

SI 630: Homework 3 Report

Name: Chenyun Tao Uniqname: cyuntao

Task 2

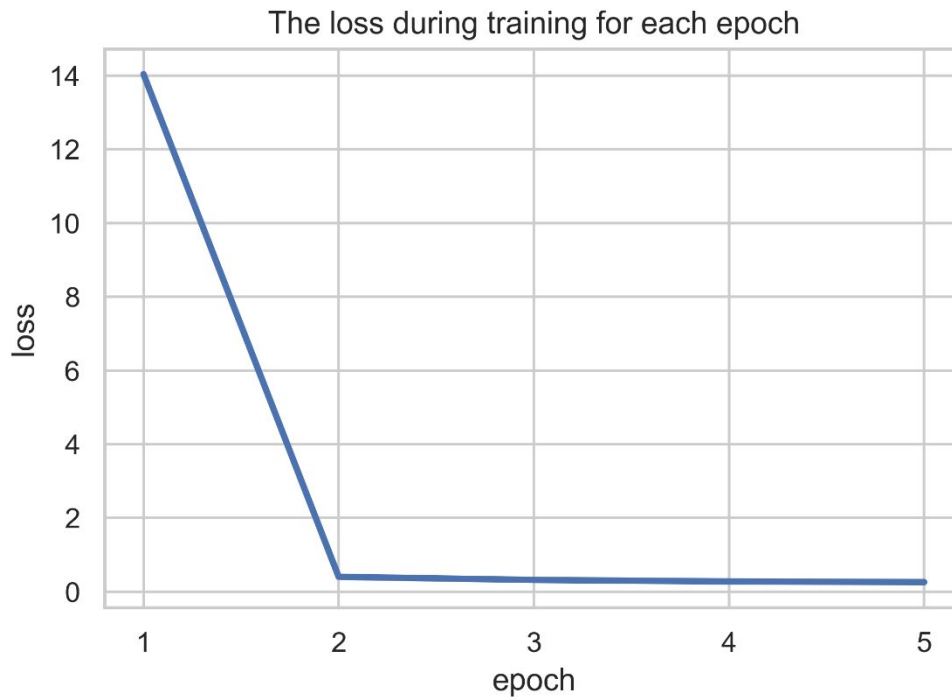


Figure 1: The loss during training for each epoch

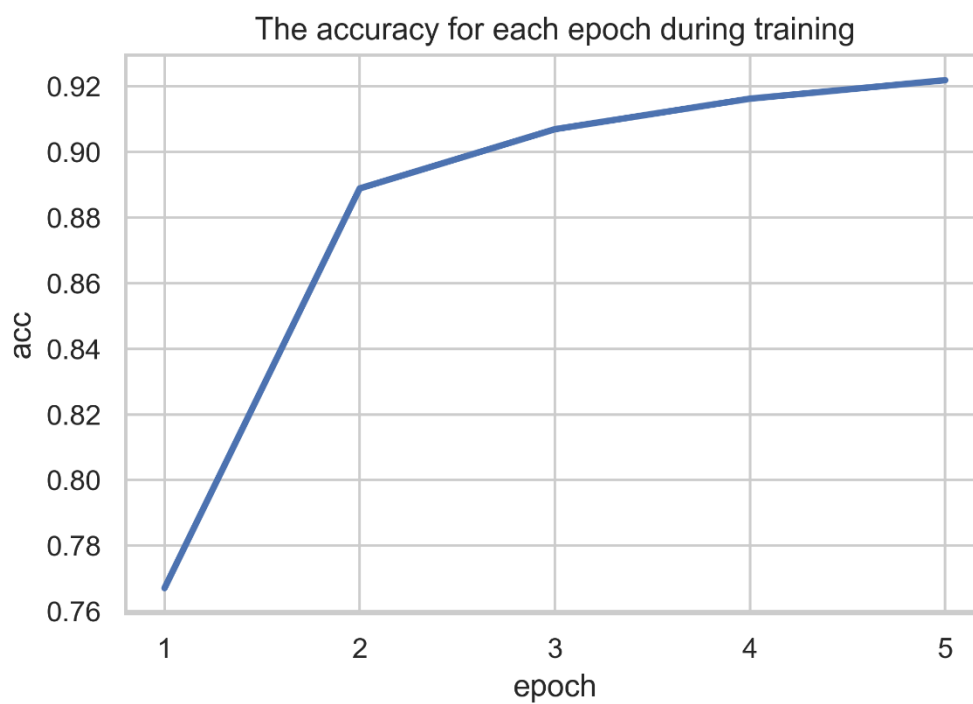


Figure 2: The accuracy for each epoch during training

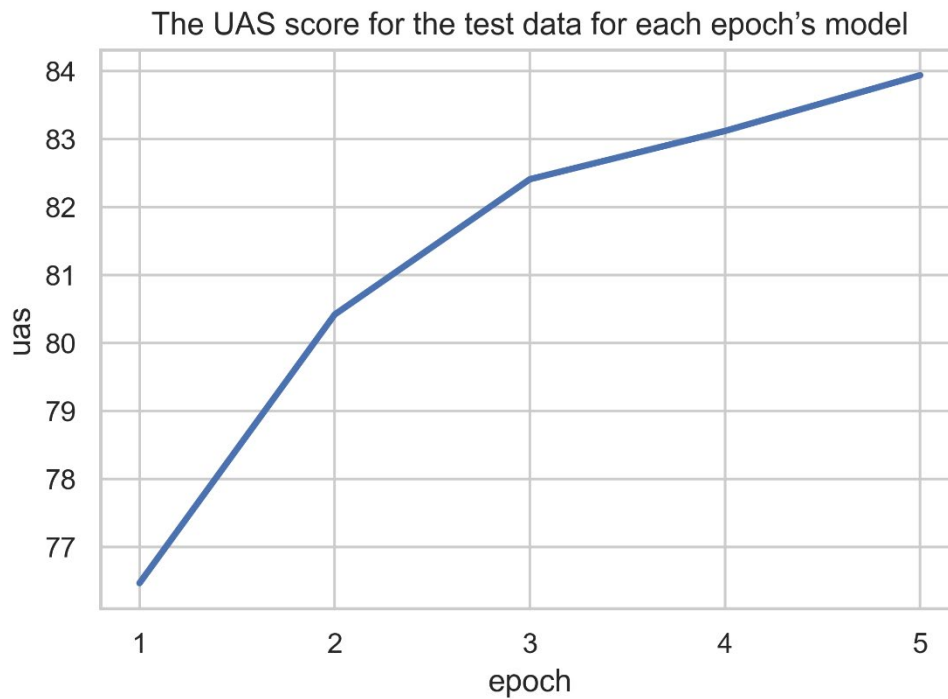


Figure 3: The UAS score for the test data for each epoch's model

The UAS score for the test data for each epoch's model is: 76.47, 80.42, 82.41, 83.12, and 83.94. The three plots: (1) the loss during training for each epoch, (2) the accuracy for each epoch during training, and (3) the UAS score for the test data for each epoch's model are shown above.

From these graphs, we can see that in general, the models are getting better performance. The loss is decreasing, the accuracy is increasing, and the UAS score is increasing. I may want to stop training at epoch 3 as the performance is good enough, and it seems that the model begins to converge at that time.

Task 3

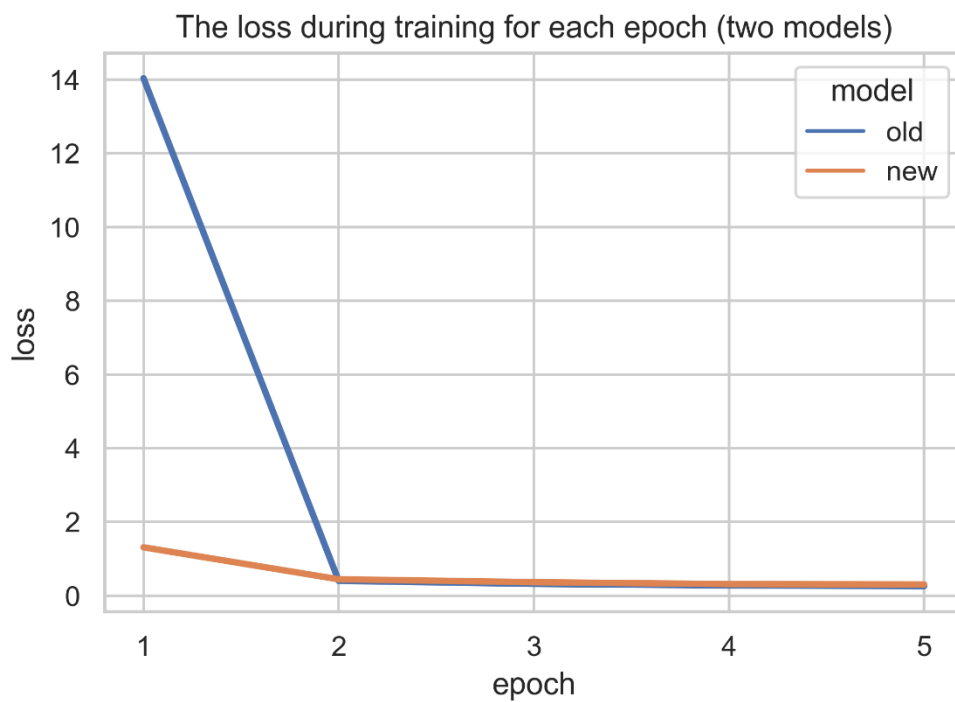


Figure 4: The loss during training for each epoch (both old and new model)

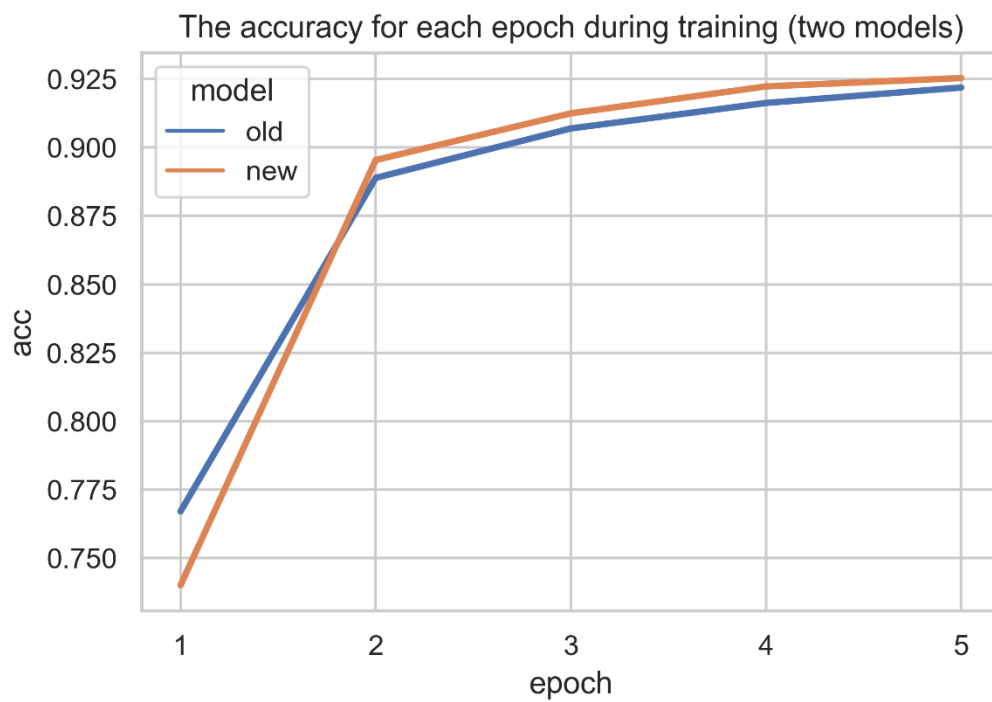


Figure 5: The accuracy for each epoch during training (both old and new model)



Figure 6: The UAS score for the test data for each epoch's model (both old and new model)

In the new model, I made the following modifications: (1) add 1 hidden layer to the network, (2) add normalization or regularization to the layers, (3) change to a different activation function in hidden layer 1, from cubic to ReLU, (4) change the size (number of neurons) in layers, from `l1_hidden_size = 200` to 300.

Similar to Task 2, the three plots: (1) the loss during training for each epoch, (2) the accuracy for each epoch during training, and (3) the UAS score for the test data for each epoch's model are shown above, but here both old model and new model's performances are included.

From the graphs, we can see the new model seems to have a better performance, with higher accuracy during training, and higher UAS score for the test data. However, as the epoch number increases, the difference between two models seems to become smaller. In addition, the new model has a much lower loss during training at first, but later the loss of the two models become close, and the loss of the old model is slightly lower.

Task 4

I used a trained model `saved_weights/parser-epoch-5.mdl` from Task 2. The shift-reduce output for the sentence "The big dog ate my homework" and the parse tree are shown below:

```
----
buffer: ['the', 'big', 'dog', 'ate', 'my', 'homework']
stack: ['<root>']
action: shift
----
buffer: ['big', 'dog', 'ate', 'my', 'homework']
stack: ['<root>', 'the']
action: shift
----
buffer: ['dog', 'ate', 'my', 'homework']
stack: ['<root>', 'the', 'big']
action: shift
----
buffer: ['ate', 'my', 'homework']
stack: ['<root>', 'the', 'big', 'dog']
action: shift
----
buffer: ['my', 'homework']
stack: ['<root>', 'the', 'big', 'dog', 'ate']
action: left arc, <d>:compound
----
buffer: ['my', 'homework']
stack: ['<root>', 'the', 'big', 'ate']
action: left arc, <d>:amod
----
buffer: ['my', 'homework']
stack: ['<root>', 'the', 'ate']
action: left arc, <d>:det
----
buffer: ['my', 'homework']
```

```

stack: ['<root>', 'ate']
action: shift
----
buffer: ['homework']
stack: ['<root>', 'ate', 'my']
action: shift
----
buffer: []
stack: ['<root>', 'ate', 'my', 'homework']
action: left arc, <d>:nmod:poss
----
buffer: []
stack: ['<root>', 'ate', 'homework']
action: right arc, <d>:dobj
----
buffer: []
stack: ['<root>', 'ate']
action: right arc, <d>:root
    <root>
      |
      ate
    _____|_____
    |   |   | homework
    |   |   |   |
the  big  dog  my

```

A mistake exists in the 4th operation, but other operations are correct. The correct operation to make at each time step should be: shift, shift, shift, left-arc, left-arc, shift, left-arc, shift, shift, left-arc, right-arc, right-arc.