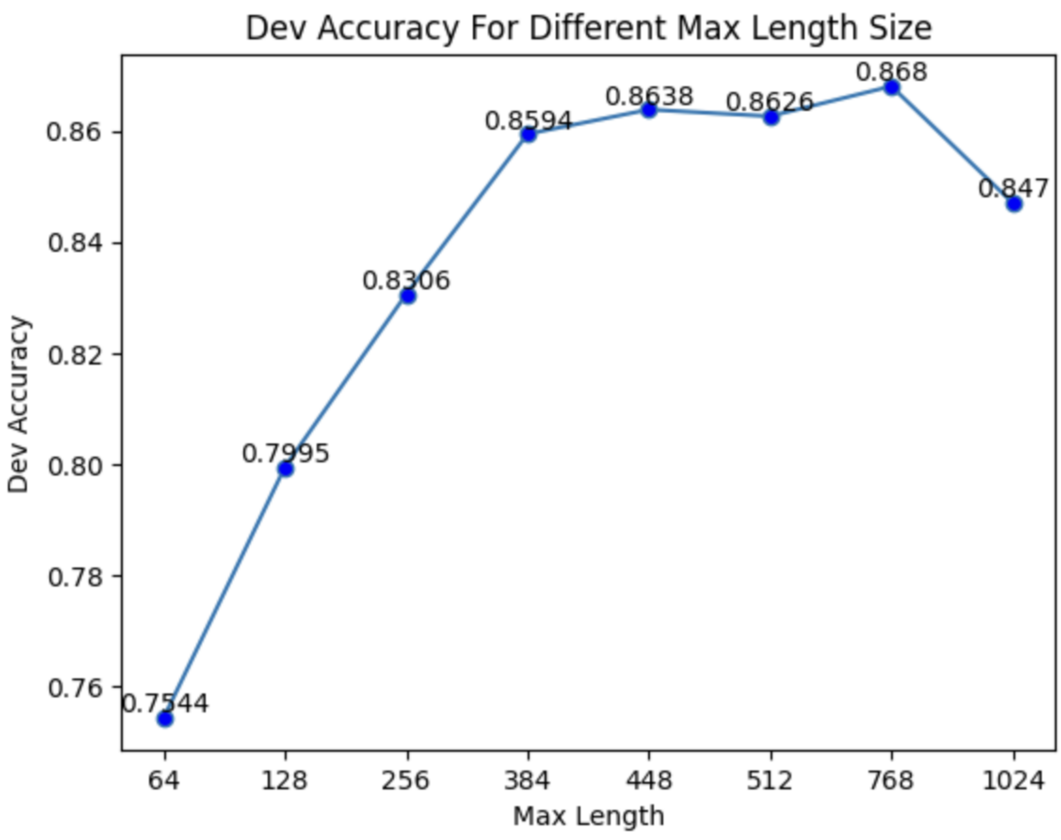
CS472 Assignment 4
Tong Guan UO ID: 951871617
Colab:https://colab.research.google.com/drive/1BvWxfMtOMtbTFHsmYibJzeuCO19I-Ty8?usp=sharing

Reports:
4.1 Fine Tuning : max_length, batch size, epoch
(1) max_length

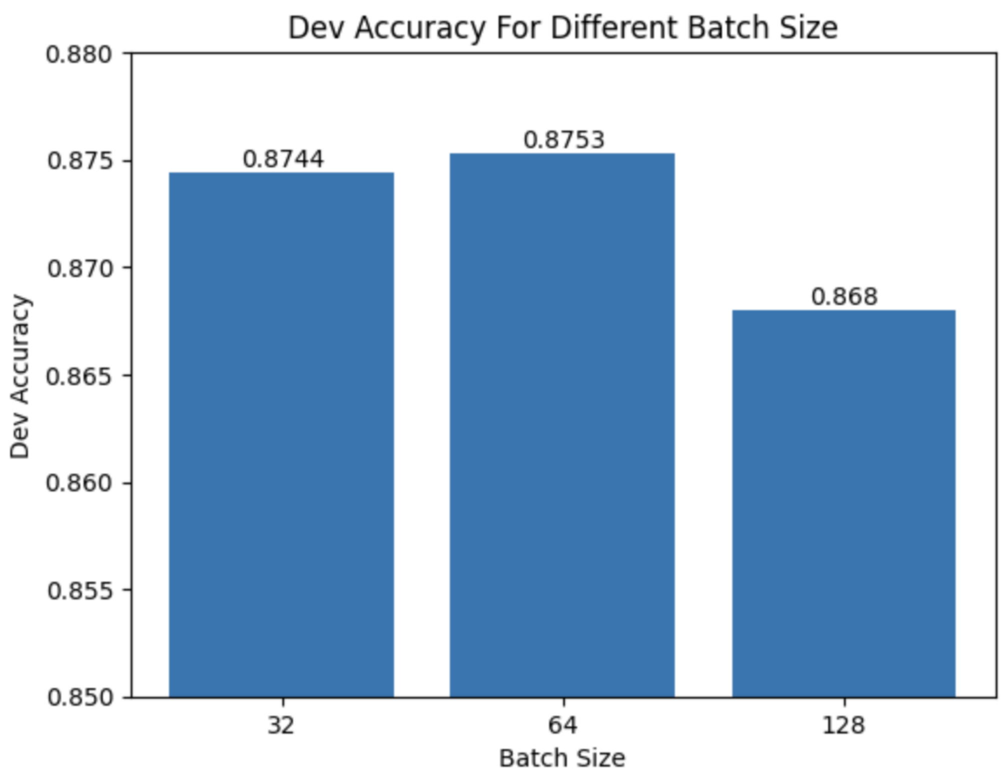
max_length	glove	dropout	hidden_size	kernel_size	epoch	lr	batch size	optimizer
tuning!	300d	0.2	256	5	40	0.01	128	SGD



After tuning, I **set the max length to 768**, as it hits the highest accuracy in performance.

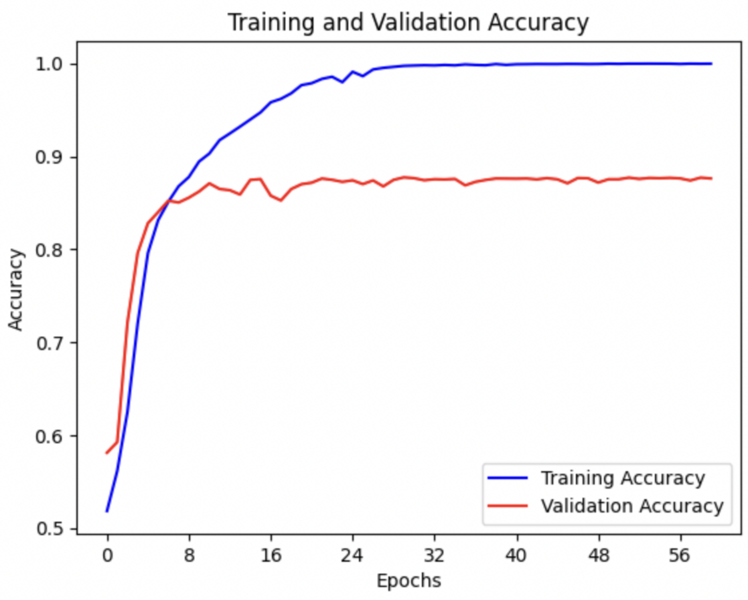
(2) Batch Size

max_length	glove	dropout	hidden_size	kernel_size	epoch	lr	batch size	optimizer
768	300d	0.2	256	5	40	0.01	**tuning!**	SGD



After tuning, I **set the batch size to 64**, as it hits the highest accuracy in performance.

(3) Epoch



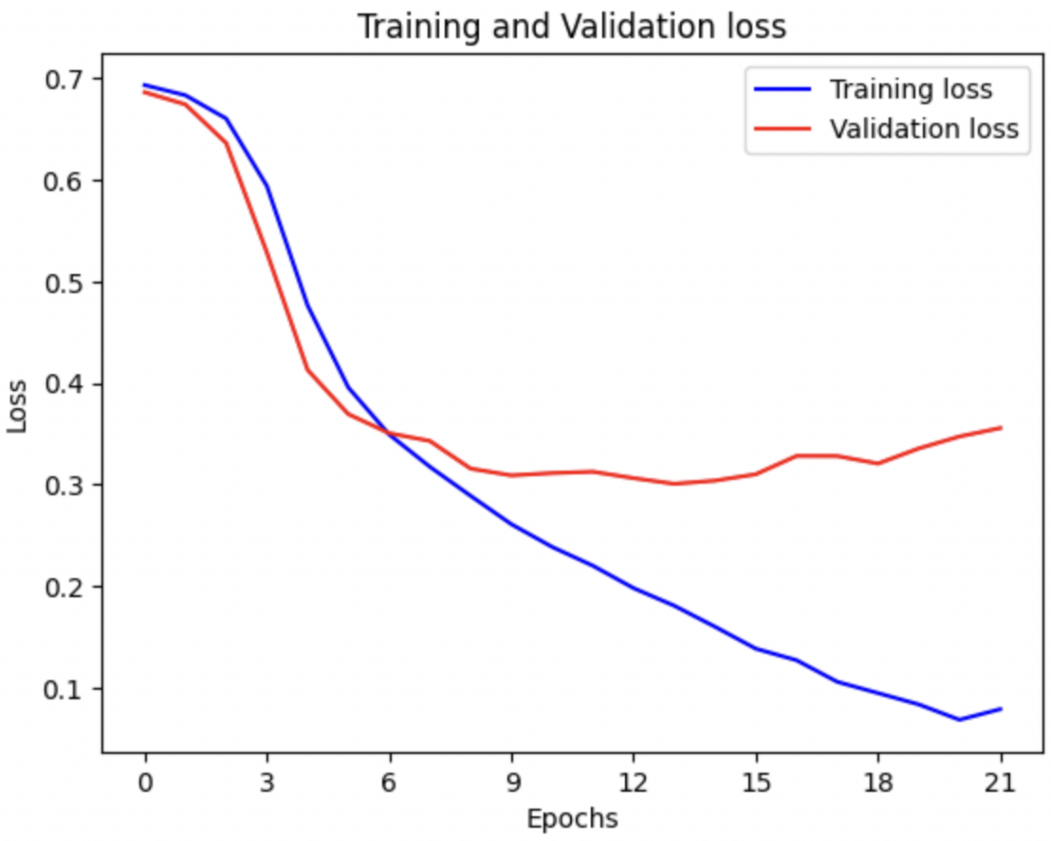
After observing the learning curve of the accuracy graph from epoch 0 to epoch 59,I choose epoch to be 22 as a balance of larger than 20 and avoid too much overfitting.
And the dev accuracy at epoch 21 is 0.8761.

4.2
After the tuning in 4.1, I've adjust my argument as below:

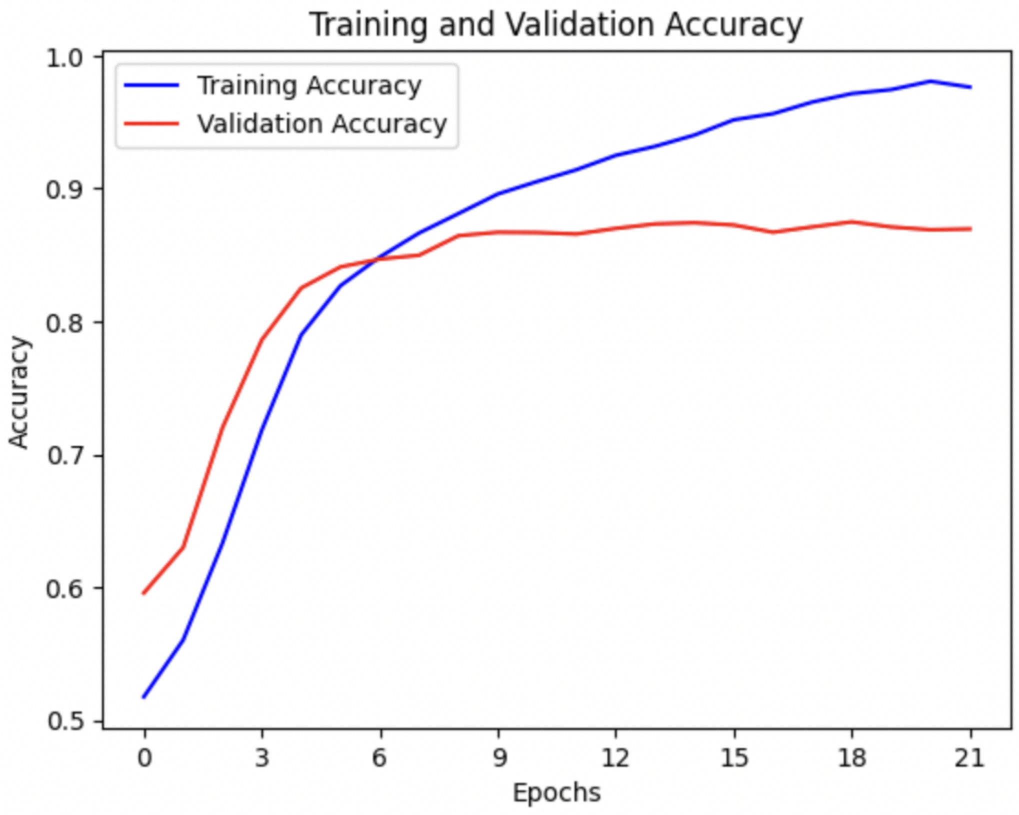
max_length	glove	dropout	hidden_size	kernel_size	epoch	lr	batch size	optimizer
768	300d	0.2	256	5	22	0.01	64	SGD

At Epoch 21, the dev accuracy is 0.8761 = 87.61%

(1) the graph for training and development loss over epochs



(2) the graph for training and development accuracy over epochs



(3) my conclusion

At first, I spent a lot of time tuning the learning rate, and I read articles about why we use [0.001:0.01] for tuning and how we tune the learning rate. However, I figured out that the learning rate didn't improve performance effectively. Instead, changing the max length size does have a great improvement. After tuning the parameters, I have a better understanding about different parameters and how they will affect the performance and how to decide if there is an overfitting and we should stop.