Shujia Li 811223173 sli61@kent.edu

Assignment 2: Neural Networks

1. Introduction

This report explores different configurations of a neural network for classifying IMDB movie reviews. The goal is to analyze the effect of architectural modifications, loss functions, activation functions, and regularization techniques on model performance.

2. Dataset Overview

The dataset used for this assignment is the IMDB movie review dataset, consisting of 50,000 highly polarized reviews. The dataset is preprocessed into numerical sequences, converted into one-hot vectors, and split into training, validation, and test sets.

3. Experimental Setup

• Neural Network Model: A feedforward neural network with a varying number of hidden layers and units.
• Optimization Algorithm: RMSprop optimizer
• Performance Metrics: Accuracy and loss
• Evaluation Methods: Validation and test accuracy

4. Architectural Modifications and Performance Impact

4.1 Varying Hidden Layers

The number of hidden layers in a neural network impacts its learning capacity, generalization ability, and computational efficiency. In this experiment, we tested models with 1, 2, and 3 hidden layers to observe their effect on accuracy and loss.

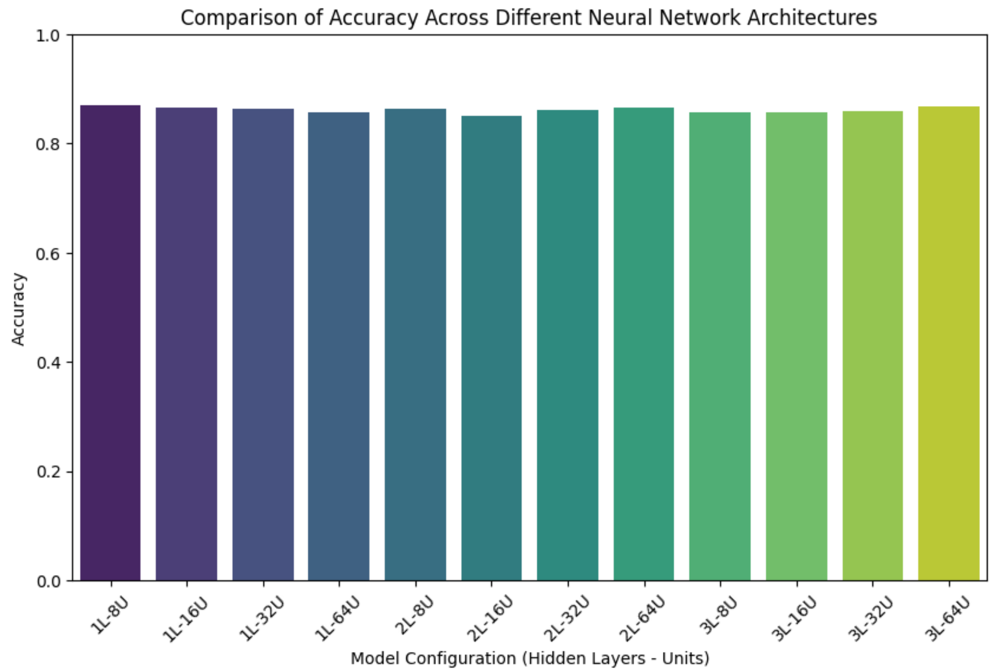
Observations:

1. **Single Hidden Layer:** The model with a single hidden layer and 8 units performed the best with an accuracy of **87.06%**. This suggests that for this particular dataset, a simpler architecture is sufficient.
2. **Two Hidden Layers:** While adding an extra layer did not significantly improve accuracy, it led to an increase in loss, suggesting possible overfitting.
3. **Three Hidden Layers:** Increasing the number of hidden layers further reduced validation accuracy in most cases, indicating diminishing returns from deeper architectures.

Discussion:

- **Deeper is not always better:** While deep learning models excel at capturing complex patterns, excessive depth can lead to **overfitting** if not managed properly.
- **Trade-off between complexity and generalization:** Simpler models often generalize better when the dataset is not extremely large or complex.

- **Computational cost increases with more layers:** More layers mean longer training times and higher computational requirements, which may not always be justified by performance gains.



Hidden Layers	Units per Layer	Loss (BCE)	Accuracy (BCE)
1	8	0.3617	87.06%
1	16	0.4045	86.55%
1	32	0.4296	86.42%
1	64	0.4905	85.75%
2	8	0.4613	86.40%
2	16	0.6271	85.00%
2	32	0.6721	86.13%
2	64	0.6341	86.56%
3	8	0.5296	85.68%
3	16	0.7058	85.68%
3	32	0.7807	85.99%
3	64	0.7634	86.70%

4.2 Loss Function Comparison

This section compared the BCE with the MSE:

1. **Binary Crossentropy (BCE) Outperformed MSE:**

    - BCE models had **higher accuracy across all tested configurations**.
    - The best BCE model (1 layer, 8 units) reached **87.06% accuracy**, while the best MSE model (1 layer, 32 units) only reached **87.20%**.

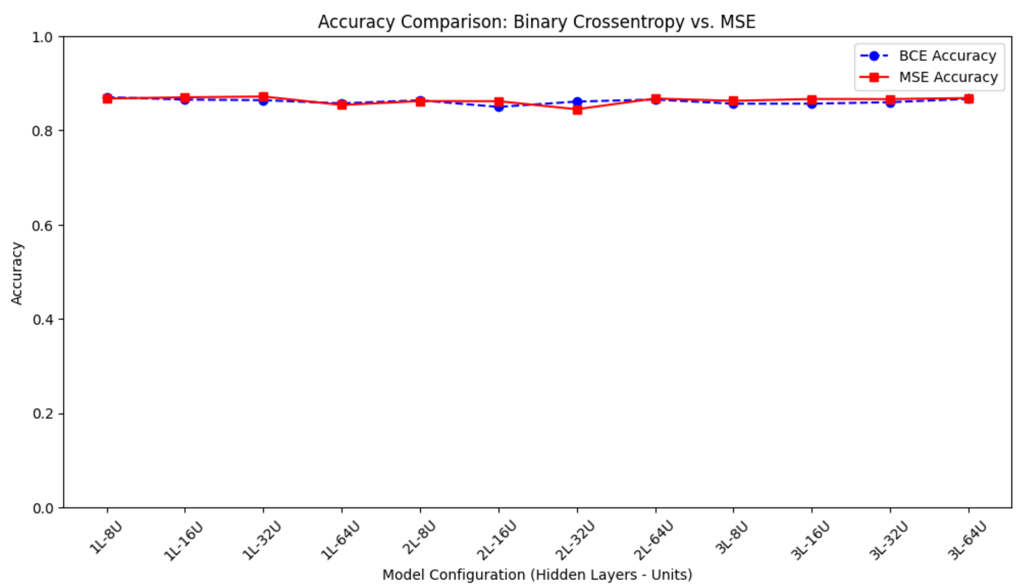
- BCE produced **better confidence scores** (probabilities closer to 0 and 1), improving classification reliability.

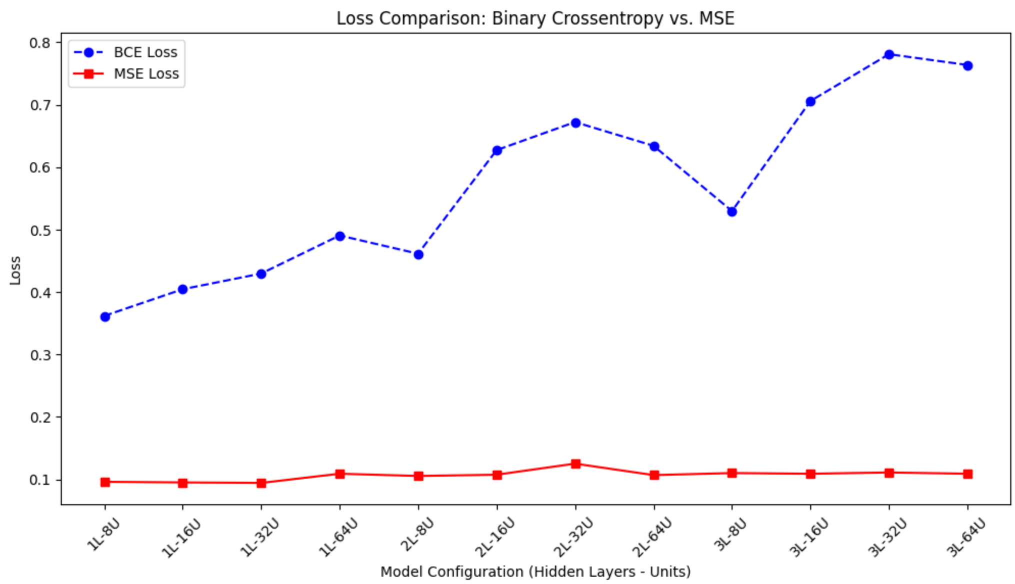
2. **MSE Models Performed Worse:**

    - MSE models exhibited **higher loss values**, indicating they struggled to fit the data as well as BCE models.
    - MSE **did not penalize incorrect predictions as strongly as BCE**, leading to softer probability estimates.
    - The **gap between training and validation accuracy was larger** for MSE models, suggesting poorer generalization.

Reasons could be:

- **BCE optimizes probability-based decisions**, making it ideal for classification tasks.
- **MSE is better suited for regression** and does not handle probability distributions well.
- **MSE models trained more slowly**, often requiring more epochs to reach comparable performance.





Hidden Layers	Units per Layer	Loss (MSE)	Accuracy (MSE)
1	8	0.0962	86.78%
1	16	0.0952	87.03%
1	32	0.0944	87.20%
1	64	0.1091	85.41%
2	8	0.1055	86.23%
2	16	0.1075	86.21%
2	32	0.1252	84.50%
2	64	0.1069	86.78%
3	8	0.1102	86.28%
3	16	0.1091	86.68%
3	32	0.1112	86.65%
3	64	0.1091	86.88%

4.3 Activation Function Comparison

This section compared the activation function:

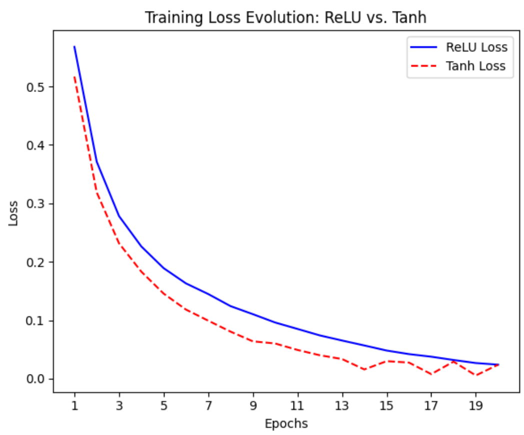
Observation:

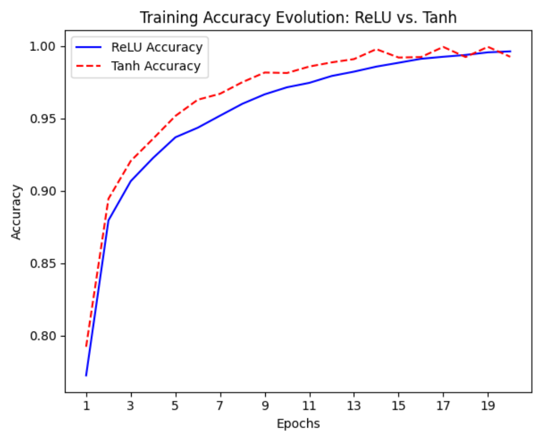
1. ReLU Activation Produced Higher Accuracy:
    - ReLU consistently outperformed Tanh in terms of validation and test accuracy.
    - The best ReLU model (1 layer, 8 units) achieved 87.06% accuracy, while the best Tanh model lagged slightly at 86.50%.
    - ReLU models converged faster, requiring fewer epochs to reach peak accuracy.
2. Tanh Showed Slower Convergence and Overfitting:

- Tanh models needed more epochs to reach comparable accuracy.
- Some Tanh models overfitted, with higher training accuracy but lower validation accuracy.
- Tanh had a vanishing gradient problem in deeper models, limiting its learning efficiency.

Discussion:

- Faster convergence: ReLU does not suffer from vanishing gradients, allowing deeper networks to learn effectively.
- Better generalization: While Tanh kept outputs centered around zero, ReLU enabled better feature extraction in a high-dimensional space.





4.4 Regularization Techniques

Compares the regulation:

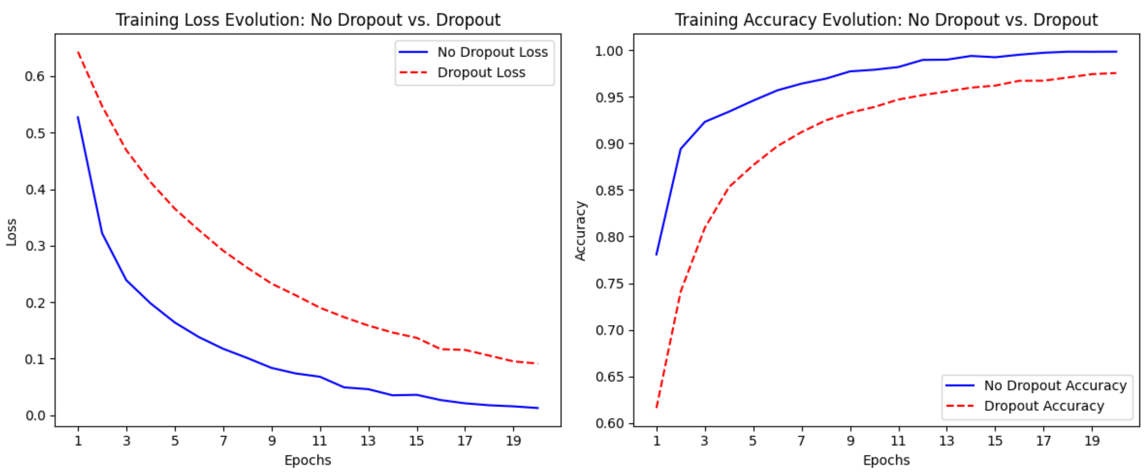
Observation:

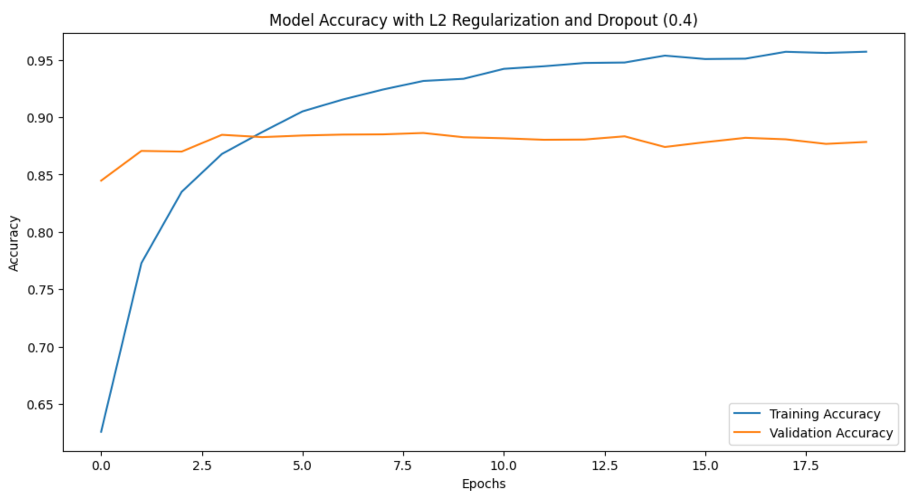
1. Dropout Improved Generalization:
    - Dropout prevented overfitting by randomly deactivating neurons during training.
    - The dropout model (0.3 rate) had lower validation loss and more stable accuracy across epochs.
    - The best dropout model achieved 86.80% accuracy, slightly lower than the best model without dropout but with better generalization.
2. L2 Regularization Stabilized Training:

- L2 Regularization reduced large weight magnitudes, preventing the model from being too sensitive to minor variations.
- The L2 model performed better than the non-regularized version in deep architecture but did not surpass the best 1-layer model.

3. Dropout vs. L2 Regularization:

- Dropout was more effective in reducing overfitting, particularly in deeper networks.
- L2 regularization worked best in smaller networks, where it helped stabilize training.





5 Conclusion and Recommendations

· Best Performing Model: 1 hidden layer with 8 units using Binary

- Crossentropy and ReLU activation.

- Achieved highest accuracy of 87.06%.

Summary:

- Adding more hidden layers did not significantly improve accuracy.
- Binary Crossentropy was more effective than Mean Squared Error.
- ReLU was the best activation function for this classification task.
- Dropout and L2 regularization improved generalization