

# 1 Calculate Probability for a Labelling Sequence

$$\text{score}(\mathbf{l}|s) = \sum_{j=1}^m \sum_{i=1}^n w_j f_j(s, i, l_i, l_{i-1})$$

$m$  feature functions  $f$  and  $m$  weights  $w$  for each feature function.  $n$  tokens in the observation sequence  $s$ .

$$p(\mathbf{l}|s) = \frac{\exp[\text{score}(\mathbf{l}|s)]}{\sum_{\mathbf{l}' \in \mathbf{L}} \exp[\text{score}(\mathbf{l}'|s)]} = \frac{\exp[\sum_{j=1}^m \sum_{i=1}^n w_j f_j(s, i, l_i, l_{i-1})]}{\sum_{\mathbf{l}' \in \mathbf{L}} \exp[\sum_{j=1}^m \sum_{i=1}^n w_j f_j(s, i, l'_i, l'_{i-1})]}$$

$\mathbf{L}$  are all the possible labellings for observation sequence  $s$ . If the size of tagset is 2 and the length of  $s$  is 3, there are  $2^3 = 8$  possible labellings (i.e.  $|\mathbf{L}| = 8$ ).

## 2 Learning Weights

Use stochastic gradient ascent to learn weight  $w_j$  for function  $f_j$  from training data  $s$ :

$$\frac{\partial \log p(\mathbf{l}|s)}{\partial w_j} = \sum_{i=1}^n f_j(s, i, l_i, l_{i-1}) - \sum_{\mathbf{l}' \in \mathbf{L}} p(\mathbf{l}'|s) \sum_{i=1}^n f_j(s, i, l'_i, l'_{i-1})$$

- $\mathbf{l}$  is the gold standard labelling of training instance  $s$ .
- $\mathbf{L}$  contains all the possible labellings for training instance  $s$ .
- $\sum_{i=1}^n f_j(s, i, l_i, l_{i-1})$ : how many times the feature function  $f_j$  is actually activated (i.e. return 1) by instance  $s$ —i.e. the contribution of  $f_j$  under the true labelling.
- $p(\mathbf{l}'|s)$ : the probability of a labelling  $\mathbf{l}'$  for  $s$ , under the current weights.
- $\sum_{\mathbf{l}' \in \mathbf{L}} p(\mathbf{l}'|s) \sum_{i=1}^n f_j(s, i, l'_i, l'_{i-1})$ : the expected contribution of  $f_j$ , under the current weights.

$$w_j = w_j + \alpha \times \frac{\partial \log p(\mathbf{l}|s)}{\partial w_j}$$

## 3 Example

Lee loves tea (t) and coffee (c). However, he is only allowed to drink one cup of tea (t) or coffee (c) per day, on Monday, Tuesday and Wednesday.

Assume Lee's drinking behaviour on a particular day is affected by many factors, including the weather (sunny or rainy) and what he drinks on the previous day.

In week one, the weather for Monday, Tuesday and Wednesday is [sunny (s), rainy (r), rainy (r)], and Lee's drinking behaviour is [coffee (c), tea (t), tea (t)]. In week two, the weather is [r, s, r] and Lee drinks [c, t, c]. In week three, the weather forecast is [s, s, r], what's Lee's drinking behaviour?

In CRF++, our training file, test file and template file will look something like this:

For this settings, the two feature templates will generate 10 feature functions:

|   |   |
|---|---|
| s | c |
| r | t |
| r | t |
| r | c |
| s | t |
| r | c |

|   |
|---|
| s |
| s |
| r |

|             |
|-------------|
| U00:%x[0,0] |
| B           |

Table 2: Test file

Table 3: Template file

Table 1: Training file

- ff0=if (f=U00:s and current\_label=c) return 1 else return 0
- ff1=if (f=U00:s and current\_label=t) return 1 else return 0
- ff2=if (f=U00:r and current\_label=c) return 1 else return 0
- ff3=if (f=U00:r and current\_label=t) return 1 else return 0
- ff4=if (previous\_label=t and current\_label=t) return 1 else return 0
- ff5=if (previous\_label=c and current\_label=t) return 1 else return 0
- ff6=if (previous\_label=t and current\_label=c) return 1 else return 0
- ff7=if (previous\_label=c and current\_label=c) return 1 else return 0
- ff8=if (previous\_label=None and current\_label=c) return 1 else return 0
- ff9=if (previous\_label=None and current\_label=t) return 1 else return 0

For convenience:

- f0: s, c
- f1: s, t
- f2: r, c
- f3: r, t
- f4: t, t
- f5: c, t
- f6: t, c
- f7: c, c
- f8: None, c
- f9: None, t

### 3.1 Learning weights: iteration one

Now, we start to learn weights by using stochastic gradient ascent. Let's start with all weights 1:  $w_0 = w_1 = \dots = w_9 = 1$ . In the first iteration, we randomly pick up a training instance which happens to be [s, r, r] with gold-standard labelling [c, t, t]. In order to calculate this:

$$\frac{\partial \log p(\mathbf{l}|s)}{\partial w_j} = \sum_{i=1}^n f_j(s, i, l_i, l_{i-1}) - \sum_{l' \in \mathbf{L}} p(l'|s) \sum_{i=1}^n f_j(s, i, l'_i, l'_{i-1})$$

for a weight  $w_j$ , we need to calculate all  $p(l'|s)$  first. There are  $2^3 = 8$  possible labelling sequences for the training instance [s, r, r]: We need to calculate a *score* for each possible

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| s | c | c | c | c | t | t | t | t |
| r | c | c | t | t | c | c | t | t |
| r | c | t | t | c | c | t | t | c |

labelling sequence, given the observation [s, r, r], using this formula:

$$score(\mathbf{l}|s) = \sum_{j=1}^m \sum_{i=1}^n w_j f_j(s, i, l_i, l_{i-1})$$

- $score(\mathbf{l}_0|s) = (w_0 * f_0 + w_8 * f_8) + (w_2 * f_2 + w_7 * f_7) + (w_2 * f_2 + w_7 * f_7)$   
 $= (1 * 1 + 1 * 1) + (1 * 1 + 1 * 1) + (1 * 1 + 1 * 1) = 6.0$
- $score(\mathbf{l}_1|s) = (w_0 * f_0 + w_8 * f_8) + (w_2 * f_2 + w_7 * f_7) + (w_3 * f_3 + w_5 * f_5)$   
 $= (1 * 1 + 1 * 1) + (1 * 1 + 1 * 1) + (1 * 1 + 1 * 1) = 6.0$
- $score(\mathbf{l}_2|s) = (w_0 * f_0 + w_8 * f_8) + (w_3 * f_3 + w_5 * f_5) + (w_3 * f_3 + w_4 * f_4)$   
 $= (1 * 1 + 1 * 1) + (1 * 1 + 1 * 1) + (1 * 1 + 1 * 1) = 6.0$
- $score(\mathbf{l}_3|s) = (w_0 * f_0 + w_8 * f_8) + (w_3 * f_3 + w_5 * f_5) + (w_2 * f_2 + w_6 * f_6)$   
 $= (1 * 1 + 1 * 1) + (1 * 1 + 1 * 1) + (1 * 1 + 1 * 1) = 6.0$
- $score(\mathbf{l}_4|s) = (w_1 * f_1 + w_9 * f_9) + (w_2 * f_2 + w_6 * f_6) + (w_2 * f_2 + w_7 * f_7)$   
 $= (1 * 1 + 1 * 1) + (1 * 1 + 1 * 1) + (1 * 1 + 1 * 1) = 6.0$
- $score(\mathbf{l}_5|s) = (w_1 * f_1 + w_9 * f_9) + (w_2 * f_2 + w_6 * f_6) + (w_3 * f_3 + w_5 * f_5)$   
 $= (1 * 1 + 1 * 1) + (1 * 1 + 1 * 1) + (1 * 1 + 1 * 1) = 6.0$
- $score(\mathbf{l}_6|s) = (w_1 * f_1 + w_9 * f_9) + (w_3 * f_3 + w_4 * f_4) + (w_3 * f_3 + w_4 * f_4)$   
 $= (1 * 1 + 1 * 1) + (1 * 1 + 1 * 1) + (1 * 1 + 1 * 1) = 6.0$
- $score(\mathbf{l}_7|s) = (w_1 * f_1 + w_9 * f_9) + (w_3 * f_3 + w_4 * f_4) + (w_2 * f_2 + w_6 * f_6)$   
 $= (1 * 1 + 1 * 1) + (1 * 1 + 1 * 1) + (1 * 1 + 1 * 1) = 6.0$

Then, we convert *scores* into probabilities by using this formula:

$$p(\mathbf{l}|s) = \frac{\exp[score(\mathbf{l}|s)]}{\sum_{l' \in \mathbf{L}} \exp[score(l'|s)]} = \frac{\exp[\sum_{j=1}^m \sum_{i=1}^n w_j f_j(s, i, l_i, l_{i-1})]}{\sum_{l' \in \mathbf{L}} \exp[\sum_{j=1}^m \sum_{i=1}^n w_j f_j(s, i, l'_i, l'_{i-1})]}$$

- $p(\mathbf{l}_0|s) = \frac{1}{8}$
- $p(\mathbf{l}_1|s) = \frac{1}{8}$
- $p(\mathbf{l}_2|s) = \frac{1}{8}$
- $p(\mathbf{l}_3|s) = \frac{1}{8}$
- $p(\mathbf{l}_4|s) = \frac{1}{8}$
- $p(\mathbf{l}_5|s) = \frac{1}{8}$
- $p(\mathbf{l}_6|s) = \frac{1}{8}$
- $p(\mathbf{l}_7|s) = \frac{1}{8}$

Now, we can calculate the gradient of each weight:

- $\frac{\partial \log p(\mathbf{l}|s)}{\partial w_0} = 1 - (0.125 * 1 + 0.125 * 1 + 0.125 * 1 + 0.125 * 1 + 0.125 * 0 + 0.125 * 0 + 0.125 * 0 + 0.125 * 0) = 0.5$
- $\frac{\partial \log p(\mathbf{l}|s)}{\partial w_1} = 0 - (0.125 * 0 + 0.125 * 0 + 0.125 * 0 + 0.125 * 0 + 0.125 * 1 + 0.125 * 1 + 0.125 * 1 + 0.125 * 1) = -0.5$
- $\frac{\partial \log p(\mathbf{l}|s)}{\partial w_2} = 0 - (0.125 * 2 + 0.125 * 1 + 0.125 * 0 + 0.125 * 1 + 0.125 * 2 + 0.125 * 1 + 0.125 * 0 + 0.125 * 1) = -1.0$
- $\frac{\partial \log p(\mathbf{l}|s)}{\partial w_3} = 2 - (0.125 * 0 + 0.125 * 1 + 0.125 * 2 + 0.125 * 1 + 0.125 * 0 + 0.125 * 1 + 0.125 * 2 + 0.125 * 1) = 1.0$
- $\frac{\partial \log p(\mathbf{l}|s)}{\partial w_4} = 1 - (0.125 * 0 + 0.125 * 0 + 0.125 * 1 + 0.125 * 0 + 0.125 * 0 + 0.125 * 0 + 0.125 * 2 + 0.125 * 1) = 0.5$
- $\frac{\partial \log p(\mathbf{l}|s)}{\partial w_5} = 1 - (0.125 * 0 + 0.125 * 1 + 0.125 * 1 + 0.125 * 1 + 0.125 * 0 + 0.125 * 1 + 0.125 * 0 + 0.125 * 0) = 0.5$
- $\frac{\partial \log p(\mathbf{l}|s)}{\partial w_6} = 0 - (0.125 * 0 + 0.125 * 0 + 0.125 * 0 + 0.125 * 1 + 0.125 * 1 + 0.125 * 1 + 0.125 * 0 + 0.125 * 1) = -0.5$
- $\frac{\partial \log p(\mathbf{l}|s)}{\partial w_7} = 0 - (0.125 * 2 + 0.125 * 1 + 0.125 * 0 + 0.125 * 0 + 0.125 * 1 + 0.125 * 0 + 0.125 * 0 + 0.125 * 0) = -0.5$
- $\frac{\partial \log p(\mathbf{l}|s)}{\partial w_8} = 1 - (0.125 * 1 + 0.125 * 1 + 0.125 * 1 + 0.125 * 1 + 0.125 * 0 + 0.125 * 0 + 0.125 * 0 + 0.125 * 0) = 0.5$
- $\frac{\partial \log p(\mathbf{l}|s)}{\partial w_9} = 0 - (0.125 * 0 + 0.125 * 0 + 0.125 * 0 + 0.125 * 0 + 0.125 * 1 + 0.125 * 1 + 0.125 * 1 + 0.125 * 1) = -0.5$

Last step of the first round: adjust the weights based on the each gradient:

$$w_j = w_j + \alpha \times \frac{\partial \log p(\mathbf{l}|s)}{\partial w_j}$$

Let's choose a small learning ratio of  $\alpha = 0.01$ :

- $w_0 = w_0 + 0.01 \times 0.5 = 1 + 0.01 \times 0.5 = 1.005$
- $w_1 = w_1 + 0.01 \times -0.5 = 1 + 0.01 \times -0.5 = 0.995$
- $w_2 = w_2 + 0.01 \times -1.0 = 1 + 0.01 \times -1.0 = 0.99$
- $w_3 = w_3 + 0.01 \times 1.0 = 1 + 0.01 \times 1.0 = 1.01$
- $w_4 = w_4 + 0.01 \times 0.5 = 1 + 0.01 \times 0.5 = 1.005$
- $w_5 = w_5 + 0.01 \times 0.5 = 1 + 0.01 \times 0.5 = 1.005$
- $w_6 = w_6 + 0.01 \times -0.5 = 1 + 0.01 \times -0.5 = 0.995$
- $w_7 = w_7 + 0.01 \times -0.5 = 1 + 0.01 \times -0.5 = 0.995$
- $w_8 = w_8 + 0.01 \times 0.5 = 1 + 0.01 \times 0.5 = 1.005$
- $w_9 = w_9 + 0.01 \times -0.5 = 1 + 0.01 \times -0.5 = 0.995$

Before changing the weights:

$$\log p(\mathbf{l}_{\text{trn}_0} | \text{trn}_0) + \log p(\mathbf{l}_{\text{trn}_1} | \text{trn}_1) = \log 0.125 + \log 0.125 = -4.159$$

After changing the weights:

$$\log p(\mathbf{l}_{\text{trn}_0} | \text{trn}_0) + \log p(\mathbf{l}_{\text{trn}_1} | \text{trn}_1) = \log 0.130 + \log 0.123 = -4.139$$

Therefore, the current (changed) weights lead to better overall prediction over training data. We go to the next iteration.

### 3.2 Learning weights: iteration two

In the second iteration, we randomly pick another instance from the training dataset. The picked instance happens to be [r, s r] with labelling sequence [c, t, c]. Again, there are 8 possible labelling sequences for this instance:

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| r | c | c | c | c | t | t | t | t |
| s | c | c | t | t | c | c | t | t |
| r | c | t | t | c | c | t | t | c |

Calculate the *score* for each possible labelling sequence:

- $score(\mathbf{l}_0 | s) = (w_2 * f_2 + w_8 * f_8) + (w_0 * f_0 + w_7 * f_7) + (w_2 * f_2 + w_7 * f_7)$   
 $= (0.99 * 1 + 1.005 * 1) + (1.005 * 1 + 0.995 * 1) + (0.99 * 1 + 0.995 * 1) = 5.98$
- $score(\mathbf{l}_1 | s) = (w_2 * f_2 + w_8 * f_8) + (w_0 * f_0 + w_7 * f_7) + (w_3 * f_3 + w_5 * f_5)$   
 $= (0.99 * 1 + 1.005 * 1) + (1.005 * 1 + 0.995 * 1) + (1.01 * 1 + 1.005 * 1) = 6.01$
- $score(\mathbf{l}_2 | s) = (w_2 * f_2 + w_8 * f_8) + (w_1 * f_1 + w_5 * f_5) + (w_3 * f_3 + w_4 * f_4)$   
 $= (0.99 * 1 + 1.005 * 1) + (0.995 * 1 + 1.005 * 1) + (1.01 * 1 + 1.005 * 1) = 6.01$
- $score(\mathbf{l}_3 | s) = (w_2 * f_2 + w_8 * f_8) + (w_1 * f_1 + w_5 * f_5) + (w_2 * f_2 + w_6 * f_6)$   
 $= (0.99 * 1 + 1.005 * 1) + (0.995 * 1 + 1.005 * 1) + (0.99 * 1 + 0.995 * 1) = 5.98$

- $score(\mathbf{l}_4|s) = (w_3 * f_3 + w_9 * f_9) + (w_0 * f_0 + w_6 * f_6) + (w_2 * f_2 + w_7 * f_7)$   
 $= (1.01 * 1 + 0.995 * 1) + (1.005 * 1 + 0.995 * 1) + (0.99 * 1 + 0.995 * 1) = 5.99$
- $score(\mathbf{l}_5|s) = (w_3 * f_3 + w_9 * f_9) + (w_0 * f_0 + w_6 * f_6) + (w_3 * f_3 + w_5 * f_5)$   
 $= (1.01 * 1 + 0.995 * 1) + (1.005 * 1 + 0.995 * 1) + (1.01 * 1 + 1.005 * 1) = 6.02$
- $score(\mathbf{l}_6|s) = (w_3 * f_3 + w_9 * f_9) + (w_1 * f_1 + w_4 * f_4) + (w_3 * f_3 + w_4 * f_4)$   
 $= (1.01 * 1 + 0.995 * 1) + (0.995 * 1 + 1.005 * 1) + (1.01 * 1 + 1.005 * 1) = 6.02$
- $score(\mathbf{l}_7|s) = (w_3 * f_3 + w_9 * f_9) + (w_1 * f_1 + w_4 * f_4) + (w_2 * f_2 + w_6 * f_6)$   
 $= (1.01 * 1 + 0.995 * 1) + (0.995 * 1 + 1.005 * 1) + (0.99 * 1 + 0.995 * 1) = 5.99$
- $score(\mathbf{l}_8|s) = (w_2 * f_2 + w_8 * f_8) + (w_1 * f_1 + w_5 * f_5) + (w_2 * f_2 + w_6 * f_6)$   
 $= (0.99 * 1 + 1.005 * 1) + (0.995 * 1 + 1.005 * 1) + (0.99 * 1 + 0.995 * 1) = 5.98$

Compute the probabilities:

- $p(\mathbf{l}_0|s) = 0.123$
- $p(\mathbf{l}_1|s) = 0.126$
- $p(\mathbf{l}_2|s) = 0.126$
- $p(\mathbf{l}_3|s) = 0.123$
- $p(\mathbf{l}_4|s) = 0.124$
- $p(\mathbf{l}_5|s) = 0.128$
- $p(\mathbf{l}_6|s) = 0.128$
- $p(\mathbf{l}_7|s) = 0.124$

Calculate the gradient for each weight:

- $\frac{\partial \log p(\mathbf{l}|s)}{\partial w_0} = 0 - (0.123 * 1 + 0.126 * 1 + 0.126 * 0 + 0.123 * 0 + 0.124 * 1 + 0.128 * 1 + 0.128 * 0 + 0.124 * 0) = -0.5$
- $\frac{\partial \log p(\mathbf{l}|s)}{\partial w_1} = 1 - (0.123 * 0 + 0.126 * 0 + 0.126 * 1 + 0.123 * 1 + 0.124 * 0 + 0.128 * 0 + 0.128 * 1 + 0.124 * 1) = 0.5$
- $\frac{\partial \log p(\mathbf{l}|s)}{\partial w_2} = 2 - (0.123 * 2 + 0.126 * 1 + 0.126 * 1 + 0.123 * 2 + 0.124 * 1 + 0.128 * 0 + 0.128 * 0 + 0.124 * 1) = 1.00999941672$
- $\frac{\partial \log p(\mathbf{l}|s)}{\partial w_3} = 0 - (0.123 * 0 + 0.126 * 1 + 0.126 * 1 + 0.123 * 0 + 0.124 * 1 + 0.128 * 2 + 0.128 * 2 + 0.124 * 1) = -1.00999941672$
- $\frac{\partial \log p(\mathbf{l}|s)}{\partial w_4} = 0 - (0.123 * 0 + 0.126 * 0 + 0.126 * 1 + 0.123 * 0 + 0.124 * 0 + 0.128 * 0 + 0.128 * 2 + 0.124 * 1) = -0.504999708359$   
 $\frac{\partial \log p(\mathbf{l}|s)}{\partial w_5} = 1 - (0.123 * 0 + 0.126 * 1 + 0.126 * 1 + 0.123 * 1 + 0.124 * 0 + 0.128 * 1 + 0.128 * 0 + 0.124 * 0) = 0.497500270808$
- $\frac{\partial \log p(\mathbf{l}|s)}{\partial w_6} = 1 - (0.123 * 0 + 0.126 * 0 + 0.126 * 0 + 0.123 * 1 + 0.124 * 1 + 0.128 * 1 + 0.128 * 0 + 0.124 * 1) = 0.502499729192$

- $\frac{\partial \log p(\mathbf{l}s)}{\partial w_7} = 0 - (0.123 * 2 + 0.126 * 1 + 0.126 * 0 + 0.123 * 0 + 0.124 * 1 + 0.128 * 0 + 0.128 * 0 + 0.124 * 0) = -0.495000291641$
- $\frac{\partial \log p(\mathbf{l}s)}{\partial w_8} = 1 - (0.123 * 1 + 0.126 * 1 + 0.126 * 1 + 0.123 * 1 + 0.124 * 0 + 0.128 * 0 + 0.128 * 0 + 0.124 * 0) = 0.502499979167$
- $\frac{\partial \log p(\mathbf{l}s)}{\partial w_9} = 0 - (0.123 * 0 + 0.126 * 0 + 0.126 * 0 + 0.123 * 0 + 0.124 * 1 + 0.128 * 1 + 0.128 * 1 + 0.124 * 1) = -0.502499979167$

Adjust the weights:

- $w_0 = w_0 + 0.01 \times -0.5 = 1.005 + 0.01 \times -0.5 = 1.0$
- $w_1 = w_1 + 0.01 \times 0.5 = 0.995 + 0.01 \times 0.5 = 1.0$
- $w_2 = w_2 + 0.01 \times 1.00999941672 = 0.99 + 0.01 \times 1.00999941672 = 1.00009999417$
- $w_3 = w_3 + 0.01 \times -1.00999941672 = 1.01 + 0.01 \times -1.00999941672 = 0.999900005833$
- $w_4 = w_4 + 0.01 \times -0.504999708359 = 1.005 + 0.01 \times -0.504999708359 = 0.999950002916$
- $w_5 = w_5 + 0.01 \times 0.497500270808 = 1.005 + 0.01 \times 0.497500270808 = 1.00997500271$
- $w_6 = w_6 + 0.01 \times 0.502499729192 = 0.995 + 0.01 \times 0.502499729192 = 1.00002499729$
- $w_7 = w_7 + 0.01 \times -0.495000291641 = 0.995 + 0.01 \times -0.495000291641 = 0.990049997084$
- $w_8 = w_8 + 0.01 \times 0.502499979167 = 1.005 + 0.01 \times 0.502499979167 = 1.01002499979$
- $w_9 = w_9 + 0.01 \times -0.502499979167 = 0.995 + 0.01 \times -0.502499979167 = 0.989975000208$

Before changing the weights:

$$\log p(\mathbf{l}_{\text{trn}_0} | \text{trn}_0) + \log p(\mathbf{l}_{\text{trn}_1} | \text{trn}_1) = \log 0.130 + \log 0.123 = -4.139$$

After changing the weights:

$$\log p(\mathbf{l}_{\text{trn}_0} | \text{trn}_0) + \log p(\mathbf{l}_{\text{trn}_1} | \text{trn}_1) = \log 0.127 + \log 0.128 = -4.119$$

Therefore, the current (changed) weights lead to better overall prediction over training data. We go to the next iteration.

### 3.3 After 212 iterations...

Current weights: [0.931, 1.069, 1.110, 0.890, 1.024, 1.602, 1.103, 0.271, 1.666, 0.334]

New weights: [0.929, 1.071, 1.116, 0.884, 1.020, 1.603, 1.108, 0.269, 1.668, 0.332]

Under the current weights:

$$\log p(\mathbf{l}_{\text{trn}_0} | \text{trn}_0) + \log p(\mathbf{l}_{\text{trn}_1} | \text{trn}_1) = \log 0.246 + \log 0.392 = -2.3397$$

Under the new weights:

$$\log p(\mathbf{l}_{\text{trn}_0} | \text{trn}_0) + \log p(\mathbf{l}_{\text{trn}_1} | \text{trn}_1) = \log 0.242 + \log 0.398 = -2.3399$$

We stop and use the current weights for our model.