Overview and Poster

BE BOUNDLESS

DATA CLEANING

Because text and visual learning models can have a longer run time between scraping, processing, etc., we decided to aggressively cut down the number of data points.

- **Filtering out all rows with < 10 million in revenue.**
- **Historical average revenue per movie was 12 million. We decided a slightly lower bar of 10 million would have more data points.**
- **Filtered out rows that did not have a valid movie poster.**

MODEL APPROACH

Multi-input deep learning model predicting gross based on the movie summary and poster image.

> **Text: text cleaning, stop words, lemmatization, tokenization, vectorization, embedding.**
> **Images: YOLOv8, contrast, RGB, GLCM, Canny, anomaly.**
> **Text and images are handled separately then the results are combined.**

MODEL APPROACH CONT'D

- Text: Embedding layer → Bidirectional LSTM → Dense Layer
- Images: Dense layer
- Combined: Concatenated single feature vector → Dense layer → Dense layers w/ ReLU → Dense layer
- Split: 80%-20% training/test
- Loss function: MSE
- Optimizer: Adam
- Regularization: Dropout layers, Early stopping

WHY SUMMARY (OVERVIEW)?

Hope to analyze how different writing styles in movie summaries impact viewership and subsequently, gross.

> **A certain theme → More engagement → Higher audience interest.**

> **Certain word combinations → More viewers → Higher box office gross.**

TEXT PRE-PROCESSING

- Convert text to lowercase, remove special characters and numbers.
- Split sentences into words, remove stopwords and apply lemmatization for consistency.

TOKENIZATION

- Use TensorFlow/Keras Tokenizer (10000-word limit with <OOV> token).
- Maps words to numbers then convert text to numerical sequences. Pad them to a fixed length of 300 words.

QUICK EXAMPLE

- {'the': 1, 'love': 2, 'story': 3, 'war': 4, 'hero': 5, '<OOV>': 6}
- text = "A love story about a hero"
- [[2, 3, 5]]
- "A love story":[2, 3] // "A war hero":[4, 5] // "An epic adventure with a hero":[7, 8, 9, 5]
- [0, 0, 2, 3] // [0, 0, 4, 5] // [7, 8, 9, 5]

WHAT IS A LSTM?

- Long Short-Term Memory networks are a special type of RNN
- LSTMs Capture long-range dependencies
- Three gates: forget, input, output

BIDIRECTIONAL LSTM EXAMPLE

- "The hero fights bravely in the battle."
- "The battle was bravely fought by the hero."
- Both have the same meaning, but the word "battle" appears earlier in one case and later in another.

TEXT PROCESSING

- Input: tokenized and padded sequences (max length 300)
- Embedding: convert words into vectors (128 dimensions)
- BiLSTM 1: Extracts contextual meaning from sequences.
- Dropout Layer 1
- BiLSTM 2: Further refines sequence understanding.
- Dropout Layer 2
- Dense Layer: Extracts higher-level text features.

EMBEDDING EXAMPLE

- "Apple" might turn into [0.2, -0.3, 0.8, ...]
- "Banana" might turn into [0.25, -0.28, 0.75, ...]

WHY IMAGES (POSTER_PATH)?

Investigate whether the number of faces and exaggerated visual characteristics correlate with higher revenue.

- **More faces on a poster → Higher chance of a popular actor or a familiar actor for the audience.**
- **Eye-catching design elements → More attention → Higher viewer engagement.**

COUNTING PEOPLE: YOLOv8

Object Detection Using YOLOv8

- **YOLOv8 detects objects in the image, but we only count people (person class).**
- **People are counted if their confidence score > 0.5 for accuracy.**


person 0.82
person 0.77
person 0.74
person 0.79
person 0.65
person 0.48
person 0.60
car 0.89
car 0.87
FAST
X

EVERYDAY IMAGE INFORMATION

Contrast and R, G, B channels

- The image is converted to grayscale.
- The standard deviation of grayscale intensities is computed as a measure of contrast.
- The mean intensity of each color channel (R, G, B) is extracted and recorded.

High Contrast

Low Contrast

Color Psychology

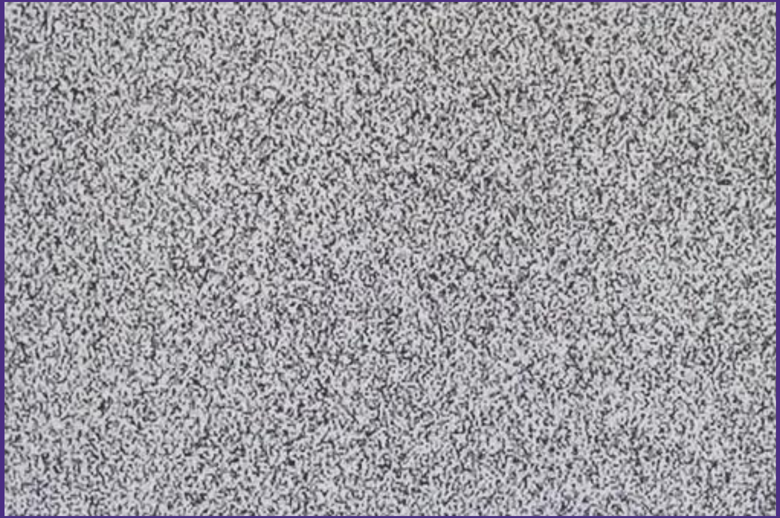
HOMOGENEITY: GLCM

Gray-Level Co-occurrence Matrix

- Analyze the texture of an image by evaluating how pixel intensity (ibrightness value of a pixel) is distributed in relation to one another.
- The grayscale image is resized to 64x64 and quantized (reduced to fewer intensity levels) to 16.
- Computer matrix for four different directions (0°, 45°, 90°, 135°) with a distance of 1. Extract “homogeneity”.

High Homogeneity

Low Homogeneity



High Homogeneity

Low Homogeneity

Grayscale Image (Resized to 64x64)

Quantized Image (8 Levels Demo - 16 Levels Application)

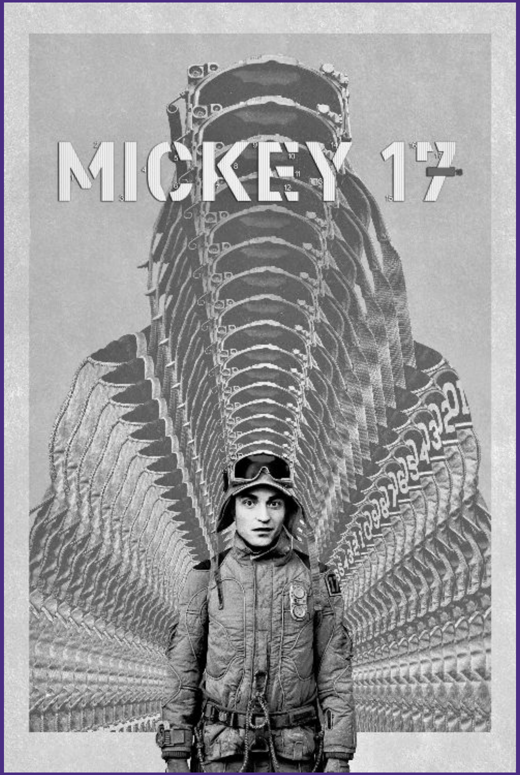
EDGE DETECTION: CANNY ALG.

Canny algorithm for edge detection

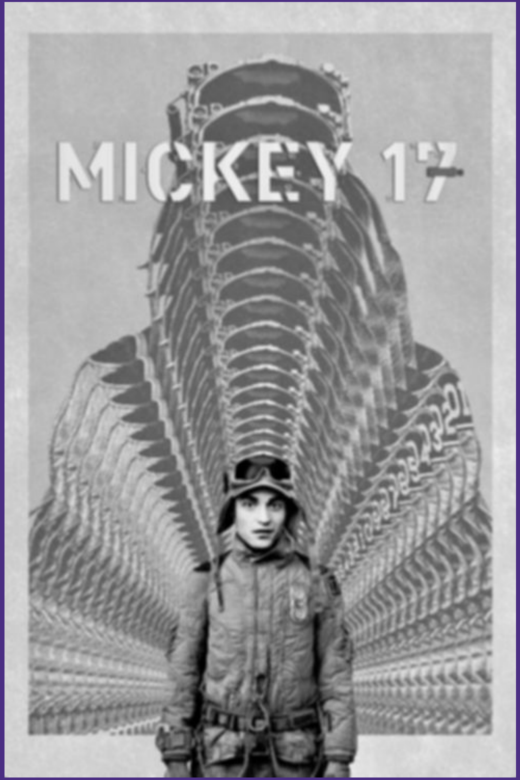
> Convert to grayscale.
> Apply Gaussian Blur (removes noise, keep significant edges).
> Find intensity gradients (detects where brightness changes sharply).
> Apply non-maximum suppression (removes weak edges).
> Apply hysteresis thresholding (keeps only the strongest edges).

Case Study

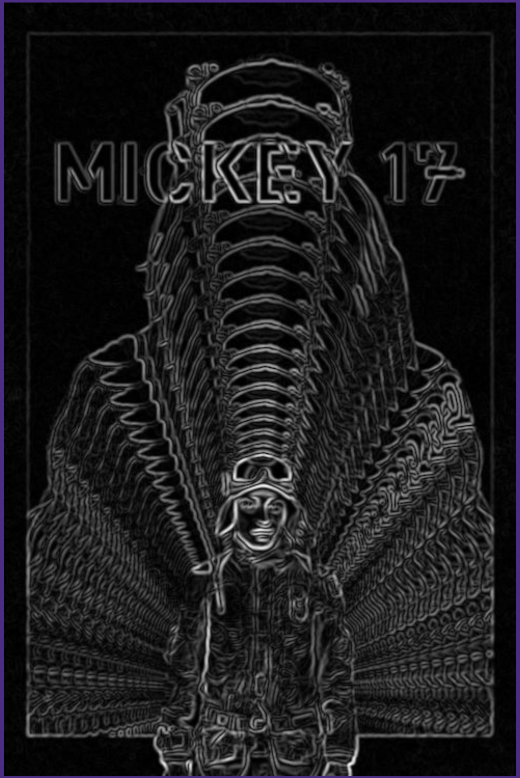
Grayscale



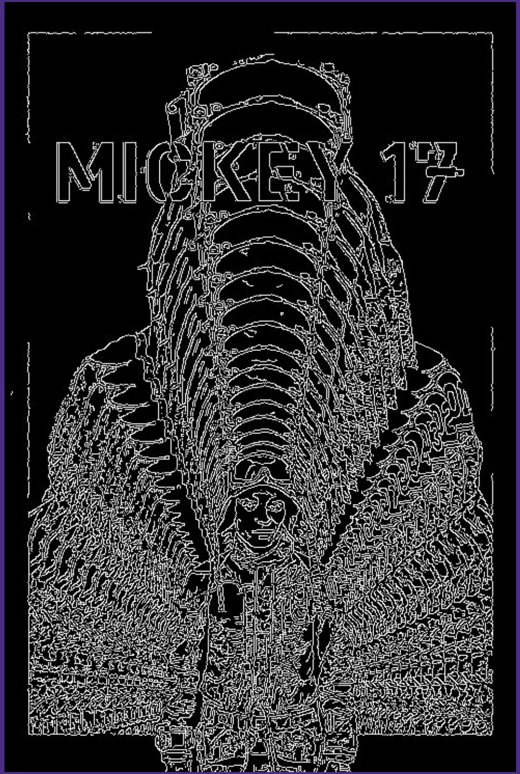
Gaussian Blur



Sobel Operator



Final Result



FINDING "ANOMALIES"

Anomaly detection with contrast and "randomness"

- Higher contrast values mean more variation in brightness.
- High entropy suggests a highly detailed, complex image, while low entropy means a smooth, simple image.
- Both values are normalized to a 0-1 scale and averaged to compute an anomaly score.

BACK TO THE MODEL

All the data we collected off of the images represent the data used for the “numerical” branch of the model.

> **Normalize all numerical data.**
> **Layer 1 (32 neurons), layer 2 (16 neurons), layer 3 (8 neurons)**
> **All have ReLU**
> **Numerical features: people count, contrast, avg. R, avg. B, avg. G, texture, edge detection, anomaly score**

COMBINE THE TWO BRANCHES

Now we have to merge everything together.

> **Text data as 64 dimension vectors.**
> **Numerical data as 8 dimension vectors.**
> **Concatenate them 64+8=72 dimension vector.**
> **Dense layer → Dropout layer → Dense layer → Dropout Layer**
> **Output layer is a single neuron with no ReLU**

TRAINING

Finally after all that work we get to actually train the model.

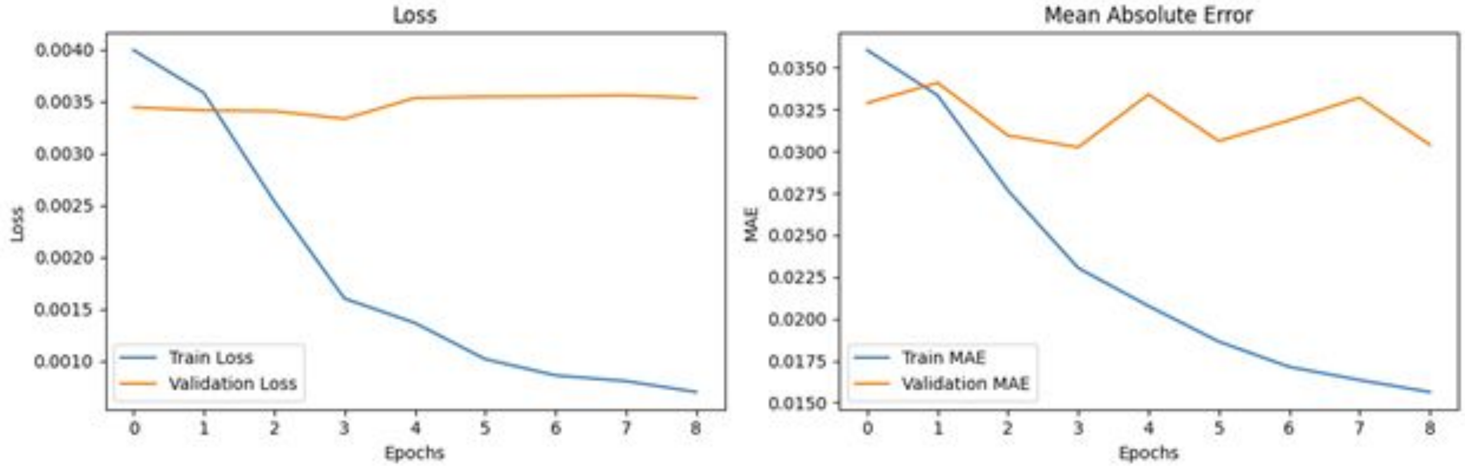
> **MSE for loss, Adam for optimization**

> **Stops training if validation loss doesn't improve for 5 epochs.**

> **Epochs: 100 // Batch size: 32**

> **Saves the best-performing model (checkpoints).**

MODEL WEAKNESS



UNIVERSITY *of* WASHINGTON

PREDICTION (Mickey 17 - 3/07/25)

Final prediction is $78,132,304.00
Current reported is $53,300,000.00

- **46.59% error**
- **More data points and longer training for accuracy.**
- **Add dense layers and/or adjust drop out values.**
- **Longer time spent training.**

Questions?

BE BOUNDLESS