Movie Genre Classification Based On Its Poster using VGG16 and DenseNet169

A MAJOR II REPORT

SUBMITTED IN FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF DEGREE OF

MASTER OF TECHNOLOGY
IN
COMPUTER SCIENCE & ENGINEERING

Submitted By
Mohd Umar Faiz
2K19/CSE/13

Under the supervision of

Mr Nipun Bansal
(Assistant Professor)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College Of engineering)

Bawana Road, Delhi-110042

DECLARATION

I, Mohd Umar Faiz, Roll No. 2K19/CSE/13 student of M.Tech (Compter Science & Engineering), hereby declare that the Project Dissertation titled "**Movie Genre Classification Based On Its Poster using VGG16 and DenseNet169**" which is submitted by me to the Department of Computer Science & Engineering, Delhi Technological University, Delhi. Report of the Major II which is being submitted to Delhi Technological University Delhi, in fulfillment for the requirement of the award of degree of Master of Technology.

Place: DTU, Delhi

Mohd Umar Faiz

(2K19/CSE/13)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College Of engineering)

Bawana Road, Delhi-110042

CERTIFICATE

I, hereby certify that the Project Dissertation titled *"***Movie Genre Classification Based On Its Poster using VGG16 and DenseNet169"** which is submitted by Mohd Umar Faiz, Roll No. 2K19/CSE/13, Department of computer Science & Engineering, Delhi Technological University, Delhi in partial fulfillment for the requirement of the award of degree of Master of Technology (Computer Science and Engineering) is a record of a project work carried out by the student under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Nipun

Place: Delhi **(Mr. Nipun Bansal)**

Date: **SUPERVISOR**

Assistant Professor
Department of Computer Engineering
Delhi Technological University

ACKNOWLEDGMENT

I am extremely grateful to **Mr Nipun Bansal** Assistant Professor, Department of Computer Science Engineering, Delhi Technological University, Delhi for providing invaluable guidance and being a constant source of inspiration throughout my research. I will always be indebted to his for the extensive support and encouragement he provided.

I also convey my heartfelt gratitude to all the research scholars of the web Research Group at Delhi Technological University, for their valuable suggestions and helpful discussions throughout the course of this research work.

Mohd Umar Faiz

Roll-No-2K19/CSE/13

ABSTRACT

As the size of the entertainment world develops, the interest for approved movie databases is additionally developing. The class of a movie supplies data on its general substance and has various qualities. Hence , it ought to be very much arranged using the qualities of movies, without exclusions in the dataset and their posters.

A clever and stylish poster can be the pivot towards the acknowledgment of a movie. Humans have a very unique ability to understand the world around them in an instance. With regards to movies, we can take a glance at a poster, break down its tones, textures, surfaces, faces, appearances, objects, and so on and rapidly figure out what be the issue here or what does the poster represent. In this way, we humans can identify the genre of a movie by just taking a glance at the poster of the movie.

We proposed a model that does the exactly same like humans i.e., take the movie poster as its input and predict its genre from the poster of the movie using VGG16 .

The dataset that we used is taken from kaggle which contains the poster of the movies released on or before 2017 across 28 different genres.

We carried out modified renditions of two standard deep learning models for image classification : VGG-16, and DenseNet-169. In the evaluation, I show that the proposed strategy yields a great performance on various movie posters.

CONTENTS

LIST OF FIGURES

1. INTRODUCTION

1.1 Overview:

Humans have a very unique ability to understand the world around them in an instance. With regards to movies, we can take a glance at a poster, break down its tones, textures, surfaces, faces, appearances, objects, and so on and rapidly figure out what be the issue here or what does the poster represent. In this way, we humans can identify the genre of a movie or what the movie is about by just taking a glance at the poster of the movie.

In the film-production industry, the poster of a movie has an unimaginably significant job. Customarily, the poster has conveyed the primary overview of a movie with watchers - through shows at Cinemas, inside paper commercials, or as DVD fronts, or in the social medial sites. All the more as of late, with the development of advanced streaming stages, the part of the poster has expanded its importance. For viewers perusing most social networking sites, the visuals in the poster are what makes a specific movie recognizable inside the abundance of the content.

The size of the worldwide entertainment world is essentially developing each year, and the techniques by which it disseminates content have gone through considerable changes. Earlier, theater-arranged organizations have been the fundamental center, however, the development of web based real time features, for example, Netflix and other OTT platforms have extraordinarily changed how crowds watch through content. Therefore, the substance stage business has expanded and different genre-arranged data sets have become a fundamental component. Homegrown and global movie data sets, for example, the Internet Movie Database (IMDB), give an assortment of data identified with movies. Such information bases use the arrangement frameworks that partition movies into a few many genres.

Genre prediction has been a profoundly investigated subject in deep learning all things considered. The genre of a piece of workmanship encodes a lot of data about the piece inside a solitary word. It is a concise and viable approach to feature the highlights and dissimilarities between various works. When it comes to movies, the genre is regularly the main consideration for a watcher to choose the movie between various different choices.

1.2 Dataset

The data for this project incorporates two components. " The first is a set of movie posters and the other, a set of their genre labels. This dataset was produced using a text file obtained from Kaggle, which contains a database of movie information including movie titles, genres, IMDB scores, and URLs to images of movie posters. The original database contains 40109 movie titles." **[9]**

The files provided are:

- moviegenre.csv - contains 40109 movie names along with their genre and links for their posters.
- Poster images - The posters are scrapped from the web by using the link provided in the dataset.

(i) moviegenre.csv

The training dataset contains 40109 rows

The following are the features in the training data:

a. imdb_*id*: Unique ID associated with each movie and their posters

b. *imdb_link* : Contains the imdb link from where the poster should be scrapped

c. *Title* : Contains the title of the movie.

d. *imdb_score*: Contains the rating of the movie in IMDB out of 10

e. *Genre*: Contains the information about what genre the movie belongs to .

f. *Poster*: Gives the complete link of the poster from where it should be scrapped.

The following are the 28 genre for which we have to predict that the movie belongs to which genre. These are mostly self-explanatory.

1. *Animation*
2. *Adventure*
3. *Action*
4. *Comedy*
5. *Family*

6. Romance

7. Drama

8. Crime

9. Thriller

10. Fantasy

11. Horror

12. Biography

13. History

14. Mystery

15. Sci-fi

16. War

17. Sport

18. Music

19. Documentary

20. Musical

21. Western

22. Short

23. Film-noir

24. Talk-show

25. News

26. Adult

27. Reality TV

28. Game show

1.3 Minimum System Requirements

1.3.1 Hardware Requirements for Development:

- **Processor:** i5 or above
- **RAM :** twice the amount of your total GPU memory
- **GPU :**It depends how many GPUs you really need according you the kind of project you are working on. Four GPUs is the max that one will need. Anything over that is just for performance. Each GPU adds a certain overhead and your system has to decide which GPU to use for which task.
- **Storage :** SSD is a good option but not an absolute requirement.
- **Power Supply :** High efficiency one . For a 4 GPU system you will require 1400 to 1600 watts.
- **Cooling :** Important as it affects the performance of the machine. Heat affect a lot.

1.3.2 Software or Module Requirements for Development

- Google Colab
- Python3.6
- matplotlib
- Sklearn
- Tensorflow
- Pickle
- VGG16
- DenseNet 169

2. Problem Identification

2.1 Problem Statement:

Movie genre classification might be extensive or assorted dependent on the standards; in addition, there are numerous genres that are comparative, and one movie may have a place with a few of them, making exact characterization troublesome. Although a few Datasets utilize a blend of genre character and movie attributes to accomplish genre grouping, this strategy may make uncertainty with respect to genres and irregularity in the absolute number of genres. This technique is likewise tedious and is presented to the danger of abstract decisions.

To conquer these issues and perform genre classification productively, numerous past examinations used AI and Deep learning on how to endeavor programmed genre grouping, in view of different information, like movie posters.

2.2 Existing Approaches:

The most common approaches I have come across are as follows:

- “One such work is from Kos, Pobar and Mikec in 2014 where they used distance ranking, Naïve Bayes and RAKEL for multi-label genre classification. They were able to achieve fairly good validation accuracy with a small dataset of only 1500 movie posters across 6 genres.” **[6]**

- “Following the same, Kos, Pobar and Ipsic came back in 2015 with another different approach to the same problem. This time they used a larger dataset of 6000 posters across 18 genres. Then they converted the multi-labels to single-labels and ran predictions using k-NN, Naïve Bayes and RAKEL to classify posters into genres. The key breakthrough in this paper was the transformation of multi-labels where they created a continuous embedded space with all labels where similar genres were clubbed together.”**[7]**

- “Jeong A. Wi; Soojin Jang; Youngbin Kim in 2020 Proposed the use of the Gram layer with a Gram matrix for multi-label genre classification by poster. They used VGG16 and DenseNet169 for classification of posters. They achieved a validation accuracy of 51,71% using VGG16 in their proposed model and a validation accuracy of 54.88% using DenseNet169. ” **[2]**

2.3 Motivation:

The motivation Behind our project is to give a model that ensures that a film banner is expected to invigorate the right group. This is done by perceiving the genre information that the visuals in the banner pass on. An especially planned banner can give the class of a film to a human, who has no prior data on the film, at first. Henceforth, we expect that a film banner ought to have visual qualities that can be connected with different sorts by a significant learning estimation.

Considering this, we implemented a model that snaps the photo of a film poster as its information, takes it through a Convolutional Neural Network (CNN), and yields the assorted film genre that the poster falls inside. A specific model would be significant for the film-creation industry to overhaul the arrangement of their banners.

2.4 Objective of the project:

The objective of this project is to catch a portion of this cycle through the development of a Convolutional Neural Network (CNN) that trains on a large number of various movie posters with different genres as their marks and at last anticipate the genre(s) of an inconspicuous movie poster.

3. Methodology

- Let's start by a more generalist understanding and how the movie posters are distributed between the different genres.

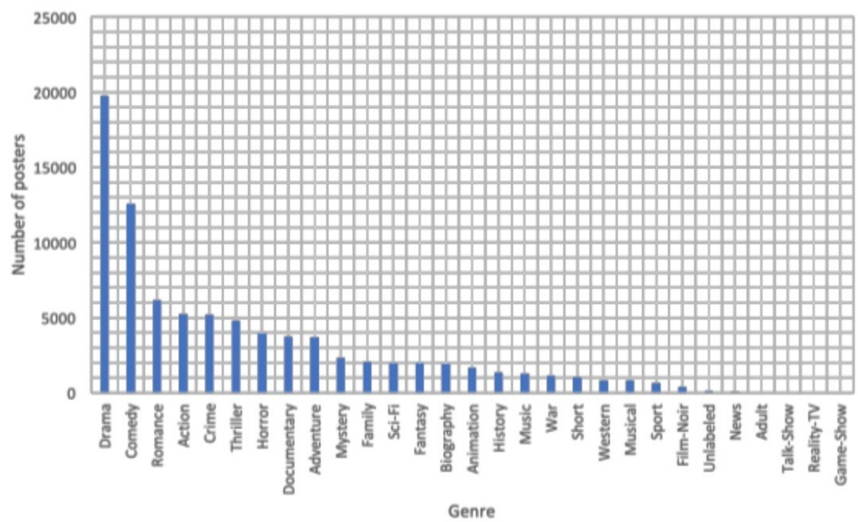


Fig 1: Movie poster distribution across different genre

Fig 1: The most common category in the movie genre is :
1 - Drama(41.8%)
2 - Comedy (33.2%)
3 - Romance(14.7%)
4 – Action(12.2%)
5 – Crime(11.9%)

In the above chart, we can see that the quantity of posters per genre are unbalanced – Drama and Comedy joined contribute for half of the dataset. To fix this, we take a gander at the genres that co-happen the most and pick ones that are illustrative of the bigger populace of the data.

To make our training from a balanced dataset we have to select those classes that give more information about the data from their posters and does not cause imbalance in the dataset.

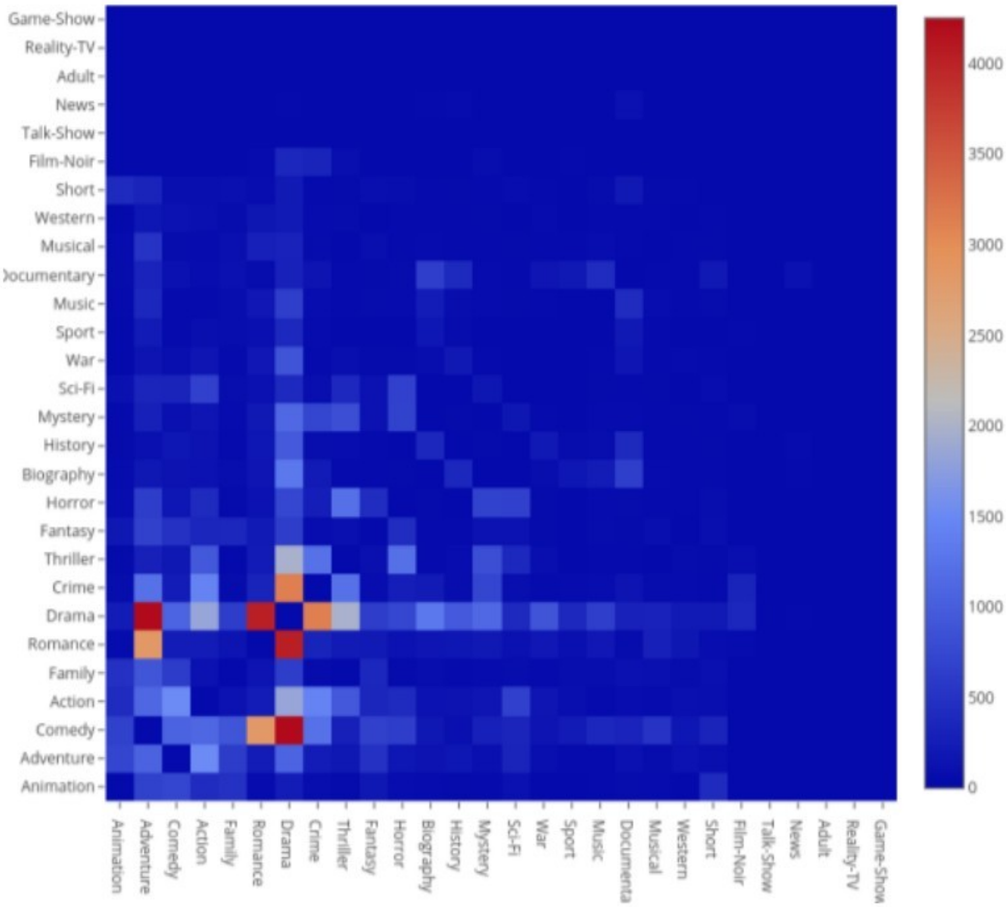


Fig 2 : Genre Co-occurrence heat map

In view of the genre co-event heat map in Fig 2, we will pick four key genres from various co-event levels that can be demonstrative of the bigger populace size. These genres are: Romance, Action, Horror and Documentary.

The model was prepared on movie posters from 1997-2017 in genres: Romance, Action, Horror, and Documentary.

Dataset Reconstruction:

"The dataset utilized for preparing the model comprises of two parts. The first is a bunch of movie posters and the other, a bunch of their genre names. This dataset was created utilizing CSV got from Kaggle, which contains a data set of movie data including movie titles, genres, IMDb scores, and URLs to pictures of movie posters. The first information base contains 40109 movie titles. Endless supply of titles that don't have any genres recorded and titles that have an absent or broken URL to the banner picture, we were left with 26429 titles. We Downloaded the poster pictures from the URLs for these and got the RGB pixel information for each."[9] The pixel information of a solitary banner was of size 268 x 182 x 3. Then the posters are resized from 268 x 182 x 3 to 256 x 256 x 3 .

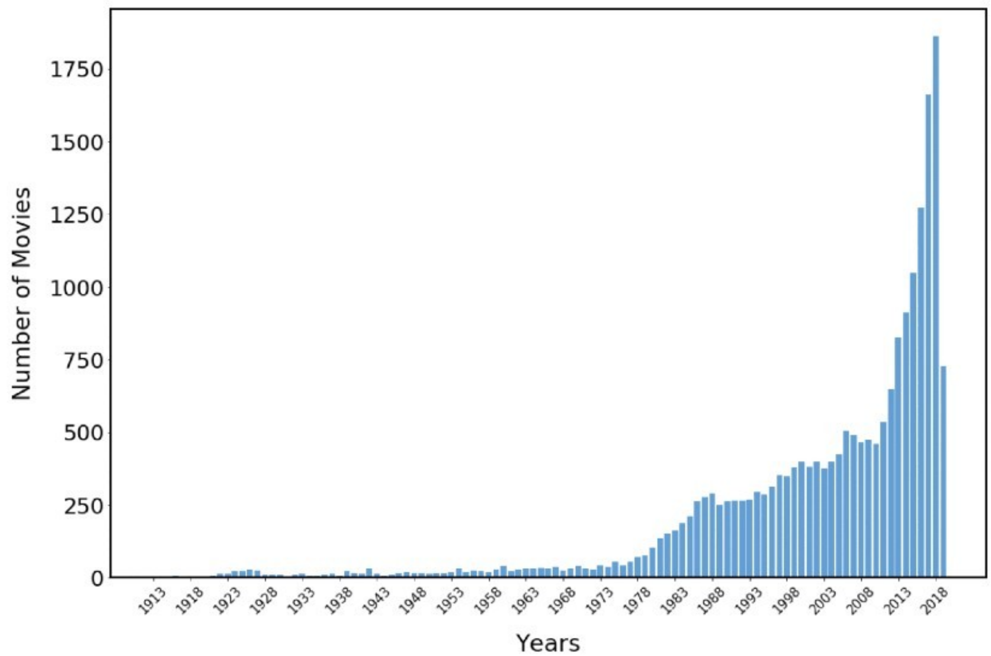


Fig 3 : Year wise distribution of movie posters in the dataset

As shown in Fig 3, The majority distribution of the movie posters in the dataset lies between 1990 to 2017, So in order to train the model we selected the movies lying between 1997 and 2017 .

We Proposed 2 models for classification of posters :

1. VGG16
2. DenseNet169

3.1 VGG16

3.1.1 Model Architecture:

VGG-16 is a Convolutional neural network that 16 layers deep.

"The input to cov1 layer is of fixed size 224 x 224 RGB image. The image is passed through a stack of convolutional (conv.) layers, where the filters were used with a very small receptive field: 3×3 (which is the smallest size to capture the notion of left/right, up/down, center). In one of the configurations, it also utilizes 1×1 convolution filters, which can be seen as a linear transformation of the input channels (followed by non-linearity). The convolution stride is fixed to 1 pixel; the spatial padding of conv. layer input is such that the spatial resolution is preserved after convolution, i.e. the padding is 1-pixel for 3×3 conv. layers. Spatial pooling is carried out by five max-pooling layers, which follow some of the conv. layers (not all the conv. layers are followed by max-pooling). Max-pooling is performed over a 2×2 pixel window, with stride 2." **[4]**

"Three Fully-Connected (FC) layers follow a stack of convolutional layers (which has a different depth in different architectures): the first two have 4096 channels each, the third performs 1000-way ILSVRC classification and thus contains 1000 channels (one for each class). The final layer is the soft-max layer. The configuration of the fully connected layers is the same in all networks.All hidden layers are equipped with the rectification (ReLU) non-linearity." **[4]**

We executed a standard VGG-16 design with some modifications. The last fully-connected layer of 1000 units was indeed supplanted by 3 consecutive fully-connected layers of 1024,128, and 7 units with ReLU, ReLU and sigmoid activation individually. The whole model comprises of 40,013,252 parameters, which were all trainable.

Fig 4:Model Architecture for modified VGG16

Training vs validation accuracy for the learning rate is then shown. The model is trained for 70 epochs .

3.1.2 Optimization

Optimizer are algorithms or strategies used to change the characteristics of your neural network, for example, loads and learning rate to lessen the losses.
We used Adam optimizer for this project.

“Adam is a replacement optimization algorithm for stochastic gradient descent for training deep learning models. Adam combines the best properties of the AdaGrad and RMSProp algorithms to provide an optimization algorithm that can handle sparse gradients on noisy problems.” **[11]**

Adam is defined by :

$$
\begin{aligned}
m &= \beta_1 \cdot m_{t-1} + (1-\beta_1) \cdot dx \\
v &= \beta_2 \cdot v_{t-1} + (1-\beta_2) \cdot dx^2 \\
loss &= loss_{t-1} - \alpha \cdot \frac{m}{\sqrt{v+\varepsilon}}
\end{aligned}
$$

Fig 5 : Formula for Adam Calculation

$\beta 1$ — For decaying the running average of gradient
$\beta 2$ — For decaying the running average of the square of gradient
α — Learning rate step size

3.1.3 Loss

Loss is a summation of error for every information point made via training and validation set. The thought is to tune the model to reduce the loss however much as could be expected.

In this project, we used a loss function called Categorical Cross entropy(CC Loss).

The motive for utilizing this Loss function is on the grounds that the objectives for classes are encoded in multi-hot encoding since one film poster can be related with various genres. The CC Loss estimates the average number of bits expected to distinguish an event drawn from a set. For this situation, the event is right set of classes of genres for a given film poster.

CC loss is given as :

$$H(p,q) = -\sum_x p(x) \cdot \log(q(x))$$

Fig 6: Formula for CC loss function

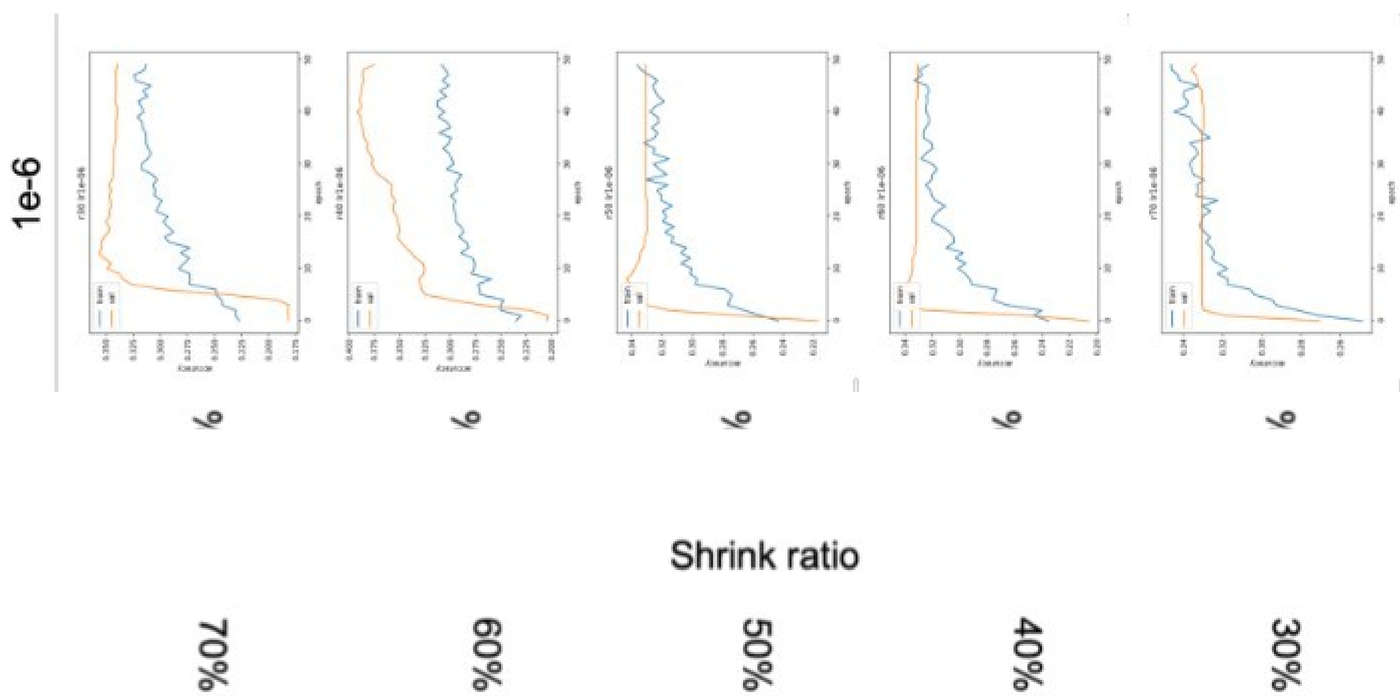
p(x) – 2-d tensor where each row representing a distribution q(x) – 2-d tensor where each element is representing the position of 1 in multi-hot encoding

3.1.4 Regularization

It is a method to keep the model from overfitting by adding additional data to it.

For this project, we used a regularization method called inverted Dropout. This procedure drops out randomly picked apparent/covered up units in a neural network at a probability of p. This diminishes related learning among the neurons and decreases overfitting.

In this project , The dropout probability is set at p = 0.4 .



3.2 DenseNet169

3.2.1 Model architecture:

“In a DenseNet architecture, each layer is connected to every other layer, hence the name Densely Connected Convolutional Network. For `L` layers, there are `L(L+1)/2` direct connections. For each layer, the feature maps of all the preceding layers are used as inputs, and its own feature maps are used as input for each subsequent layers.” [5]

Each architecture in DenseNet-169 consists of four Dense Blocks with varying number of layers. DenseNet-169 has [6, 12, 32, 32] layers in the four dense blocks .

we executed the standard DenseNet-169 model with some changes. The last fully connected layer of 1000 units was indeed changed by 3 successive completely associated layers of 3 1024, 128, and 7 units with ReLU, ReLU, and sigmoid activations individually. The whole model comprises 16,055,876 boundaries, out of which, 15,897,476 were trainable.

Layers	Output Size	DenseNet-169			
Convolution	112×112	7×7 conv, stride 2			
Pooling	56×56	3×3 max pool, stride 2			
Dense Block (1)	56×56	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$
Transition Layer	56×56	1×1 conv			
(1)	28×28	2×2 average pool, stride 2			
Dense Block (2)	28×28	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$
Transition Layer	28×28	1×1 conv			
(2)	14×14	2×2 average pool, stride 2			
Dense Block (3)	14×14	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 24$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 64$
Transition Layer	14×14	1×1 conv			
(3)	7×7	2×2 average pool, stride 2			
Dense Block (4)	7×7	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 16$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$
Classification	1×1	7×7 global average pool			
Layer		1000D fully-connected, softmax			

Fig 8: DenseNet169 model architecture

The DenseNet model is also trained for 70 epochs with learning rate 1e-6 .

3.2.2 Optimization:

Optimizer are algorithms or strategies used to change the characteristics of your neural network, for example, loads and learning rate to lessen the losses.

We used Adam optimizer for the DenseNet model too.

3.2.3 Loss:

Loss is a summation of error for every information point made via training and validation set. The thought is to tune the model to reduce the loss however much as could be expected.

In this project, we used a loss function called Categorical Cross entropy(CC Loss).

The motive for utilizing this Loss function is on the grounds that the objectives for classes are encoded in multi-hot encoding since one film poster can be related with various genres. The CC Loss estimates the average number of bits expected to distinguish an event drawn from a set. For this situation, the event is right set of classes of genres for a given film poster.

3.2.4 Regularization:

For this project, we used a regularization method called inverted Dropout. This procedure drops out randomly picked apparent/covered up units in a neural network at a probability of p. This diminishes related learning among the neurons and decreases overfitting.

4. Result

The task was for a given *Movie poster* the model has to predict its genre based out of the 28 genre classes present in the dataset.

As we can notice see a few models produce a validation accuracy that is higher than the training precision . This is basically in light of the fact that we have utilized Dropout regularization for training. When training, half of highlights are set to 0 utilizing Dropout.

We note that the model proposed excels on certain movie posters where the components in the posters are unmistakably noticeable and huge enough.

we evaluated the performance of the model based on the F1 scores and the AUC of the Receiver Operating Characteristic (ROC) score.

The F1 score can be calculated as :

$$F1\,score = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}}$$

$$Precision = \frac{True\,positives}{True\,positives + False\,positives}; Recall = \frac{True\,positives}{True\,positives + False\,negatives}$$

Fig 9: Formula for calculation of F1 score

"ROC is a probability curve and AUC represents the degree or measure of separability. It tells how much the model is capable of distinguishing between classes." [10]

	precision	recall	f1-score
0	0.49	0.35	0.41
1	0.57	0.66	0.61
2	0.43	0.69	0.53
3	0.41	0.57	0.49

roc auc score, 0.602855145243443

Fig 10: F1 score and ROC-AUC score of VGG16

	precision	recall	f1-score
0	0.38	0.59	0.46
1	0.56	0.67	0.61
2	0.51	0.43	0.47
3	0.41	0.52	0.46

roc auc score, 0.6169649401732731

Fig 11: F1 score and AUC score of DenseNet169.

References

[1] Samuel sung, Rahul choksi : Classification of Movie Posters to Movie Genres

[2] Nirman Dave : Predicting movie genres from movie posters

[3] Jeong A WI , Soojin Jang , Youngbin kim :Poster-based Multiple Movie Genre Classification Using Inter-Channel Features

[4] K. Simonyan , A. Zisserman : Very Deep Convolutional Networks for Large-Scale Image Recognition

[5] Gao Huang, Zhuang Liu, Laurens van der Maaten, Kilian Q. Weinberger : Densely Connected Neural Networks

[6] Marina Ivasic-Kos, Miran Pobar, Luka Mikec : Movie Poster classification into genres based on low level features

[7] Ivašić-Kos, Marina Pobar, Miran, Ipsic, Ivo: Automatic Movie Posters Classification into Genres. Advances in Intelligent Systems and Computing

[8] Kjartansson, Sigtryggur, Ashavsky, Alexander: Can you Judge a Book by its Cover?

[9] Dataset : Kaggle https://www.kaggle.com/c/movie-genre-classification/data

[10] Andrew P.Bradley : The use of the area under the ROC curve in the evaluation of machine learning algorithms

[11] Diederik P. Kingma, Jimmy Ba : Adam: A Method for Stochastic Optimization