Multimodal IMDB Analysis with Keras (CNN and LSTM)

Juan Manuel Gonzalez Rincon

ID 23031523

Introduction: The present project aims to design and implement two classification models based on Neural Networks a CNN and an LSTM to classify movies into different genres (a multilabel problem) based on their posters (in the case of CNN) and their overviews (in the case of LSTM). The results are analyzed in this report to compare the performance of both models.

Models Selected

CNN: Used for image classification, the CNN is based on convolutional layers that extract important features by applying matrix filters, pooling layers that reduce the dimensionality or size of the image data through specific functions, and dropout layers that mitigate overfitting by deactivating certain neurons during training. Finally, classification layers are added to address the multilabel nature of the problem, requiring an output value between 0 and 1 as a probability for each class. A sigmoid activation function was used for the output. This is a deep network consisting of 21 layers.

LSTM: Designed for natural language processing (NLP) classification in a multilabel problem, the LSTM is based on a Recurrent Neural Network (RNN) but incorporates mechanisms to control the vanishing gradient problem during training. The last layer uses a ReLU activation function to improve learning, and the final layer is activated with a sigmoid function to output probabilities for each class.

Preprocessing

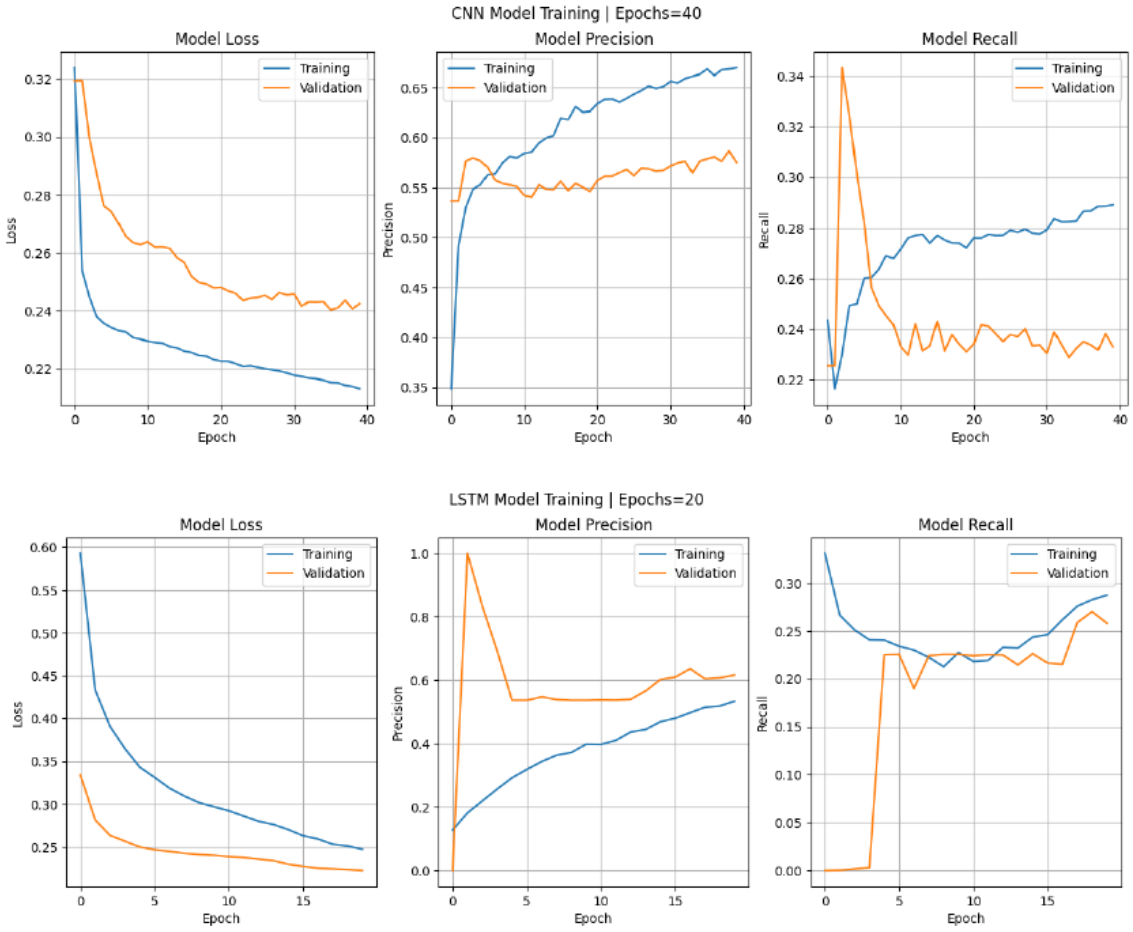
CNN: The data was first split into two groups for training and validation, with 20% reserved for validation. For each group, the data was loaded into a TensorFlow dataset class. Each dataset was then split into batches of 64 samples, vectorized, and standardized to ensure uniform image size, followed by conversion to floating-point values. The transformations were cached to save time, and prefetching was employed to prepare data in the background during training.

LSTM: Similarly, the data was split into training and validation groups with the same 20% validation split. The data was loaded into a TensorFlow dataset class, divided into batches of 64 samples, and preprocessed to save resources during training. A vocabulary of 10,000 words was created based on the overviews, which was used to encode the text into numeric representations.

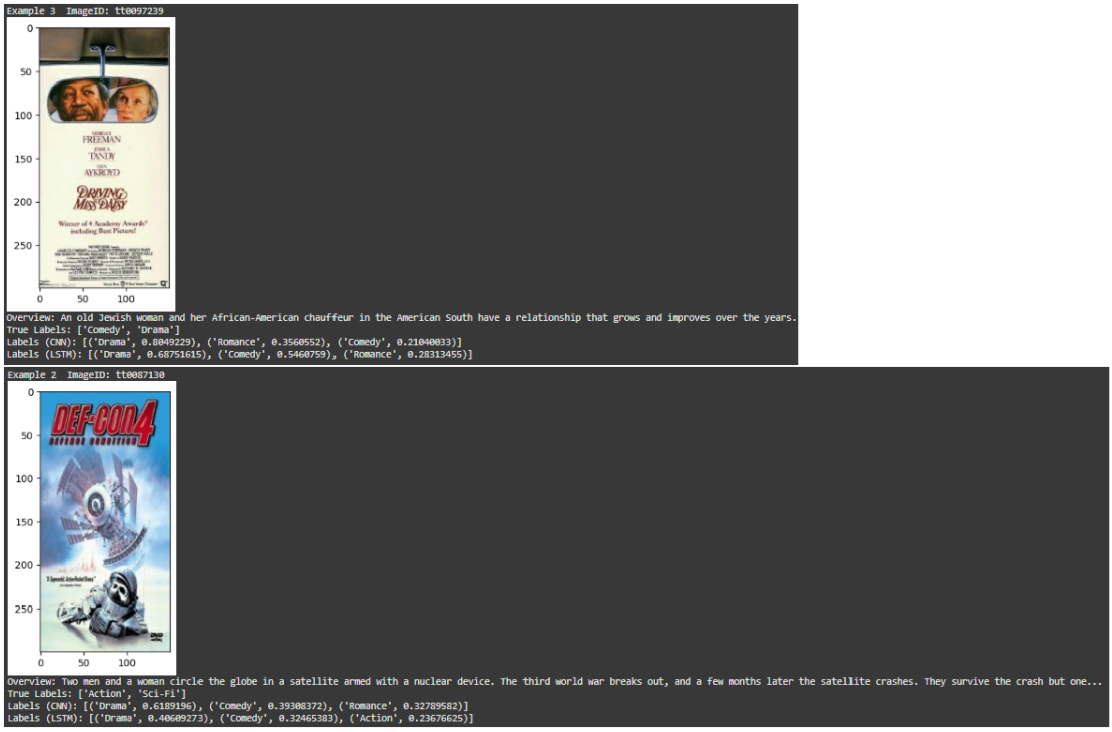
Model Configuration: The models were compiled using the “Adam” optimizer, which effectively adapts the learning rate. The loss function used was BinaryCrossentropy, suitable for the multilabel classification problem with probabilistic outputs. The models were evaluated using **precision**, which measures the proportion of positive predictions that are correct, and **recall**, which measures the proportion of actual positive cases correctly classified.

Training: During training, model checkpoints were created to save the weights of the models with the lowest loss values across epochs, ensuring the best model configurations were retained.

Results: For both models, the training loss improved steadily until the final epoch, with the CNN achieving a slightly better training loss (0.21) compared to the LSTM (0.24). However, the LSTM outperformed the CNN on validation loss (0.22 vs. 0.24). Precision values were very similar: CNN (0.57) and LSTM (0.61), slightly better than the 50% threshold of random guessing per class. However, recall values were relatively low, indicating the models struggled to classify certain classes. Recall was 0.23 for CNN and 0.26 for LSTM, reflecting a significant number of undetected classes.



Analyzing the random examples, when drama is part of the true labels, it has the highest probability in both models, in the case of the IDtt0097239 with probabilities CNN (0.80) and LSTM (0.68), in this case just the LSTM got the correct prediction of comedy (0.54). But even when drama is not part of the true label, both models predict drama probability as the higher as with the IDtt0087130 with CNN (0.62) and LSTM (0.41) and although they don't pass the 0.5 threshold, comedy is also the second higher CNN (0.4) and LSTM (0.32) which is a misclassification of this class and lack of classification of the true classes: action and science fiction.



It is evident that the samples are not balanced during training. There is a very high concentration of the *drama* (22.6%) and *comedy* (18.0%) classes, which are much higher than the third most common class, *crime* (7.4%). For the *drama* class, the precision is very similar to the overall model precision for LSTM (0.62) and CNN (0.56). However, the CNN model tends to classify *drama* in 90% of the cases, and a similar pattern occurs with the *comedy* label, where it is predicted in 92% of cases. This affects the recall, particularly when misclassifications occur, and is largely due to the small sample size of less common classes.

The model prioritizes predictions of the most frequent classes because they have a greater impact on the global loss function. This issue highlights an overfitting problem for the *drama* and *comedy* classes compared to the less common ones. A possible technique to address this problem includes implementing data augmentation or subsampling to balance the class distribution. Alternatively, using a different loss function that assigns greater weight to less common classes could also mitigate this issue.