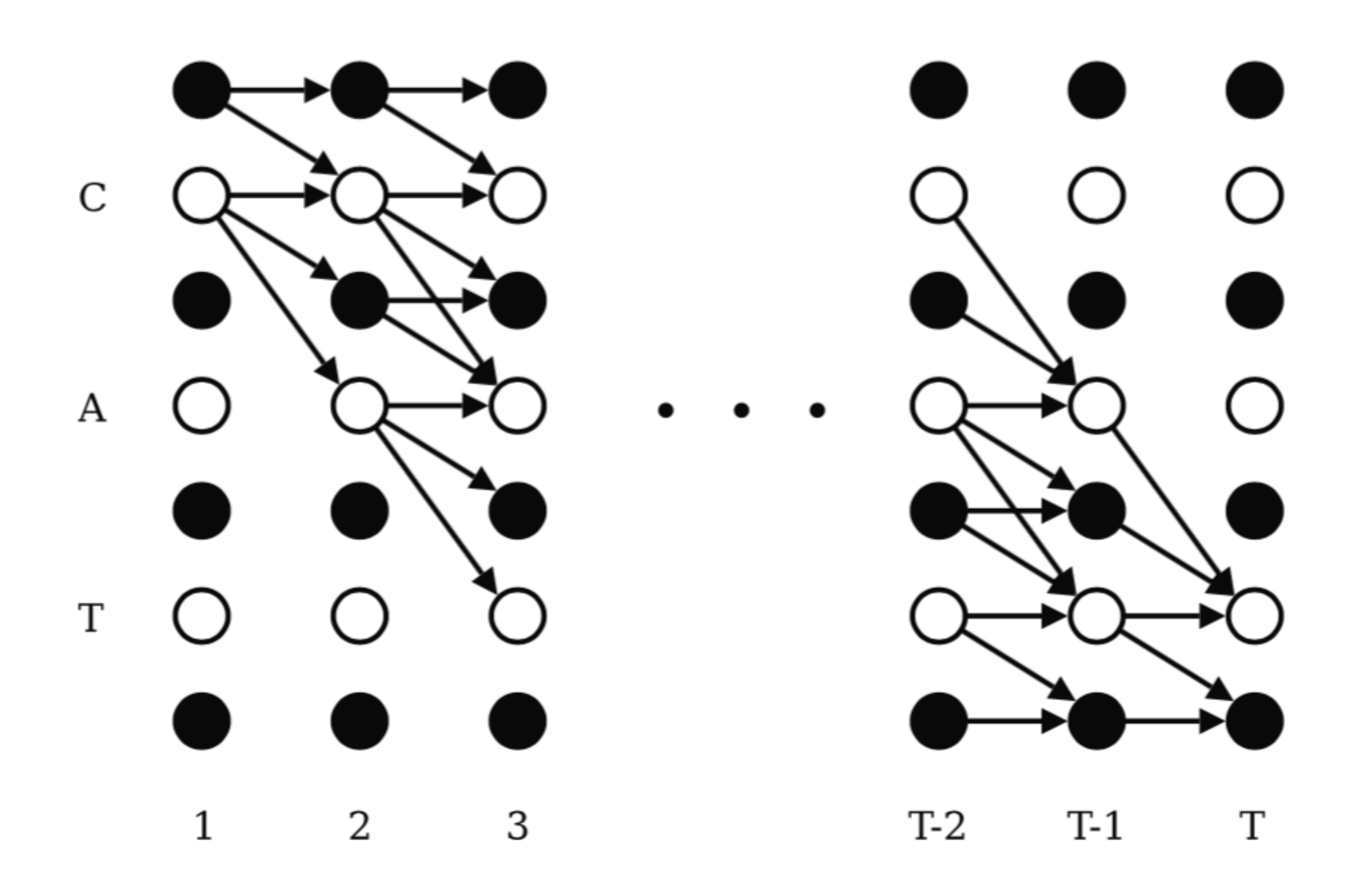
$$T = 4$$
$$l = (c,a)$$
$$l' = (\_,c,\_,a,\_)$$

	A	C	T	_
1	0.2	0.1	0.1	0.6
2	0.0	0.7	0.2	0.1
3	0.2	0.0	0.0	0.8
4	0.6	0.1	0.1	0.2

	_	c	_	a	_
1	0.6	0.7	0	0	0
2	0.1*(0+0.6) = 0.06	0.7*(0+0.6+0.7) = 0.91	0.1*(0.7+0) = 0.07	0*(0.7+0+0) = 0	0.1*(0+0) = 0
3	0.8*(0+0.06) = 0.048	0.0*(0+0.06+0.91) = 0.0	0.8*(0.91+0.07) = 0.784	0.2*(0.91+0.07+0) = 0.196	0.8*(0+0) = 0
4	0.2*(0+0.048) = 0.0096	0.1*(0+0.048+0) = 0.0048	0.2*(0.0+0.784) = 0.157	0.6*(0+0.784+0.196) = 0.588	0.2*(0.196+0) = 0.0392

CTC - Forward-Backward

- Backward variable:
  - $\beta(t,u)$
  - Summed probabilities of all paths that start at $t+1$ and finish u



CTC - Forward-Backward

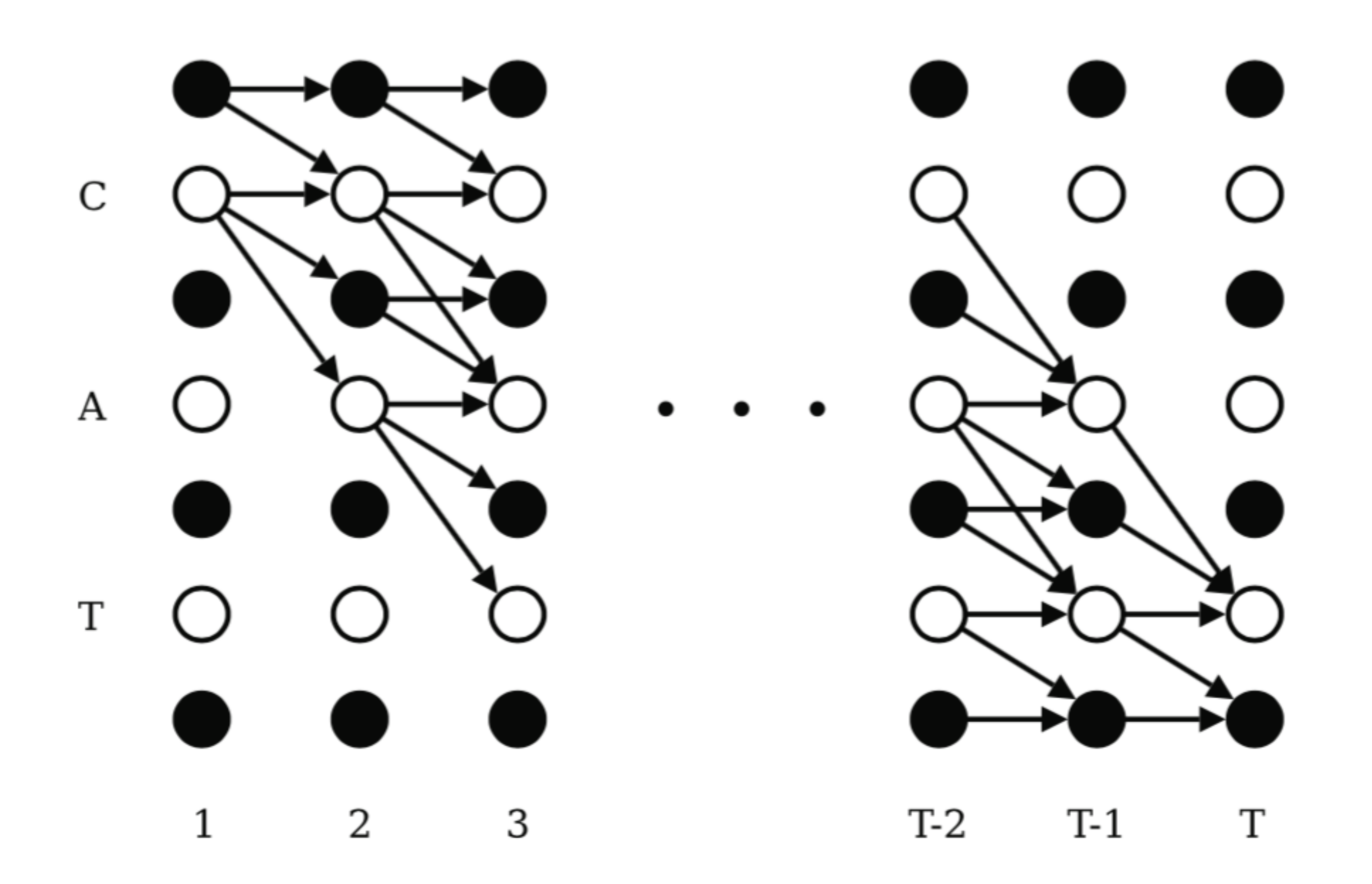
- Initialization:

$$\beta(T, U') = \beta(T, U' - 1) = 1$$
$$\beta(T, u) = 0, \ \forall u < U' - 1$$

- Recurrence:

$$\beta(t, u) = \sum_{i=u}^{g(u)} \beta(t+1, i) y_{l'_i}^{t+1}$$

$$g(u) = \begin{cases} u+1 & \text{if } l'_u = \text{blank or } l'_{u+2} = l'_u \\ u+2 & \text{otherwise} \end{cases}$$



Example

$$\beta(T, U') = \beta(T, U'-1) = 1$$
$$\beta(T, u) = 0,\ \forall u < U'-1$$

$$g(u) = \begin{cases} u+1 & \text{if } l'_u = \text{blank or } l'_{u+2} = l'_u \\ u+2 & \text{otherwise} \end{cases}$$

$$\beta(t, u) = \sum_{i=u}^{g(u)} \beta(t+1, i) y^{t+1}_{l'_i}$$

$$T = 4$$
$$l = (c,a)$$
$$l' = (\_,c,\_,a,\_)$$

	A	C	T	_
1	0.2	0.1	0.1	0.6
2	0.0	0.7	0.2	0.1
3	0.2	0.0	0.0	0.8
4	0.6	0.1	0.1	0.2

	_	c	_	a	_
1					
2					
3					
4	0	0	0	1	1

Example

$$\beta(T, U') = \beta(T, U' - 1) = 1$$
$$\beta(T, u) = 0, \ \forall u < U' - 1$$

$$g(u) = \begin{cases} u + 1 & \text{if } l'_u = \text{blank or } l'_{u+2} = l'_u \\ u + 2 & \text{otherwise} \end{cases}$$

$$\beta(t, u) = \sum_{i=u}^{g(u)} \beta(t + 1, i) y_{l'_i}^{t+1}$$

$$T = 4$$
$$l = (c,a)$$
$$l' = (\_,c,\_,a,\_)$$

	A	C	T	_
1	0.2	0.1	0.1	0.6
2	0.0	0.7	0.2	0.1
3	0.2	0.0	0.0	0.8
4	0.6	0.1	0.1	0.2

	_	c	_	a	_
1					
2					
3				1*0.6+1*0.2+0 = 0.8	1*0.2+0 = 0.2
4	0	0	0	1	1

Example

$$\beta(T,U') = \beta(T,U'-1) = 1$$
$$\beta(T,u) = 0,\ \forall u < U'-1$$

$$g(u) = \begin{cases} u+1 & \text{if } l'_u = \text{blank or } l'_{u+2} = l'_u \\ u+2 & \text{otherwise} \end{cases}$$

$$\beta(t,u) = \sum_{i=u}^{g(u)} \beta(t+1,i) y^{t+1}_{l'_i}$$

$$T = 4$$
$$l = (c,a)$$
$$l' = (\_,c,\_,a,\_)$$

	A	C	T	_
1	0.2	0.1	0.1	0.6
2	0.0	0.7	0.2	0.1
3	0.2	0.0	0.0	0.8
4	0.6	0.1	0.1	0.2

	_	c	_	a	_
1	0*0.1+0.64*0.7 = 0.448	0.64*0.7+0.64*0.1+0.32*0 = 0.512	0.64*0.1+0.32*0 = 0.064	0.32*0+0.16*0.1+0 = 0.016	0.16*0.1+0 = 0.016
2	0*0.8+0.6*0 = 0	0.6*0+0.6*0.8+0.8*0.2 = 0.64	0.6*0.8+0.8*0.2 = 0.64	0.8*0.2+0.2*0.8+0= 0.32	0.2*0.8+0 = 0.16
3	0*0.2+0*0.1 = 0	0*0.1+0*0.2+1*0.6 = 0.6	0*0.2+1*0.6 = 0.6	1*0.6+1*0.2+0 = 0.8	1*0.2+0 = 0.2
4	0	0	0	1	1

CTC - Label Probability

$$p(\mathbf{z}|\mathbf{x}) = \sum_{u=1}^{|\mathbf{z}'|} \alpha(t,u)\beta(t,u)$$

1	2	3	4

Forward:

	_	c	_	a	_
1	0.6	0.7	0	0	0
2	0.1*(0+0.6) = 0.06	0.7*(0+0.6+0.7) = 0.91	0.1*(0.7+0) = 0.07	0*(0.7+0+0) = 0	0.1*(0+0) = 0
3	0.8*(0+0.06) = 0.048	0.0*(0+0.06+0.91) = 0.0	0.8*(0.91+0.07) = 0.784	0.2*(0.91+0.07+0) = 0.196	0.8*(0+0) = 0
4	0.2*(0+0.048) = 0.0096	0.1*(0+0.048+0) = 0.0048	0.2*(0.0+0.784) = 0.157	0.6*(0+0.784+0.196) = 0.588	0.2*(0.196+0) = 0.0392

Backward:

	_	c	_	a	_
1	0*0.1+0.64*0.7 = 0.448	0.64*0.7+0.64*0.1+0.32*0 = 0.512	0.64*0.1+0.32*0 = 0.064	0.32*0+0.16*0.1+0 = 0.016	0.16*0.1+0 = 0.016
2	0*0.8+0.6*0 = 0	0.6*0+0.6*0.8+0.8*0.2 = 0.64	0.6*0.8+0.8*0.2 = 0.64	0.8*0.2+0.2*0.8+0= 0.32	0.2*0.8+0 = 0.16
3	0*0.2+0*0.1 = 0	0*0.1+0*0.2+1*0.6 = 0.6	0*0.2+1*0.6 = 0.6	1*0.6+1*0.2+0 = 0.8	1*0.2+0 = 0.2
4	0	0	0	1	1

CTC - Label Probability

$$p(\mathbf{z}|\mathbf{x}) = \sum_{u=1}^{|\mathbf{z}'|} \alpha(t,u)\beta(t,u)$$

1	2	3	4
0.6272	0.6272	0.6272	0.6272

Forward:

	_	c	_	a	_
1	0.6	0.7	0	0	0
2	0.1*(0+0.6) = 0.06	0.7*(0+0.6+0.7) = 0.91	0.1*(0.7+0) = 0.07	0*(0.7+0+0) = 0	0.1*(0+0) = 0
3	0.8*(0+0.06) = 0.048	0.0*(0+0.06+0.91) = 0.0	0.8*(0.91+0.07) = 0.784	0.2*(0.91+0.07+0) = 0.196	0.8*(0+0) = 0
4	0.2*(0+0.048) = 0.0096	0.1*(0+0.048+0) = 0.0048	0.2*(0.0+0.784) = 0.157	0.6*(0+0.784+0.196) = 0.588	0.2*(0.196+0) = 0.0392

Backward:

	_	c	_	a	_
1	0*0.1+0.64*0.7 = 0.448	0.64*0.7+0.64*0.1+0.32*0 = 0.512	0.64*0.1+0.32*0 = 0.064	0.32*0+0.16*0.1+0 = 0.016	0.16*0.1+0 = 0.016
2	0*0.8+0.6*0 = 0	0.6*0+0.6*0.8+0.8*0.2 = 0.64	0.6*0.8+0.8*0.2 = 0.64	0.8*0.2+0.2*0.8+0= 0.32	0.2*0.8+0 = 0.16
3	0*0.2+0*0.1 = 0	0*0.1+0*0.2+1*0.6 = 0.6	0*0.2+1*0.6 = 0.6	1*0.6+1*0.2+0 = 0.8	1*0.2+0 = 0.2
4	0	0	0	1	1

CTC - Loss

- Minimize negative log likelihood of data

$$\mathcal{L}(\mathbf{x}, \mathbf{z}) = -\ln \sum_{u=1}^{|\mathbf{z}'|} \alpha(t, u)\beta(t, u)$$

$l = (c,a)$ $\quad$ $\mathrm{L} = 0.4665$

CTC - Gradient

Helper Function: $B(\mathbf{z}, k) = \{u : \mathbf{z}'_u = k\}$

$l = (c,a)$

$l' = (\_,c,\_,a,\_)$

$B(l,a) = \{4\}$ $B(l,c) = \{2\}$ $B(l,\_) = \{1,3,5\}$

$B(l,t) = \{\}$

Gradient: $$\frac{\partial \mathcal{L}(\mathbf{x}, \mathbf{z})}{\partial a_k^t} = y_k^t - \frac{1}{p(\mathbf{z}|\mathbf{x})} \sum_{u \in B(\mathbf{z},k)} \alpha(t,u)\beta(t,u)$$

CTC - Gradient

$$\frac{\partial \mathcal{L}(\mathbf{x},\mathbf{z})}{\partial a_k^t} = y_k^t - \frac{1}{p(\mathbf{z}|\mathbf{x})} \sum_{u \in B(\mathbf{z},k)} \alpha(t,u)\beta(t,u)$$

	A	C	T	_
1	0.2	0.1	0.1	0.6
2	0.0	0.7	0.2	0.1
3	0.2	0.0	0.0	0.8
4	0.6	0.1	0.1	0.2

$p(l \mid x) = 0.6272$

B(l,a) = {4}   B(l,c) = {2}

B(l,_) = {1,3,5}

B(l,t) = {}

Forward:

	_	c	_	a	_
1	0.6	0.7	0	0	0
2	0.1*(0+0.6) = 0.06	0.7*(0+0.6+0.7) = 0.91	0.1*(0.7+0) = 0.07	0*(0.7+0+0) = 0	0.1*(0+0) = 0
3	0.8*(0+0.06) = 0.048	0.0*(0+0.06+0.91) = 0.0	0.8*(0.91+0.07) = 0.784	0.2*(0.91+0.07+0) = 0.196	0.8*(0+0) = 0
4	0.2*(0+0.048) = 0.0096	0.1*(0+0.048+0) = 0.0048	0.2*(0.0+0.784) = 0.157	0.6*(0+0.784+0.196) = 0.588	0.2*(0.196+0) = 0.0392

Backward:

	_	c	_	a	_
1	0*0.1+0.64*0.7 = 0.448	0.64*0.7+0.64*0.1+0.32*0 = 0.512	0.64*0.1+0.32*0 = 0.064	0.32*0+0.16*0.1+0 = 0.016	0.16*0.1+0 = 0.016
2	0*0.8+0.6*0 = 0	0.6*0+0.6*0.8+0.8*0.2 = 0.64	0.6*0.8+0.8*0.2 = 0.64	0.8*0.2+0.2*0.8+0= 0.32	0.2*0.8+0 = 0.16
3	0*0.2+0*0.1 = 0	0*0.1+0*0.2+1*0.6 = 0.6	0*0.2+1*0.6 = 0.6	1*0.6+1*0.2+0 = 0.8	1*0.2+0 = 0.2
4	0	0	0	1	1

CTC - Gradient

$$\frac{\partial \mathcal{L}(\mathbf{x},\mathbf{z})}{\partial a_k^t} = y_k^t - \frac{1}{p(\mathbf{z}|\mathbf{x})} \sum_{u \in B(\mathbf{z},k)} \alpha(t,u)\beta(t,u)$$

	A	C	T	_
1	0.2	0.1	0.1	0.6
2	0.0	0.7	0.2	0.1
3	0.2	0.0	0.0	0.8
4	0.6	0.1	0.1	0.2

$p(l\,|\,x) = 0.6272$

B(l,a) = {4}   B(l,c) = {2}
B(l,_) = {1,3,5}
B(l,t) = {}

= -0.2286

Forward:

	_	c	_	a	_
1	0.6	0.7	0	0	0
2	0.1*(0+0.6) = 0.06	0.7*(0+0.6+0.7) = 0.91	0.1*(0.7+0) = 0.07	0*(0.7+0+0) = 0	0.1*(0+0) = 0
3	0.8*(0+0.06) = 0.048	0.0*(0+0.06+0.91) = 0.0	0.8*(0.91+0.07) = 0.784	0.2*(0.91+0.07+0) = 0.196	0.8*(0+0) = 0
4	0.2*(0+0.048) = 0.0096	0.1*(0+0.048+0) = 0.0048	0.2*(0.0+0.784) = 0.157	0.6*(0+0.784+0.196) = 0.588	0.2*(0.196+0) = 0.0392

Backward:

	_	c	_	a	_
1	0*0.1+0.64*0.7 = 0.448	0.64*0.7+0.64*0.1+0.32*0 = 0.512	0.64*0.1+0.32*0 = 0.064	0.32*0+0.16*0.1+0 = 0.016	0.16*0.1+0 = 0.016
2	0*0.8+0.6*0 = 0	0.6*0+0.6*0.8+0.8*0.2 = 0.64	0.6*0.8+0.8*0.2 = 0.64	0.8*0.2+0.2*0.8+0= 0.32	0.2*0.8+0 = 0.16
3	0*0.2+0*0.1 = 0	0*0.1+0*0.2+1*0.6 = 0.6	0*0.2+1*0.6 = 0.6	1*0.6+1*0.2+0 = 0.8	1*0.2+0 = 0.2
4	0	0	0	1	1

CTC - Decoding

- Given a new input $\boldsymbol{x}$
  - Want $l^* = \text{argmax } p(l \mid \boldsymbol{x})$

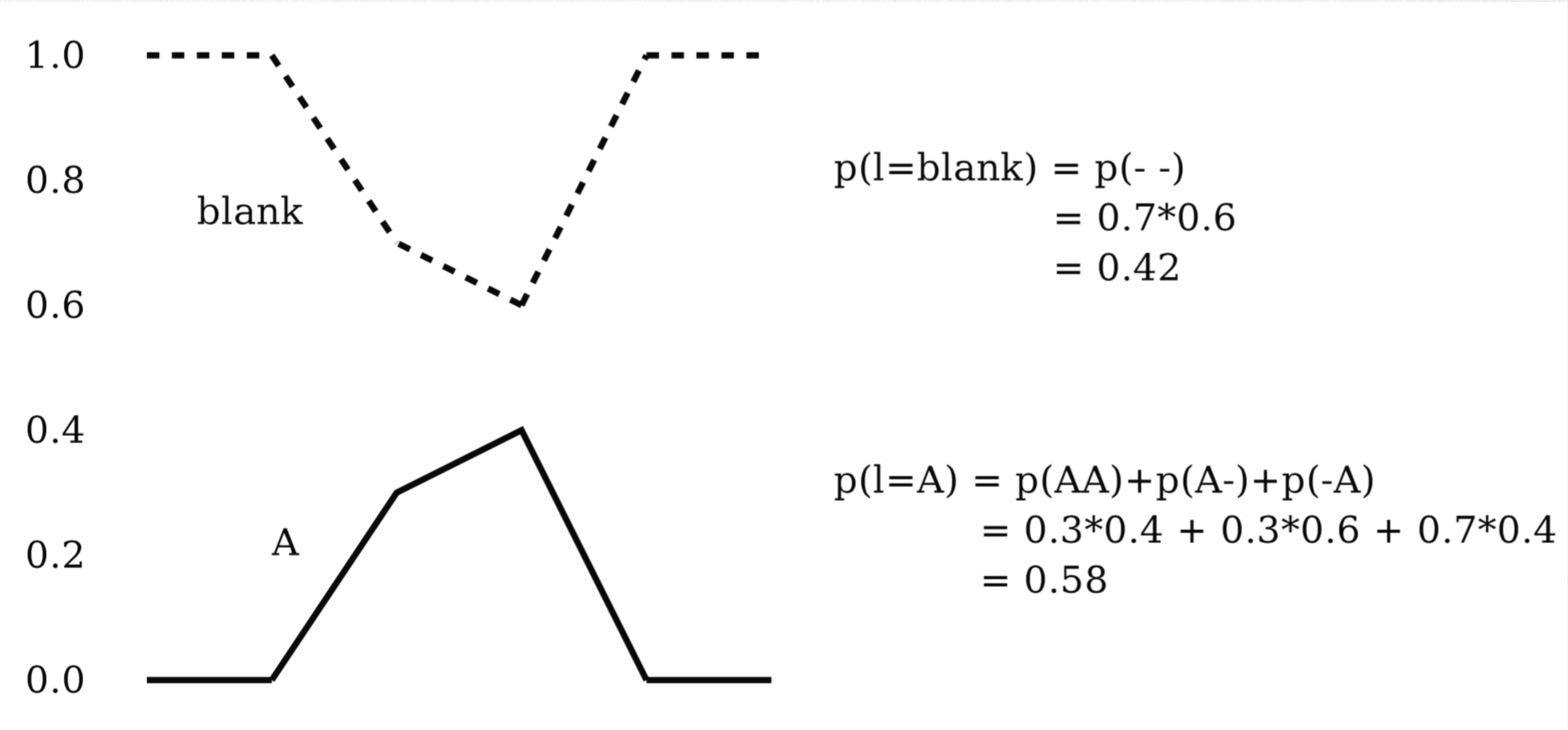
	A	C	T	_
1	0.2	0.1	0.1	0.6
2	0.0	0.7	0.2	0.1
3	0.2	0.0	0.0	0.8
4	0.6	0.1	0.1	0.2

Ideas?

CTC - Decoding

- Given a new input $\boldsymbol{x}$
  - Want $l^* = \text{argmax } p(l \mid \boldsymbol{x})$
- Greedy
- Beam Search
- Prefix Search

	A	C	T	_
1	0.2	0.1	0.1	0.6
2	0.0	0.7	0.2	0.1
3	0.2	0.0	0.0	0.8
4	0.6	0.1	0.1	0.2



Experiments

Experiments

- 2 datasets
  - Digit and Ascii
  - Avg Sequence Length: 45

Digit

10 Labels

9 Features

Digit

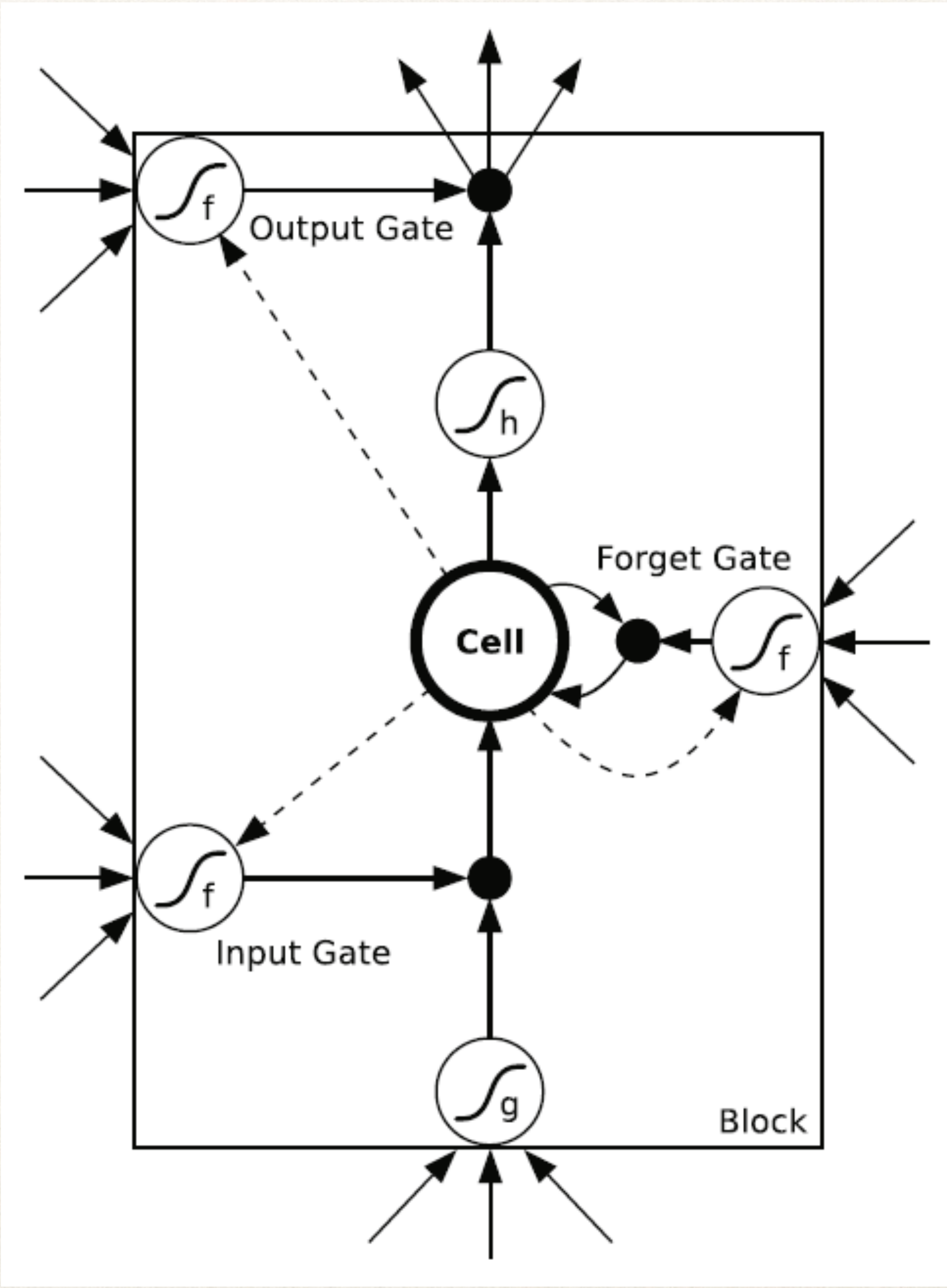
94 Labels

11 Features

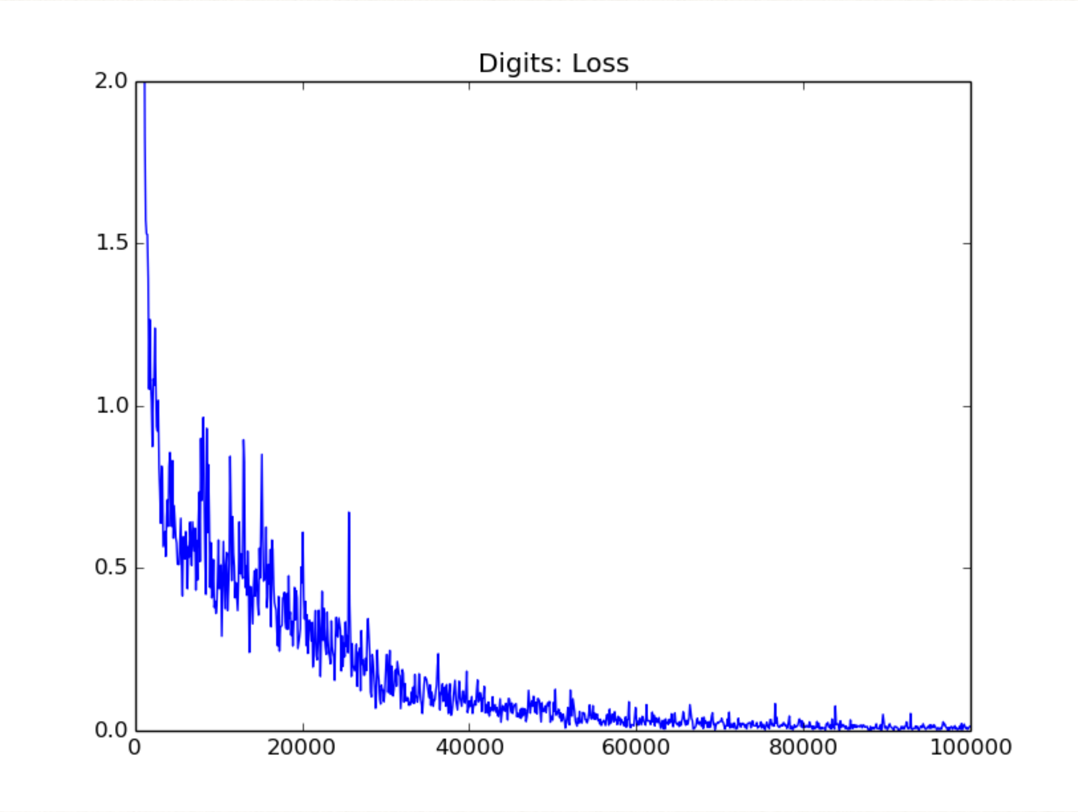
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~

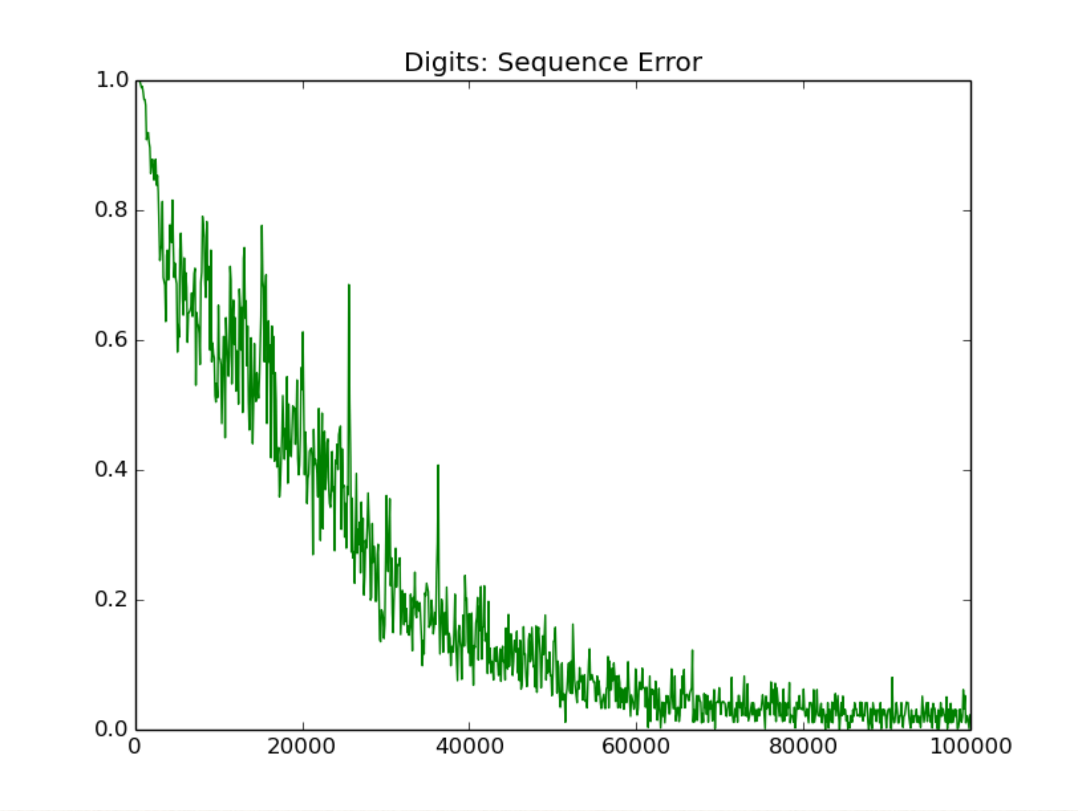
Network Parameters

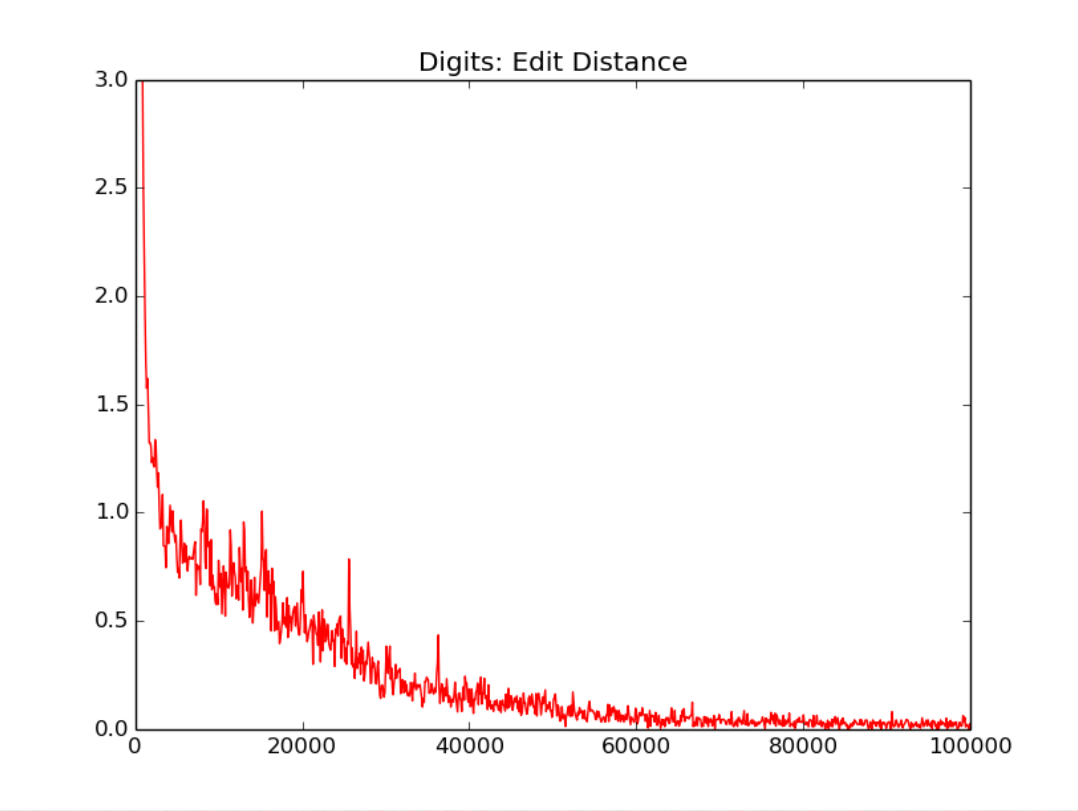
- 25 Bidirectional LSTM Nodes
- Mini-Batch size of 20
- Training set size of 7000
- Validation set size of 3000
- 100 epochs of training
- No momentum
- Decaying learning rate
- Gradient clipping at 10
- Trained with RMSProp



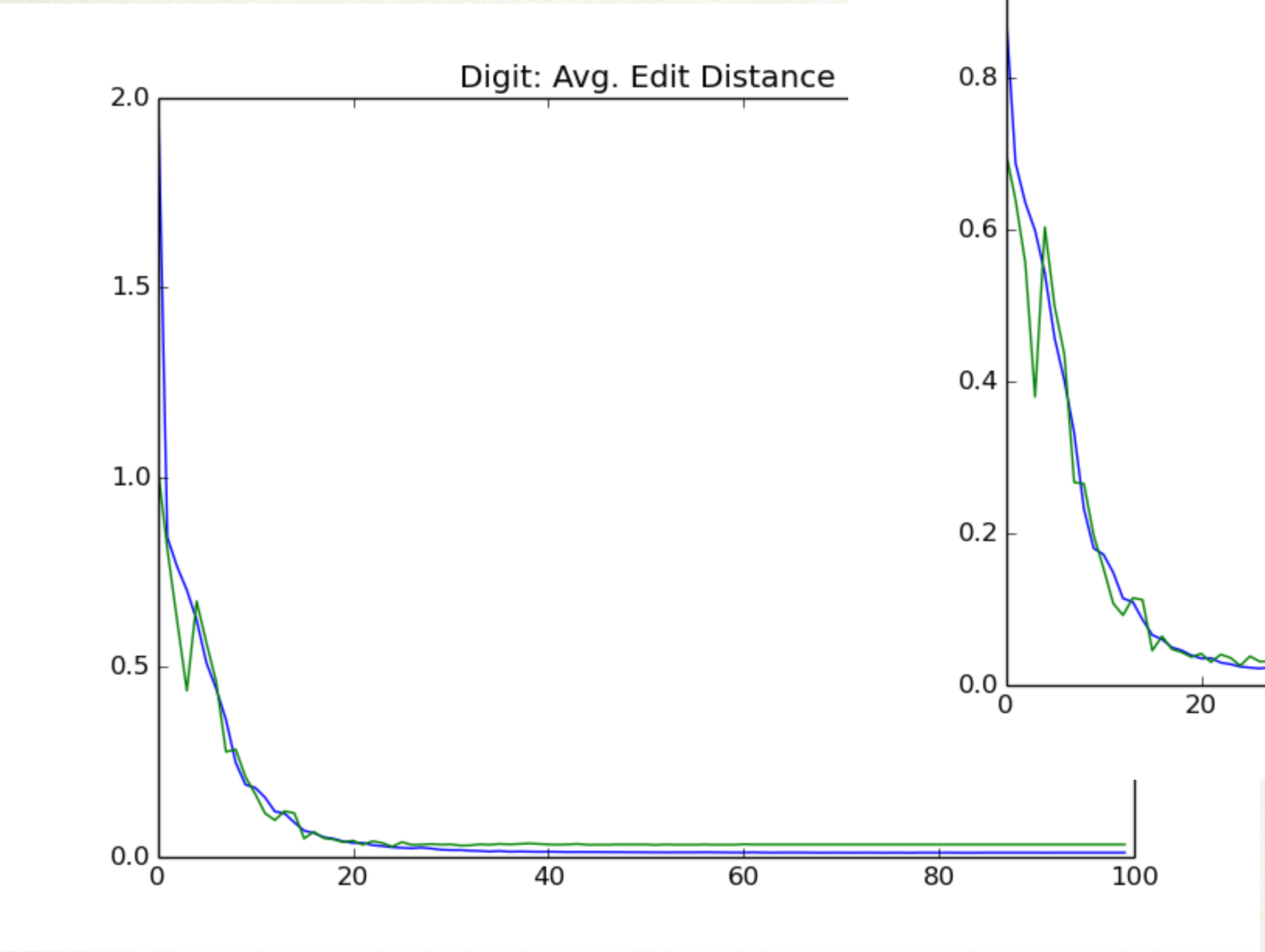
Results: Digit

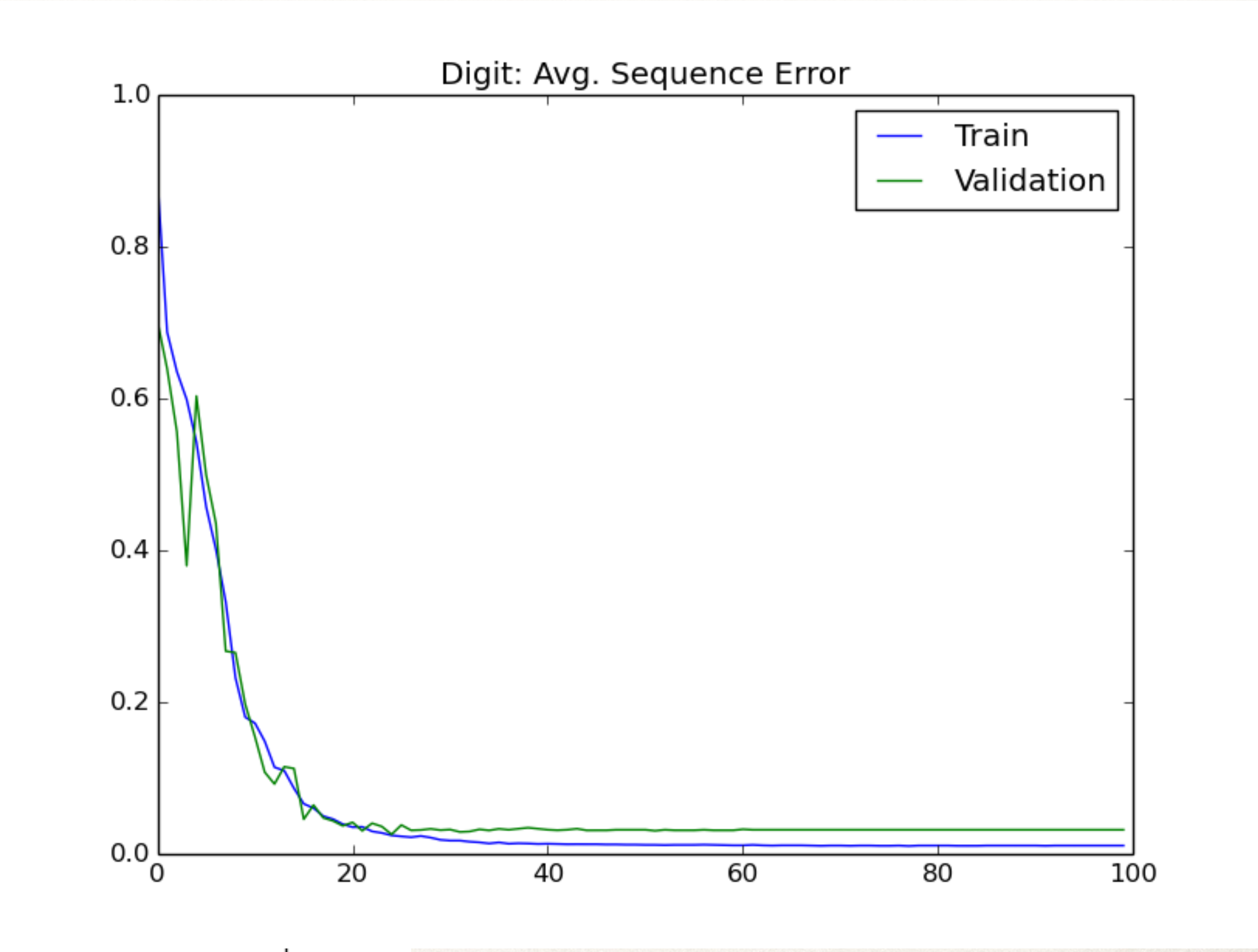




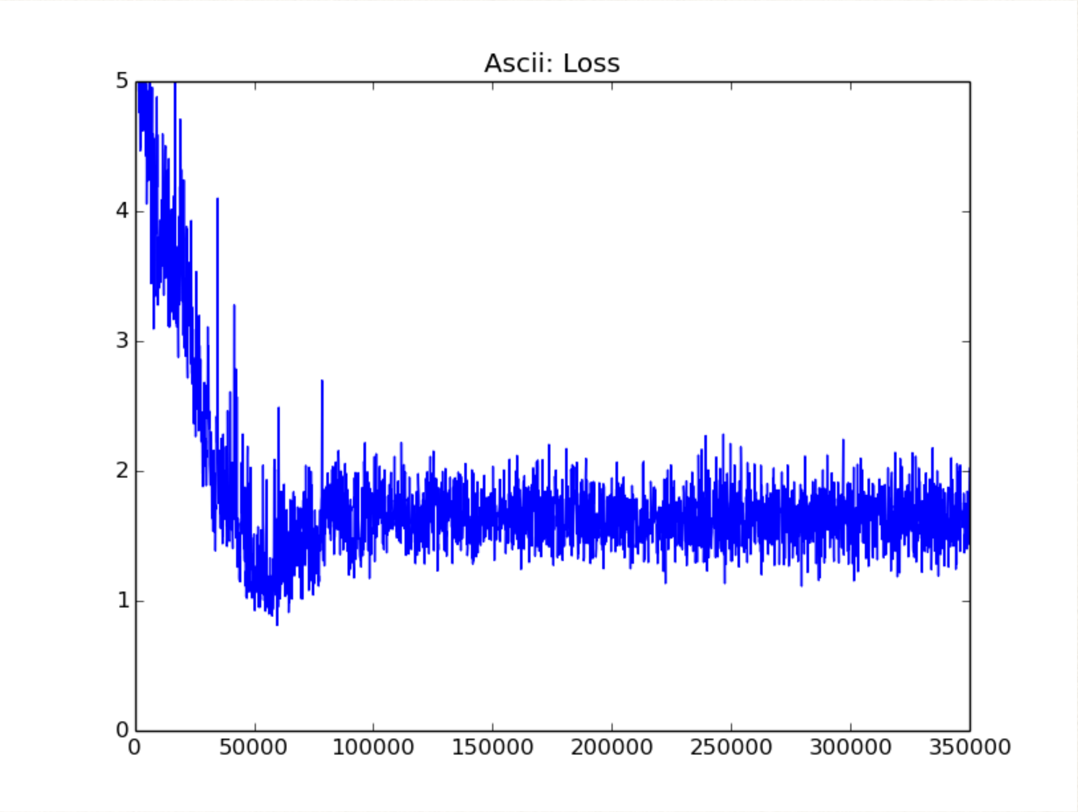


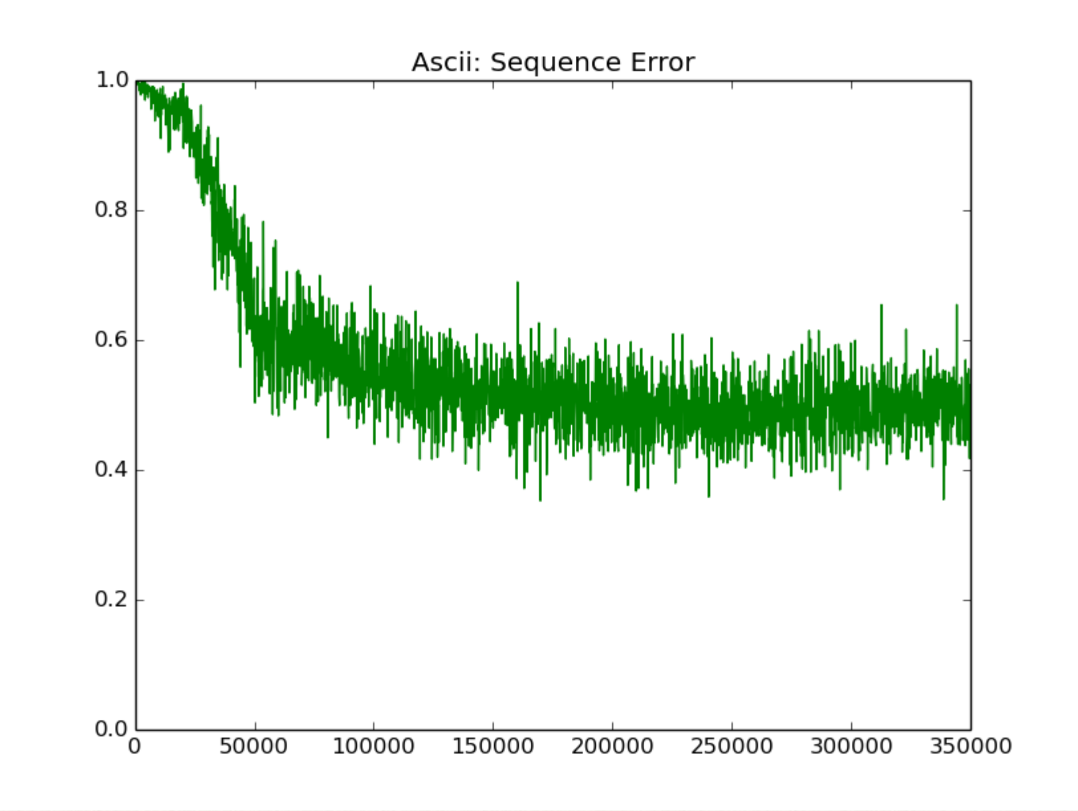
Results: Digit

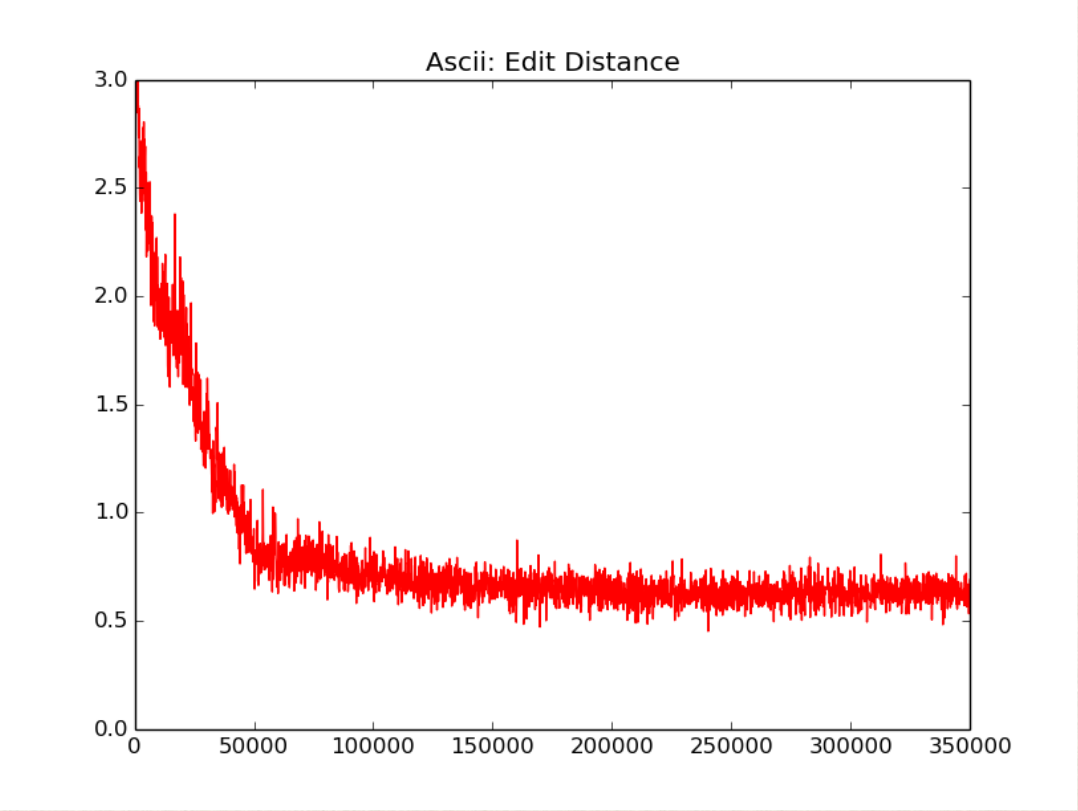




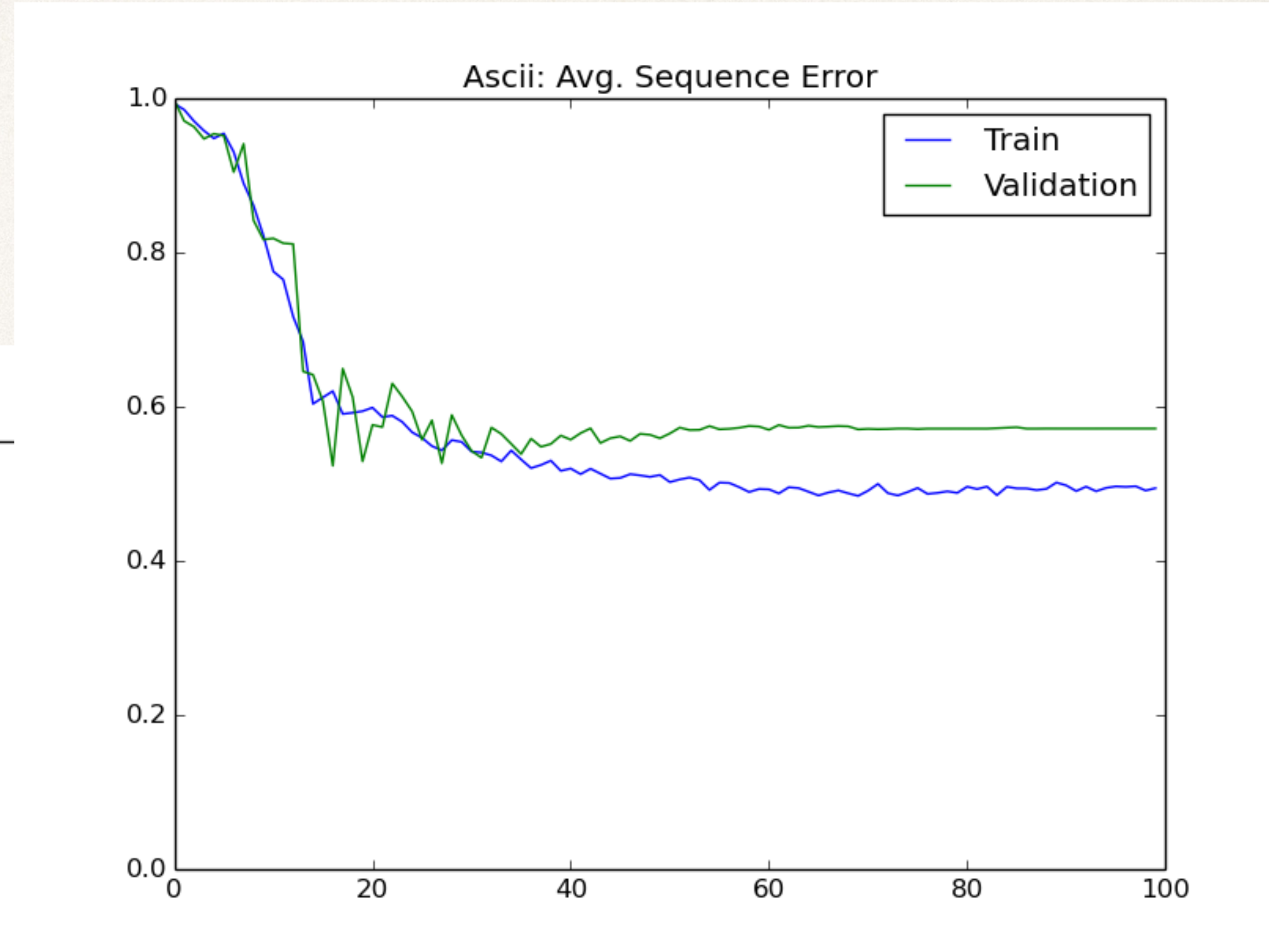
Results: Ascii

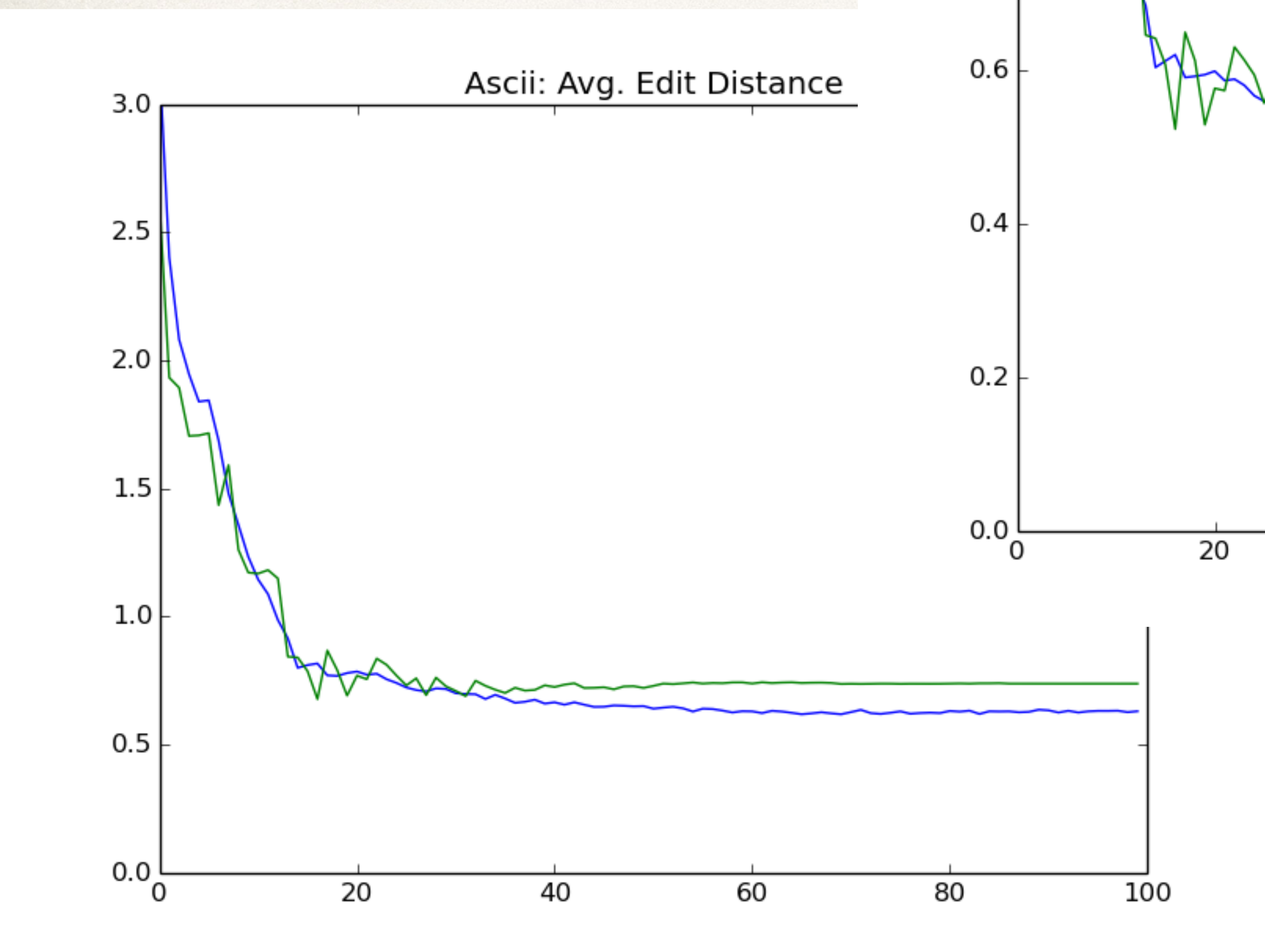






Results: Ascii





Other Applications

Multidimensional LSTMs with CTC

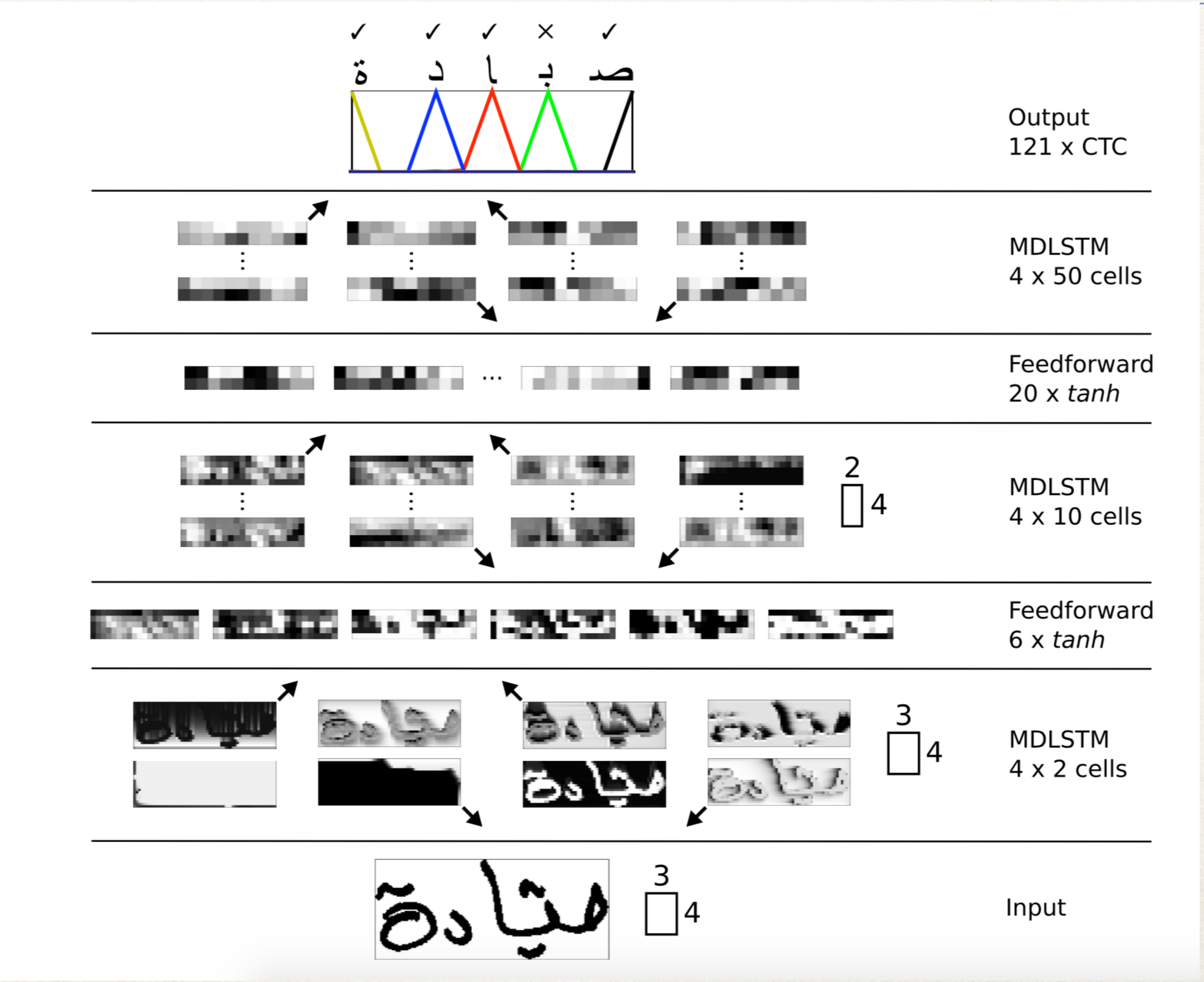
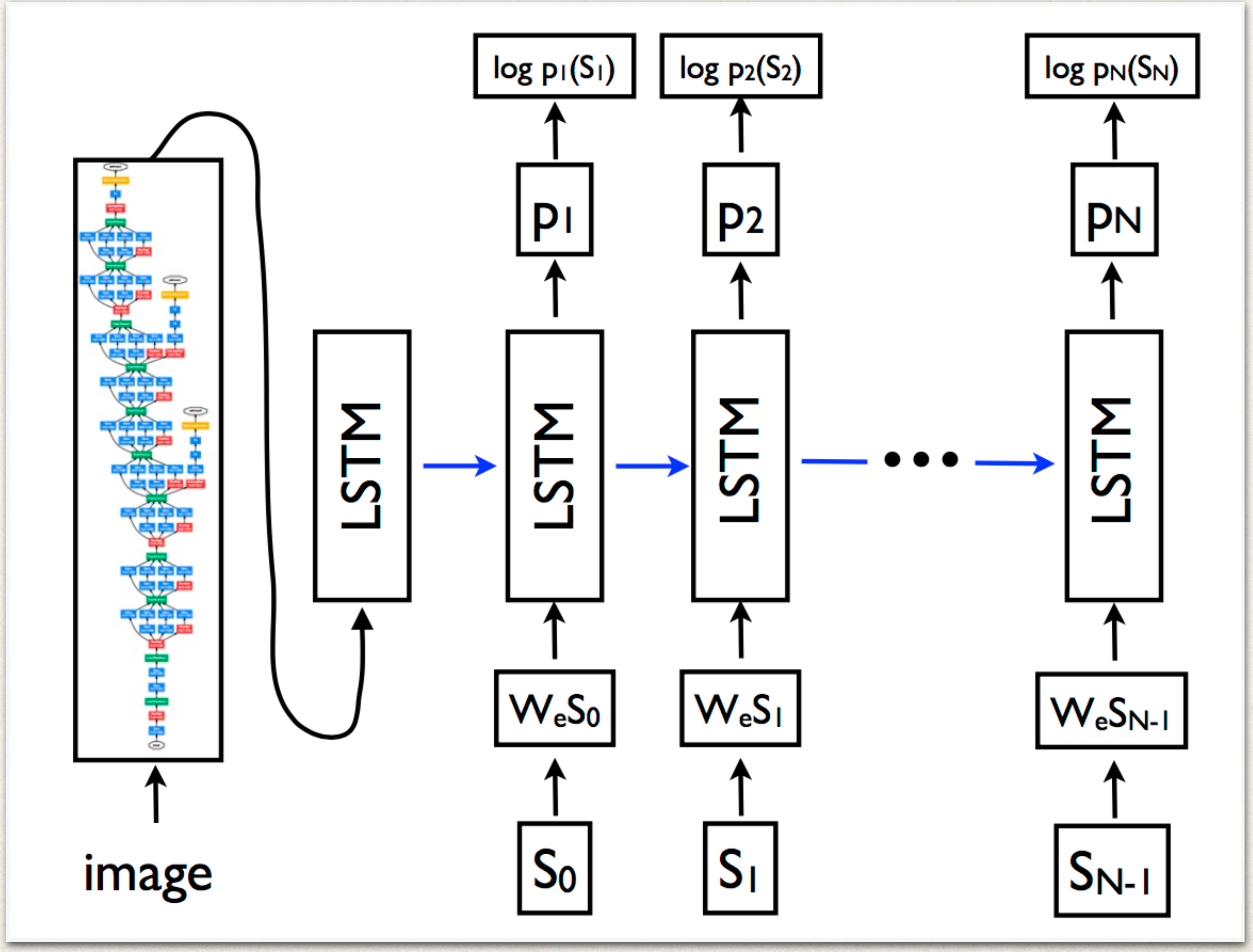
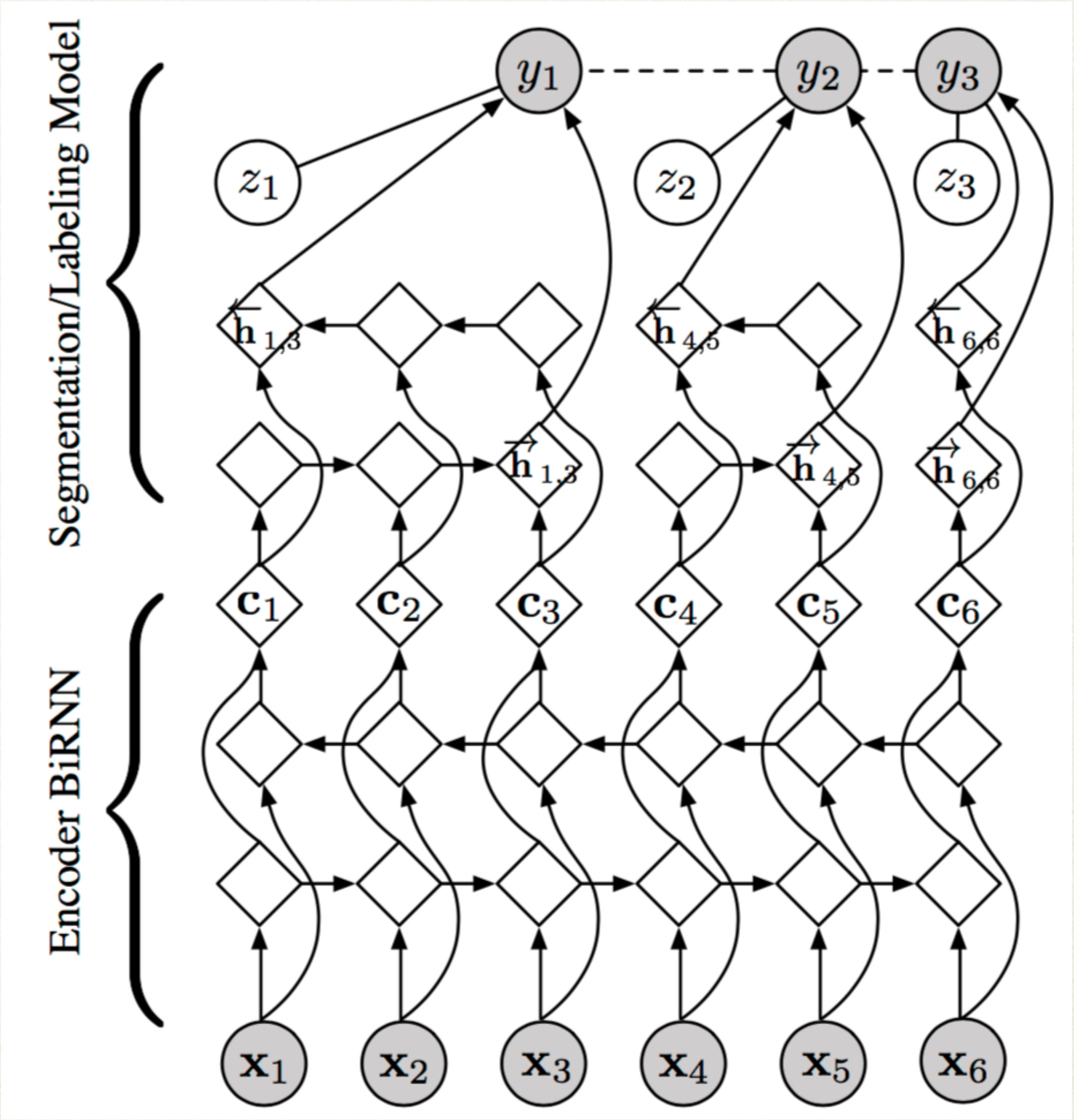


Image Captioning



Segmental Recurrent Neural Networks

- Explicitly model segmentation as latent variable
- Uses 2 layers of Bidirectional LSTMs
- Dynamic programing to compute hidden values for every possible segmentation
- Claim better results than CTC on handwriting and Chinese Segmentation/POS tagging



Bibliography

Graves, Alex, et al. "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks." *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006.
Vinyals, Oriol, et al. "Show and tell: A neural image caption generator." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015.
Graves, Alex. "Offline arabic handwriting recognition with multidimensional recurrent neural networks." *Guide to OCR for Arabic Scripts*. Springer London, 2012. 297-313.
Kong, Lingpeng, Chris Dyer, and Noah A. Smith. "Segmental Recurrent Neural Networks." *arXiv preprint arXiv:1511.06018* (2015).
Graves, Alex. *Supervised sequence labelling*. Springer Berlin Heidelberg, 2012.

https://www.uea.ac.uk/computing/research-at-the-uea-speech-group
https://people.cs.umass.edu/~mccallum/courses/inlp2004/lect10-tagginghmm1.pdf
http://karpathy.github.io/2015/05/21/rnn-effectiveness/
http://eric-yuan.me/rnn2-lstm/
http://colah.github.io/posts/2015-08-Understanding-LSTMs/
https://developer.valvesoftware.com/wiki/Phoneme_Tool
http://turing.iimas.unam.mx/~ivanvladimir/slides/fonologia_forense/identification_cont.html#/
http://my.fit.edu/~vkepuska/ece5527/Projects/Fall2011/Burgos,%20Wilson/sphinx4-1.0beta6/sphinx4-1.0beta6/
https://www.tensorflow.org/versions/r0.7/tutorials/seq2seq/index.html

Questions?