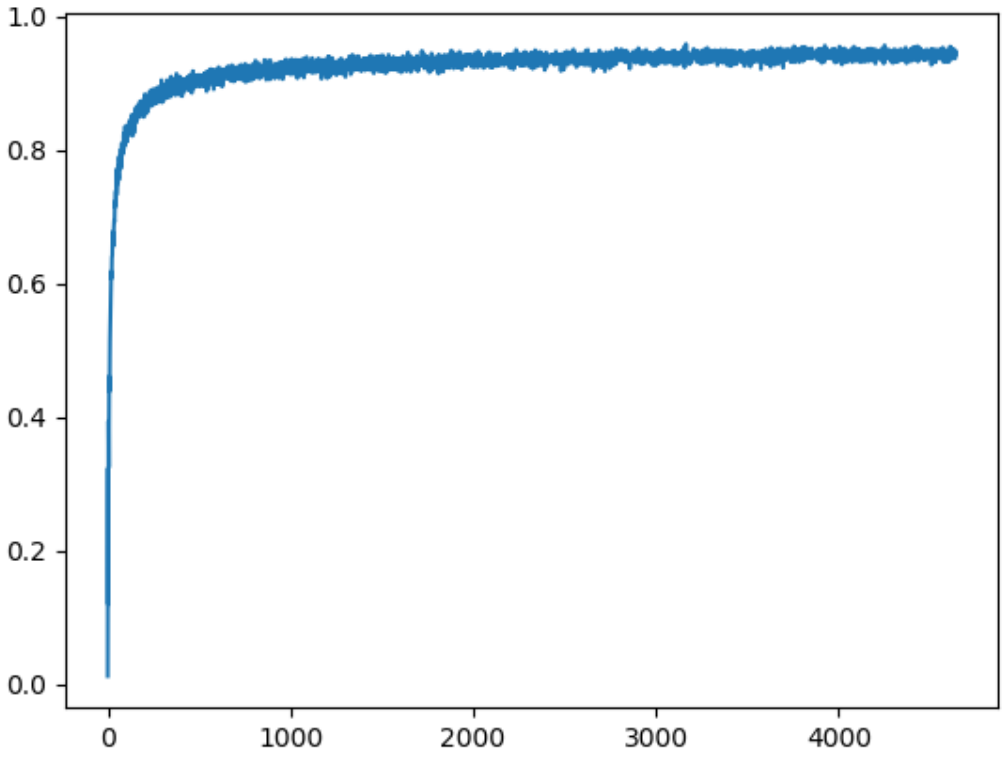
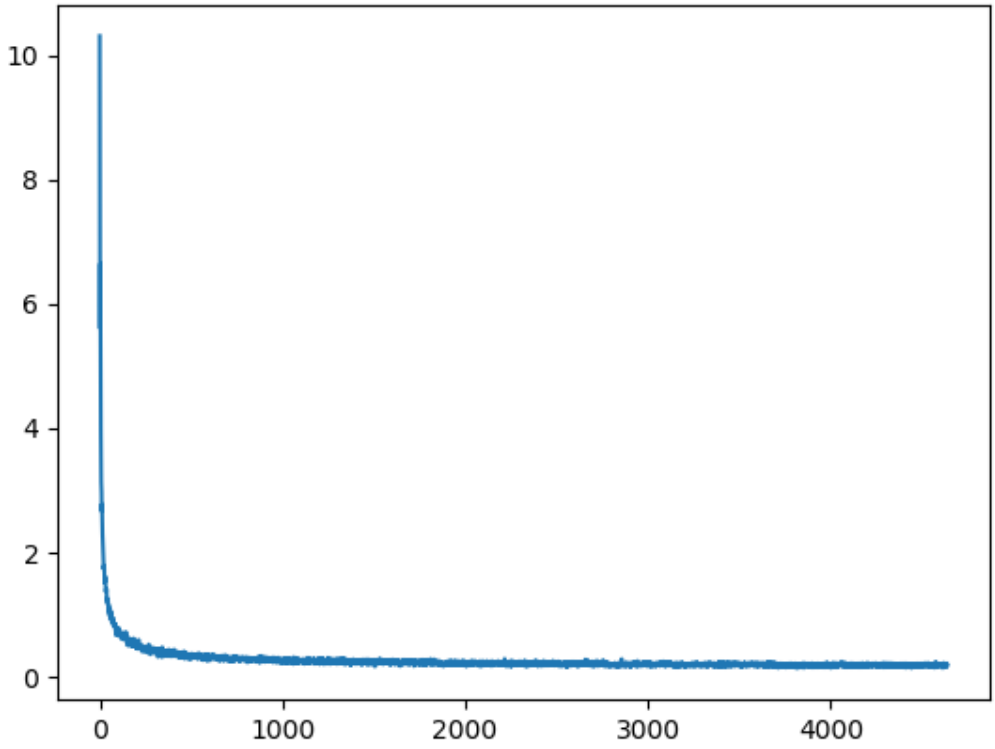
Problem 2.1

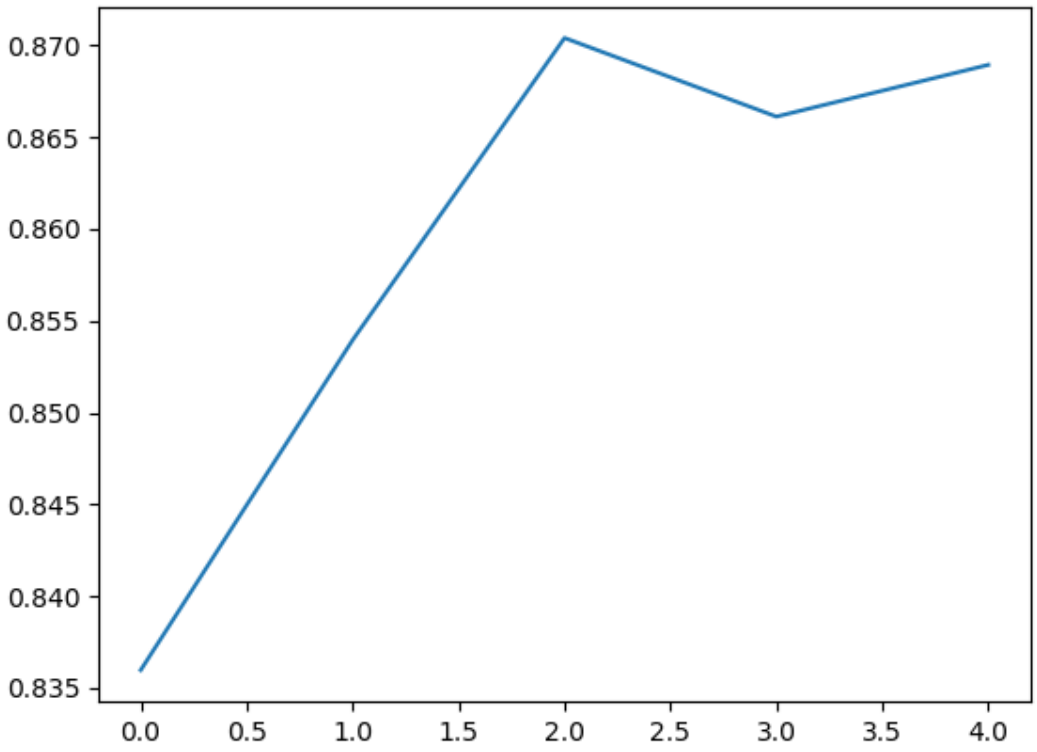
Accuracy versus iteration



Loss versus iteration



UAS versus epochs

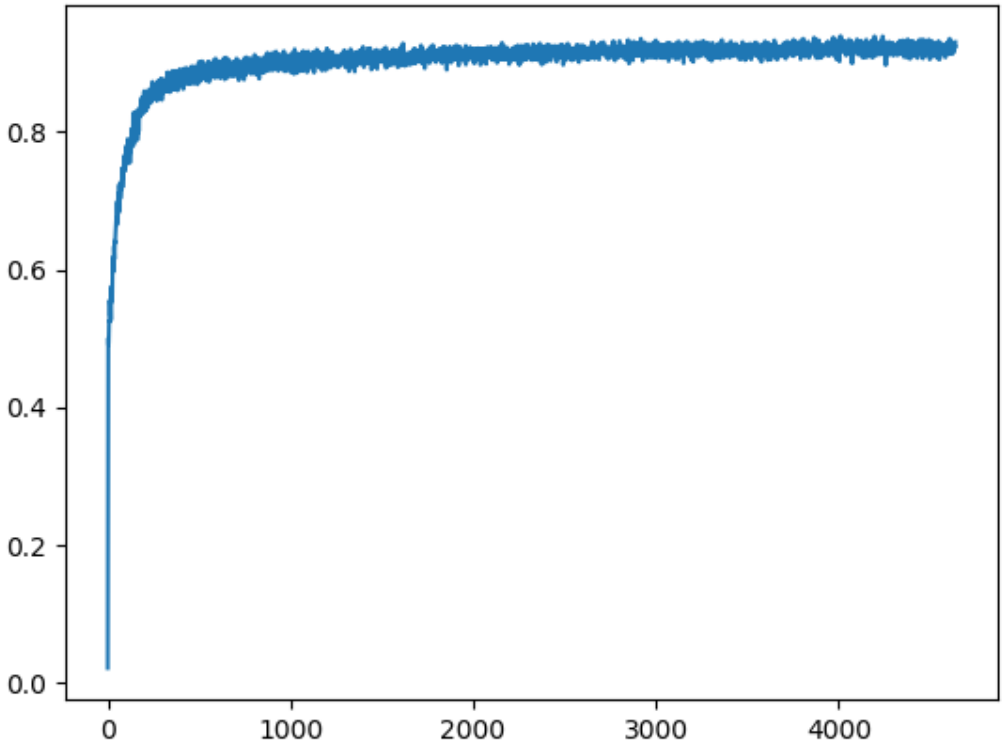


Problem 2.2.

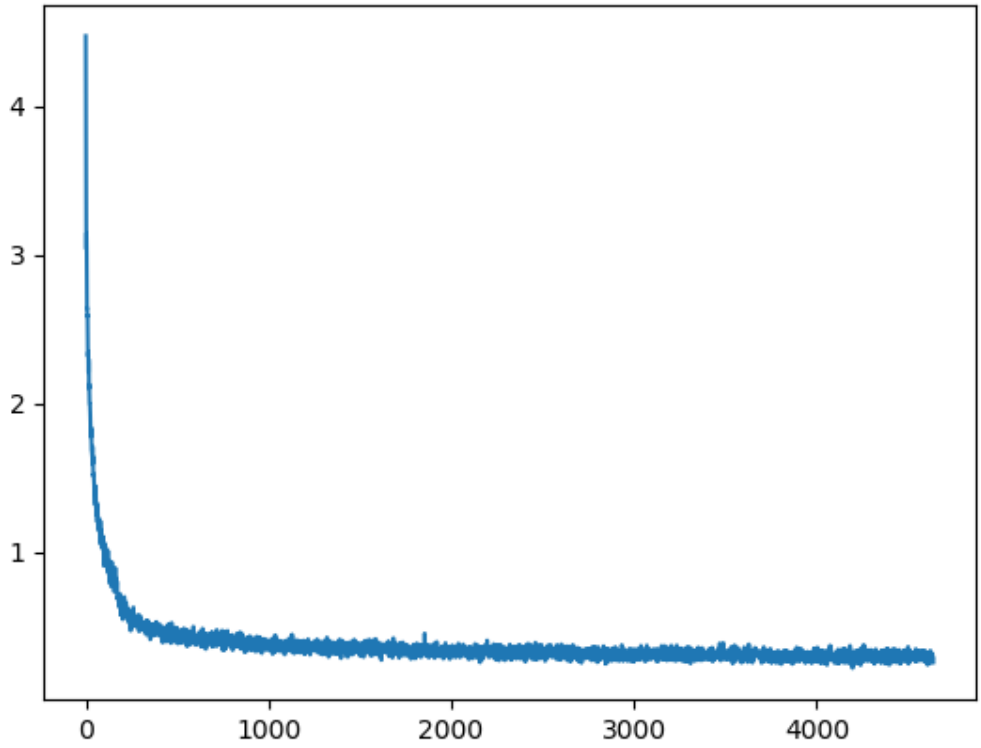
I see after 2000 (2 epoch)iteration the accuracy graph did not change too much and it was almost fix.
I see after 2000 (2 epoch)iteration the loss graph did not change too much and it was almost fix.
I see the maximum UAS happened in second epoch; therefore, it seems two epochs is enough or we have to train more than five epochs.

Problem 3.1

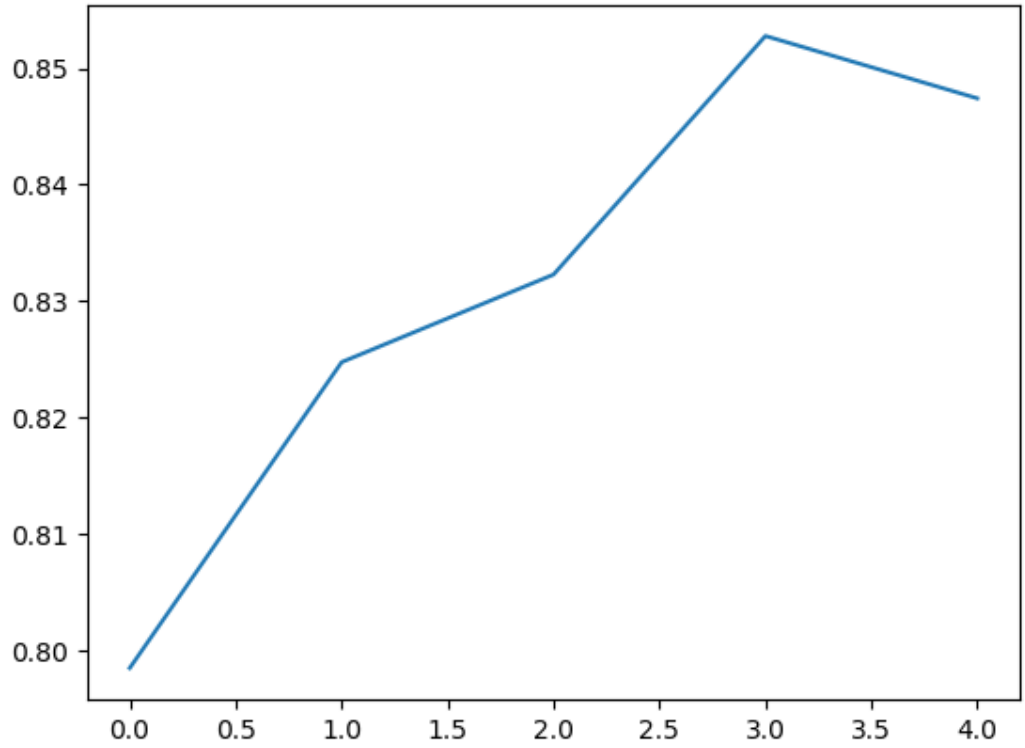
I changed the cube activation function to relu activation function, and I changed the one hidden layer to two hidden layers with l1_hidden_size = 200
l2_hidden_size = 15. The results are below. As we see on these figures compares to the figure of problem 2, the accuracy increases slower than the previous structure in problem 2. The loss drops slower than the loss of the previous model. UAS is smaller than the UAS of the previous model. One reason could be the increase of hidden units needs more time for training. Another reason is that cubic activation is more suitable for this problem.
Accuracy versus iteration. Best loss I get was .16. Best accuracy is .95 , and best UAS was 87 in five epochs.



Loss versus iteration



UAS versus epochs



Problem 4.1

```
----
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>:dep
----
buffer: []
stack:  ['<root>', 'ate']
action: right arc, <d>:root
    <root>
      |
     ate
  ____|___________
 |    |     |  homework
 |    |     |     |
the  big   dog    my
```

Problem 4.2

Bellow I write the correct action in each time step and what is the right parse tree.

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:leftarc

Buffer:[’ate’ , ’my’ , ’homework’]
Stack:[‘root’,’the’,’dog’]
Action:leftarc

Buffer:[’ate’ , ’my’ , ’homework’]
Stack:[‘root’,’dog’]
Action:shift

Buffer:[’my’ , ’homework’]
Stack:[‘root’,’dog’,’ate’]
Action:leftarc

Buffer:[’my’ , ’homework’]
Stack:[‘root’,’ate’]
Action:shift

Buffer:[’homework’]
Stack:[‘root’,’ate’,’my’]
Action:shift

Buffer:[]

Stack:[‘root’,’ate’,’my’,’homework’]
Action:leftarc

Buffer:[]
Stack:[‘root’,’ate’,’homework’]
Action:rightarc

Buffer:[]
Stack:[‘root’,’ate’]
Action:rightarc

‘root’
|
ate
|
_____
| |
dog homework
| |
_____ my
| |
the big