# Judging a Book By Its Cover

Kamil Ahmed Aslam and Isaac Manly

***Abstract -- This paper classifies books by genres and ratings based solely on their covers. The method used is a CNN with a pre-trained DenseNet121, which is then trained on a subset of a dataset of 32,000 book cover images web scraped from Amazon.***

## INTRODUCTION

The value of tailoring the consumer experience is increasing as contemporary shopping becomes more algorithmic and refined by digital platforms. A facet of tailoring experience are recommender systems, which use genre as an important feature for accurate recommendation of music, TV, movies, books, etc. "Don't judge a book by its cover" is a longstanding idiom that is thrown around to discourage judgement based on outward appearance. Ironically, covers are the first interaction consumers have with books and they create a lasting impression which guides our purchase decisions. We wanted to investigate what book covers conveyed and how accurate of a depiction they are of the content they encapsulate. Surprisingly, it is even difficult for humans to distinguish between the genres of different covers if they have vague titles or illustrations without any prior context of the book. There have been studies that predict genres using book descriptions, however, we wanted to arm ourselves with machine learning tools to explore large datasets of book covers to understand the nature of visual design. A study by Gudinavicius and Suminas [1] has shown that the cover is an important part of customer selection. They used eye tracking to determine that the color of a given cover influences the customer, which was pronounced in the women's data. A second study by Nakahata et. al [2] also used eye tracking to show that consumers have a font preference and the perceived warmth of colors

Another facet of tailoring customer experience is the quality of products sold which in this scenario can be quantified with book ratings. Alongside genre, we also explore quality prediction using the ratings of books. With that in mind, we propose a method to predict the genre of a book based solely on its cover using a Convolutional Neural Network.

One of the major difficulties that we expected to run into was that unlike other object detection and classification tasks, genres don't have a concrete definition. From intentional vagueness or satire, to other artistry approaches, we saw a large data set of books with a plethora of designs and styles which resulted in misleading covers.

When CNNs go very deep, problems may arise because the path for information from input layer until the output layer (and for the gradient in the opposite direction) can vanish before it reaches the other side as it gains size. However, we tackle that issue using Densely Connected Convolutional Networks, DenseNets, to simplify the connectivity pattern between layers of architecture.

## RELATED WORKS

There is a lack of literature on predicting book genres from their respective covers. However, our research and paper was inspired by the work that had been done for predicting movie genres from different movie attributes.

There were attempts in the literature at models that predict a movie's genre using non-visual promotional materials. Hoang's study [8] used various machine learning methods such as Naive Bayes and RNN to predict a movie's genre using plot summaries. It found that a Gated Recurrent Units neural network was able to identify genre in 80.5% of cases. Another study conducted by Makita and Lenskiy [7] used a multivariate Bernoulli event model to learn likelihood of genre based off of a movie's ratings. It had a success rate of 50%, which is a reasonably significant result.

The prediction of features based on a work's presentation has a budding literature. In 2019, Barney and Kaya [5] used a ResNet34 model to predict multiple genres of movies and could accurately predict all associated genres 14% of the time. Their "At least one match" accuracy ranged between 19.5% to 50% using genres Animation, Comedy, Drama, Horror and family.

For visual design specifically, there is also a lack of work implementing machine learning because it remains a relatively newer field. However, different techniques have been used to

identify artistic styles and qualities of photographs as well as paintings e.g. Gatys, et al. [9] employed deep CNNs to learn and copy the unique artistic styles of different paintings.

Similarly, Classifying Paintings by Artistic Genre [10]. Zujovic et al. tried to automatically classify digital pictures of paintings by artistic genre. They used a simple approach of feature extraction from grayscale and coloured images which they inputted in different classifiers such as SVM, AdaBoost, ANN etc.

There was more work done on categorical prediction than a numerical classification. For predicting book rating and quality, we turned to the work of Zhou, et al. [11]. By pre-training the CNN, they learned features that were relevant to movie box-office and predicted future revenues. They found that using a multi-modal deep neural network yielded superior accuracy.

## DATASET AND FEATURES

The data used here was webscraped and posted to Kaggle.com [4]. The data was originally web scraped from bookdepository.com. The full dataset classified the books into 33 categories (genres) and each category contained approximately 1000 images.

However, the distribution of ratings were greatly skewed towards ratings above 3. Approximately 95% of the books had a rating above 3. Due to computational limitations we used a subset of the data and opted to use five genres as classes. The five genres are:

1. Science-Fiction-Fantasy-Horror
2. Biography
3. Graphic-Novels-Anime-Manga
4. Poetry-Drama
5. Business-Finance-Law

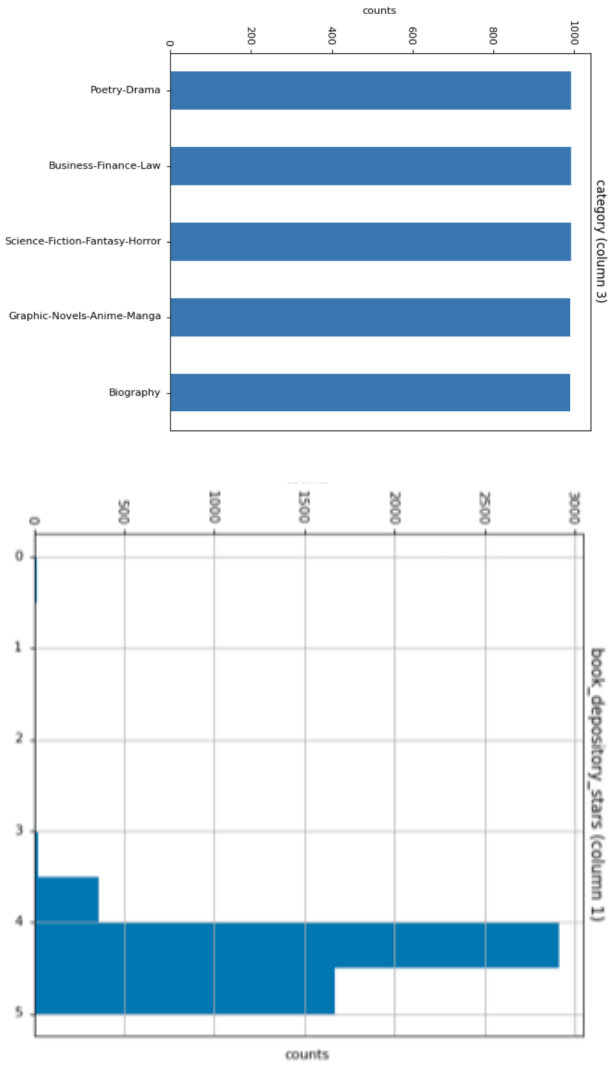


Figure. 1: Genre and Rating Histogram

From the histograms of genres and ratings above in Fig. 1, it is shown that the chosen subset of data is well balanced.

The images obtained from Kaggle had already been reduced in varying size and generally had low resolution. See fig. 2 below.

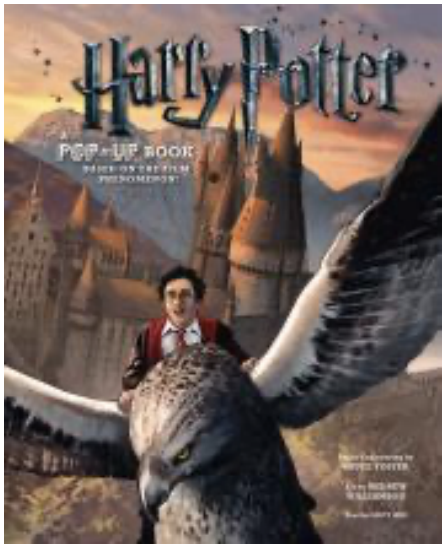


Figure. 2: Sample book cover from Fantasy

Due to the non-standard sizing, we resized all of the images to 128x128 pixels to ensure image consistency and lower computational requirements. We chose to implement random horizontal flips because we are not focusing on the text of book covers, but the patterns that are common to certain genres.


Figure. 3: Sample Manga book cover

Initially, we converted all images to grayscale to produce images like those seen in Fig. 3 above. After further reading in the literature, it became apparent that the RGB values may be important to overall model accuracy. We trained and tested greyscale and non-greyscale models for a comparison.

## METHODS

Convolutional neural nets(CNN) are a powerful class of algorithms which take images as input and assign importance to features of that image by learning weights and biases to classify images or features. These algorithms have convolution layers, pooling layers and fully-connected layers. Traditional CNN's have X layers with X connections, one between each layer and its subsequent layer, while DenseNet-121 has X(X+1)/2 connections. By using DenseNets we can ensure maximum information (and gradient) flow. To do it, we simply connect every layer directly with each other. These extra connections give us a number of advantages:

1. Alleviation of the vanishing gradient problem
2. Strengthened feature propagation
3. Encourage feature reuse and reduce the number of parameters
4. Each layer has direct access to the gradients from the loss function

The architecture for DenseNets can be seen below in table 1. Instead of drawing representational power from extremely deep or wide architectures, DenseNets exploit the potential of the network through feature reuse. Fig. 4 illustrates this structure of DenseNets schematically. And Fig. 5

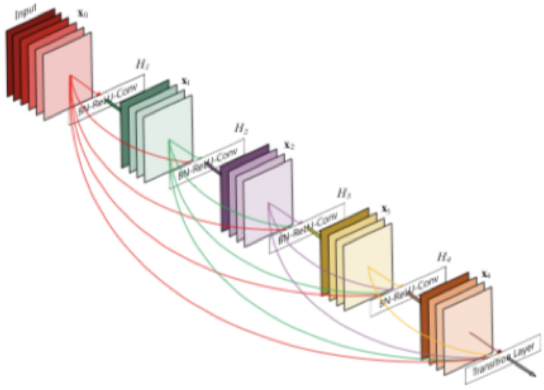

Figure. 4: DenseNet with 5 layers with expansion of 4

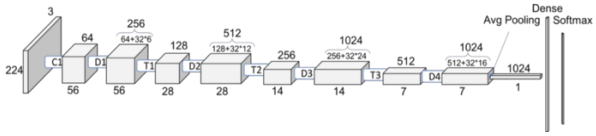


Figure. 5: DenseNet-121 architecture

The optimizer chosen is Adam [6], a gradient based optimization of stochastic objective functions. The loss function chosen was the negative log likelihood loss and we trained for 11 epochs with a learning rate of 0.1.

## RESULTS

Our model achieved an accuracy of 0.47 on the greyscale testing set with five genres. From figure 6, it appears that the model is underfitting the data because the validation losses don't converge to the training losses. On the RGB training and test sets, the model performed similarly, achieving an accuracy of 0.48, but with lower training losses than the greyscale set seen in figure 7. Both accuracies are much stronger than if the model were stochastically assigning genres which would result in approximately 0.20 accuracies.

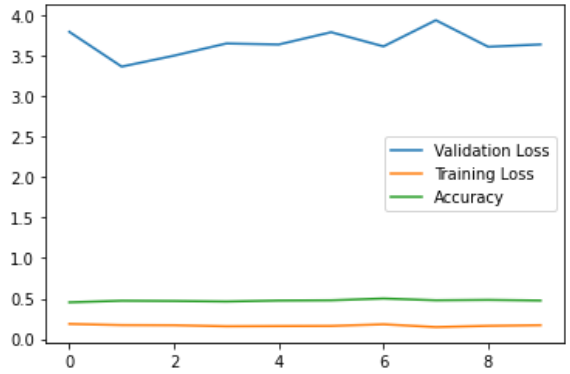


Figure. 6: Losses and accuracy of grayscale set

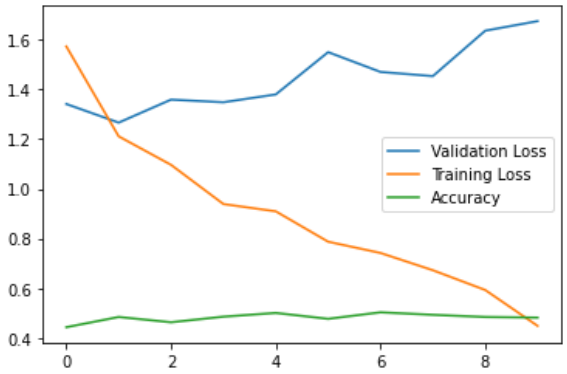


Figure. 7: Losses and accuracy of RGB set

Since the processing of images as RGB or grayscale did not greatly improve accuracy, we chose to use grayscale posters to predict book ratings. The results for the training and testing of the ratings is shown in figure 8. The model attained an accuracy of 0.81 on predicting the rating of a book.

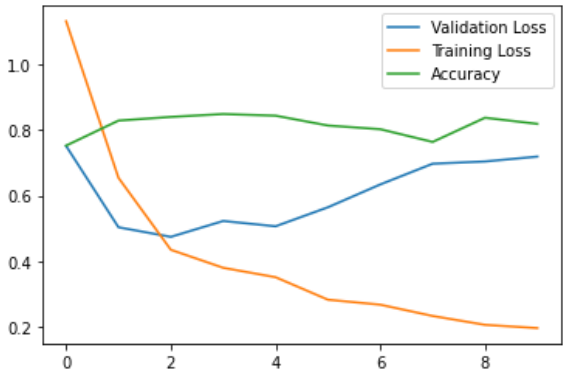


Figure 8: Losses and accuracy of ratings

Accuracy Performances by Training Sets

| Training Sets | Category | Accuracy Score |
|---|---|---|
| Grayscale | Genre | 0.47 |
| RGB | Genre | 0.48 |
| Grayscale | Rating | 0.81 |

## CONCLUSION

We found that DenseNet-121 could predict the genre of a book based on its cover and that transforming the images to grayscale had little effect on accuracy in this experiment. Additionally, we found that the model was significantly more accurate when predicting the rating of a book. In the future, we would use a more balanced data set as there is certainly bias due to the imbalanced ratings. We would also suggest using more computation resources than were at our disposal to train and test on the entire dataset.

## REFERENCES


1. Arūnas Gudinavičius, Andrius Šuminas. (2018) Choosing a book by its cover: analysis of a reader's choice. Emerald. ISSN: 0022-0418
2. Shoko Nakahata, Emiko Sakamoto, Akiho Oda, Noriko Kobata, Sho Sato. (2016). Effects of color of book cover and typeface of title and author name on gaze duration and choice behavior for books: Evidence from an eye‐tracking experiment. Proceedings of the Association for Information Science and Technology. Vol 53.
3. Gao Huang, Zhuang Liu, Laurens van der Maaten, Kilian Q. Weinberger. (2018). arXiv: 1608.06993
4. Luke Anicin. (2019). Book Covers Dataset. Kaggle.
5. Gabriel Barney (barneyga) and Kris Kaya (kkaya23). (2019). Predicting Genre from Movie Posters. Stanford.
6. Diederik P. Kingma, Jimmy Ba. (2014). Adam: A Method for Stochastic Optimization. arXiv:1412.6980
7. Hoang, Q. (2018). Predicting movie genres based on plot summaries. arXiv preprint arXiv:1801.04813.
8. Eric Makita and Artem Lenskiy. A multinomial probabilistic model for movie genre predictions. arXiv preprint arXiv:1603.07849, 2016.
9. L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," arXiv preprint arXiv:1508.06576, 2015.
10. Zujovic, Jana, et al. "Classifying paintings by artistic genre: An analysis of features & classifiers." Multimedia Signal Processing, 2009. MMSP'09. IEEE International Workshop on. IEEE, 2009.
11. Zhou, Yao & z, l & Yi, Zhang. (2019). Predicting movie box-office revenues using deep neural networks. Neural Computing and Applications. 31. 10.1007/s00521-017-3162-x.

# TABLES

Table 1:

| Layers | Output Size | DenseNet-121 | D |
|---|---|---|---|
| Convolution | 112 × 112 | | |
| Pooling | 56 × 56 | | |
| Dense Block (1) | 56 × 56 | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$ | [ 1 3 |
| Transition Layer (1) | 56 × 56 | | |
| | 28 × 28 | | |
| Dense Block (2) | 28 × 28 | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$ | [ 1 3 |
| Transition Layer (2) | 28 × 28 | | |
| | 14 × 14 | | |
| Dense Block (3) | 14 × 14 | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 24$ | [ 1 3 |
| Transition Layer (3) | 14 × 14 | | |
| | 7 × 7 | | |
| Dense Block (4) | 7 × 7 | $\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 16$ | [ 1 3 |
| Classification Layer | 1 × 1 | | |
| | | | |