Myelin Foundry Winter Internship (2019)

-Arjun Andra

Introduction

- At Myelin Foundry I worked on the OTT Service, Mayabazaar.
- Mayabazaar is an on-demand video streaming platform primarily focused on Telugu content delivery.
- My initial project was to aid in the general envisioning of the product.
- I then moved on to work on a specific area of the product.
- I chose to work on the Recommendation Engine.

Recommendation Engines

Basic Recommendation Engines

- Text-based meta data analysis
  - Grouping based on keywords

- Collaborative Filtering
  - User-User Filtering
  - Item-Item Filtering

- Time-based Recommendation Engines
  - Removes low impact parameters for characteristics that more drastically impact users’ moods.

Recommendation Engines

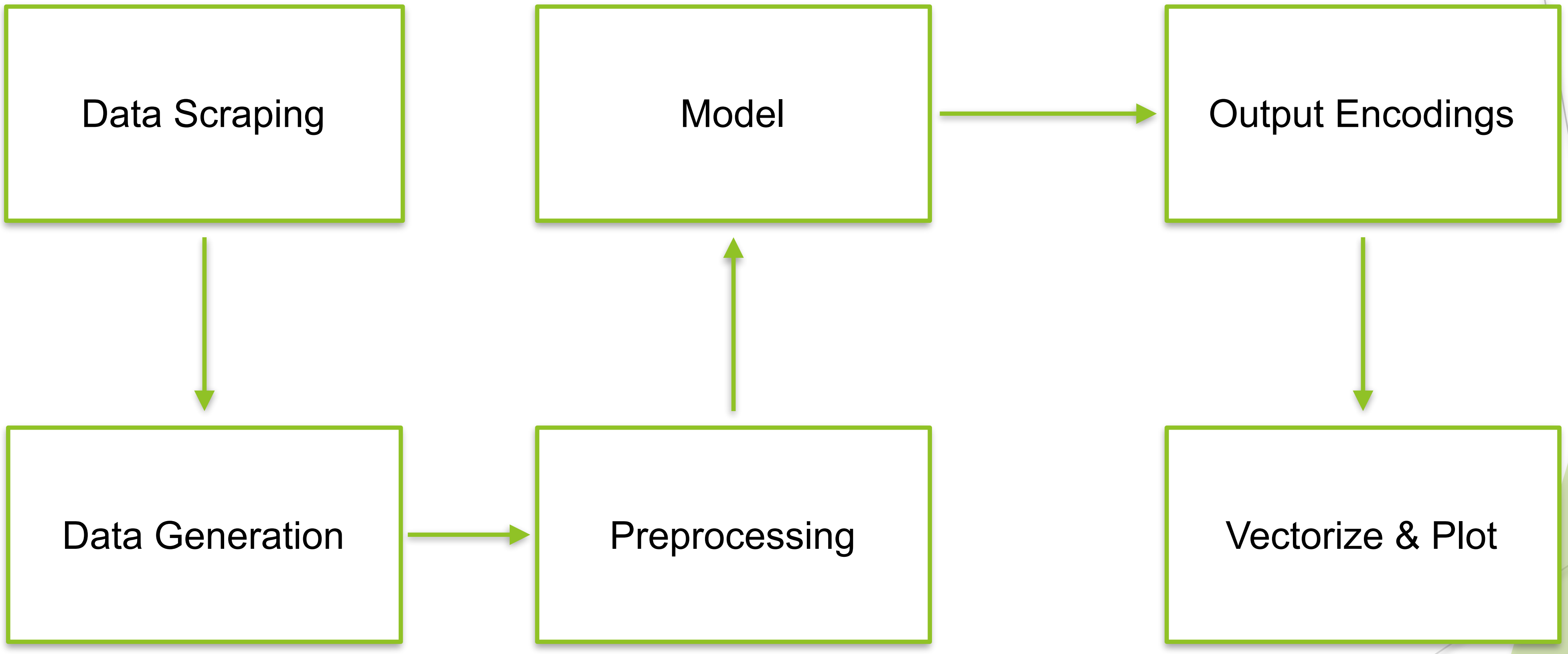
Advanced Recommendation Engines

- Visual Content Based Recommendation Engines
  - Poster object detection helps tag & cluster image (Illustration2Vec Model)

Objectives

- To make movie recommendations based on trailers of movies.
- To capture the spatio-temporal information present within clips of trailers, and use this information to cluster & recommend movies.

Pipeline



Data Scraping

Data Generation

Preprocessing

Model

Output Encodings

Vectorize & Plot

Data Scraping

- Iterating through the imdb movie dataset & downloading .mp4 files of trailers (using BeautifulSoup)
- Iterating through the sports 1m dataset & downloading .mp4 / .mkv files of youtube videos (using youtube-DL)

Data Generation

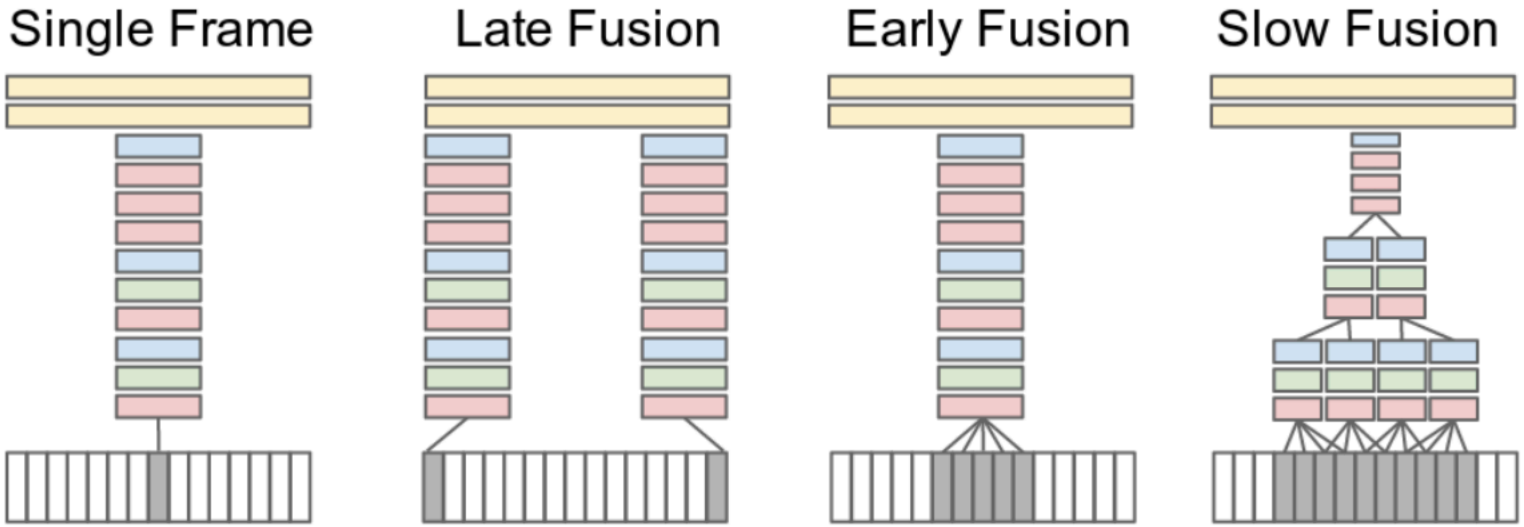
- Splitting of the .mp4 / .mkv files into K Iframes (based on the size of video) for easy inputting into the model

Preprocessing

Applied consistently to all frames that are a part of the same clip:

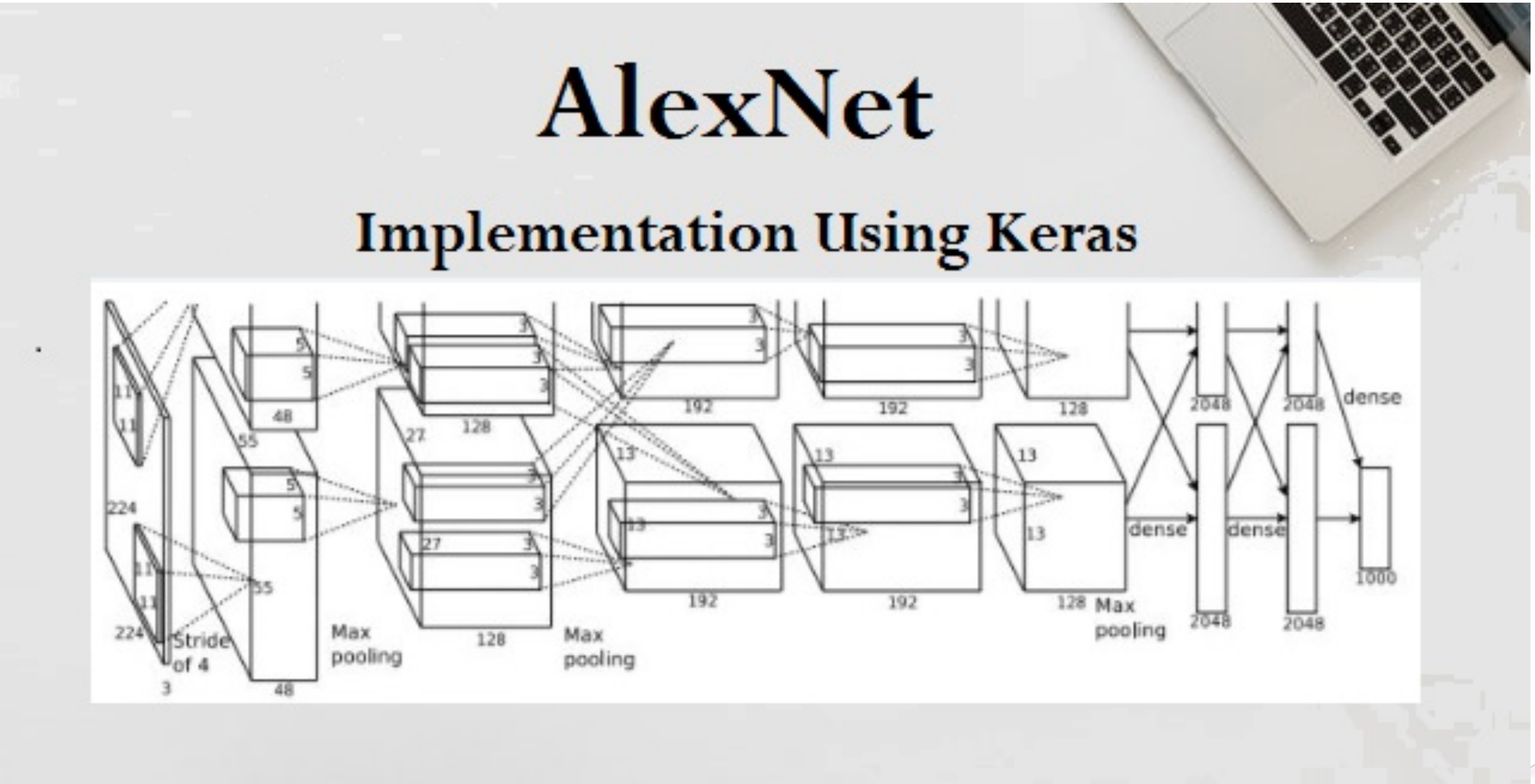
- Crop to center region of frame.
- Resize to 200 x 200 pixels.
- Randomly sample a 170 x 170 region.
- Randomly flip the images horizontally with 50 % probability

Model



- The Single Frame model processes each frame individually and takes an average prediction amongst all frames.
- The Late Fusion model utilises the predictions of 2 single-frame models on the first & last clip and aggregates its predictions.
- The Early Fusion model takes a temporal input (multiple contiguous frames at once) and processes them all at once.
- The Slow Fusion model takes 4 such contiguous inputs & aggregates between all of them.

Model


AlexNet
Implementation Using Keras
224
224
11
11
3
Stride of 4
55
55
48
48
5
5
Max pooling
27
27
128
128
3
Max pooling
13
13
192
192
192
192
128
128
Max pooling
dense
dense
dense
2048
2048
2048
2048
1000

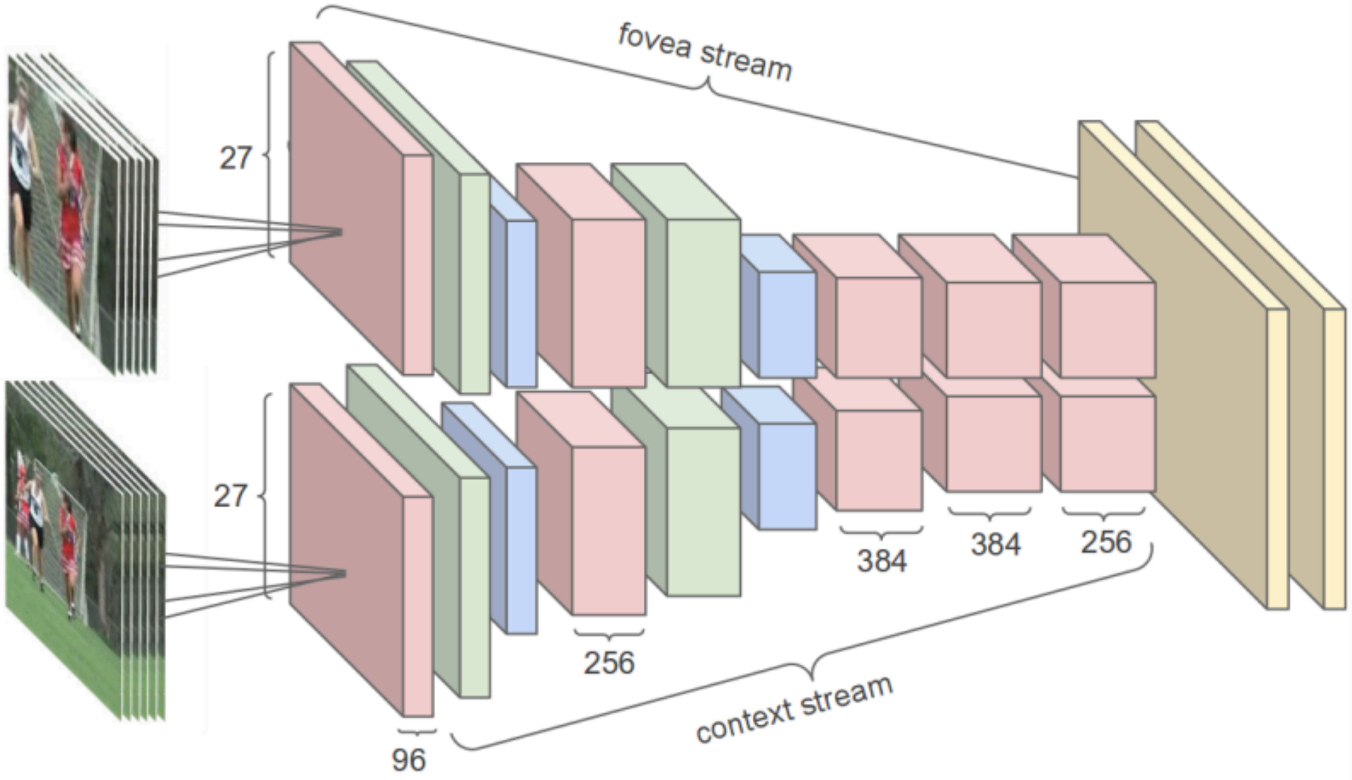
Model

Modified Layers

- First Convolutional Layer (96 Filters, 11 x 11 Spatial Size, 3 x 3 Input Stride)
- First Normalization Layer
- First Max Pooling Layer (Stride of 2 x 2)
- Second Convolutional Layer (256 Filters, 5 x 5 Spatial Size, 1 x 1 Input Stride)
- Second Normalization Layer
- Second Max Pooling Layer (Stride of 2 x 2)
- Third Convolutional Layer (384 Filters, 3 x 3 Spatial Size, 1 x 1 Input Stride)
- Fourth Convolutional Layer (384 Filters, 3 x 3 Spatial Size, 1 x 1 Input Stride)
- Fifth Convolutional Layer (256 Filters, 3 x 3 Spatial Size, 1 x 1 Input Stride)
- Dense Layer (4096 nodes)
- Dense Layer (4096 nodes)

Model Enhancements

Multiresolution CNN



Model Enhancements

Multiresolution CNN

The proposed multi resolution architecture aims to strike a compromise by having two separate streams of processing over two spatial resolutions.

It is broken up into a context & fovea stream.

Context Stream:

- The context stream receives the downsampled frames at half the original spatial resolution (89 x 89)

Fovea Stream:

- The fovea stream receives the center 89 × 89 region at the original resolution.

This approach is used to significantly reduce the training time for the model.

Present Status :

- Using the azure cloud VM to run the single frame model on scraped & preprocessed data from sports1m to understand the contribution of static appearance to the classification accuracy.

Future Path :

- Transform model into slow fusion model (that utilizes multi resolution CNNs).
- Train model on collected sports dataset.
- Transfer model onto trailer dataset.
- Retrieve & plots the encodings of trailers from hidden layer of model.

Learnings

Conceptual Learnings

- Multiresolution CNNs
- Time Information Fusion in CNNs

Implementational Learnings

- Data Scraping (using BeautifulSoup, youtube-dl, & Selenium)
- Preprocessing Videos (Iframe Conversion & Pixel Manipulation)
- Custom Model Construction (using TF & Keras)

References

- https://towardsdatascience.com/introduction-to-video-classification-6c6acbc57356
- https://www.bluepiit.com/blog/classifying-recommender-systems/
- https://github.com/gtoderici/sports-1m-dataset
- https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/42455.pdf