LAB4 AI REPORT

MADE BY:

Ahmad Said Nouh (7086)

Hossam Eldin Ahmed Eissa (6747)

Hussein Mourad Kassem (6729)

Ahmed Alaa Tawfik (6842)

Contents

	Predicted (n)	Predicted (p)
Actual (n)	TN	FP
Actual (p)	FN	TP

Preprocessed:

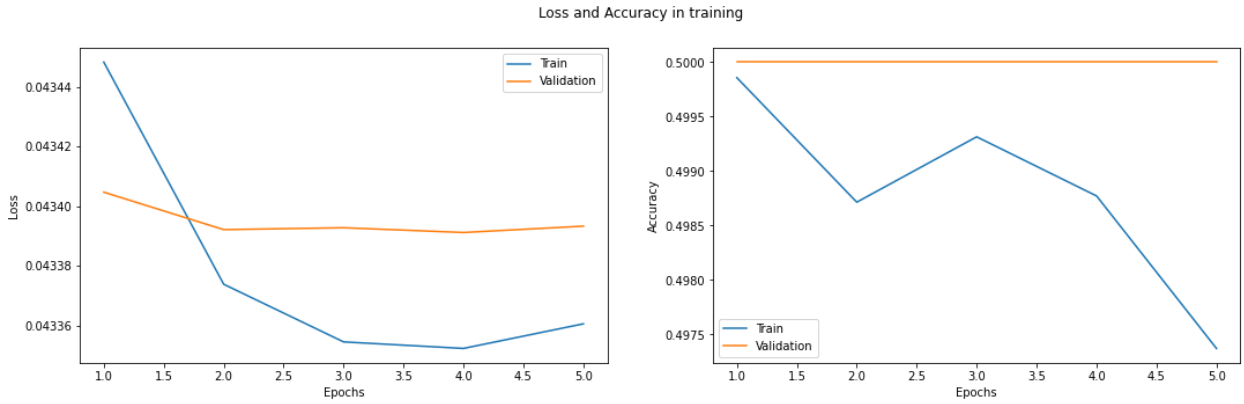
(1e-3):

Confusion Matrix:

	Predicted n	Predicted p
Actual n	0.0	5000.0
Actual p	0.0	5000.0

```
Test Accuracy:  0.500
Percision : 0.0
Sensitivity (Recall) : nan
Specifity : 0.5
F1 Score = nan
```

Graph:



(5e-4):

Confusion Matrix:

	Predicted n	Predicted p
Actual n	5000.0	0.0
Actual p	5000.0	0.0

```
Test Accuracy:  0.500
Percision : 1.0
Sensitivity (Recall) : 0.5
Specifity : nan
F1 Score = 0.6666666666666666
```

Graphs:

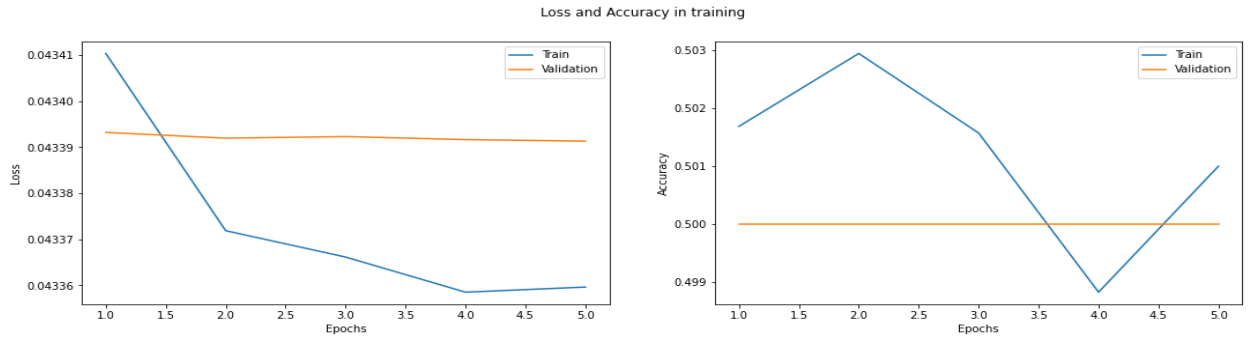
(1e-4):

Confusion Matrix:

	Predicted n	Predicted p
Actual n	5000.0	0.0
Actual p	5000.0	0.0

```
Test Accuracy:  0.500
Percision : 1.0
Sensitivity (Recall) : 0.5
Specifity : nan
F1 Score = 0.6666666666666666
```

Graphs:



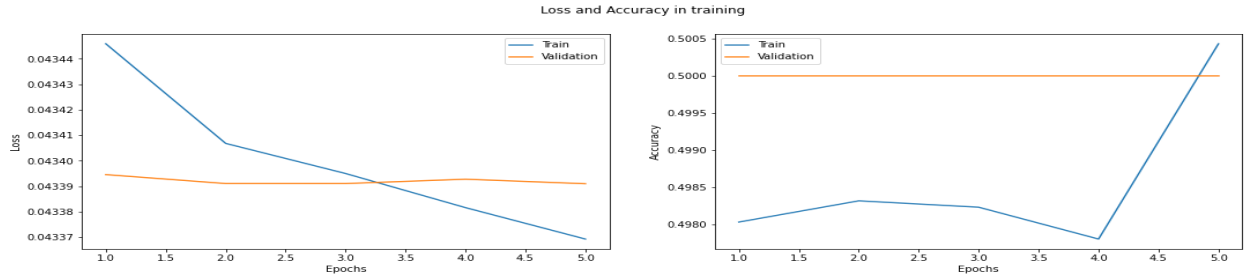
(5e-5):

Confusion Matrix:

	Predicted n	Predicted p
Actual n	5000.0	0.0
Actual p	5000.0	0.0

```
Test Accuracy:  0.500
Percision : 1.0
Sensitivity (Recall) : 0.5
Specifity : nan
F1 Score = 0.6666666666666666
```

Graphs:



(1e-5):

Confusion Matrix:

	Predicted n	Predicted p
Actual n	5000.0	0.0
Actual p	4999.0	1.0

```
Test Accuracy:  0.500
Percision : 1.0
Sensitivity (Recall) : 0.5000500050005
Specifity : 1.0
F1 Score = 0.6667111140742716
```

Graphs:

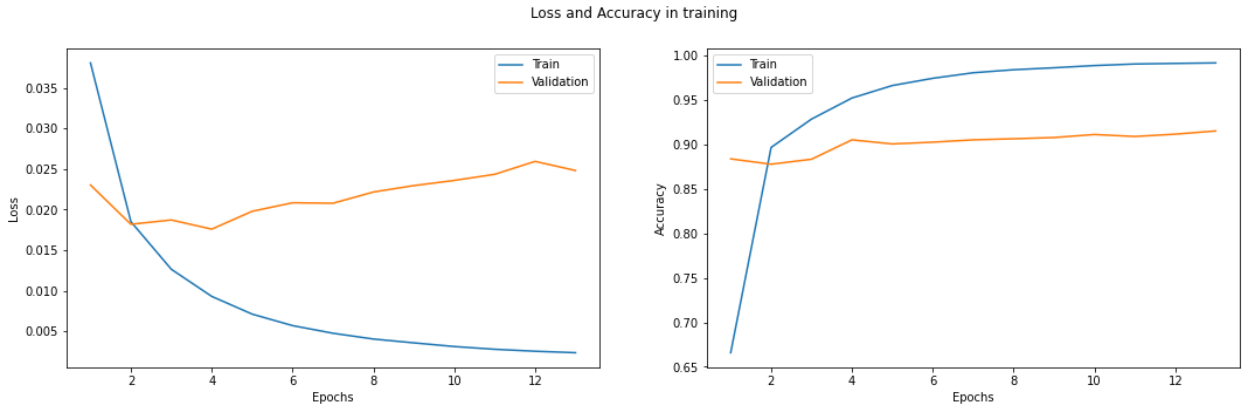
(5e-6):

Confusion Matrix:

	Predicted n	Predicted p
Actual n	4516.0	484.0
Actual p	395.0	4605.0

```
Test Accuracy:  0.912
Percision : 0.9032
Sensitivity (Recall) : 0.9195683160252495
Specifity : 0.9048929062684221
F1 Score = 0.9113106649177681
```

Graphs:



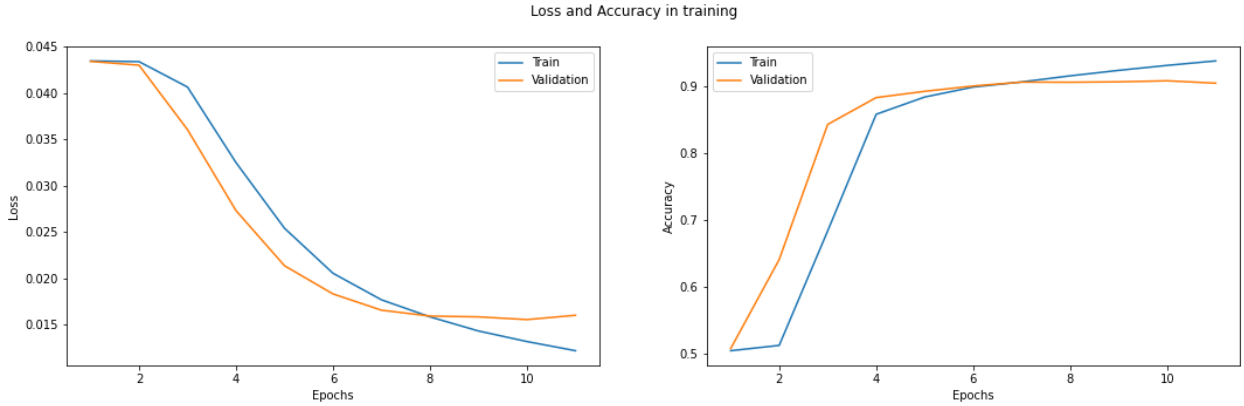
(1e-6):

Confusion Matrix:

	Predicted n	Predicted p
Actual n	4657.0	343.0
Actual p	615.0	4385.0

```
Test Accuracy:  0.904
Percision : 0.9314
Sensitivity (Recall) : 0.8833459787556904
Specifity : 0.9274534686971235
F1 Score = 0.9067367601246106
```

Graphs:



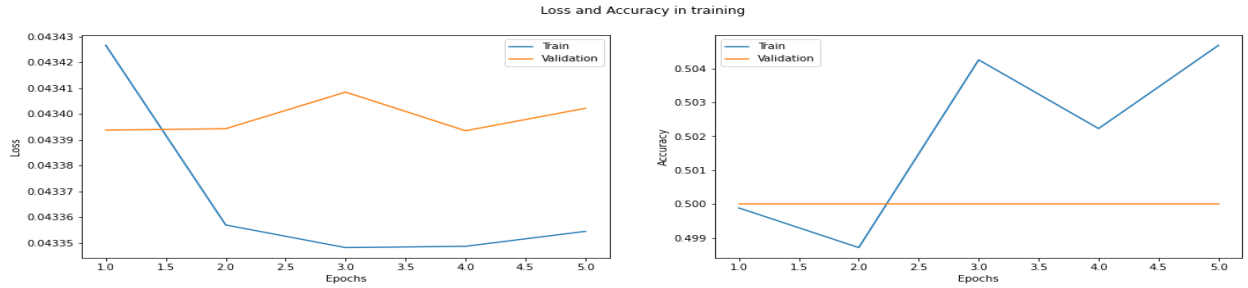
Raw:

(1e-3):

Confusion Matrix:

```
Test Accuracy:  0.500
Percision : 1.0
Sensitivity (Recall) : 0.5
Specifity : nan
F1 Score = 0.6666666666666666
Confusion Matrix :
[[5000.    0.]
 [5000.    0.]]
```

Graphs:

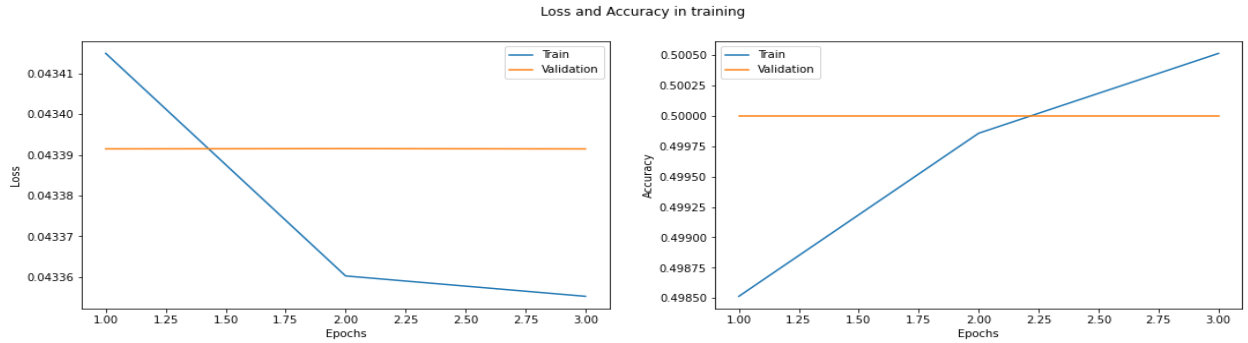


(5e-4):

Confusion Matrix:

```
Test Accuracy:  0.500
Percision : 0.0
Sensitivity (Recall) : nan
Specifity : 0.5
F1 Score = nan
Confusion Matrix :
[[   0. 5000.]
 [   0. 5000.]]
```

Graphs:

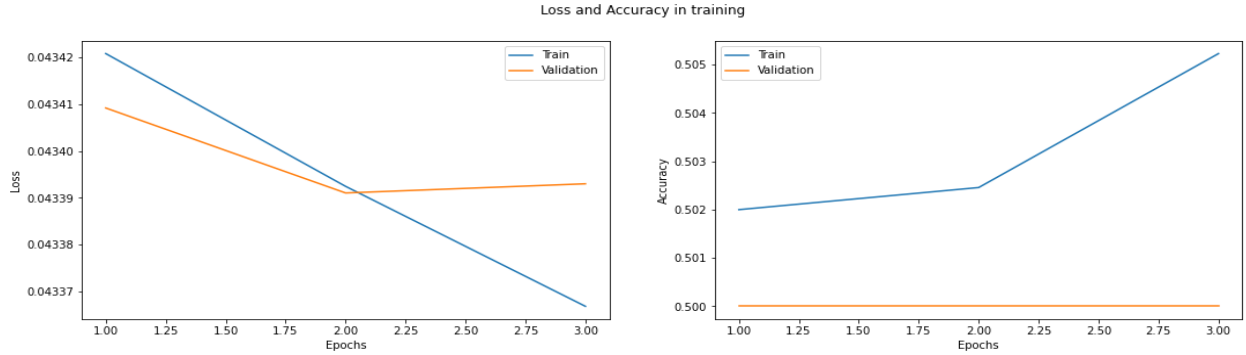


(1e-4):

Confusion Matrix:

```
Test Accuracy:  0.500
Percision : 1.0
Sensitivity (Recall) : 0.5
Specifity : nan
F1 Score = 0.6666666666666666
Confusion Matrix :
[[5000.    0.]
 [5000.    0.]]
```

Graphs:

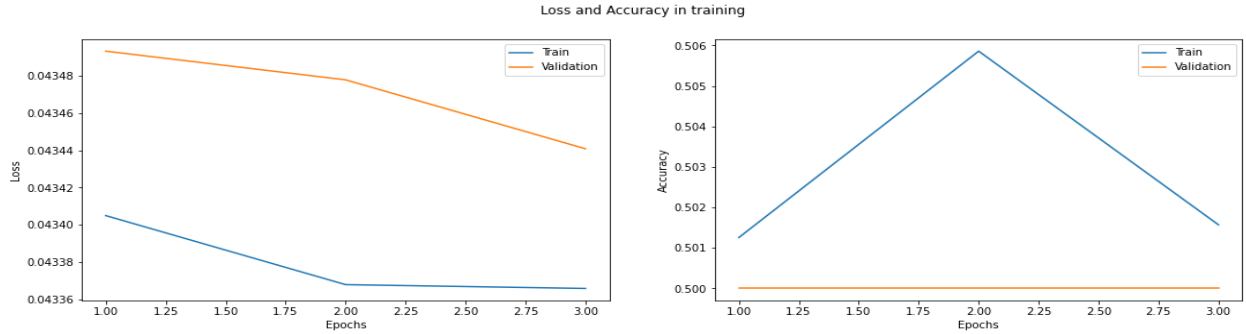


(5e-5):

Confusion Matrix:

```
Test Accuracy:  0.500
Percision : 1.0
Sensitivity (Recall) : 0.5
Specifity : nan
F1 Score = 0.6666666666666666
Confusion Matrix :
[[5000.    0.]
 [5000.    0.]]
```

Graphs:



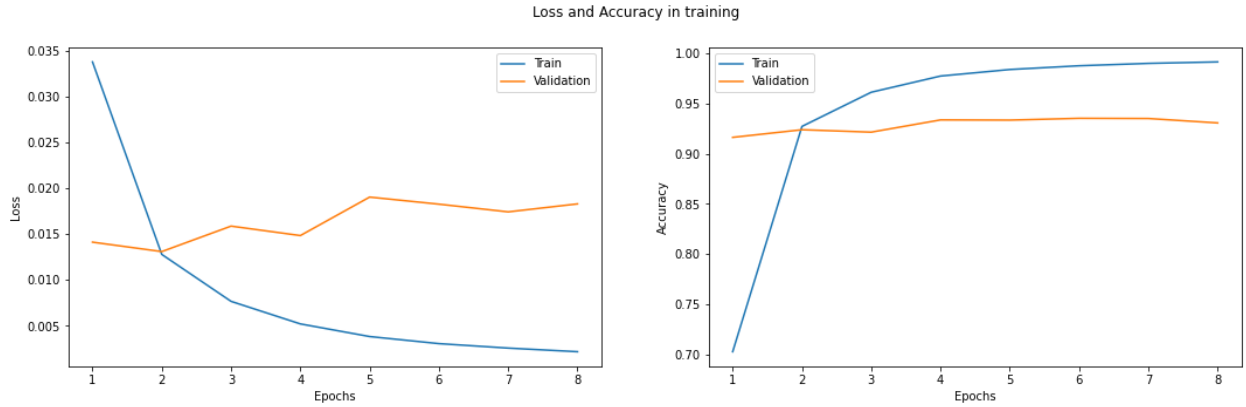
(1e-5):

Confusion Matrix:

	Predicted n	Predicted p
Actual n	4688.0	312.0
Actual p	371.0	4629.0

```
Test Accuracy:  0.932
Percision : 0.9376
Sensitivity (Recall) : 0.9266653488831785
Specifity : 0.9368548876745598
F1 Score = 0.9321006064221096
```

Graphs:



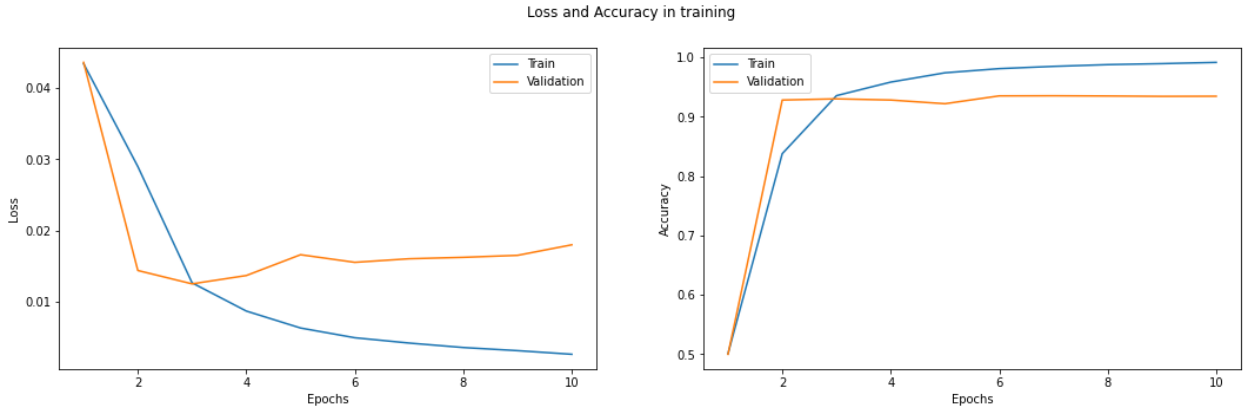
(5e-6):

Confusion Matrix:

	Predicted n	Predicted p
Actual n	4638	362
Actual p	306	4694

```
Test Accuracy:  0.933
Percision : 0.9276
Sensitivity (Recall) : 0.9381067961165048
Specifity : 0.9284018987341772
F1 Score = 0.9328238133547868
```

Graphs:



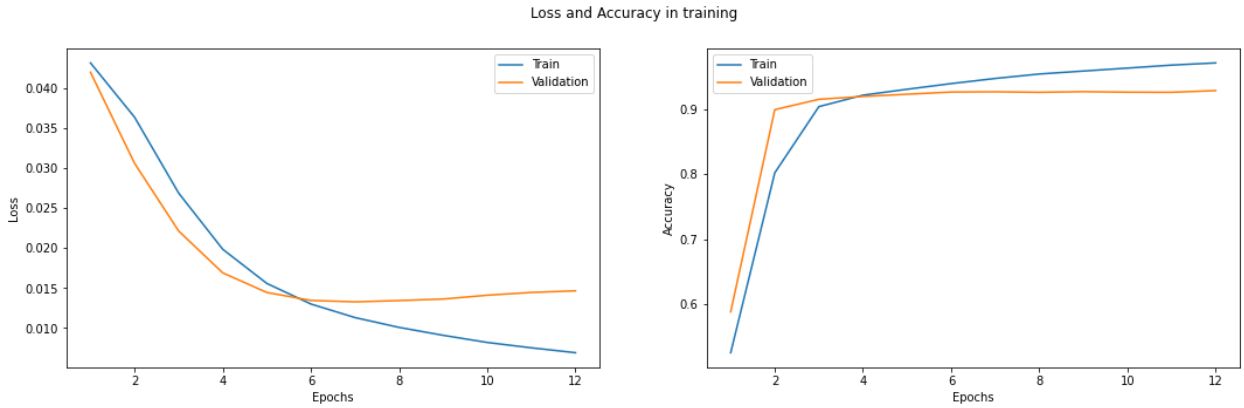
(1e-6):

Confusion Matrix:

	Predicted n	Predicted p
Actual n	4589.0	411.0
Actual p	275.0	4725.0

```
Test Accuracy:  0.931
Percision : 0.9178
Sensitivity (Recall) : 0.9434621710526315
Specifity : 0.9199766355140186
F1 Score = 0.9304541768045417
```

Graphs:



Bonus:

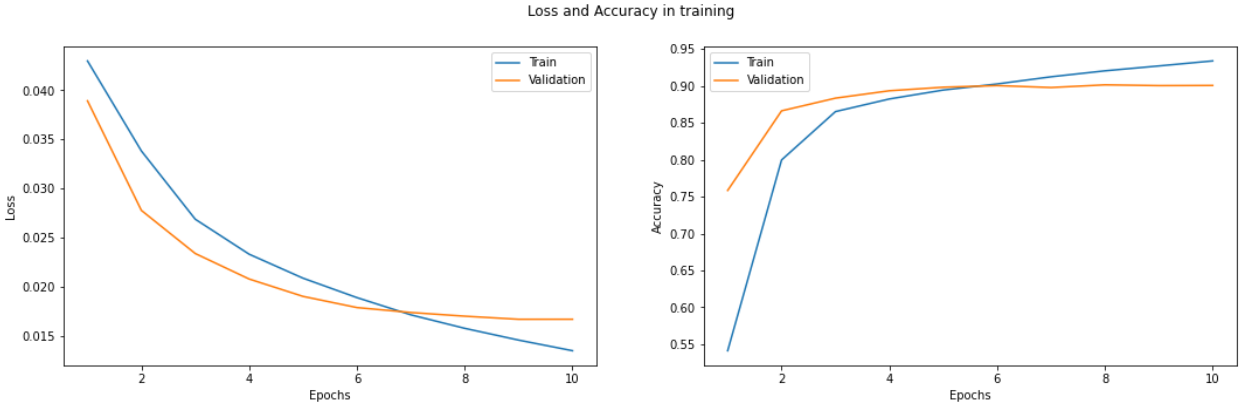
(1e-6), processed, decoder (/4):

Confusion Matrix:

	Predicted n	Predicted p
Actual n	4706.0	294.0
Actual p	748.0	4252.0

```
Test Accuracy:  0.896
Percision : 0.9412
Sensitivity (Recall) : 0.8628529519618628
Specifity : 0.9353277606687197
F1 Score = 0.9003252343600535
```

Graphs:



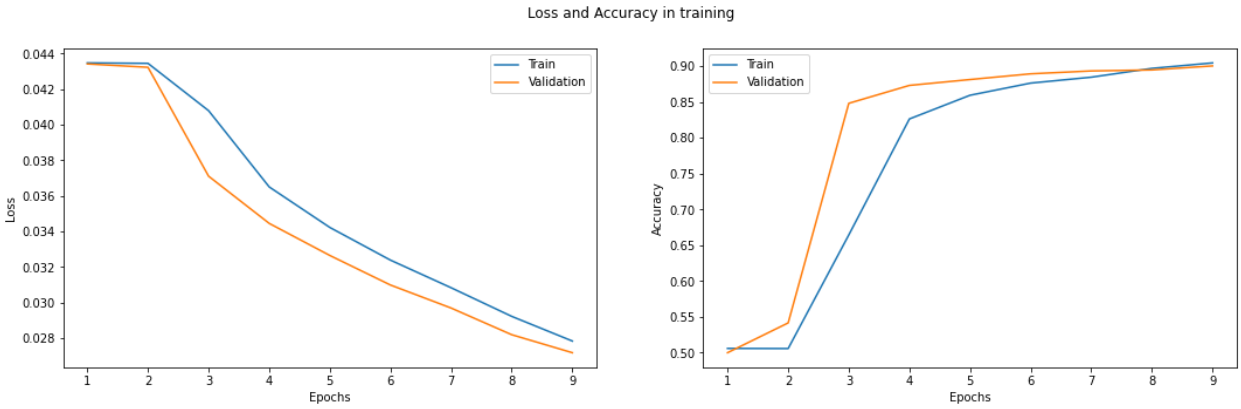
(1e-6), processed, decoder (/8):

Confusion Matrix:

	Predicted n	Predicted p
Actual n	4466.0	534.0
Actual p	457.0	4543.0

```
Test Accuracy:  0.901
Percision : 0.8932
Sensitivity (Recall) : 0.9071704245378834
Specifity : 0.8948197754579476
F1 Score = 0.9001310087675098
```

Graphs:



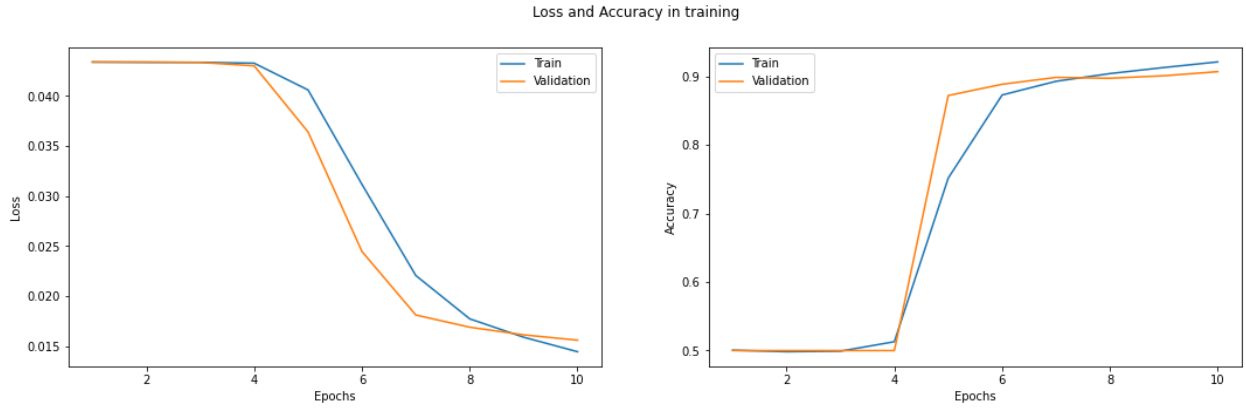
(1e-6), processed, encoder + decoder (/4):

Confusion Matrix:

	Predicted n	Predicted p
Actual n	4662.0	338.0
Actual p	650.0	4350.0

```
Test Accuracy:  0.901
Percision : 0.9324
Sensitivity (Recall) : 0.8776355421686747
Specifity : 0.927901023890785
F1 Score = 0.904189294026377
```

Graphs:



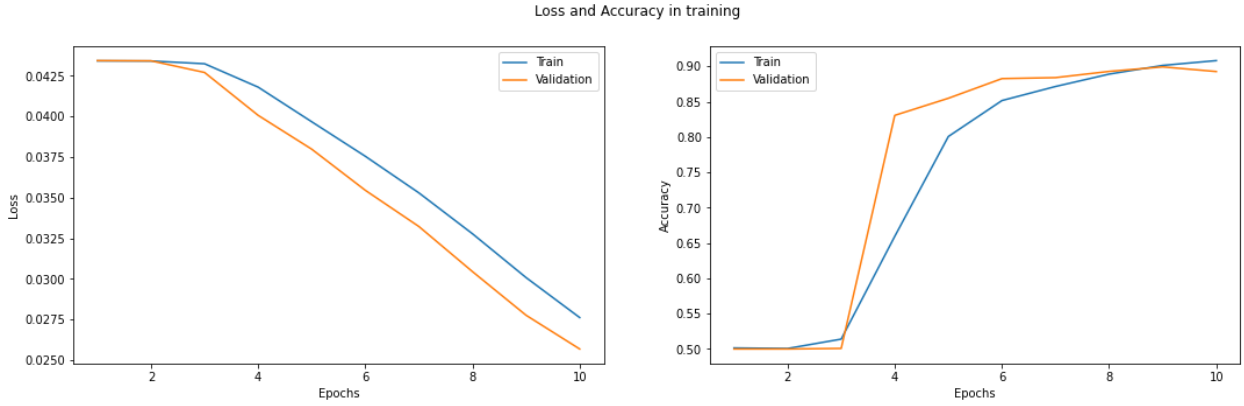
(1e-6), processed, encoder + decoder (/8):

Confusion Matrix:

	Predicted n	Predicted p
Actual n	4661.0	339.0
Actual p	718.0	4282.0

```
Test Accuracy:  0.894
Percision : 0.9322
Sensitivity (Recall) : 0.866517940137572
Specifity : 0.9266392555723869
F1 Score = 0.898159745640235
```

Graphs:



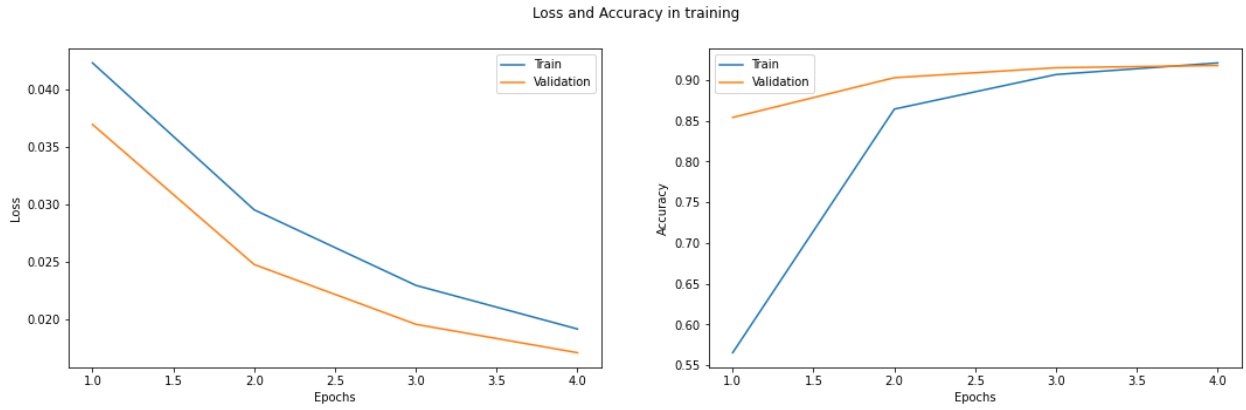
(1e-6), raw, decoder (/4):

Confusion Matrix:

	Predicted n	Predicted p
Actual n	4697.0	303.0
Actual p	494.0	4506.0

```
Test Accuracy:  0.920
Percision : 0.9394
Sensitivity (Recall) : 0.9048352918512811
Specifity : 0.9369931378665003
F1 Score = 0.921793739574134
```

Graphs:



Comments:

It is clear that all models trained using the raw data performed better than the models that trained using the preprocessed data.

There are several reasons why our NLP BERT-based model may have performed better without preprocessing your dataset. One possibility is that preprocessing steps such as lowercasing, stemming, and stopword removal may have removed important information from the text that the model needs to make accurate predictions. Additionally, BERT is trained on a large corpus of unprocessed text, so it may be better suited to handle raw text than preprocessed text. Another possible reason is that our preprocessing steps may have introduced errors into the dataset that hindered the model's performance. Finally, it could be the case that our dataset is small and the added noise from the preprocessing steps caused overfitting. Therefore, it's always a good idea to try both preprocessed and unprocessed data to see which works better for a specific task.

More detailed analysis for several potential reasons why our model may have performed better without preprocessing the IMDB dataset:

The dataset is already cleaned and preprocessed: The IMDB dataset is a popular dataset and has been used in many studies, it's likely that it's already cleaned and preprocessed to some extent.

Preprocessing steps introduced errors: Preprocessing steps such as lowercasing, stemming, and stopword removal may have introduced errors into the dataset that hindered the model's performance.

Preprocessing removed important information: Preprocessing steps may have removed important information from the text that the model needs to make accurate predictions.

The dataset is large: BERT is trained on a large corpus of text and it can handle the noise and the out of vocabulary words well.

Regularization: The preprocessing may have removed some of the noise that the model relies on as a regularization technique.

It's always a good practice to try both preprocessed and unprocessed data to see which works better for a specific task. And also fine-tune our preprocessing steps to see which one works best.