Deloitte.

Book Cover Judger

ML Guild Apprentice Capstone – Grant Tesdahl

September 24, 2020

Contents

1 **About Me**

2 **Problem Statement**

3 **Methodology**

4 **Lessons Learned & Next Steps**

About Me

Grant Tesdahl

Analyst – CBO CCG
Minneapolis

Education

B.S. in Industrial Engineering from the University of Wisconsin-Madison

Project Experience

Data Modernization at Wells Fargo (rolling off next Friday!)

Interests

Reading, skiing, bouldering, soccer, & Freakonomics podcast

ML Journey

Before Deloitte:

Data Rich Internships at UW Health & ExxonMobil…
…but without the tools to uncover the best insights

›

Data Mining & **Machine Learning Course**…
…where I realized that ML enables those insights

At Deloitte:

Building **business cases** for predictive models
Banking Profitability Insights & Diabetes Mobile App

›

ML Guild Apprentice Program
Training + Capstone

›

Building **predictive models**
Cognitive Guided Tour: Revenue Cycle Modeling

Problem Statement

People Judge Books By Their Covers.

In fact, *Publishers Weekly* found that 75% of booksellers feel that the cover's design is the most important element in book promotion

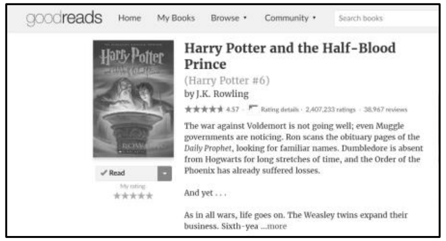
How can publishers ensure a given cover will maximize a book's popularity?

Methodology

Using book rating data and book cover images from Goodreads, I built a
CNN that predicts book popularity

DATA

1. Pull book review data & book cover images
2. **Calculate potential target values** related to a book's average rating and rating count



Harry Potter and the Half-Blood Prince

(Harry Potter #6)

by J.K. Rowling

★★★★★ 4.57 · Rating details · 2,407,233 ratings · 38,967 reviews

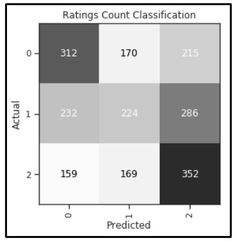
The war against Voldemort is not going well; even Muggle governments are noticing. Ron scans the obituary pages of the *Daily Prophet*, looking for familiar names. Dumbledore is absent from Hogwarts for long stretches of time, and the Order of the Phoenix has already suffered losses.

And yet . . .

As in all wars, life goes on. The Weasley twins expand their business. Sixth-yea ...more

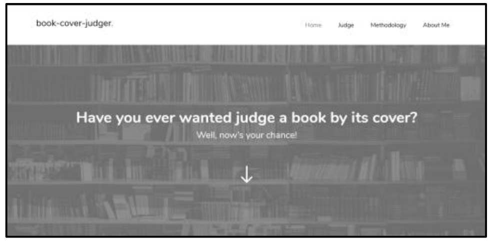
MODELING

1. Train a **CNN for each potential target value**
2. Compare models and pick the best target variable
3. Run additional training



DEPLOYMENT

1. Design and code an interactive website
2. **Integrate the model** into the website



Have you ever wanted judge a book by its cover?

Well, now's your chance!

Data: Acquisition

Approach

Two types of data required for this computer vision task:

Book Cover Images + Book Rating Data

Captured 10k of each via Goodreads, which assigns a **bookID** to each book

Book Rating Data: **Average Rating** (indicates quality) & **Rating Count** (indicates popularity)

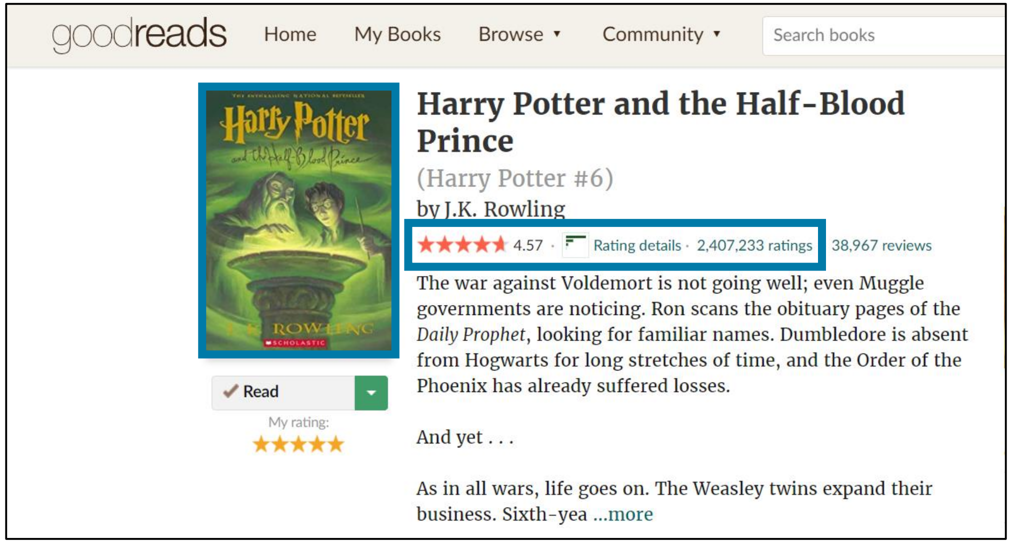
Example

bookID = **1**

↓

https://www.goodreads.com/book/show/**1**

↓



Harry Potter and the Half-Blood Prince

(Harry Potter #6)

by J.K. Rowling

★★★★★ 4.57 · Rating details · 2,407,233 ratings 38,967 reviews

The war against Voldemort is not going well; even Muggle governments are noticing. Ron scans the obituary pages of the *Daily Prophet*, looking for familiar names. Dumbledore is absent from Hogwarts for long stretches of time, and the Order of the Phoenix has already suffered losses.

And yet . . .

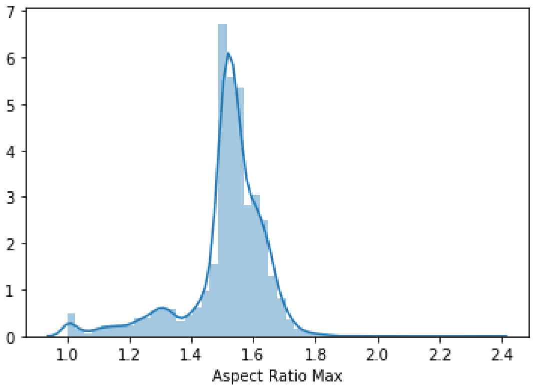
As in all wars, life goes on. The Weasley twins expand their business. Sixth-yea ...more

Data: Transformation

Book Cover Images

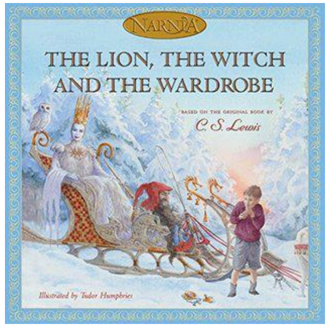
Transformed images due to unequal aspect ratios

Image Aspect Ratio

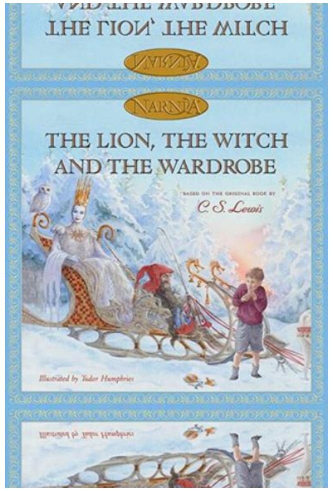


Set aspect ratio to 1.5; Padded images with a reflection; Normalized pixel values

Original Image (Ratio = 1)



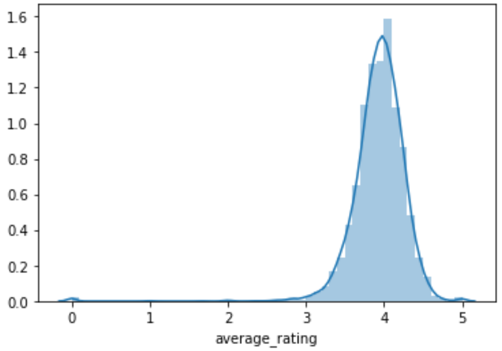
Transformed Image (Ratio = 1.5)



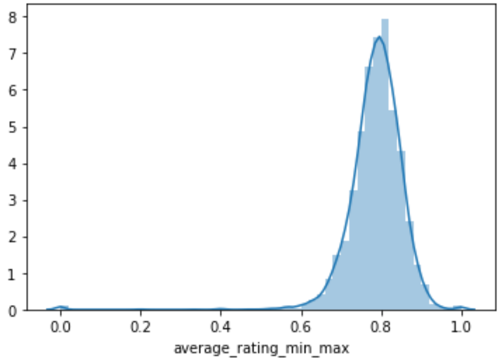
Book Rating Data

Transformed data to create 6 potential targets (4 regression, 2 classification)

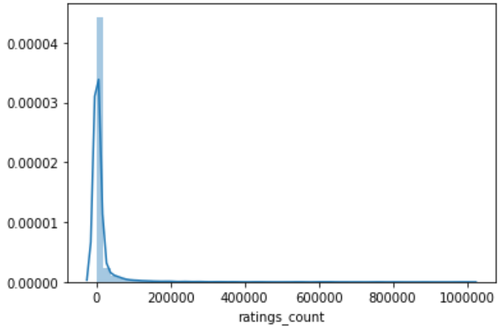
Average Rating



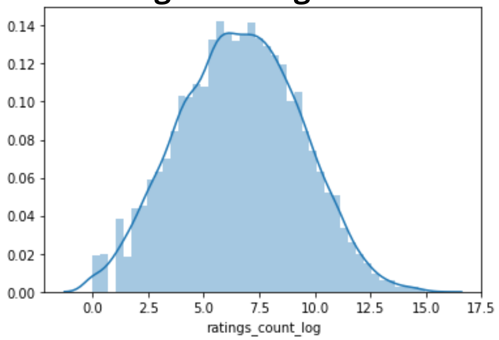
Min-Max Norm of Rating



Rating Count



Log of Rating Count



Classification Segment Ranges

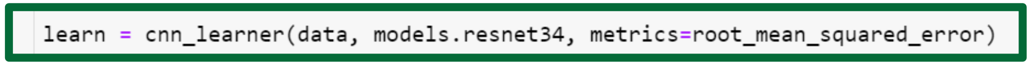
Segment	Average Rating
High	4.07 to 5.00
Medium	3.84 to 4.07
Low	0.00 to 3.84

Segment	Rating Count
High	2,627 to 4,597,666
Medium	223 to 2,627
Low	0 to 223

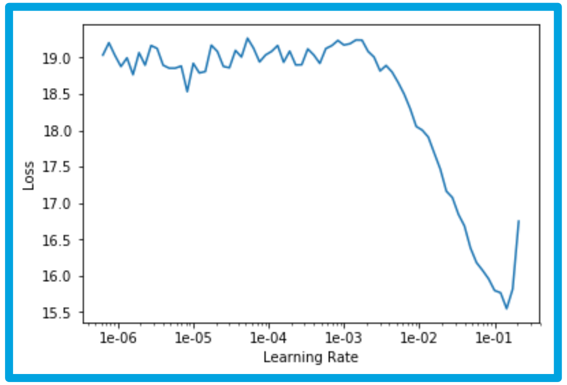
Modeling: Initial Training

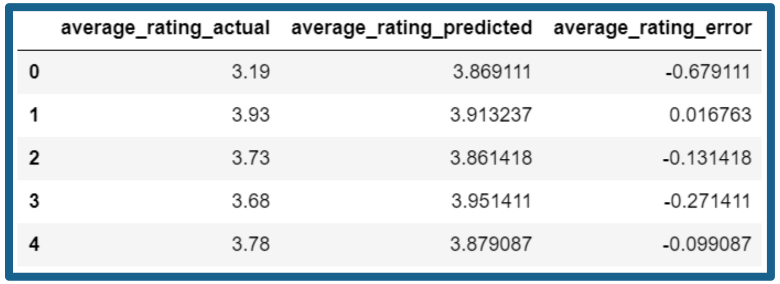
For Each Target Variable:

- Randomly split 20% of the samples for the **validation set**
- Create **mini-batches with 12 images** of size 192x128 (determined via experimentation)
- Create **CNN based on the ResNet34 Architecture** to start with a pre-trained model rather than from scratch
- Identify a **learning rate** to ensure rapid convergence
- Train for **5-8 epochs** depending on whether the loss continued decreasing
- Save validation set **predictions** to compare target variables



```
learn = cnn_learner(data, models.resnet34, metrics=root_mean_squared_error)
```





	average_rating_actual	average_rating_predicted	average_rating_error
0	3.19	3.869111	-0.679111
1	3.93	3.913237	0.016763
2	3.73	3.861418	-0.131418
3	3.68	3.951411	-0.271411
4	3.78	3.879087	-0.099087

Modeling: Target Variable Selection

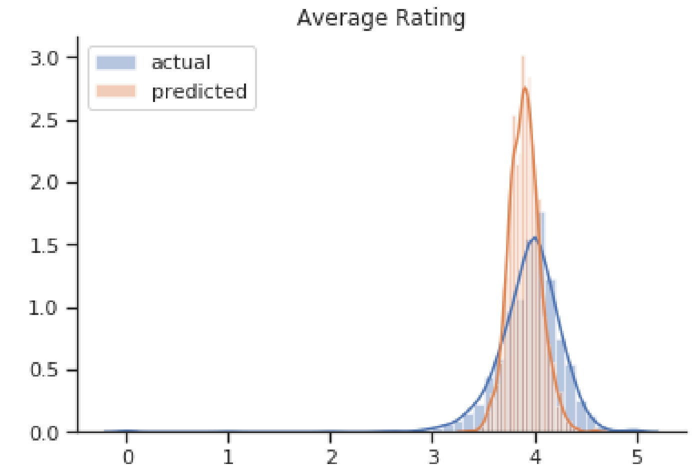
Rating Count Classification looks the least bad (42% accuracy); regression models center heavily around the mean

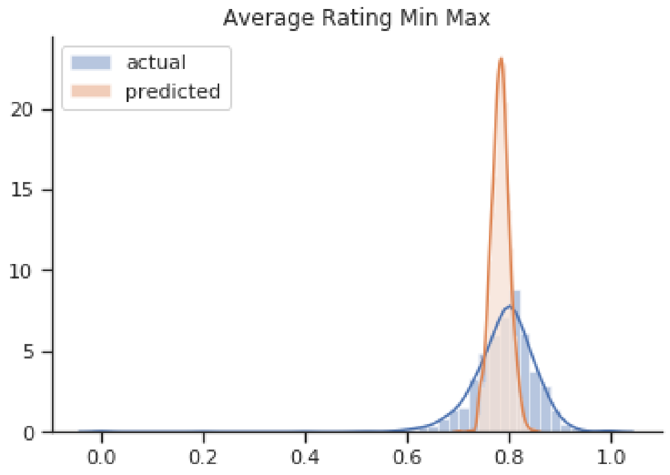
Regression: Raw Values

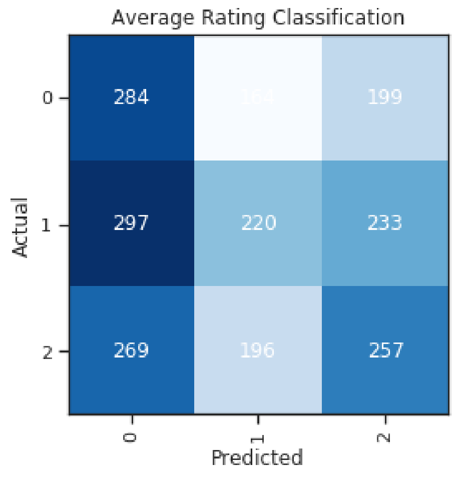
Regression: Transformed Values

Classification: High, Medium, & Low

Average Rating



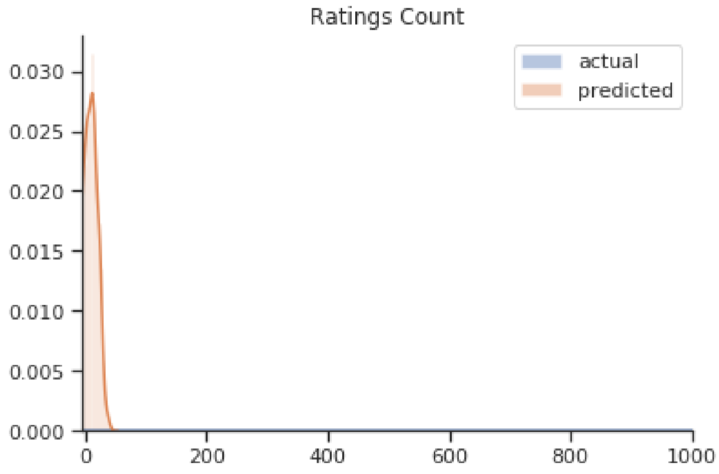


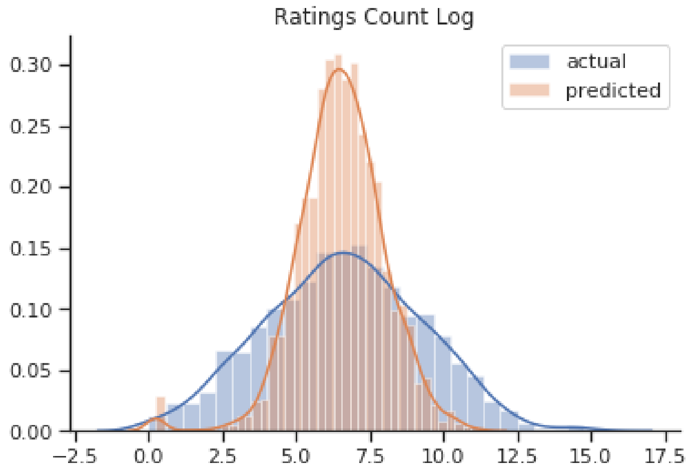


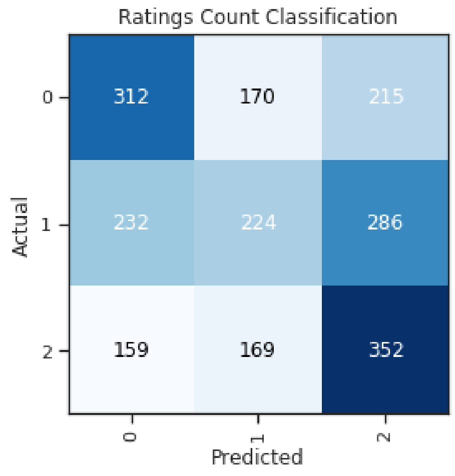
Average Rating Classification

Actual \ Predicted	0	1	2
0	284	164	199
1	297	220	233
2	269	196	257

Rating Count







Ratings Count Classification

Actual \ Predicted	0	1	2
0	312	170	215
1	232	224	286
2	159	169	352

Modeling: Additional Training

Enhancing the Classification Model:

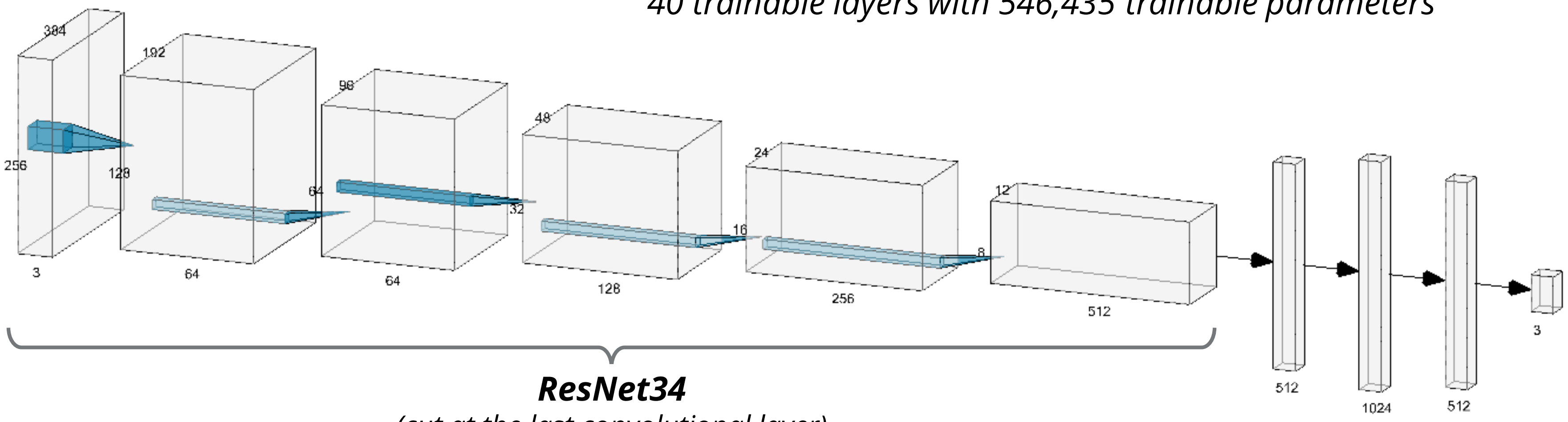
Unfreeze ResNet34 Layers & Train for 20 Epochs

Recreate Dataset with Larger Images

Refreeze & Train Last Layers for 3 Epochs

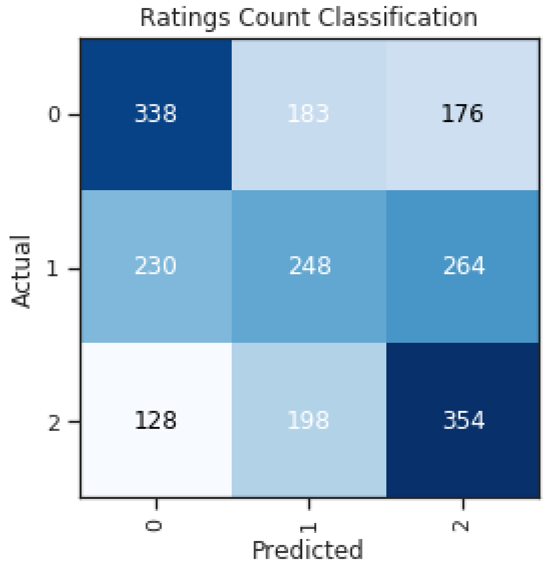
Final Architecture & Results:

40 trainable layers with 546,435 trainable parameters



ResNet34
(cut at the last convolutional layer)

44% Accuracy on the Validation Set
(Compared to 33% Baseline; P-Value < 0.0001)



Ratings Count Classification

Actual \ Predicted	0	1	2
0	338	183	176
1	230	248	264
2	128	198	354

Deployment

www.book-cover-judger.com

Lessons Learned & Next Steps

LESSONS LEARNED

1

Programming in Python

- My previous programming experience mainly consisted of DataCamp exercises, so a full project in Jupyter was a constant learning experience

2

Deep Learning

- The Neural Net lecture during the Bootcamp flew way over my head, so I really enjoyed having extra time to learn about CNNs, transfer learning, and other details

3

Importance of Experimentation

- Looking back, I used the recommended hyperparameters and architecture structures too frequently. In future projects, I plan to experiment with different combinations so I can see what works best for my specific use case

NEXT STEPS

1

Determine Most Important Features

- Extract and visualize the feature maps of each CNN block
- Translate activations into importance

2

Generate Book Covers with a GAN

- Collect more book covers & subset high popularity ones
- Train GAN on subset

3

Build in PyTorch from Scratch

- Leverage complex architectures (e.g. ResNet101) & customize
- Transition to an ordinal multi-label problem

Thank you!