



# **USING RNN'S TO CATEGORIZE MOVIE REVIEWS**

# INTRODUCTION

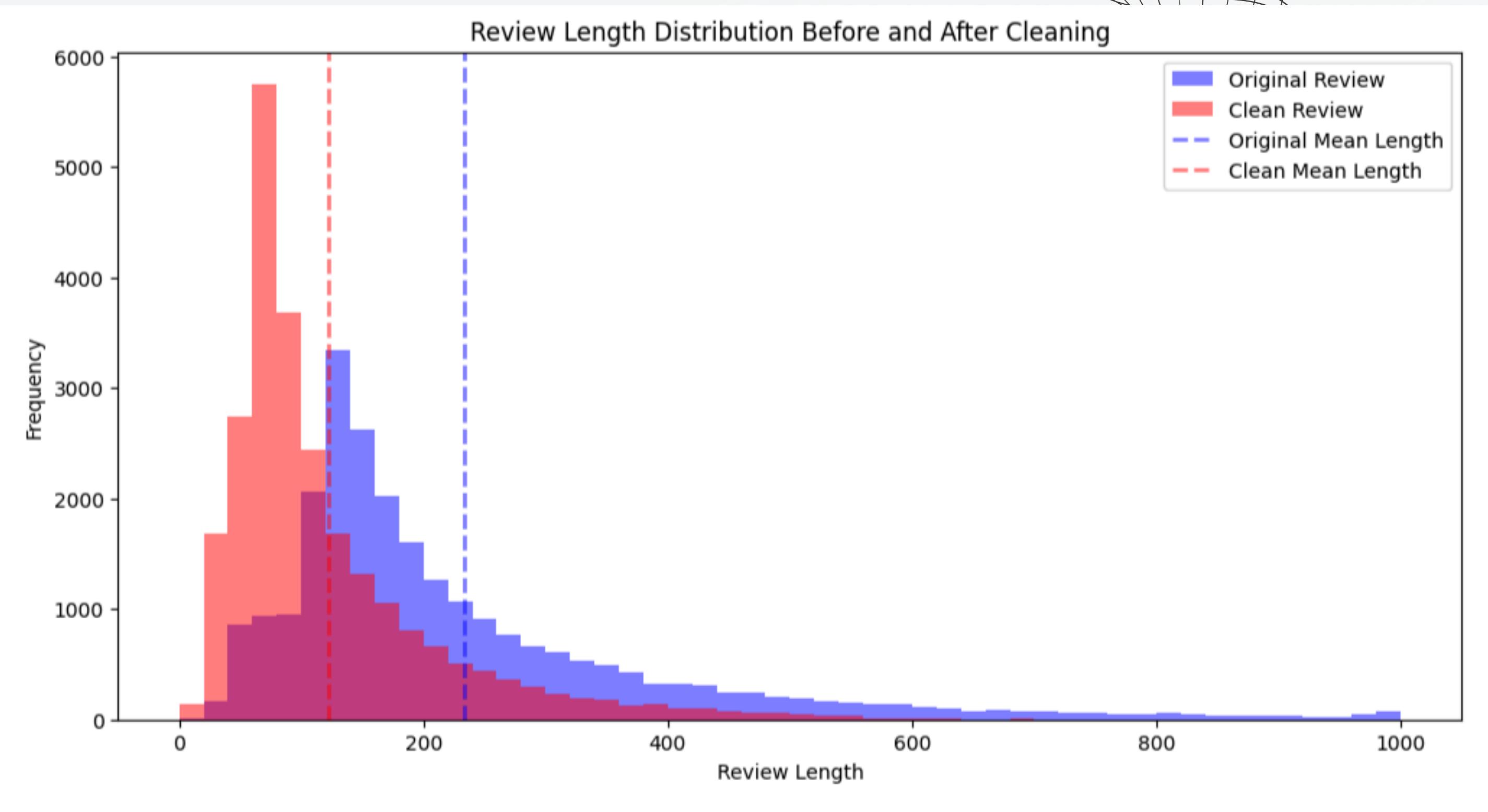
- ◆ **Topic:** Sentiment analysis on IMDb movie reviews
- ◆ **Project goal:** Use Natural Language Processing and Recurrent Neural Networks to classify movie reviews as positive or negative with high accuracy

# DATA CLEANING / EDA

- ◆ **Dataset:** Kaggle dataset with 25,000 observations
- ◆ **Text cleaning:** Remove special characters, lemmatize text, remove stop words
- ◆ **NLP:** Tokenize, pad/truncate to 200 words, Stanford GloVe 100d word embeddings

### Randomly Sampled Positive Review

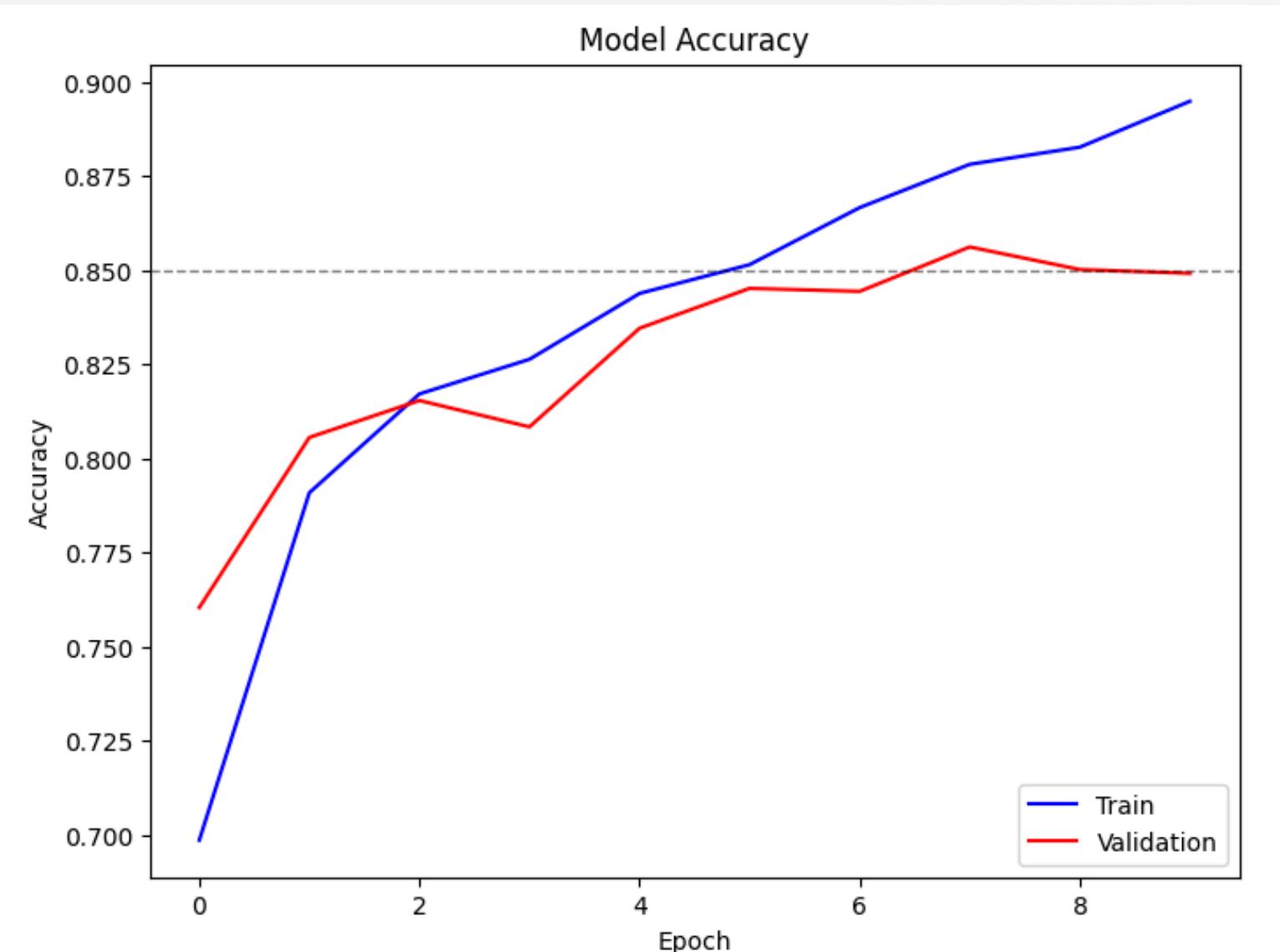
quite  
especially  
kind  
woman  
Year Old  
live  
work  
around  
human  
got  
star  
ending  
audience  
Xica  
will  
best  
first  
score  
actually  
say  
everyone  
two  
found  
heart  
next  
point  
big  
family  
now  
definitely  
seen  
part  
performance  
sure  
put  
voice  
done  
interesting  
watching  
Angie  
Kennedy  
soon  
rather  
without  
course  
Polanski  
remember  
friend  
life  
series  
black  
back  
must  
minute  
saw  
possible  
soldier  
second  
Jackson  
word  
plot  
beautiful  
Johanson  
time  
member  
year  
greater  
man  
see  
one  
ne  
find  
thing  
go  
match  
funny  
particularly  
begin  
going  
spot  
never  
army  
place  
tell  
win  
right  
ladder  
Culp  
little  
job  
side  
day  
playing  
joke  
every  
whole  
bit  
action  
much  
make  
look  
shot  
acting  
cast  
turn  
know  
give  
real  
without  
course  
remember  
friend  
life  
series  
black



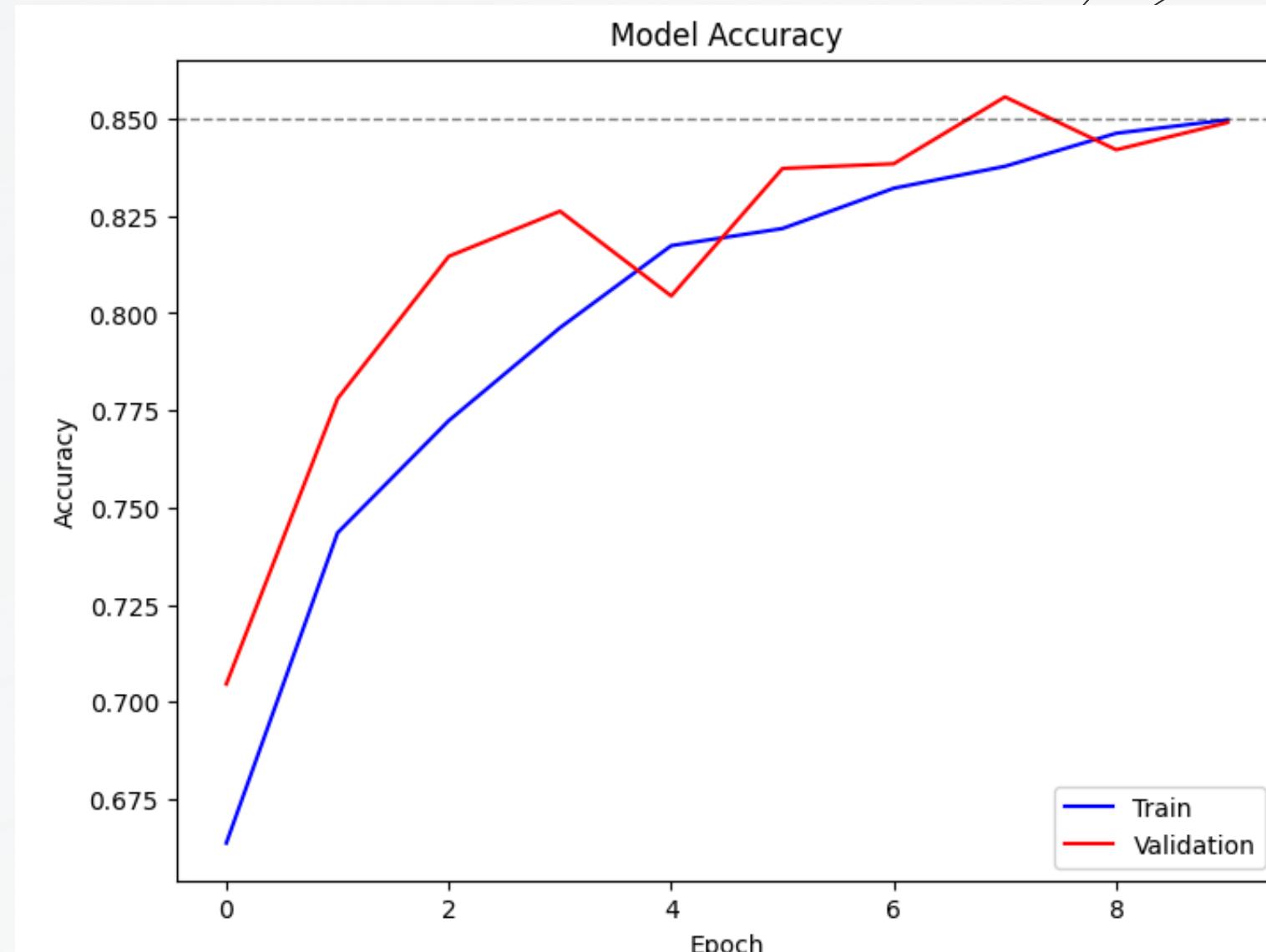
# MODELING AND ANALYSIS

- ◆ **DL approach:** Recurrent Neural Networks
- ◆ Evaluated models using validation accuracy with an 80-20 training-validation split
- ◆ Used combinations of different model architectures, hyperparameters, and NLP hyperparameters to try to tune model performance

# NO DROPOUT



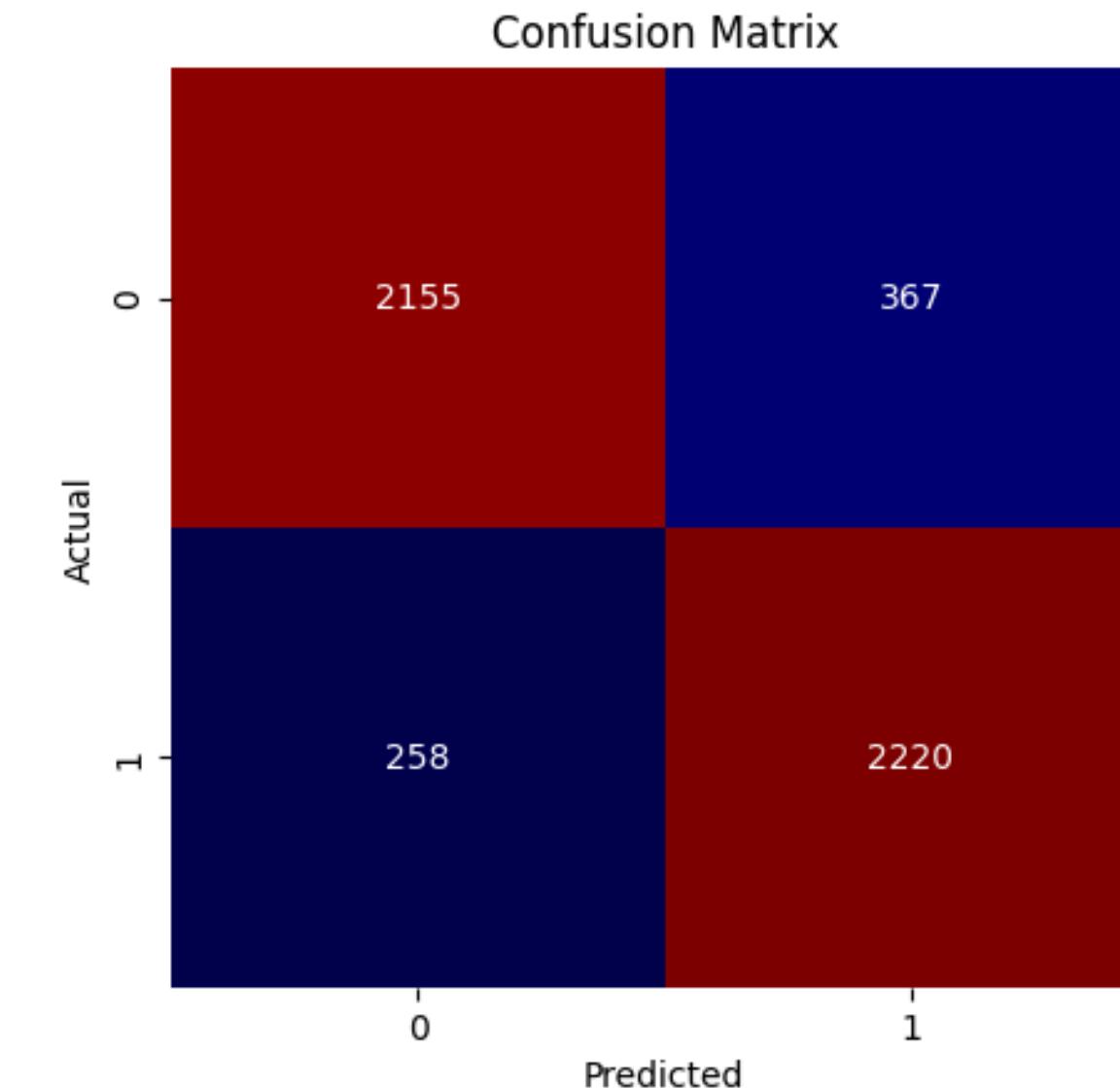
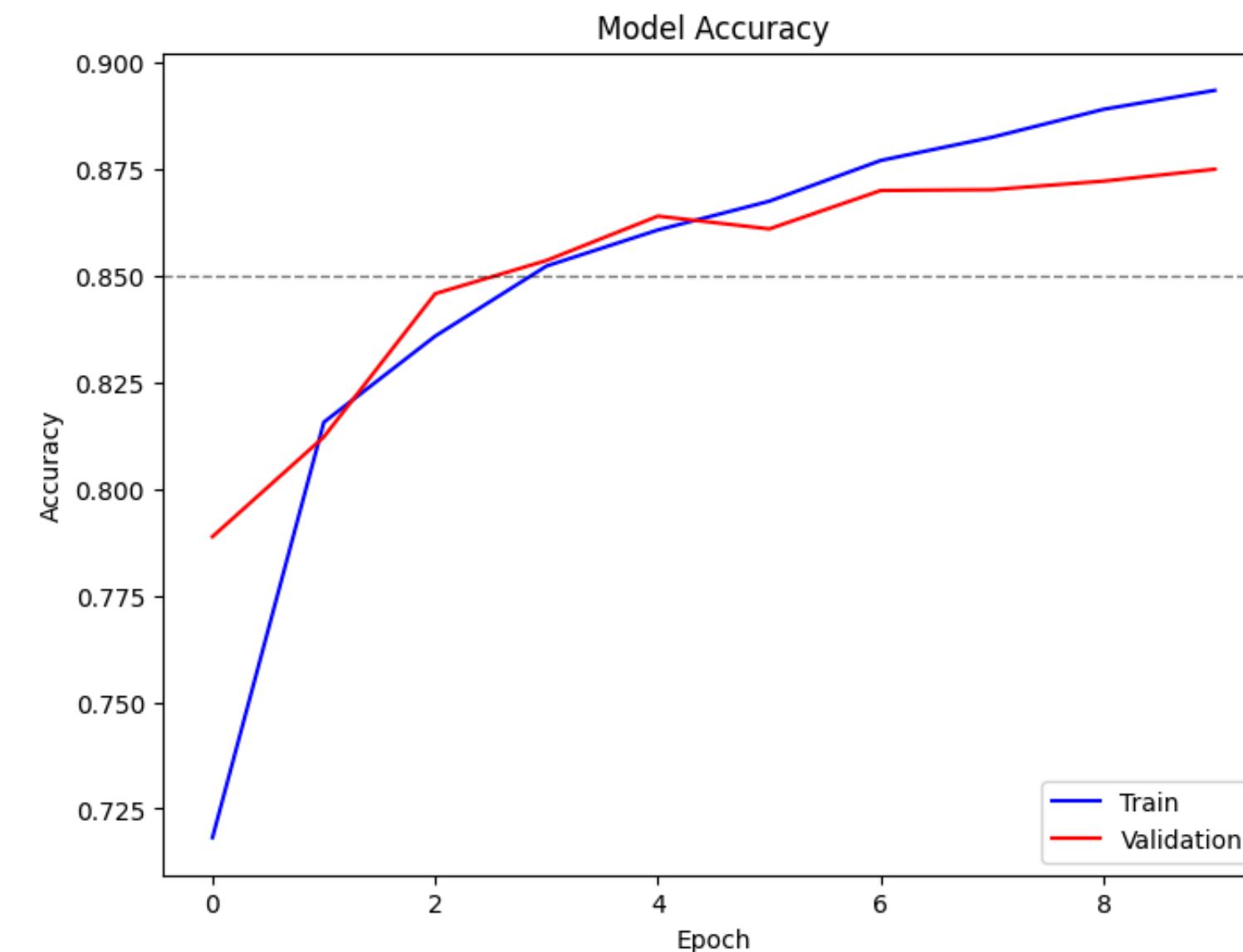
# DROPOUT=0.2



# MODEL ARCHITECTURES

Model	Architecture or Other Adjustments	Average Time/Epoch	Training Acc	Validation Acc
1	Embedding, Bi-LSTM, Sigmoid	4 seconds	89.51%	84.92%
2	Added dropout and recurrent dropout = 0.2	52 seconds	84.97%	84.9%
3	Added second Bi-LSTM layer	134 seconds	78.97%	80.5%
4	Added GlobalMaxPool1D layer	62 seconds	89.37%	86.3%
5	Max review length 200 -> 400	125 seconds	89.91%	<b>87.5%</b>
6	Limit tokenizer to 10,000 words	62 seconds	89.28%	86.38%

# MODEL 5 - 87.5%



# CONCLUSION

- ◆ GloVe and Bi-LSTM worked decently well for sentiment analysis of movie reviews with a best validation accuracy of 87.5%
- ◆ **Future work:** Explore embedding techniques, try other deep learning architectures, compare to broader supervised learning models