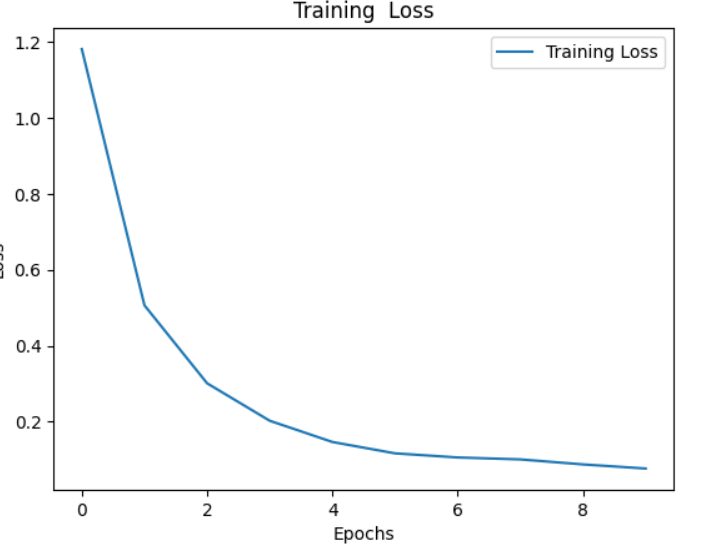
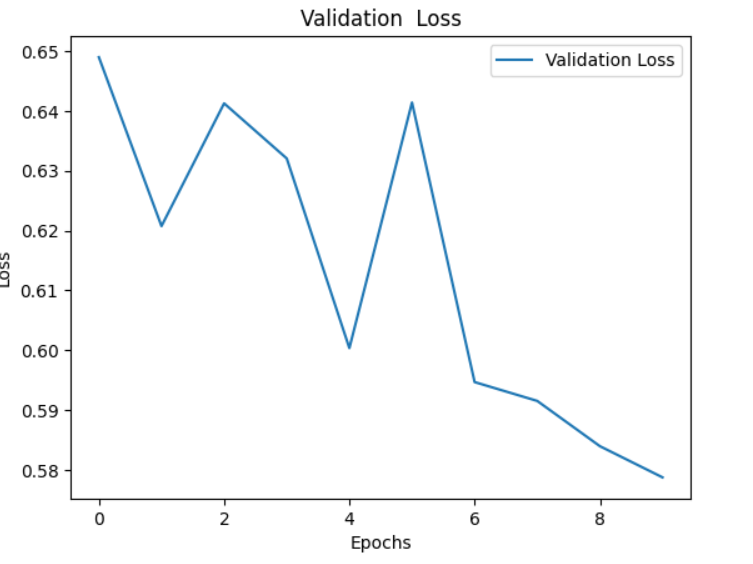
**Task 1A:**

Pearson Score on validation data: 0.8641  


The training loss decreases with each epoch, which means that the model is learning, i.e., it is able to capture the patterns in the data.



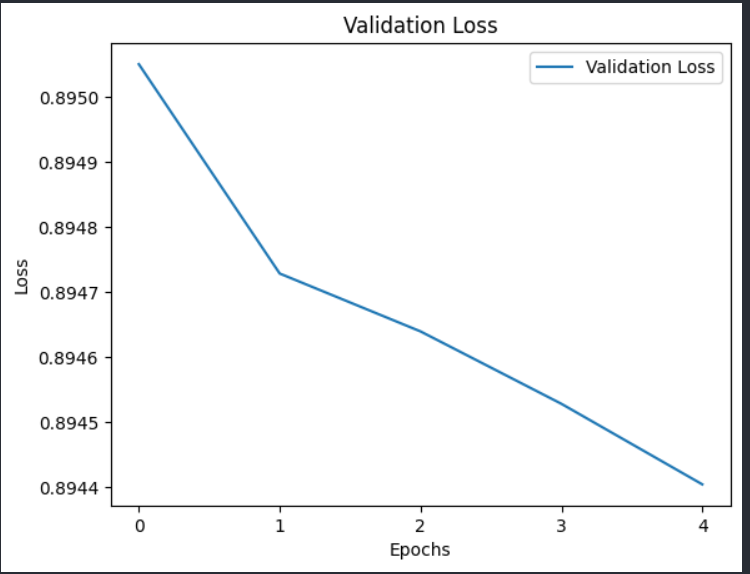
The graph of validation loss gives a decreasing curve, which means that the model is increasing its performance with each epoch. It is not overfitting since it performs better on unseen data.

**Task 1B:**

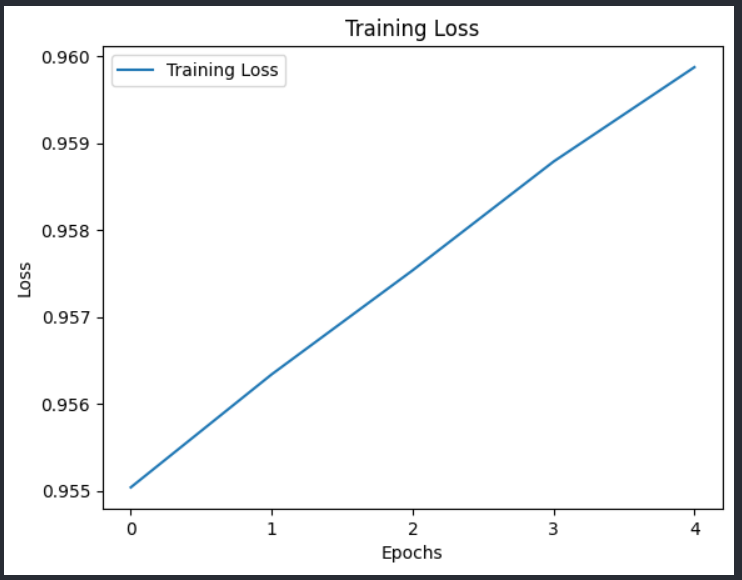
Pearson Score on validation data: 0.7920248413077999

**Task 1C:**

Pearson score on validation data: 0.8126659569980687



The graph of validation loss gives a decreasing curve, which means that the model is increasing its performance with each epoch. It is not overfitting since it performs better on unseen data.



The training rate increases with each epoch, which means that the model is underfitting, i.e., it is unable to capture the patterns in the data. We also tried decreasing the learning rate, but the outcome was not changed. Increasing the number of epochs also didn’t help.

**Provide a brief comparison and explanation for the performance differences between the**

**three setups in the report.**

::

For Task1B, since we used pre-trained ‘all-MiniLM-L6-v2’, it gave us the Pearson correlation value of 0.792. In the fine-tuned model, the Pearson Correlation on the same test dataset(validation set) was 0.812. This may have happened because the pre-trained model is trained on a more extensive set of data, but fine-tuning made the model more accurate. Fine-tuning the model is also one of the reasons why it gave a better Pearson correlation value.

Contributions:

[Saumil Lakra](mailto:saumil21097@iiitd.ac.in): Task1B, Task1C, plots of mentioned tasks, Demo part for 1C