

Fine tuning the entire network: No freezing done on any of the transformer layers:

Choosing hyperparameters:

1. For learning rate, 3 values $1e-3$, $1e-4$, $1e-5$ were tried against a validation set for a single epoch. $1e-3$ and $1e-4$ did not show any signs of converging, hence $1e-5$ was chosen
2. Batch size was chosen to be 8 due to GPU RAM size constraints.
3. The fine tuning ran for 5 epochs

Result of this network after fine tuning on the test data set ended up being

		Predicted			
Actual	0	1814	17	41	28
	1	10	1882	4	4
	2	54	9	1715	122
	3	41	7	106	1746

From this we can calculate the following values:

Label 0 (World): Recall: 0.95 Precision: 0.95 f1: 0.95

Label 1 (Sports): Recall: 0.99 Precision: 0.98 f1: 0.99

Label 2 (Business): Recall: 0.90 Precision: 0.92 f1: 0.91

Label 3 (Sci/Tech): Recall: 0.92 Precision: 0.92 f1: 0.92

When we fine tune the network after freezing the 10 initial transformer layers we get the following results:

		Predicted			
Actual	0	1802	6	56	36
	1	31	1847	12	10
	2	35	3	1742	120
	3	35	5	115	1745

From this we can calculate the following values:

Label 0 (World): Recall: 0.95 Precision: 0.95 f1: 0.95

Label 1 (Sports): Recall: 0.97 Precision: 0.99 f1: 0.98

Label 2 (Business): Recall: 0.92 Precision: 0.90 f1: 0.91

Label 3 (Sci/Tech): Recall: 0.92 Precision: 0.91 f1: 0.92

Looking at the results, we can see that technology and business news are harder to tell apart compared to world and sports news, sports news in particular being easiest to classify.

Also regarding fine tuning, the learning rates when fine tuning transformer based networks need to be much lower than values that can be used for RNNs and CNNs as with high learning rates, the networks do not converge.