Student Name: Hsing-Hao Wang
Student ID: 116855587

Evaluation Report for NLP Homework 3: Comparative Analysis of RNN Architectures for Sentiment Classification

---

GitHub Repository Link:
https://github.com/jerrryw/NLP_Homework3

1. Dataset Summary

The preprocessing process splits the IMDB dataset into 50/50, turning the whole dataset into 25,000 training and 25,000 testing samples. Then, we normalize texts by converting it to lowercase and removing special characters. Each review is tokenized using the nltk tokenizer, and we keep a vocabulary of only the top 10,000 most frequent words. Tokens are mapped to integer IDs, and reviews are padded into fixed lengths of 25, 50, and 100 words. The vocabulary size is 10,000, and the raw review length is around 230 to 240 tokens.

2. Model Configuration

- Embedding dimension: 100
- Hidden size: 64
- Number of layers: 2
- Dropout: 0.5
- Optimizer Settings: adam, sgd, rmsprop
- Loss function: binary cross-entropy loss

3. Comparative Analysis

The table for RNN models with different configurations are included in this table, further results for LSTM and BiLSTM are truncated, check metrics.csv at GitHub for more details.

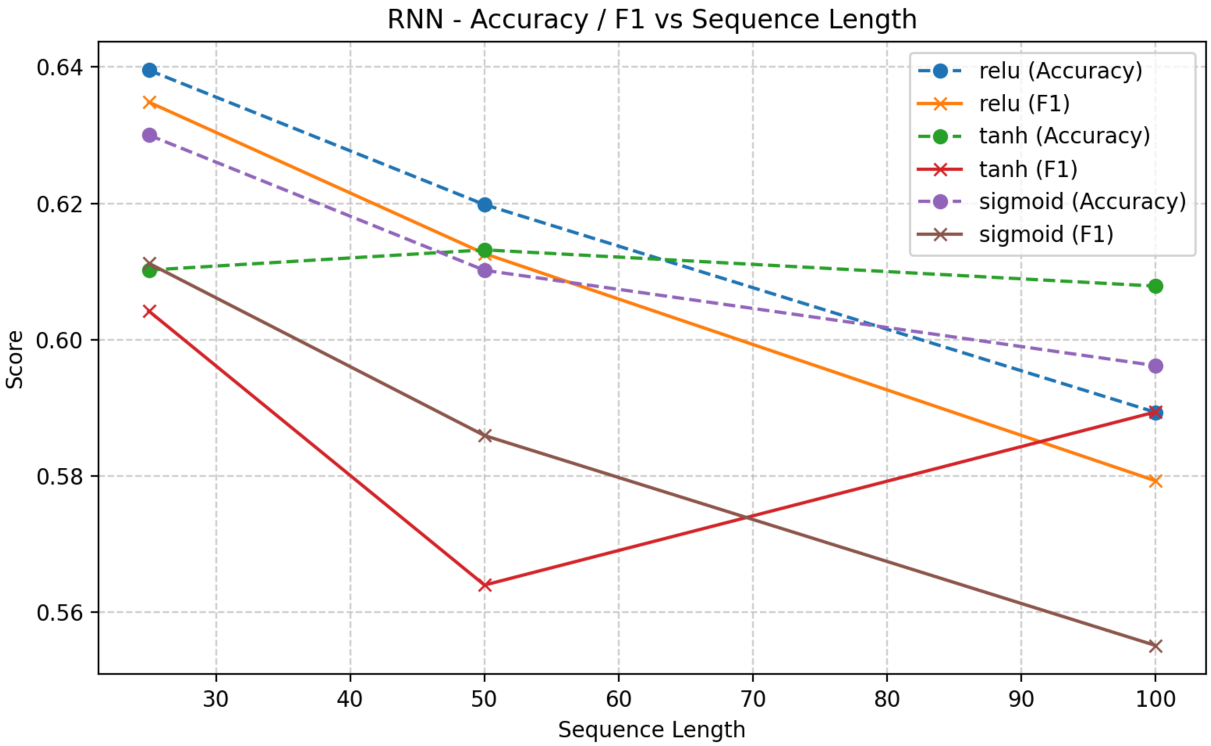
Summary Table for results:

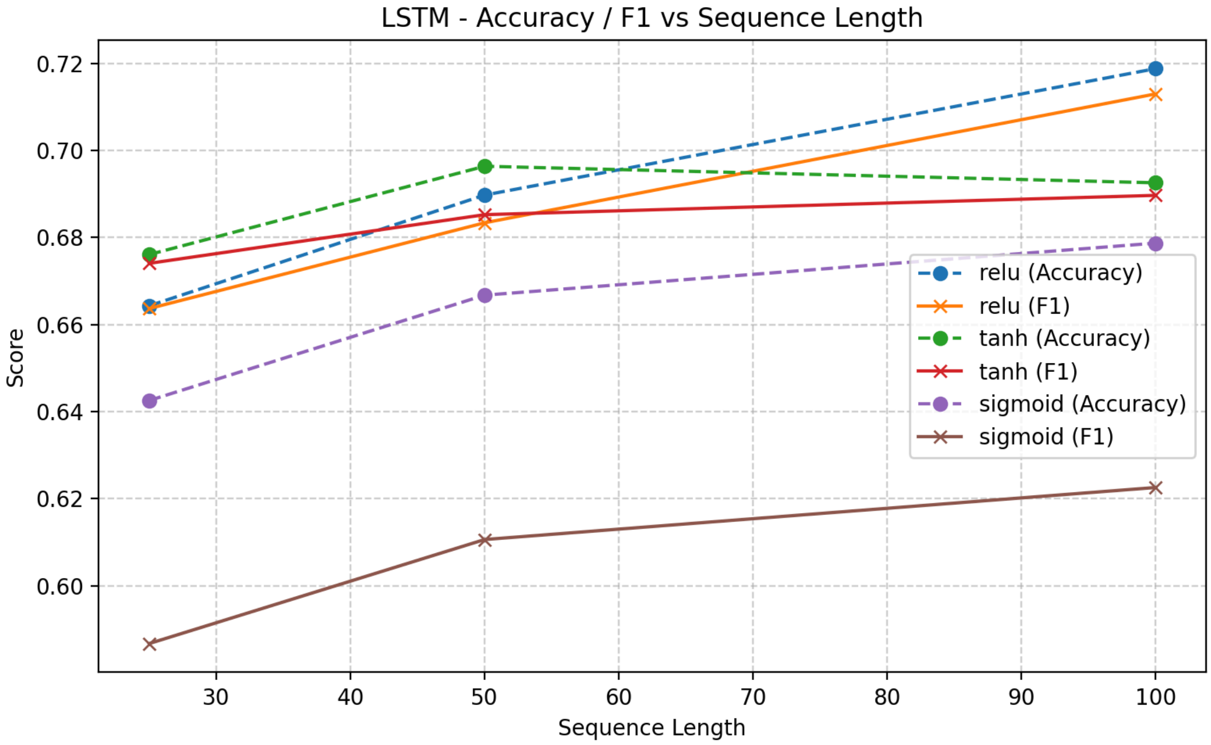
Model	Activation	Optimizer	Sequence Length	Gradient Clipping	Accuracy	F1-Score	Epoch Time (s)
RNN	relu	adam	25	None	0.6543	0.6532	5.49

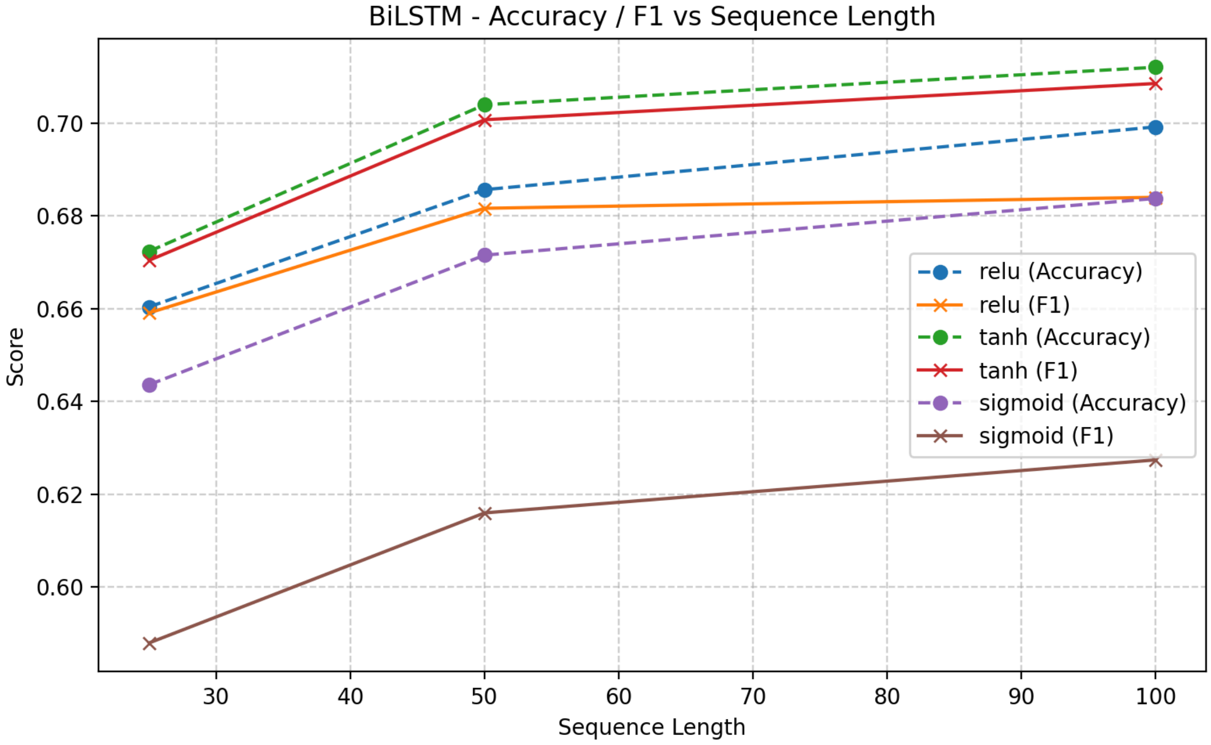
RNN	relu	adam	25	Yes	0.6827	0.6812	4.97
RNN	relu	adam	50	None	0.5968	0.5903	4.07
RNN	relu	adam	50	Yes	0.7004	0.699	4.54
RNN	relu	adam	100	None	0.5547	0.5485	4.79
RNN	relu	adam	100	Yes	0.6181	0.6173	5.08
RNN	relu	sgd	25	None	0.5798	0.574	3.79
RNN	relu	sgd	25	Yes	0.5834	0.5663	4.34
RNN	relu	sgd	50	None	0.5534	0.5299	4.47
RNN	relu	sgd	50	Yes	0.5548	0.5483	4.53
RNN	relu	sgd	100	None	0.543	0.522	4.54
RNN	relu	sgd	100	Yes	0.5317	0.5079	4.73
RNN	relu	rmsprop	25	None	0.6556	0.6549	3.96
RNN	relu	rmsprop	25	Yes	0.6814	0.6795	4.81
RNN	relu	rmsprop	50	None	0.62	0.6163	4.14
RNN	relu	rmsprop	50	Yes	0.6935	0.692	4.57
RNN	relu	rmsprop	100	None	0.621	0.6204	4.77
RNN	relu	rmsprop	100	Yes	0.6676	0.6597	5.1
RNN	tanh	adam	25	None	0.5314	0.5179	4.28
RNN	tanh	adam	25	Yes	0.6751	0.6751	5.02
RNN	tanh	adam	50	None	0.654	0.6539	4.27
RNN	tanh	adam	50	Yes	0.689	0.6881	4.61
RNN	tanh	adam	100	None	0.5602	0.5124	5.0

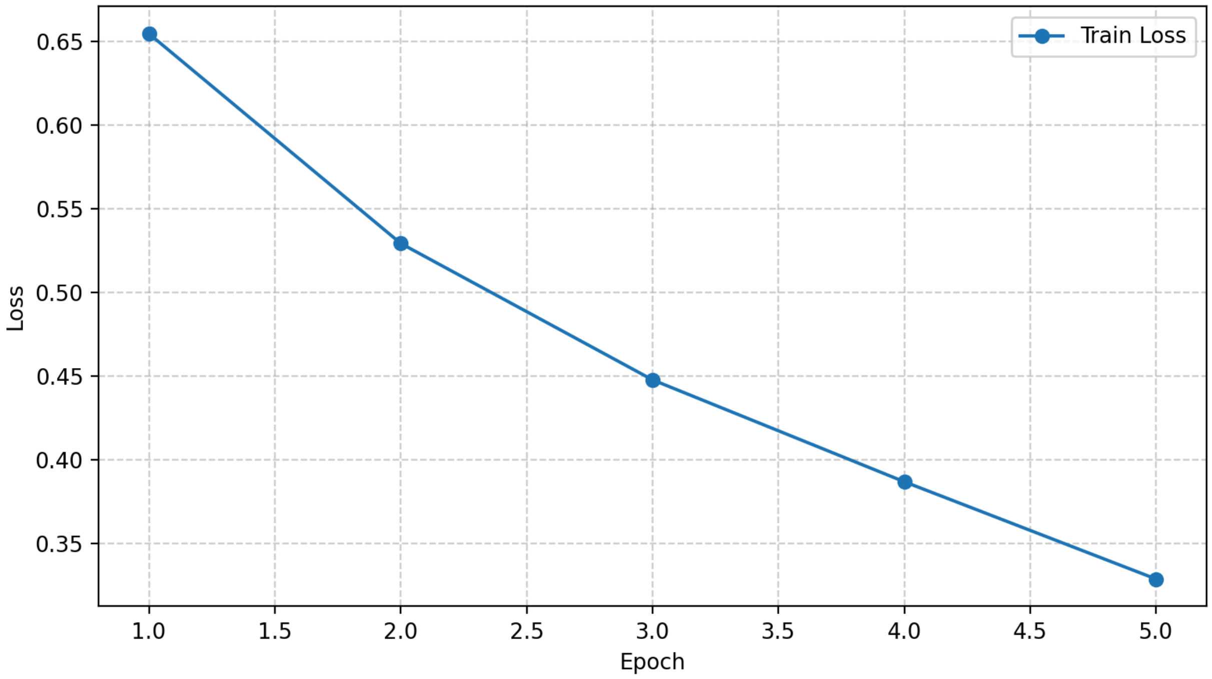
RNN	tanh	adam	100	Yes	0.6848	0.6848	4.95
RNN	tanh	sgd	25	None	0.5699	0.5559	3.97
RNN	tanh	sgd	25	Yes	0.5826	0.58	4.63
RNN	tanh	sgd	50	None	0.499	0.333	4.43
RNN	tanh	sgd	50	Yes	0.517	0.4057	4.5
RNN	tanh	sgd	100	None	0.5168	0.4902	5.55
RNN	tanh	sgd	100	Yes	0.5833	0.5634	5.52
RNN	tanh	rmsprop	25	None	0.6293	0.6235	4.17
RNN	tanh	rmsprop	25	Yes	0.673	0.6728	5.12
RNN	tanh	rmsprop	50	None	0.6259	0.6098	4.15
RNN	tanh	rmsprop	50	Yes	0.6942	0.6936	5.05
RNN	tanh	rmsprop	100	None	0.6159	0.5996	5.28
RNN	tanh	rmsprop	100	Yes	0.6862	0.6861	5.14
RNN	sigmoid	adam	25	None	0.6522	0.6522	4.69
RNN	sigmoid	adam	25	Yes	0.671	0.6709	4.66
RNN	sigmoid	adam	50	None	0.5925	0.5921	4.05
RNN	sigmoid	adam	50	Yes	0.6757	0.6753	4.5
RNN	sigmoid	adam	100	None	0.5552	0.4948	4.87
RNN	sigmoid	adam	100	Yes	0.662	0.662	5.11
RNN	sigmoid	sgd	25	None	0.5164	0.4085	4.1
RNN	sigmoid	sgd	25	Yes	0.5848	0.5848	4.45
RNN	sigmoid	sgd	50	None	0.5441	0.4937	3.93

RNN	sigmoid	sgd	50	Yes	0.5351	0.4421	4.35
RNN	sigmoid	sgd	100	None	0.56	0.5294	4.71
RNN	sigmoid	sgd	100	Yes	0.5062	0.3539	4.86
RNN	sigmoid	rmsprop	25	None	0.6791	0.6783	4.29
RNN	sigmoid	rmsprop	25	Yes	0.6766	0.6725	4.74
RNN	sigmoid	rmsprop	50	None	0.6195	0.6185	4.26
RNN	sigmoid	rmsprop	50	Yes	0.6942	0.694	4.43
RNN	sigmoid	rmsprop	100	None	0.5978	0.5949	4.93
RNN	sigmoid	rmsprop	100	Yes	0.696	0.6959	5.19

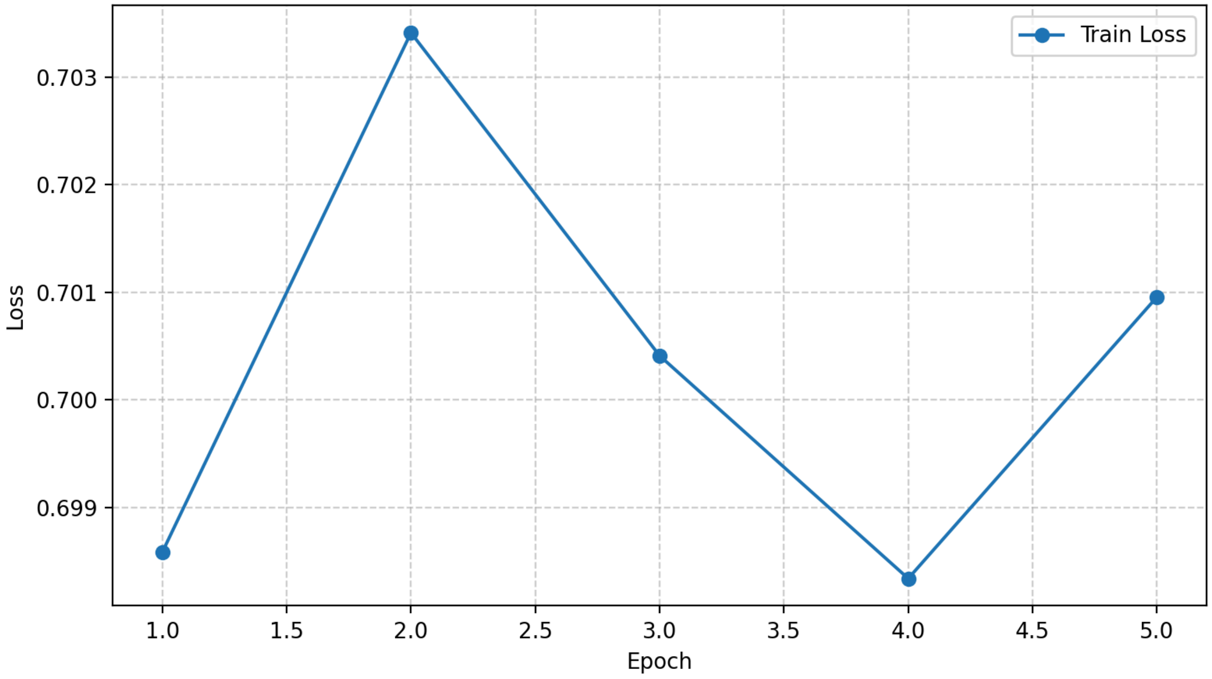



LSTM - Accuracy / F1 vs Sequence Length
Score
0.72
0.70
0.68
0.66
0.64
0.62
0.60
30
40
50
60
70
80
90
100
Sequence Length
relu (Accuracy)
relu (F1)
tanh (Accuracy)
tanh (F1)
sigmoid (Accuracy)
sigmoid (F1)


BiLSTM - Accuracy / F1 vs Sequence Length
Score
0.70
0.68
0.66
0.64
0.62
0.60
30
40
50
60
70
80
90
100
Sequence Length
relu (Accuracy)
relu (F1)
tanh (Accuracy)
tanh (F1)
sigmoid (Accuracy)
sigmoid (F1)



Training Loss vs Epochs (Best model)
LSTM-relu-adam-Seq:50-Clip:1.0



Training Loss vs Epochs (Worst model)
BiLSTM-sigmoid-sgd-Seq:100-Clip:None

4. Discussion

4.1 Which configuration performed best?

Model: LSTM
Activation: relu
Optimizer: adam
Sequence length: 100
Clipping: 1.0
Accuracy: 0.8132
F1-score: 0.8131
Epoch Time (s): 5.41

4.2 How did sequence length or optimizer affect performance?

The sequence length and optimizer choice strongly affect performance. Shorter sequences (25 tokens) train faster but only capture limited context, causing a lower accuracy and F1-scores. Medium length sequences (50 tokens) provide a good balance between efficiency and contextual depth. Longer sequences (100 tokens) include more information and high F-1 scores but can have slow convergence and can cause overfitting.

Among optimizers, RMSprop delivers the most stable and accurate results. The sequence lengths produce longer temporal dependencies that RMSprop handles better. It adapts more effectively to text sequences and loss landscapes.

4.3 How did gradient clipping impact stability?

Applying gradient clipping prevents the gradients from growing excessively large during backpropagation. This helps stabilize training by avoiding the exploding gradient problem. When enabled, gradient clipping keeps the updated values within a controlled range, leading to smoother loss curves and more consistent accuracy improvements. Without gradient clipping, the training process could be unstable.

5. Conclusion

With GPU and CUDA acceleration, the code is able to compute faster and handle longer sequence length. If run with only CPU, the optimal configuration may be top 5000 words and sequence lengths (10, 15, 20). Gradient clipping could be removed as well to avoid extra norm computations. The current number of epochs is 5, reducing to 3 could improve performance.