NANYANG TECHNOLOGICAL UNIVERSITY
AI6127-DEEP NEURAL NETWORKS FOR NATURAL LANGUAGE PROCESSING
Assignment1 Individual
Topic: Deep Learning models for Sentiment Classification
NTAMBARA Etienne G2304253K
April 30, 2024

NTAMBARA Etienne G2304253K [1 2 3] April 30, 2024

1. Introduction

In the given assignment, we were asked to experiment with deep-learning models for sentiment analysis with different optimizers such as SGD (table: 1), Adam(table: 2) and Adagrad (table: 3) where the conducted results will be shown in the next section of this report. I have made the second experiment to Adam optimizer about 5(table: 4),10(table: 5),20 (table: 6) and 50 (table: 7) epochs the results will also be shown in this report. Both results were conducted from a given IMDB dataset and RNN (Recurrent Neural Network) model.

After conducting the experiment I have also used a pre-trained model of Word2Vec to improve the performance of previous results in the RNN model, the experimented outcomes will be shown later.

Finally, I used Adam optimiser with 50 epochs and randomly initialized embeddings to the IMDB dataset and produced the experiments with different models such as One-layer feed-forward neural network, Two-layer feed-forward neural network, Three-layer feed-forward neural network, CNN, LSTM, and Bi-LSTM.

2. Conducted Results

2.1. Task 1

Now, let's visualize the experimental results conducted from SGD, Adam and Adagrad optimisers with 20 epochs in the IMDB dataset.

2.1.1. SGD Optimiser Results

SGD Optimizer: Demonstrated moderate effectiveness, with all losses around 0.68 and a training accuracy of 56.15%, validation accuracy 54.36%, and testing accuracy 54.57%. The results conducted suggest that SGD (table: 1) convergence is slower, and careful hyperparameter tuning, possibly including momentum, is required.

Table 1. SGD Optimiser Results

20TH Epoch	Loss	Accuracy (%)
Train	0.682	56.15
Val	0.686	54.36
Test	0.684	54.57

2.1.2. Adam Optimiser Results

Adam Optimizer: Outperformed SGD (table: 1) in fitting the training data, as indicated by the lower training loss of 0.346 and a higher accuracy of 85.91%. However, the performance on the validation (72.60%) and test (69.25%) sets dropped significantly, indicating a potential overfitting issue.

Table 2. Adam Optimiser Results

20TH Epoch	Loss	Accuracy (%)
Train	0.346	85.91
Val	0.651	72.60
Test	0.638	69.25

2.1.3. Adagrad Optimiser Results

Adagrad Optimizer: Achieved better generalization than Adam (table: 2), with a training loss of 0.556 and more stable validation (0.591) and test (0.586) losses. The model's accuracies (training, validation and testing(72.20%, 69.73%,69.80%)) were consistent across the datasets, suggesting a more robust generalization compared to Adam [table: 2].

Table 3. Adagrad Optimiser Results

20TH EPOCH	LOSS	ACCURACY (%)
TRAIN	0.556	72.20
VAL	0.591	69.73
TEST	0.586	69.80

In summary, while Adam optimiser [2] provided high accuracy on the training set but may require regularization to improve generalization by avoiding overfitting. Adagrad (table: 3) demonstrates more consistent performance across training, validation, and test datasets, indicating better generalization compared to SGD (table: 1) and Adam optimisers.

2.2. Task 2

Experimental results of Adam optimiser with 5, 10, 20, and 50 epochs in the IMDB dataset.

2.2.1. ADAM OPTIMISER FOR 5 EPOCHS

For the 5th epoch, the model shows initial fitting with training and validation losses over 0.65 and a higher test accuracy of 61.61%.

Table 4. Adam optimiser for 5 epochs

5TH EPOCH	LOSS	ACCURACY (%)
TRAIN	0.655	59.50
VAL	0.659	59.56
TEST	0.651	61.61

2.2.2. ADAM OPTIMISER FOR 10 EPOCHS

For the 10th epoch, training accuracy enhances significantly to 81.72%, though validation 68.79% and test accuracies 68.00% are lower, indicating the model might be starting to overfit (validation and testing losses are higher than training loss).

Table 5. Adam optimiser for 10 epochs

10TH EPOCH	LOSS	ACCURACY (%)
TRAIN	0.408	81.72
VAL	0.660	68.79
TEST	0.616	68.00

2.2.3. ADAM OPTIMISER FOR 20 EPOCHS

At the 20th epoch in (table: 6), training accuracy of 85.91%, but the validation and test accuracies show a notable drop compared to training, further suggesting overfitting

Table 6. Adam optimiser for 20 epochs

20TH EPOCH	LOSS	ACCURACY (%)
TRAIN	0.346	85.91
VAL	0.651	72.60
TEST	0.638	69.25

2.2.4. ADAM OPTIMISER FOR 50 EPOCHS

Lastly, at the 50th epoch in (table: 7), while training accuracy is high at 94.31%, validation loss jumps to 1.18, and validation, test accuracies are much lower, and this is an understandable sign of overfitting.

Table 7. Adam optimiser for 50 epochs

50TH EPOCH	LOSS	ACCURACY (%)
TRAIN	0.147	94.31
VAL	1.18	65.48
TEST	0.616	68.00

In summary, while training accuracy increases with more epochs, the Adam Optimizer's performance on unseen data does not improve correspondingly, indicating that the model benefits from regularization techniques to prevent overfitting and improve generalization.

2.3. Task 3

For the improvement of the performance of the RNN model, I have presented the conducted results from the IMDB dataset and pre-trained Word2Vec embeddings as initialization of the model using Adam optimiser with 50 epochs(table: 8).

2.3.1. RESULTS FOR ADAM OPTIMIZER AND 50 EPOCH FOR PRETRAINED WORD2VEC EMBEDDING

With pre-trained Word2Vec embeddings and 50 epochs of Adam, the RNN model demonstrates a high training accuracy of 99.45% and a low training loss of 0.023. The test accuracy is 75.29%, but the validation accuracy is 74.51%, with a high validation loss of 1.150 and 0.548 test loss which are higher compared to training loss. In contrast to earlier epochs without pre-trained embeddings, when the model had tighter validation/test accuracies and a more moderate training accuracy, this suggests a significant overfitting. This indicates that the pre-trained embeddings model matches the training data too well, sacrificing its ability to generalize to new data, results are shown in (table: 8).

Table 8. Results for Adam optimizer and 50 epoch for Pretrained Word2Vec Embedding with Adam Optimiser

50^{TH} EPOCH	LOSS	ACCURACY (%)
TRAIN	0.023	99.45
VAL	1.150	74.51
TEST	0.548	75.29

2.4. Task 4

I have used Adam optimizer, 50 epochs and randomly initialized embedding with different models and gained the different results as shown in (tables: 9, 10, 11, 12, 13, 14):

2.4.1. ONE-LAYER FEED-FORWARD NEURAL NETWORK (FFNN), HIDDEN DIMENSION IS 500

One-layer FFNN (table: 9): Displayed nearly perfect training performance but showed possible signs of overfitting as evidenced by higher validation and test losses compared to training loss.

Table 9. One-layer FFNN with Adam Optimiser, hidden dimension 500

50^{TH} EPOCH	LOSS	ACCURACY (%)
TRAIN	0.002	99.94
VAL	0.851	88.36
TEST	0.333	86.58

2.4.2. TWO-LAYER FEED-FORWARD NEURAL NETWORK, HIDDEN DIMENSIONS ARE 500 AND 300

Two-layer FFNN (table: 10): Also demonstrated high training accuracy, but with significant overfitting, indicated by huge validation and test losses.

Table 10. Two-layer FFNN with Adam Optimiser, hidden dimensions are 500 and 300

50^{TH} EPOCH	LOSS	ACCURACY (%)
TRAIN	0.034	99.38
VAL	1.514	87.27
TEST	0.361	85.71

2.4.3. THREE-LAYER FEED-FORWARD NEURAL NETWORK, HIDDEN DIMENSIONS ARE 500, 300, AND 200

Three-layer FFNN (table: 11): Similar to the two-layer FFNN model, it had high training accuracy and signs of overfitting, with validation and testing losses notably higher than training loss.

Table 11. Three-layer FFNN with Adam Optimiser, hidden dimensions are 500, 300, and 200

50^{TH} EPOCH	LOSS	ACCURACY (%)
TRAIN	0.009	99.55
VAL	1.187	87.92
TEST	0.379	84.18

2.4.4. CNN MODEL (USING THREE FEATURE MAPS WITH THE SIZES 1, 2, AND 3)

CNN (table: 12) Achieved perfect training accuracy and maintained comparatively lower validation and test losses, indicating strong generalization.

Table 12. CNN model with Adam Optimiser

50^{TH} EPOCH	LOSS	ACCURACY (%)
TRAIN	0.000	100.00
VAL	0.526	89.07
TEST	0.286	88.45

2.4.5. LSTM MODEL

LSTM (table: 13): Displayed high training accuracy with slightly elevated validation and test losses, suggesting good generalization with a slight overfit.

Table 13. LSTM model with Adam Optimiser

50^{TH} EPOCH	LOSS	ACCURACY (%)
TRAIN	0.004	99.87
VAL	0.813	86.97
TEST	0.328	85.85

2.4.6. BI-LSTM MODEL

Bi-LSTM (table: 14): Showed high training accuracy, yet the validation loss was high, which could indicate overfitting, though the test loss was somewhat lower.

Table 14. Bi-LSTM with Adam Optimiser

50^{TH} EPOCH	LOSS	ACCURACY (%)
TRAIN	0.003	99.78
VAL	1.029	86.59
TEST	0.353	85.50

2.4.7. RESULT'S SUMMARY

The performance of various neural network designs trained with the Adam optimizer over 50 epochs is detailed in the above tests. On both the validation and test datasets, the CNN [table: 12] model exhibits the least overfitting and excellent accuracy, demonstrating strong generalization. Although the one-layer, two-layer, and three-layer FFNNs achieve high training accuracy, a considerable decline in validation and test accuracies indicates overfitting. Compared to FFNNs, the LSTM and Bi-LSTM models exhibit better training performance with less overfitting.

Acknowledgement

Credits to (Maas et al., 2011) for preparing IMBD dataset used in this assignment, (Rehurek, 2024) for Word2vec as pre-trained model used, and (contributors, 2018) for preparing some codes to initialize the model

References

contributors, S. O. *PyTorch / Gensim - How do I load pre-trained word embeddings?* 2018. URL `https://stackoverflow.com/questions/49710537/pytorch-gensim-how-do-i-load-pre-trained-word-embeddings`. Question asked on Stack Overflow.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/P11-1015`.

Rehurek, R. *Word2vec embeddings*. Gensim, 2024. URL `https://radimrehurek.com/gensim/models/word2vec.html`. Gensim: Topic modelling for humans.

[1]Nanyang Technological University (NTU), Singapore [2]School of Computer Science and Engineering (SCSE) [3]Master of Science in Artificial Intelligence. Correspondence to: Ntambara Etienne <ntam0001@e.ntu.edu.sg>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.