Multimodal Genre Classification with CNN and LSTM Performance Evaluation

Name: Vishakanan Sivarajah || Student ID: 23090811

Introduction

Film genre classification is a key task in computer vision and natural language processing, used in applications such as content recommendation and media organization. This task is challenging due to the multi label nature of films many belong to more than one genre and the variability in visual and textual features. In this project, we use Convolutional Neural Networks (CNNs) to classify genres based on film posters and LSTM networks to process textual overviews. CNNs extract visual patterns from images, while LSTMs capture the context of text sequences. The goal is to build and evaluate both models using a TensorFlow pipeline and critically analyze their performance in predicting film genres.

Data Processing

The project uses a dataset containing film posters and corresponding textual overviews, each labeled with one or more genres. Several key preprocessing steps were undertaken:

Image Parsing: A custom parse_image function was used to load and decode image files. Each poster was assigned a label based on a predefined dictionary that mapped film genres to numerical values.
Dataset Splitting: The dataset was divided into training and validation subsets to train the models and evaluate their generalization on unseen data.
Image Preprocessing: A function, img_process, was implemented to convert image data to float32 format and resize images to 64x64 pixels. This ensures uniform input dimensions for CNN and improves computational efficiency.
Text Tokenization: For textual overviews, the Text Vectorization layer was used to tokenize the text and build a vocabulary using the encoder. adapt() method. This prepares the text for input into the LSTM-based model.
TF Data Pipeline: TensorFlow's tf.data API was utilized to build an efficient data pipeline with caching, shuffling, batching, and prefetching, enhancing training performance.

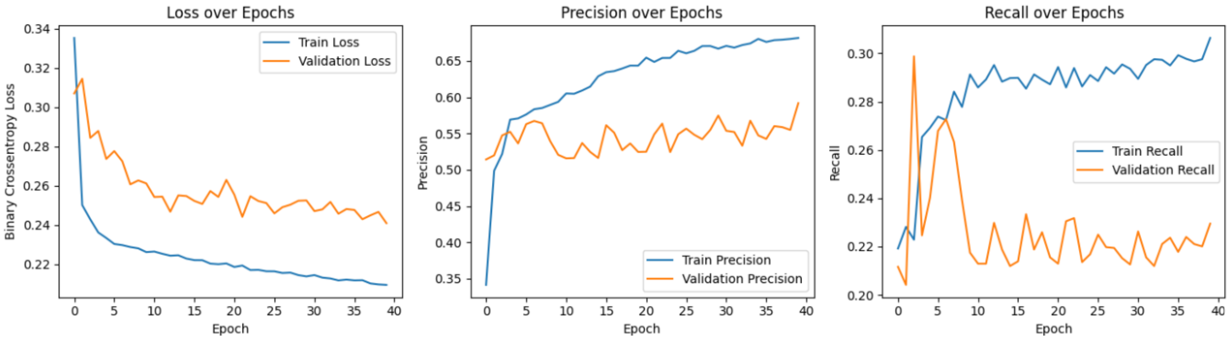
Model Architecture

The CNN model for film posters begins with an input layer that accepts 64x64 pixel preprocessed images. Convolutional layers extract features using filters, with each layer capturing increasingly complex patterns. ReLU activation introduces nonlinearity, enhancing feature learning. Pooling layers reduce spatial dimensions, lowering computational cost and minimizing overfitting. Fully connected layers combine extracted features, and the output layer uses a softmax activation function for multi-class classification.

The LSTM model for film overviews takes tokenized and padded word sequences. An embedding layer transforms word indices into dense vectors, capturing semantic relationships. The LSTM layer learns sequential dependencies and context. A dropout layer prevents overfitting by randomly deactivating units during training. The fully connected layer integrates learned features, and the output layer uses a sigmoid activation function to predict genres from 19 possible labels.

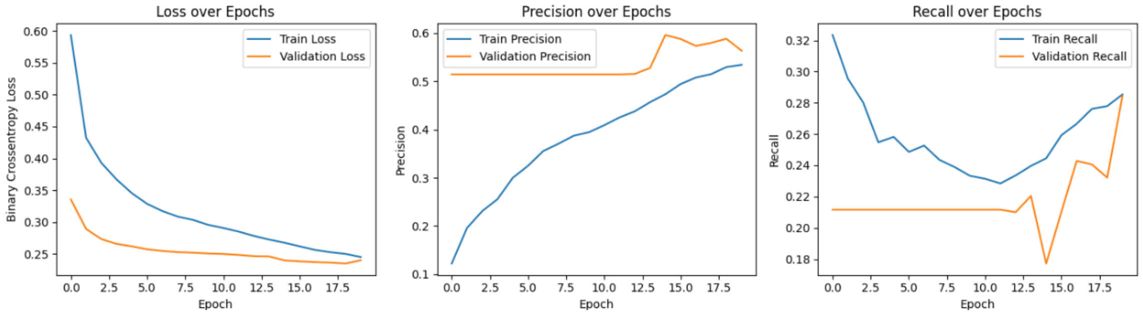
Model Evaluation & Performance

CNN model



The CNN model showed a steady decrease in training and validation loss, indicating effective learning. Training precision improved consistently, reaching above 0.67, while validation precision remained stable around 0.55. However, recall metrics revealed a gap, with training recall increasing slightly but validation recall staying low around 0.22. This suggests the model made accurate predictions but missed some true positives, indicating high precision but lower recall performance.

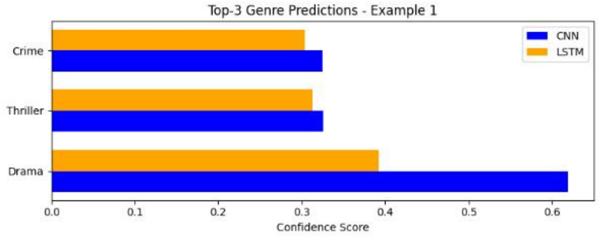
LSTM model

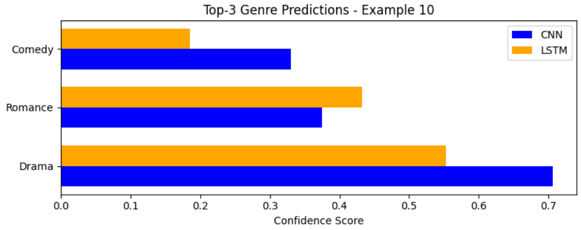


The LSTM model showed a steady decrease in training loss, while validation loss remained slightly higher and fluctuated mildly after the initial epochs. Training precision improved consistently, reaching around 0.67, whereas validation precision hovered around 0.55 with minor variations. Training recall gradually increased, but validation recall remained low and unstable. This indicates the model performed well on the training set but had challenges generalizing to unseen data, particularly in terms of recall.

Finally, The CNN and LSTM models both achieved good training precision (0.67), but struggled with validation recall (0.22), indicating high accuracy yet poor generalization in identifying all relevant genres. CNN showed slightly more stable learning, while LSTM faced mild fluctuations in validation loss, suggesting CNN may generalize slightly better.

Prediction sample Analysis





Overview: Three blue-collar acquaintances come across millions of dollars in lost cash and make a plan to keep their find from the authorities, but it isn't long before complications and mistrust weave their way into the plan.

Overview: A champion high school cheerleading squad discovers its previous captain stole all their best routines from an inner-city school and must scramble to compete at this year's championships.

Example	True Genres	CNN Predictions	LSTM Predictions	CNN Missed	LSTM Missed
1	Crime, Drama, Thriller	Drama, Thriller, Crime	Drama, Action, Comedy	None	Crime, Thriller
10	Comedy, Sport	Drama, Romance, Comedy	Drama, Comedy, Romance	Sport	Sport

In Example 1, the CNN model accurately identified all true genres Crime, Drama, and Thriller demonstrating its effectiveness in capturing visual cues from the poster, while the LSTM model only detected Drama and misclassified the rest. In Example 10, both models correctly predicted Comedy, but missed the Sport genre. The CNN leaned towards Drama and Romance, while the LSTM predicted Drama and Romance with similar confidence, again missing Sport. These examples reflect the CNN's overall stronger genre recognition capabilities. Across all 10 tested samples, the CNN consistently aligned better with ground truth genres and showed higher confidence scores, indicating greater reliability in multi-label genre classification compared to the LSTM.

Conclusion

In this assignment, I developed and evaluated two deep learning models a CNN for movie poster classification and an LSTM for movie overview classification to perform multi-label genre prediction. Each model was tested on the same dataset to compare performance across visual and textual inputs. The CNN model outperformed the LSTM, showing higher accuracy and more confident predictions, especially in identifying the correct top-3 genres. The LSTM occasionally misclassified overlapping genres. Example evaluations supported these findings, with the CNN correctly matching more true genres than the LSTM. Overall, the project highlights the effectiveness of CNNs in genre prediction tasks using visual data. Future improvements could include combining both models into a multimodal approach for better performance and generalization.