

## Summary

Given the 50000 reviews in the IMDB dataset. The word count for these reviews is 10,000. All these reviews have labels already. The methods I took to determine the ideal mix of hyperparameters to improve test and validation accuracy are listed below.

I started by importing every module needed to create and run a model. They are.

```
“Import os  
from operator import itemgetter  
  
import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
import tensorflow as tf  
  
  
from keras import models, regularizers, layers, optimizers, losses, metrics  
from keras.models import Sequential  
from keras.layers import Dense  
from keras.utils import np_utils  
from tensorflow.keras.utils import to_categorical  
  
from keras.datasets import imdb”
```

The IMDB dataset was then loaded:

The dataset is split into two sections: a training section and a testing section. There are 25000 reviews in each component. The data is then vectorized because we cannot input integers into neural networks after that. Additional training set is broken into partial training and validation sets in two sections.

### Model Architectures with Varying Hidden Layers and Units

Combination	Training Accuracy	Validation Accuracy	Test Accuracy
One Hidden Layer	0.9450	0.8894	0.8824
Two Hidden Layers	0.9983	0.8696	0.8582
Three Hidden Layers	0.9984	0.8440	0.8326

- The model with one hidden layer outperforms others in terms of validation and test accuracy.
- Multiple hidden layers result in overfitting.

The following two models are constructed using a single hidden layer, the ReLU activation function, the rmsprop optimizer, and the binary cross entropy loss function. The hidden units are now 32 and 64 in number. The outcomes are shown in the following table.

#### Models with Different Hidden Units

Combination	Training Accuracy	Validation Accuracy	Test Accuracy
16 Hidden Units	0.9450	0.8894	0.8824
32 Hidden Units	0.9925	0.8731	0.8615
64 Hidden Units	0.9906	0.8698	0.8591

- 16 hidden units perform the best, avoiding overfitting.

Increasing the hidden units has the same overfitting issue as increasing the hidden layers. The optimal combination to employ is 16 hidden units since it produces the most accuracy. Change the loss function to mse and compare it to binary cross entropy to assess the accuracy now. One hidden layer, 16 hidden units, the ReLU activation function, and the rmsprop optimizer make up these models.

#### Loss Function Comparison

Combination	Training Accuracy	Validation Accuracy	Test Accuracy
Binary Cross Entropy	0.9450	0.8894	0.8824
Mean Squared Error (MSE)	0.9830	0.8782	0.8701

- Binary cross-entropy loss is better suited for this dataset.
- A further model is created utilizing one hidden layer, 16 hidden units, the rmsprop optimizer, and the binary cross entropy loss function to compare the effectiveness of the tanh and relu activation functions.

#### Activation Function Comparison

Combination	Training Accuracy	Validation Accuracy	Test Accuracy
ReLU	0.9450	0.8894	0.8824
Tanh	0.9925	0.8737	0.8637

- ReLU activation function outperforms Tanh.

#### Dropout Regularization

In the next model, I used the Drop out (0.5). In this model, I used one hidden layer, 16 hiddenunits, rmsprop optimizer, Binary cross entropy loss function and relu activation function

Combination	Training Accuracy	Validation Accuracy	Test Accuracy
Without Dropout	0.9450	0.8894	0.8824

With Dropout (0.5)	0.9772	0.8844	0.8753
--------------------	--------	--------	--------

- Using dropout leads to a slight decrease in accuracy, indicating it may not be necessary in this case.

### Regularizer Comparison

Combination	Training Accuracy	Validation Accuracy	Test Accuracy
No Regularizer	0.9450	0.8894	0.8824
L1 Regularizer	0.9993	0.8759	0.8667
L2 Regularizer	0.9925	0.8798	0.8706

- Regularizers (L1 and L2) do not significantly improve accuracy in this scenario.

### Optimizer Comparison

Combination	Training Accuracy	Validation Accuracy	Test Accuracy
RMSprop	0.9450	0.8894	0.8824
Adam	0.9816	0.8833	0.8721

- RMSprop performs better than Adam for the given IMDB dataset.

### Conclusion:

- With limited data samples, one hidden layer with 16 hidden units, ReLU activation, binary cross-entropy loss, and RMSprop optimizer performs the best.
- Optimizers, regularizers, and dropouts may be more useful with larger datasets.
- Obtaining more data could further improve model accuracy.