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
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

![Diagram of a Simple RNN](image)
Sequences

One-One Models

\[ h_t \quad A \quad X_t \quad = \quad A \quad A \quad A \quad \ldots \quad A \quad X_t \]
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

扈 Split audio into 10 ms clips (overlapping)

扈 Create Cepstral Coefficients

扈 Convert audio into Frequency Domain

扈 Do it again

扈 Compute 1st and 2nd derivatives

扈 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 ms. 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 \( O_1...O_t \) and being in \( S_i \) at step \( t \)

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:
\[
P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i).
\]
**Forward Algorithm**

\[ \pi = \{0.3, 0.3, 0.4\} \]

What is \( P("F1 F3 F3" \mid \lambda) \)?

<table>
<thead>
<tr>
<th></th>
<th>Transition</th>
<th>Emission</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.2 0.5 0.3</td>
<td>0.5 0.2 0.3</td>
</tr>
<tr>
<td></td>
<td>0.4 0.4 0.2</td>
<td>0.2 0.3 0.5</td>
</tr>
<tr>
<td></td>
<td>0.1 0.4 0.5</td>
<td>0.1 0.1 0.8</td>
</tr>
</tbody>
</table>

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:**
\[
P(O \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i).
\]

<table>
<thead>
<tr>
<th></th>
<th>( t = 1, O_t = F1 )</th>
<th>( t = 2, O_t = F3 )</th>
<th>( t = 3, O_t = F3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>( 0.3 \cdot 0.5 = 0.15 )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C3</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Forward Algorithm

\[ \pi = \{0.3, 0.3, 0.4\} \]

<table>
<thead>
<tr>
<th>A Transition</th>
<th>B Emission</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2 0.5 0.3</td>
<td>0.5 0.2 0.3</td>
</tr>
<tr>
<td>0.4 0.4 0.2</td>
<td>0.2 0.3 0.5</td>
</tr>
<tr>
<td>0.1 0.4 0.5</td>
<td>0.1 0.1 0.8</td>
</tr>
</tbody>
</table>

What is \( P("F1\ F3\ F3"\ | \lambda) \)?

1) Initialization:
\[ \alpha_1(i) = \pi_i b_i(O_1), \quad 1 \leq i \leq N. \]

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

3) Termination:
\[ P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i). \]

<table>
<thead>
<tr>
<th>( t = 1, \ O_t = F1 )</th>
<th>( t = 2, \ O_t = F3 )</th>
<th>( t = 3, \ O_t = F3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1 ( 0.3 \cdot 0.5 = 0.15 )</td>
<td>( )</td>
<td>( )</td>
</tr>
<tr>
<td>C2 ( 0.3 \cdot 0.2 = 0.06 )</td>
<td>( )</td>
<td>( )</td>
</tr>
<tr>
<td>C3 ( 0.4 \cdot 0.1 = 0.04 )</td>
<td>( )</td>
<td>( )</td>
</tr>
</tbody>
</table>
Forward Algorithm

\[ \pi = \{0.3, 0.3, 0.4\} \]

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

<table>
<thead>
<tr>
<th></th>
<th>A Transition</th>
<th>B Emission</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>0.2 0.5 0.3</td>
<td>0.5 0.2 0.3</td>
</tr>
<tr>
<td>C2</td>
<td>0.4 0.4 0.2</td>
<td>0.2 0.3 0.5</td>
</tr>
<tr>
<td>C3</td>
<td>0.1 0.4 0.5</td>
<td>0.1 0.1 0.8</td>
</tr>
</tbody>
</table>

1) Initialization:
\[ \alpha_1(i) = \pi_i b_i(O_1), \quad 1 \leq i \leq N. \]

2) Induction:
\[ \alpha_t(j) = \left[ \sum_{i=1}^{N} \alpha_{t-1}(i) a_{ij} \right] b_j(O_t), \quad 1 \leq t \leq T - 1 \]
\[ 1 \leq j \leq N. \]

3) Termination:
\[ P(O \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i). \]

<table>
<thead>
<tr>
<th></th>
<th>( t = 1, O_t = \text{F1} )</th>
<th>( t = 2, O_t = \text{F3} )</th>
<th>( t = 3, O_t = \text{F3} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>0.3 \cdot 0.5 = 0.15</td>
<td>((0.15 \cdot 0.2 + 0.06 \cdot 0.4 + 0.04 \cdot 0.1) \cdot 0.3 = 0.017)</td>
<td></td>
</tr>
<tr>
<td>C2</td>
<td>0.3 \cdot 0.2 = 0.06</td>
<td>((0.15 \cdot 0.5 + 0.06 \cdot 0.4 + 0.04 \cdot 0.4) \cdot 0.5 = 0.058)</td>
<td></td>
</tr>
<tr>
<td>C3</td>
<td>0.4 \cdot 0.1 = 0.04</td>
<td>((0.15 \cdot 0.3 + 0.06 \cdot 0.2 + 0.04 \cdot 0.5) \cdot 0.8 = 0.062)</td>
<td></td>
</tr>
</tbody>
</table>
Forward Algorithm

\[ \pi = \{.3, .3, .4\} \]

What is \( P("F1 \ F3 \ F3" \mid \lambda) \)?

\[ .010 + .028 + .038 = .076 \]

<table>
<thead>
<tr>
<th></th>
<th>Transition</th>
<th>Emission</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>0.5</td>
<td>0.3</td>
</tr>
<tr>
<td>0.4</td>
<td>0.4</td>
<td>0.2</td>
</tr>
<tr>
<td>0.1</td>
<td>0.4</td>
<td>0.5</td>
</tr>
<tr>
<td>A</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1) Initialization:
\[ \alpha_t(i) = \pi_i b_i(O_t), \quad 1 \leq i \leq N. \]

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:
\[ P(O \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i). \]
Backward Algorithm

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

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

1) Initialization:

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

2) Induction:

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

$t = T - 1, T - 2, \ldots, 1, \quad 1 \leq i \leq N.$
Backward Algorithm

\[ \pi = \{.3, .3, .4\} \]

What is \( P("F1 F3 F3" | \lambda) \)?

<table>
<thead>
<tr>
<th></th>
<th>A Transition</th>
<th>B Emission</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.2 0.5 0.3</td>
<td>0.5 0.2 0.3</td>
</tr>
<tr>
<td></td>
<td>0.4 0.4 0.2</td>
<td>0.2 0.3 0.5</td>
</tr>
<tr>
<td></td>
<td>0.1 0.4 0.5</td>
<td>0.1 0.1 0.8</td>
</tr>
</tbody>
</table>

1) Initialization:
\[ \beta_t(i) = 1, \quad 1 \leq i \leq N. \]

2) Induction:
\[ \beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j), \]
\[ t = T - 1, T - 2, \cdots, 1, \quad 1 \leq i \leq N. \]

<table>
<thead>
<tr>
<th>( t = 1, O_{t+1} = F3 )</th>
<th>( t = 2, O_{t+1} = F3 )</th>
<th>( T = 3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>C_1</td>
<td>( 0.2 \cdot 0.3 \cdot 1 + 0.5 \cdot 0.5 \cdot 1 + 0.3 \cdot 0.8 \cdot 1 = 0.55 )</td>
<td>1</td>
</tr>
<tr>
<td>C_2</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>C_3</td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>
Backward Algorithm

\[ \pi = \{0.3, 0.3, 0.4\} \]

What is \( P("F1 F3 F3" | \lambda) \)?

1) Initialization:
\[ \beta_t(i) = 1, \quad 1 \leq i \leq N. \]

2) Induction:
\[ \beta_t(i) = \sum_{j=1}^{N} \alpha_{i} b_{j}(O_{t+1}) \beta_{t+1}(j), \]
\[ t = T - 1, T - 2, \ldots, 1, 1 \leq i \leq N. \]

<table>
<thead>
<tr>
<th>( t = 1, O_{t+1} = F3 )</th>
<th>( t = 2, O_{t+1} = F3 )</th>
<th>( T = 3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{C}_1 )</td>
<td>( .2 \cdot 0.3 \cdot 1 + .5 \cdot 0.5 \cdot 1 + .3 \cdot 0.8 \cdot 1 = .55 )</td>
<td>1</td>
</tr>
<tr>
<td>( \text{C}_2 )</td>
<td>( .4 \cdot 0.3 \cdot 1 + .4 \cdot 0.5 \cdot 1 + .2 \cdot 0.8 \cdot 1 = .48 )</td>
<td>1</td>
</tr>
<tr>
<td>( \text{C}_3 )</td>
<td>( .1 \cdot 0.3 \cdot 1 + .4 \cdot 0.5 \cdot 1 + .5 \cdot 0.8 \cdot 1 = .63 )</td>
<td>1</td>
</tr>
</tbody>
</table>
Backward Algorithm

\[ \pi = \{0.3, 0.3, 0.4\} \]

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

\[ \sum \pi_i b_i(O_1) \beta_t(i) = 
0.3 \cdot 0.3 \cdot 0.5 + 0.3 \cdot 0.26 \cdot 0.2 + 0.4 \cdot 0.36 \cdot 1 = 
0.045 + 0.016 + 0.014 = 0.076 \]

<table>
<thead>
<tr>
<th></th>
<th>( t = 1, O_{t+1} = \text{F3} )</th>
<th>( t = 2, O_{t+1} = \text{F3} )</th>
<th>T=3</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>( 0.2 \cdot 0.5 \cdot 0.3 + 0.4 \cdot 0.4 \cdot 0.3 + 0.1 \cdot 0.4 \cdot 0.5 ) = 0.30</td>
<td>( 0.2 \cdot 0.3 \cdot 0.5 + 0.5 \cdot 0.5 \cdot 0.3 ) = 0.55</td>
<td>1</td>
</tr>
<tr>
<td>C2</td>
<td>( 0.1 \cdot 0.5 \cdot 0.48 + 0.4 \cdot 0.5 \cdot 0.48 + 0.2 \cdot 0.8 \cdot 0.63 ) = 0.36</td>
<td>( 0.4 \cdot 0.3 \cdot 1 + 0.4 \cdot 0.5 \cdot 1 + 0.8 \cdot 1 ) = 0.48</td>
<td>1</td>
</tr>
</tbody>
</table>

1) Initialization:
\[ \beta_t(i) = 1, \quad 1 \leq i \leq N. \]

2) Induction:
\[ \beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j), \quad t = T - 1, T - 2, \ldots, 1, 1 \leq i \leq N. \]
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:
\[ \delta_t(i) = \pi_i b_i(O_t), \quad 1 \leq i \leq N \]
\[ \psi_t(i) = 0. \]

2) Recursion:
\[ \delta_t(j) = \max_{1 \leq i \leq N} \left[ \delta_{t-1}(i)a_{ij}b_j(O_t) \right], \quad 2 \leq t \leq T \]
\[ 1 \leq j \leq N \]
\[ \psi_t(j) = \arg\max_{1 \leq i \leq N} \left[ \delta_{t-1}(i)a_{ij} \right], \quad 2 \leq t \leq T \]
\[ 1 \leq j \leq N. \]

3) Termination:
\[ P^* = \max_{1 \leq i \leq N} [\delta_T(i)] \]
\[ q^*_T = \arg\max_{1 \leq i \leq N} [\delta_T(i)]. \]

4) Path (state sequence) backtracking:
\[ q^*_t = \psi_{t+1}(q^*_{t+1}), \quad t = T - 1, T - 2, \ldots, 1. \]
Viterbi Algorithm

\[ \pi = \{0.3, 0.3, 0.4\} \]

What is most probable state sequence given "F1 F3 F3" and \( \lambda \)? C1, C3, C3

<table>
<thead>
<tr>
<th>A Transition</th>
<th>B Emission</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2 0.5 0.3</td>
<td>0.5 0.2 0.3</td>
</tr>
<tr>
<td>0.4 0.4 0.2</td>
<td>0.2 0.3 0.5</td>
</tr>
<tr>
<td>0.1 0.4 0.5</td>
<td>0.1 0.1 0.8</td>
</tr>
</tbody>
</table>

1) Initialization:
\[
\delta_i(i) = \pi_i b_i(O_t), \quad 1 \leq i \leq N
\]
\[
\psi_i(i) = 0.
\]

2) Recursion:
\[
\delta_i(j) = \max_{1 \leq i \leq N} [\delta_{i-1}(i) a_{ij}] b_j(O_{t+1}), \quad 2 \leq t \leq T
\]
\[
\psi_i(j) = \arg \max_{1 \leq i \leq N} [\delta_{i-1}(i) a_{ij}], \quad 2 \leq t \leq T
\]
\[
1 \leq j \leq N.
\]

3) Termination:
\[
P^* = \max_{1 \leq i \leq N} [\delta_T(i)]
\]
\[
q_T^* = \arg \max_{1 \leq i \leq N} [\delta_T(i)].
\]

4) Path (state sequence) backtracking:
\[
q_t^* = \psi_{t+1}(q_{t+1}^*), \quad t = T-1, T-2, \ldots, 1.
\]
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 /æt/ 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
Model Types

- **One to One**
- **One to Many**
- **Many to One**
- **Many to Many**

Each diagram represents a different type of model relationship, illustrating the connections between entities.
Model Types

- **One to One**
- **One to Many**
- **Many to One**
- **Many to Many**
Translation Models

- Uses 2 RNNs
- State of first RNN initializes second
- Inputs fed into first RNN
- Can generate arbitrary length sequences
I like to sleep.
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

Waveform

Framewise

CTC

"the"

"sound"

"of"
Let us assume an Alphabet $A$

We define an alphabet $A' = \{A \cup \_\}$

The output layer will contain $|A'|$ nodes.

Lets have $A' = \{a,c,t,\_\}$
CTC - Paths

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

- For $T = 1$

- $A^T = \{(c),(a),(t),(_)}$

- Given some network output $y = N(x)$ for some input $x$, what is the probability of some $\pi$?
CTC - Path Probabilities

- For $T = 1$

- $A^T = \{(c),(a),(t),(_,)\}$

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

\[ p(\pi \mid x) = y_\pi \]
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 $y$, what is the probability of some $\pi$?

\[
p(\pi|\mathbf{x}) = \prod_{t=1}^{T} y_{\pi_t}^t
\]
CTC - Labelings

- Paths $\pi$ are as long as the input sequence
- Labels $l$ are the actual predictions and could be shorter than $\pi$
- A label $l \in A^{\leq T}$
- Need a mapping $F: A^T \rightarrow A^{\leq T}$
Define $F$ to simply remove repeated symbols and all blanks

- $(c,c,c,a,a,a,t,t,t)$
- $(c,c,\_a,a,t,\_)$
- $(c,a,t)$
- $(c,\_\_a,a,\_t,t)$
- $(c,a,a,a,a,a,a,t,t)$
CTC - Label Probability

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

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

Exponential Number of Paths

\[ p(1|x) = \sum_{\pi \in F^{-1}(1)} p(\pi|x) \]
Let $l'$ be $l$ with added blanks

$(c,a,t) \rightarrow (\_c,\_a,\_t,\_)$

Forward variable:

$\alpha(t,u)$

Summed probabilities of all paths of time $t$ with the first $u$ elements of $l'$
CTC - Forward-Backward

- **Initialization:**
  \[
  \alpha(1, 1) = y^{1}_b \\
  \alpha(1, 2) = y^{1}_{i_1} \\
  \alpha(1, u) = 0, \ \forall u > 2
  \]

- **Recurrence:**
  \[
  \alpha(t, u) = y^{t}_{i_u} \sum_{i=f(u)}^{u} \alpha(t - 1, i)
  \]

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

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

\[ \alpha(1, 1) = y^1_b \]
\[ \alpha(1, 2) = y^1_i \]
\[ \alpha(1, u) = 0, \quad \forall u > 2 \]

\[ \alpha(t, u) = y^{t}_{i_u} \sum_{i=f(u)}^{u} \alpha(t-1, i) \]

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

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

\[ \alpha(1, 1) = y_b^1 \]
\[ \alpha(1, 2) = y_i^1 \]
\[ \alpha(1, u) = 0, \ \forall u > 2 \]

\[ \alpha(t, u) = y_{i_u}^t \sum_{i=f(u)}^u \alpha(t-1, i) \]

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

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>C</th>
<th>T</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.2</td>
<td>0.1</td>
<td>0.1</td>
<td>0.6</td>
</tr>
<tr>
<td>2</td>
<td>0.0</td>
<td>0.7</td>
<td>0.2</td>
<td>0.1</td>
</tr>
<tr>
<td>3</td>
<td>0.2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.8</td>
</tr>
<tr>
<td>4</td>
<td>0.6</td>
<td>0.1</td>
<td>0.1</td>
<td>0.2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>_</th>
<th>c</th>
<th>_</th>
<th>_</th>
<th>a</th>
<th>_</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.6</td>
<td>0.7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0.1 \times (0+0.6) = 0.06</td>
<td>0.7 \times (0+0.6+0.7) = 0.91</td>
<td>0</td>
<td>0</td>
<td>0 _</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Example

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

\[ \alpha(1, 1) = y^1_b \]
\[ \alpha(1, 2) = y^1_{i_1} \]
\[ \alpha(1, u) = 0, \ \forall u > 2 \]

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

\[ \alpha(t, u) = y^t_{i_u} \sum_{i=f(u)}^u \alpha(t-1, i) \]

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>C</th>
<th>T</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.2</td>
<td>0.1</td>
<td>0.1</td>
<td>0.6</td>
</tr>
<tr>
<td>2</td>
<td>0.0</td>
<td>0.7</td>
<td>0.2</td>
<td>0.1</td>
</tr>
<tr>
<td>3</td>
<td>0.2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.8</td>
</tr>
<tr>
<td>4</td>
<td>0.6</td>
<td>0.1</td>
<td>0.1</td>
<td>0.2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>–</th>
<th></th>
<th>c</th>
<th>–</th>
<th>a</th>
<th>–</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>–</td>
<td>0.6</td>
<td>0.7</td>
<td>0</td>
<td>0.1</td>
</tr>
<tr>
<td>2</td>
<td>0.1*(0+0.6) = 0.06</td>
<td>0.7*(0+0.6+0.7) = 0.91</td>
<td>0.1*(0.7+0) = 0.07</td>
<td>0*(0.7+0+0) = 0</td>
<td>0.1*(0+0) = 0</td>
</tr>
<tr>
<td>3</td>
<td>0.8*(0+0.06) = 0.048</td>
<td>0.0*(0+0.06+0.91) = 0.0</td>
<td>0.8*(0.91+0.07) = 0.784</td>
<td>0.2*(0.91+0.07+0) = 0.196</td>
<td>0.8*(0+0) = 0</td>
</tr>
<tr>
<td>4</td>
<td>0.2*(0+0.048) = 0.0096</td>
<td>0.1*(0+0.048+0) = 0.0048</td>
<td>0.2*(0.0+0.784) = 0.157</td>
<td>0.6*(0+0.784+0.196) = 0.588</td>
<td>0.2*(0.196+0) = 0.0392</td>
</tr>
</tbody>
</table>
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:**

\[
\beta(T, U') = \beta(T, U' - 1) = 1 \\
\beta(T, u) = 0, \quad \forall u < U' - 1
\]

**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

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

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

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

\[ \beta(t, u) = \sum_{i=u}^{g(u)} \beta(t+1, i)y_{i+1}^{t+1} \]

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>C</th>
<th>T</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.2</td>
<td>0.1</td>
<td>0.1</td>
<td>0.6</td>
</tr>
<tr>
<td>2</td>
<td>0.0</td>
<td>0.7</td>
<td>0.2</td>
<td>0.1</td>
</tr>
<tr>
<td>3</td>
<td>0.2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.8</td>
</tr>
<tr>
<td>4</td>
<td>0.6</td>
<td>0.1</td>
<td>0.1</td>
<td>0.2</td>
</tr>
</tbody>
</table>
Example

\[ T = 4 \]

\[ l = (c, a) \]

\[ l' = (_, c, _, a, _) \]

\[ \beta(T, U') = \beta(T, U' - 1) = 1 \]

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

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

\[ \beta(t, u) = \sum_{i=u}^{g(u)} \beta(t + 1, i) y_{i,t+1}^i \]

|   | A | C | T |  
|---|---|---|---|---|
| 1 | 0.2 | 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 |

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>c</th>
<th></th>
<th>a</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1<em>0.6+1</em>0.2+0 = 0.8</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

1*0.2+0 = 0.2
Example

\[ T = 4 \]
\[ l = (c,a) \]
\[ l' = (__,c,__,a,__) \]

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

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

\[
\beta(t, u) = \sum_{i=u}^{g(u)} \beta(t + 1, i)y_{ti}^{t+1}
\]

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>C</th>
<th>T</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.2</td>
<td>0.1</td>
<td>0.1</td>
<td>0.6</td>
</tr>
<tr>
<td>2</td>
<td>0.0</td>
<td>0.7</td>
<td>0.2</td>
<td>0.1</td>
</tr>
<tr>
<td>3</td>
<td>0.2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.8</td>
</tr>
<tr>
<td>4</td>
<td>0.6</td>
<td>0.1</td>
<td>0.1</td>
<td>0.2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>_</th>
<th></th>
<th>c</th>
<th></th>
<th></th>
<th>a</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0<em>0.1+0.64</em>0.7 = 0.448</td>
<td>0.64<em>0.7+0.64</em>0.1+0.32*0 = 0.512</td>
<td>0.64<em>0.1+0.32</em>0 = 0.064</td>
<td>0.32<em>0+0.16</em>0.1+0 = 0.016</td>
<td>0.16*0.1+0 = 0.016</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0<em>0.8+0.6</em>0 = 0</td>
<td>0.6<em>0+0.6</em>0.8+0.8*0.2 = 0.64</td>
<td>0.6<em>0.8+0.8</em>0.2 = 0.64</td>
<td>0.8<em>0.2+0.2</em>0.8+0 = 0.32</td>
<td>0.2*0.8+0 = 0.16</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0<em>0.2+0</em>0.1 = 0</td>
<td>0<em>0.1+0</em>0.2+1*0.6 = 0.6</td>
<td>0<em>0.2+1</em>0.6 = 0.6</td>
<td>1<em>0.6+1</em>0.2+0 = 0.8</td>
<td>1*0.2+0 = 0.2</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
### CTC - Label Probability

\[
p(\mathbf{z} | \mathbf{x}) = \sum_{u=1}^{\mid \mathbf{z} \mid} \alpha(t, u) \beta(t, u)
\]

#### Forward:

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.6</td>
<td>0.7</td>
<td>0.07</td>
</tr>
<tr>
<td>2</td>
<td>0.1*(0+0.6) = 0.06</td>
<td>0.7*(0+0.6+0.7) = 0.91</td>
<td>0.1*(0.7+0) = 0.07</td>
</tr>
<tr>
<td>3</td>
<td>0.8*(0+0.06) = 0.048</td>
<td>0.0*(0+0.06+0.91) = 0.0</td>
<td>0.8*(0.91+0.07) = 0.784</td>
</tr>
<tr>
<td>4</td>
<td>0.2*(0+0.048) = 0.0096</td>
<td>0.1*(0+0.048+0) = 0.0048</td>
<td>0.2*(0.0+0.784) = 0.157</td>
</tr>
</tbody>
</table>

#### Backward:

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.64<em>0.7+0.64</em>0.1+0.32*0 = 0.512</td>
<td>0.64<em>0.1+0.32</em>0 = 0.064</td>
<td>0.32<em>0+0.16</em>0.1+0 = 0.016</td>
</tr>
<tr>
<td>2</td>
<td>0.8+0.6*0 = 0</td>
<td>0.6<em>0+0.6</em>0.8+0.8*0.2 = 0.64</td>
<td>0.8<em>0.2+0.2</em>0.8+0 = 0.32</td>
</tr>
<tr>
<td>3</td>
<td>0.2+0*0.1 = 0</td>
<td>0.1+0<em>0.2+1</em>0.6 = 0.6</td>
<td>1<em>0.6+1</em>0.2+0 = 0.8</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

64
CTC - Label Probability

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

### Forward:

<table>
<thead>
<tr>
<th></th>
<th>c</th>
<th></th>
<th>a</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.6</td>
<td>0.7</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0.1*(0+0.6) = 0.06</td>
<td>0.7*(0+0.6+0.7) = 0.91</td>
<td>0.1*(0.7+0) = 0.07</td>
<td>0*(0.7+0+0) = 0</td>
</tr>
<tr>
<td>3</td>
<td>0.8*(0+0.06) = 0.048</td>
<td>0.0*(0+0.06+0.91) = 0.0</td>
<td>0.8*(0.91+0.07) = 0.784</td>
<td>0.2*(0.91+0.07+0) = 0.196</td>
</tr>
<tr>
<td>4</td>
<td>0.2*(0+0.048) = 0.0096</td>
<td>0.1*(0+0.048+0) = 0.0048</td>
<td>0.2*(0.0+0.784) = 0.157</td>
<td>0.6*(0+0.784+0.196) = 0.588</td>
</tr>
</tbody>
</table>

### Backward:

<table>
<thead>
<tr>
<th></th>
<th>c</th>
<th></th>
<th>a</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0<em>0.1+0.64</em>0.7 = 0.448</td>
<td>0.64<em>0.7+0.64</em>0.1+0.32*0 = 0.512</td>
<td>0.64<em>0.1+0.32</em>0 = 0.064</td>
<td>0.32<em>0+0.16</em>0.1+0 = 0.016</td>
</tr>
<tr>
<td>2</td>
<td>0<em>0.8+0.6</em>0 = 0</td>
<td>0.6<em>0+0.6</em>0.8+0.8*0.2 = 0.64</td>
<td>0.6<em>0.8+0.8</em>0.2 = 0.64</td>
<td>0.8<em>0.2+0.2</em>0.8+0 = 0.32</td>
</tr>
<tr>
<td>3</td>
<td>0<em>0.2+0</em>0.1 = 0</td>
<td>0<em>0.1+0</em>0.2+1*0.6 = 0.6</td>
<td>0<em>0.2+1</em>0.6 = 0.6</td>
<td>1<em>0.6+1</em>0.2+0 = 0.8</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
CTC - Loss

- Minimize negative log likelihood of data

\[ L = \frac{1}{z'} \sum_{u=1}^{z'} \alpha(t, u) \beta(t, u) \]

\[ l = (c, a) \quad L = 0.4665 \]
CTC - Gradient

Helper Function:

\[ l = (c, a) \]
\[ l' = (\_, c, \_, a, \_) \]

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

Gradient:

\[ \frac{\partial L(x, z)}{\partial a_{k}^{t}} = y_{k}^{t} - \frac{1}{p(z|x)} \sum_{u \in B(z, k)} \alpha(t, u) \beta(t, u) \]
**CTC - Gradient**

\[ p(l \mid x) = 0.6272 \]

\[ B(l,a) = \{4\} \quad B(l,c) = \{2\} \]

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

\[ B(l,t) = \{\} \]

- **Forward:**

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.6</td>
<td>0.7</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0.1*(0+0.6) = 0.06</td>
<td>0.7*(0+0.6+0.7) = 0.91</td>
<td>0.1*(0.7+0) = 0.07</td>
<td>0*(0.7+0+0) = 0</td>
</tr>
<tr>
<td>3</td>
<td>0.8*(0+0.06) = 0.048</td>
<td>0.0*(0+0.06+0.91) = 0.0</td>
<td>0.8*(0.91+0.07) = 0.784</td>
<td>0.2*(0.91+0.07+0) = 0.196</td>
</tr>
<tr>
<td>4</td>
<td>0.2*(0+0.048) = 0.0096</td>
<td>0.1*(0+0.048+0) = 0.0048</td>
<td>0.2*(0.0+0.784) = 0.157</td>
<td>0.6*(0+0.784+0.196) = 0.588</td>
</tr>
</tbody>
</table>

- **Backward:**

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0<em>0.1+0.64</em>0.7 = 0.448</td>
<td>0.64<em>0.7+0.64</em>0.1+0.32*0 = 0.512</td>
<td>0.64<em>0.1+0.32</em>0 = 0.064</td>
<td>0.32<em>0+0.16</em>0.1+0 = 0.016</td>
</tr>
<tr>
<td>2</td>
<td>0<em>0.8+0.6</em>0 = 0</td>
<td>0.6<em>0+0.6</em>0.8+0.8*0.2 = 0.64</td>
<td>0.6<em>0.8+0.8</em>0.2 = 0.64</td>
<td>0.8<em>0.2+0.2</em>0.8+0 = 0.32</td>
</tr>
<tr>
<td>3</td>
<td>0<em>0.2+0</em>0.1 = 0</td>
<td>0<em>0.1+0</em>0.2+1*0.6 = 0.6</td>
<td>0<em>0.2+1</em>0.6 = 0.6</td>
<td>1<em>0.6+1</em>0.2+0 = 0.8</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

\[
\frac{\partial L(x,z)}{\partial a_k^t} = y_k^t - \frac{1}{p(z|x)} \sum_{u \in B(z,k)} \alpha(t,u)\beta(t,u)
\]
### CTC - Gradient

\[ p(l \mid x) = 0.6272 \]

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

**Forward:**

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>_</td>
<td>c</td>
<td>_</td>
<td>a</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>0.6</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0.1*(0+0.6) = 0.06</td>
<td>0.7*(0+0.6+0.7) = 0.91</td>
<td>0.1*(0.7+0) = 0.07</td>
<td>0*(0.7+0+0) = 0</td>
</tr>
<tr>
<td>3</td>
<td>0.8*(0+0.06) = 0.048</td>
<td>0.0*(0+0.06+0.91) = 0.0</td>
<td>0.8*(0.91+0.07) = 0.784</td>
<td>0.2*(0.91+0.07+0) = 0.196</td>
</tr>
<tr>
<td>4</td>
<td>0.2*(0+0.048) = 0.0096</td>
<td>0.1*(0+0.048+0) = 0.0048</td>
<td>0.2*(0.0+0.784) = 0.157</td>
<td>0.6*(0+0.784+0.196) = 0.588</td>
</tr>
</tbody>
</table>

**Backward:**

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>_</td>
<td>c</td>
<td>_</td>
<td>a</td>
</tr>
<tr>
<td>1</td>
<td>0<em>0.1+0.64</em>0.7 = 0.448</td>
<td>0.64<em>0.7+0.64</em>0.1+0.32*0 = 0.512</td>
<td>0.64<em>0.1+0.32</em>0 = 0.064</td>
<td>0.32<em>0+0.16</em>0.1+0 = 0.016</td>
</tr>
<tr>
<td>2</td>
<td>0<em>0.8+0.6</em>0 = 0</td>
<td>0.6<em>0+0.6</em>0.8+0.8*0.2 = 0.64</td>
<td>0.6<em>0.8+0.8</em>0.2 = 0.64</td>
<td>0.8<em>0.2+0.2</em>0.8+0 = 0.32</td>
</tr>
<tr>
<td>3</td>
<td>0<em>0.2+0</em>0.1 = 0</td>
<td>0<em>0.1+0</em>0.2+1*0.6 = 0.6</td>
<td>0<em>0.2+1</em>0.6 = 0.6</td>
<td>1<em>0.6+1</em>0.2+0 = 0.8</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

\[
\frac{\partial L(x, z)}{\partial a_k^t} = y_k^t - \frac{1}{p(z \mid x)} \sum_{u \in B(z,k)} \alpha(t, u) \beta(t, u)
\]
CTC - Decoding

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

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>C</th>
<th>T</th>
<th>_</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.2</td>
<td>0.1</td>
<td>0.1</td>
<td>0.6</td>
</tr>
<tr>
<td>2</td>
<td>0.0</td>
<td>0.7</td>
<td>0.2</td>
<td>0.1</td>
</tr>
<tr>
<td>3</td>
<td>0.2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.8</td>
</tr>
<tr>
<td>4</td>
<td>0.6</td>
<td>0.1</td>
<td>0.1</td>
<td>0.2</td>
</tr>
</tbody>
</table>

Idea??
CTC - Decoding

- Given a new input $x$
- Want $l^* = \arg\max p(l \mid x)$
- Greedy
- Beam Search
- Prefix Search

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>C</th>
<th>T</th>
<th>_</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.2</td>
<td>0.1</td>
<td>0.1</td>
<td>0.6</td>
</tr>
<tr>
<td>2</td>
<td>0.0</td>
<td>0.7</td>
<td>0.2</td>
<td>0.1</td>
</tr>
<tr>
<td>3</td>
<td>0.2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.8</td>
</tr>
<tr>
<td>4</td>
<td>0.6</td>
<td>0.1</td>
<td>0.1</td>
<td>0.2</td>
</tr>
</tbody>
</table>

$\text{p}(l=\text{blank}) = \text{p}(- -) = 0.7 \times 0.6 = 0.42$

$p(l=A) = p(AA) + p(A-) + p(-A)$

$= 0.3 \times 0.4 + 0.3 \times 0.6 + 0.7 \times 0.4$

$= 0.58$
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

Digit: Avg. Edit Distance

Digit: Avg. Sequence Error

- Train
- Validation
Results: Ascii
Results: Ascii

Ascii: Avg. Edit Distance

Ascii: Avg. Sequence Error

- Train
- Validation
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


https://www.uea.ac.uk/computing/research-at-the-uea-speech-group
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?