Assignment 1 Executive Summary

The goal of this analysis was to optimize a deep learning model to classify IMDB Reviews. We tested different model types, activation functions, and loss functions to improve the accuracy of the model while preventing overfitting.

Findings for Each Step in Model Development

1. **Baseline Model**
    - The baseline model, using 2 hidden layers, relu activation function, and 16 hidden units per layer, achieved ~87.9% accuracy. This is what we will compare future model changes against.
2. **Adding a Hidden Layer**
    - Improved accuracy slightly (~88%), but at the cost of increased complexity and training time. The slight improvement to accuracy should not justify the added layer.
3. **Reducing Hidden Units**
    - When modifying the number of hidden units per layer, in addition to the baseline of 16, I tested 8, 32, and 64 units. The original 16 hidden unit version provided a good balance of performance and efficiency.
4. **Switching to Mean Squared Error (MSE) Loss**
    - Switching to the MSE loss function increased the accuracy from baseline very slightly, but the true improvement was the large decrease in our loss function, from 29.5% in the baseline, to 9%.
5. **Changing Activation Functions**
    - Replacing the relu function with tanh actually decreased model accuracy to 86.7%. Not a great choice to move forward with.
6. **Adding a dropout function**
    - In an attempt to reduce overfitting of the testing data, a dropout function of 0.3 was introduced to each hidden layer. This did significantly reduce overfitting, as the difference between the training results and validation/test results was much smaller for both loss and accuracy. Also, the accuracy was the highest of all the models, at 88.42%.

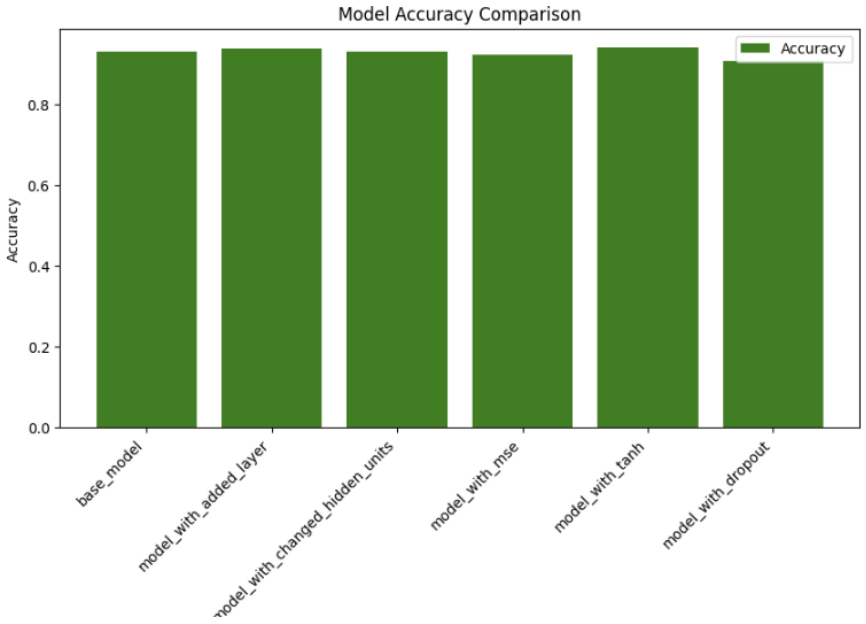
Recommendations:

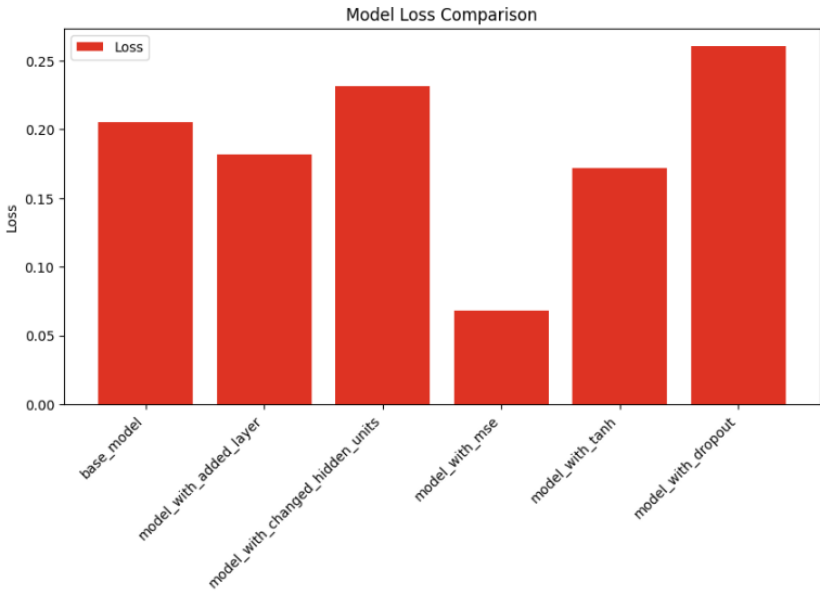
In conclusion, my final recommended model would combine all the most successful parts of the above. I would choose to keep two hidden layers, 16 hidden units per layer, use an MSE loss function, keep the relu activation function, and add a dropout function to each hidden layer. This would give us the most reliable model, while keeping computational complexity and loss functions low, and give us a higher accuracy while avoiding overfitting on our test data.

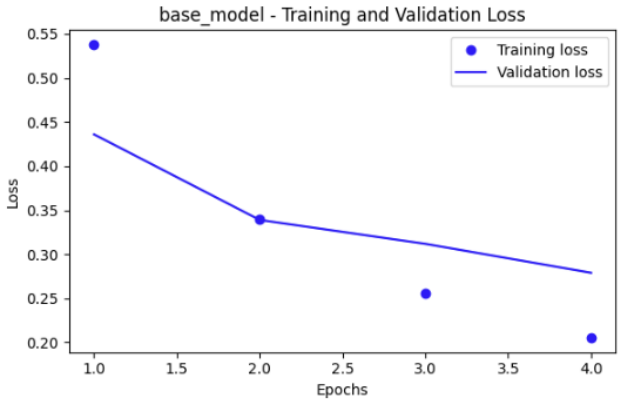
Appendix:

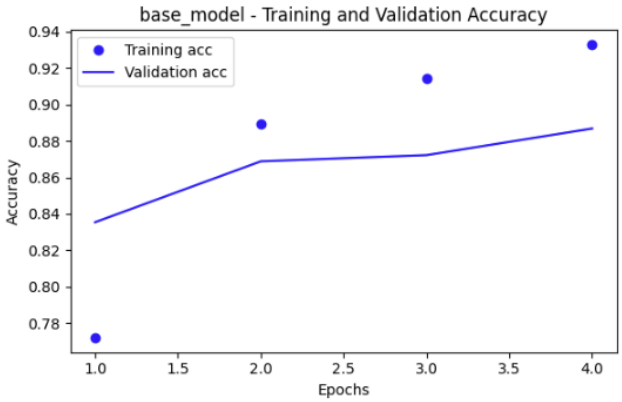
Relevant charts

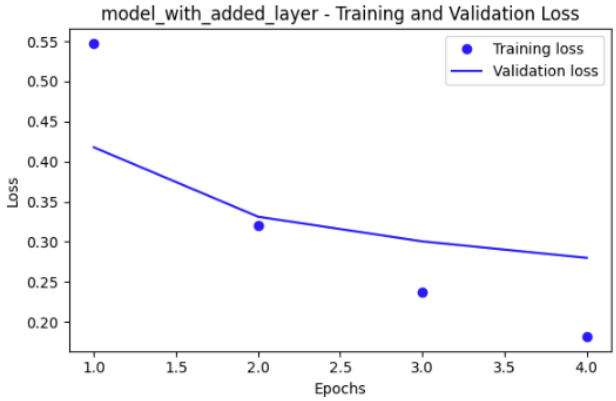
	loss	accuracy
base_model	0.294816	0.87980
model_with_added_layer	0.297208	0.88068
model_with_changed_hidden_units	0.298604	0.88480
model_with_mse	0.090495	0.88300
model_with_tanh	0.326985	0.86728
model_with_dropout	0.287226	0.88420

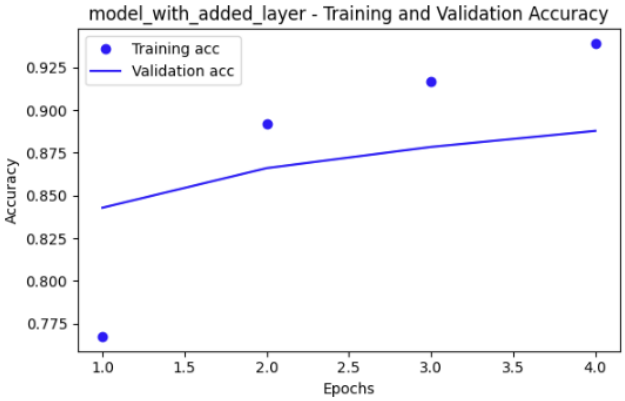


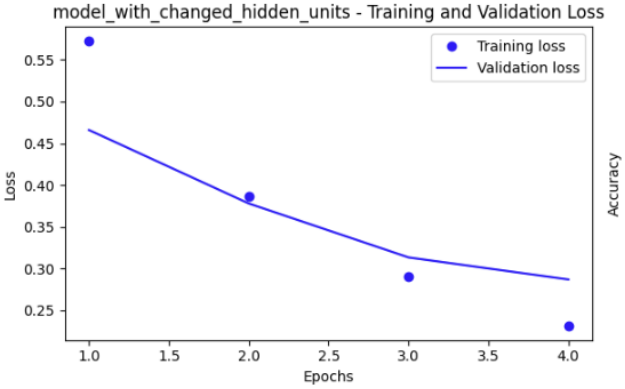


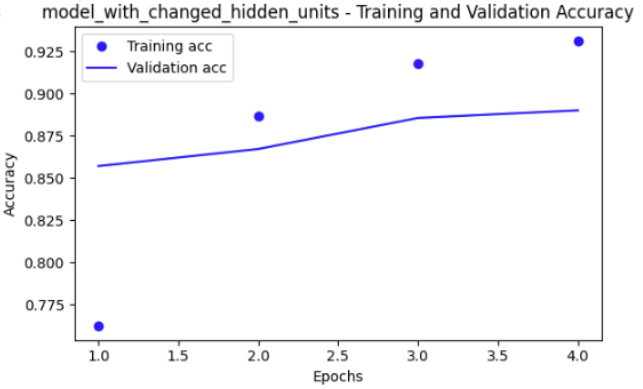

base_model - Training and Validation Loss
Training loss
Validation loss
Loss
Epochs

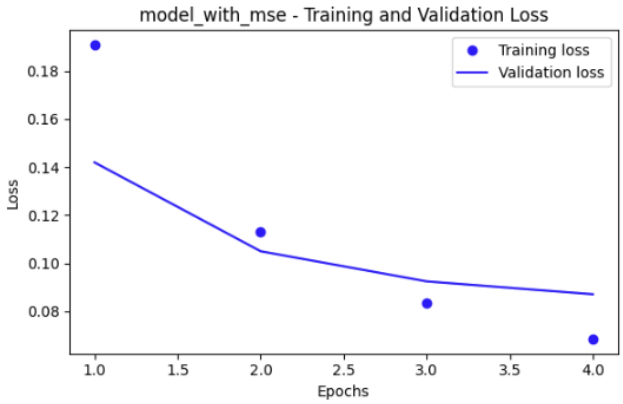

base_model - Training and Validation Accuracy
Training acc
Validation acc
Accuracy
Epochs

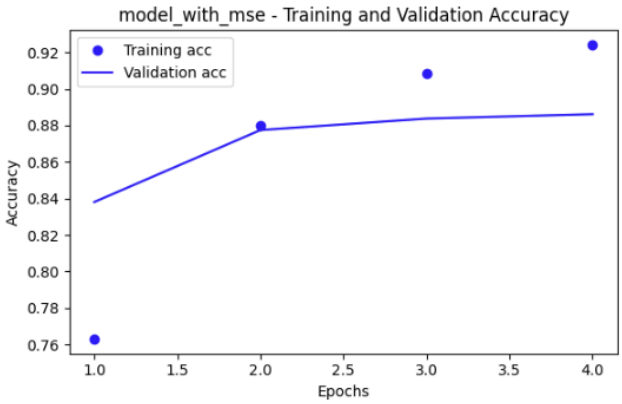

model_with_added_layer - Training and Validation Loss
Training loss
Validation loss
Loss
Epochs

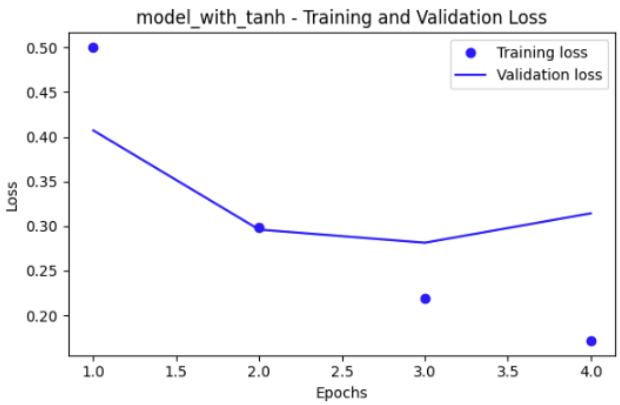

model_with_added_layer - Training and Validation Accuracy
Training acc
Validation acc
Accuracy
Epochs

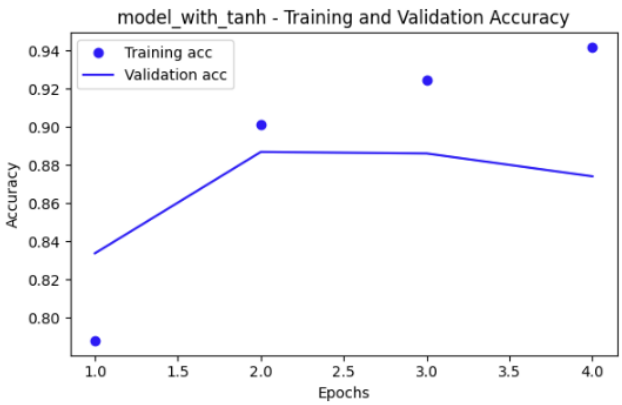

model_with_changed_hidden_units - Training and Validation Loss
Training loss
Validation loss
Loss
Epochs

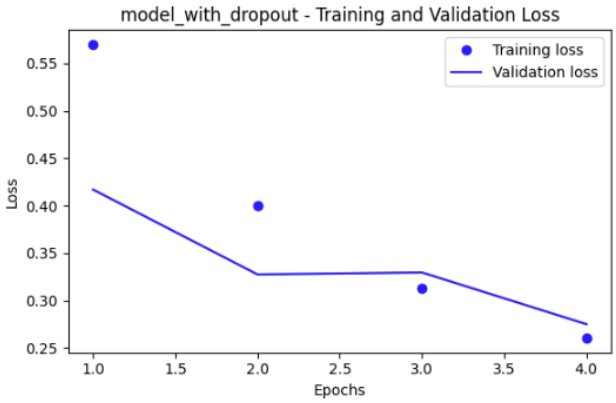

model_with_changed_hidden_units - Training and Validation Accuracy
Training acc
Validation acc
Accuracy
Epochs

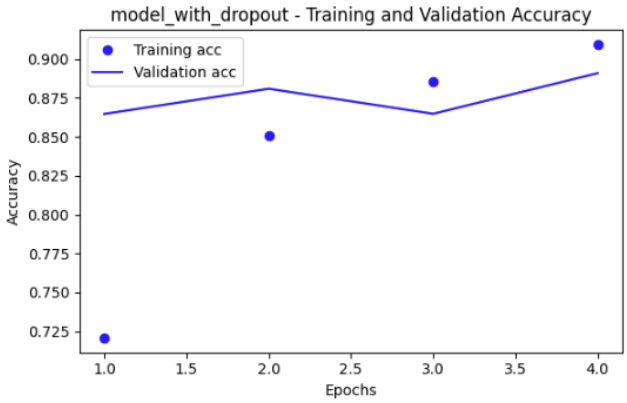

model_with_mse - Training and Validation Loss
Training loss
Validation loss
Loss
Epochs


model_with_mse - Training and Validation Accuracy
Training acc
Validation acc
Accuracy
Epochs


model_with_tanh - Training and Validation Loss
Training loss
Validation loss
Loss
Epochs


model_with_tanh - Training and Validation Accuracy
Training acc
Validation acc
Accuracy
Epochs


model_with_dropout - Training and Validation Loss
Training loss
Validation loss
Loss
Epochs


model_with_dropout - Training and Validation Accuracy
Training acc
Validation acc
Accuracy
Epochs