

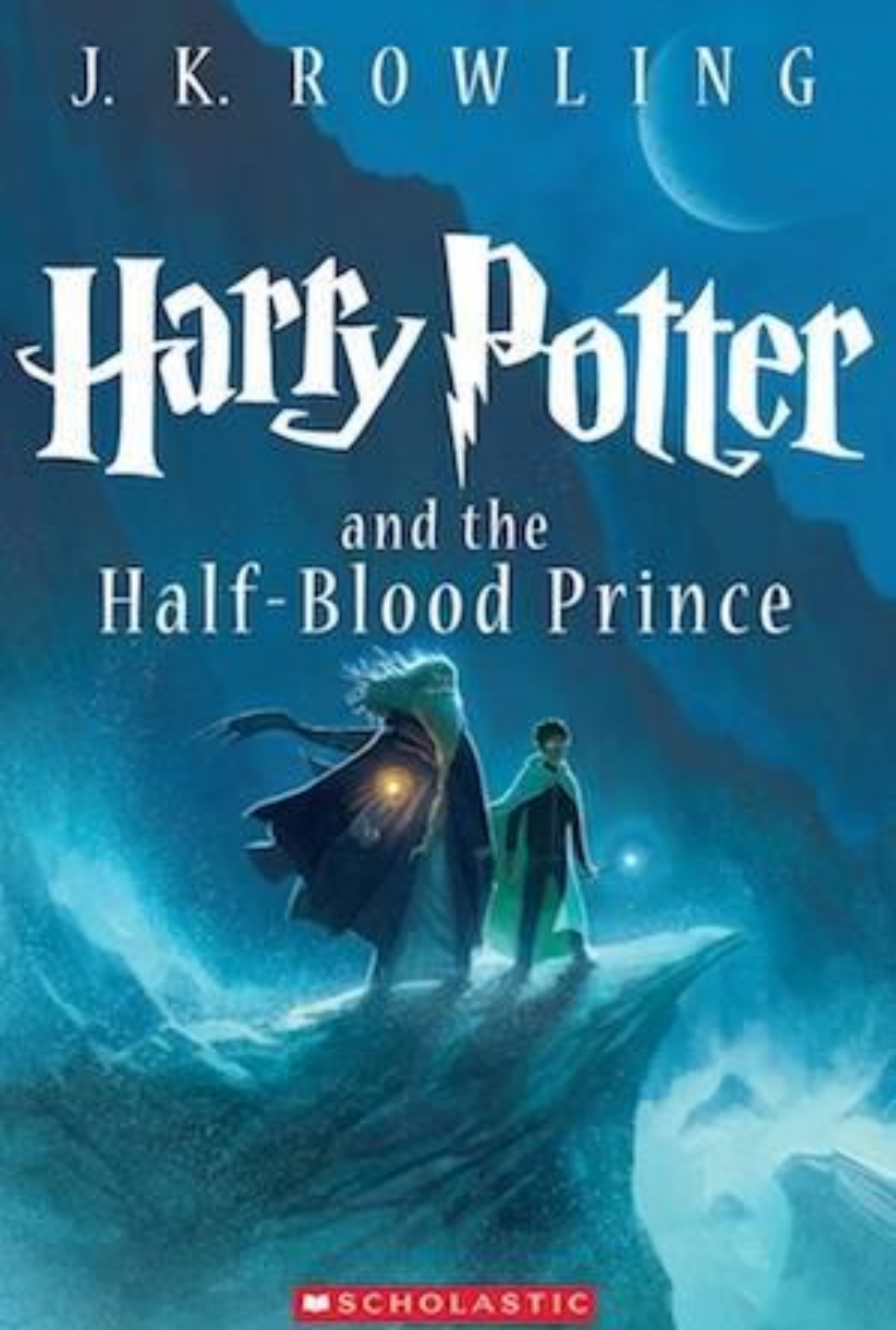
Judge a book by its cover

Submitted by:

Sayali Walke

Guided by:

Prof. Ram Hariharan



Problem Statement

- Book covers communicate information to potential readers, but can that same information be learned by computers?
- The goal is to determine if genre information can be learned based on the visual aspects of a cover created by the designer.

Difficulties Involved

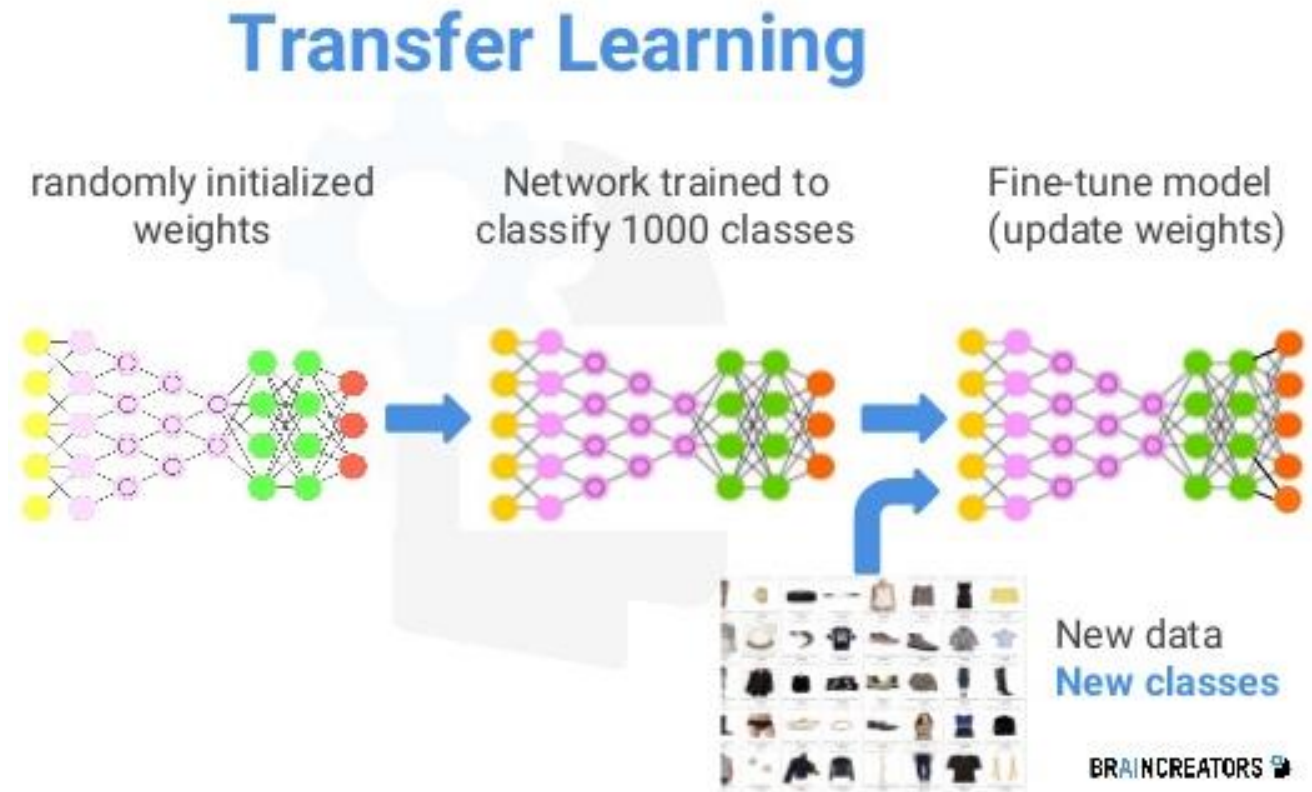
Books come with a wide variety of book covers and styles, including nondescript and misleading covers.

Unlike other object detection and classification tasks, genres are not concretely defined.

Another problem is that there is a massive amount of books exist and it is not suitable for exhaustive search methods.

Solution

- To tackle this problem I used the concept of transfer learning and developed a Convolutional Neural Networks (CNN) based system for book cover genre classification
- I used a pre-trained network on ImageNet. By pre-training VGG16 and ResNet50 models on a very large dataset such as ImageNet, its possible to take advantage of the learned features and transfer it to other applications.



About Dataset

- The dataset for this task is downloaded from Amazon.com
- The dataset contains 57,000 book cover images divided into 30 classes

Out[90]: Text(0.5, 1.0, 'Images by Category')

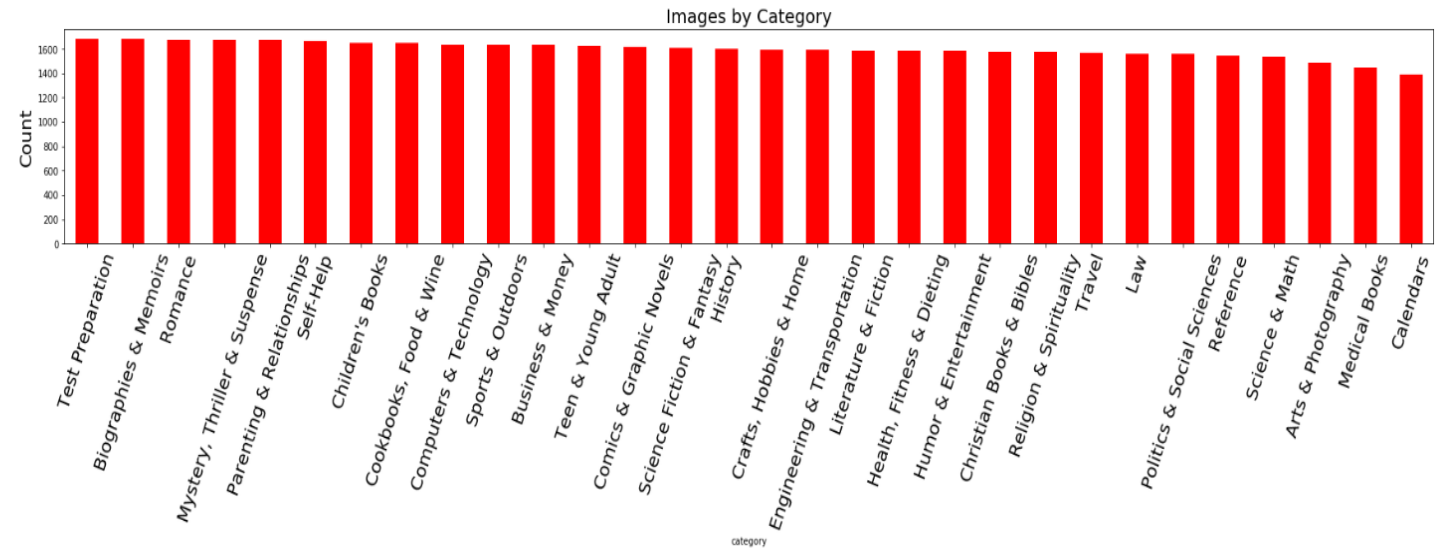


Image Augmentation & Transformation

To build a powerful image classifier using very little training data, image augmentation is usually required to boost the performance of deep networks.

Image augmentation artificially creates training images through different ways of processing or combination of multiple processing, such as random rotation, shifts, shear and flips, etc.

For increasing training and validation dataset images I used random rotation, random horizontal flips

To normalize the image I calculated mean and std deviation on entire dataset for each RGB channel

However, imagenet standard values outperformed my calculated values

After investigation I found `imagenet_stats` should be applied whenever using a model that was pretrained with imagenet.

Convolutional Neural Network

First, I started with image classification using a simple CNN

CNN stands for **Convolutional Neural Network**, where each image goes through a series of convolution and max pooling for features extraction.

With so many images, it took almost 8 hours to train the model, and achieved an accuracy of 23%.

Transfer Learning

After training a CNN from scratch, I used a pre-built and pre-trained model applying transfer learning.

The basic premise of transfer learning is simple: take a model trained on a large dataset and *transfer* its knowledge to a smaller dataset.

For this problem I used VGG16 and ResNet50 pre-trained models. These models are pre trained on huge dataset so, has learned a good representation of low-level features like spatial, edges, rotation, lighting, shapes and these features can be shared across to enable the knowledge transfer and act as a feature extractor for new images in different computer vision problems.

Results

The table shows the accuracy in percentage on the test data for each model

CNN	23 %
VGG16	59.375 %
ResNet50	62.5 %

Results

Accuracy for a prediction and a target in terms of topk for each category:

ResNet50



	class	top1	top5	loss
0	Biographies & Memoirs	69.062500	100.0	0.916479
1	Mystery, Thriller & Suspense	60.322581	100.0	1.104819
2	Parenting & Relationships	57.200000	100.0	1.142326
3	Romance	53.666667	100.0	1.167043
4	Test Preparation	72.666667	100.0	0.753208

So we can see that the category Romance has the least accuracy

VGG16



	class	top1	top5	loss
0	Biographies & Memoirs	50.937500	100.0	1.199545
1	Mystery, Thriller & Suspense	66.451613	100.0	1.045316
2	Parenting & Relationships	54.400000	100.0	1.129545
3	Romance	57.666667	100.0	1.151498
4	Test Preparation	70.666667	100.0	0.776581

So we can see that the category Biographies & Memoirs has the least accuracy

Conclusion

- Image classification can be done using neural network models. Identifying patterns and extracting features on images are what deep learning models can do, and they do it very well. So, I got best possible accuracy of 62% using Resnet50.
- I will explore different pre-trained problems in future like Alexnet.
- I will also experiment on tuning the model performance.





Thank you!