Amazon Price Prediction:

Deep Learning Through Images

JaeHo Bahng, Claire Kim, Sheeba Moghal, Powell Sheagren, Ofure Udabor

Department of Data Science & Analytics, Georgetown University

May 2024

Introduction

Selling on Amazon is a strategic, and somewhat tedious, endeavor. While sellers are given the choice to engage in a first-party relationship, where Amazon handles the logistics of selling the inventory customers provide, sellers can also opt for more independence with third-party sellers. As “61 percent of paid units were sold by third-party sellers” in the last three months of 2023, it is evident many sellers prefer less limits put on their services (Coppola, 2024). One of the advantages of being a third-party seller on Amazon is being able to have more influence over price setting (Brown, 2023). While appealing, this added benefit comes with expectations that sellers spend an extensive amount of time learning what their products would be worth in the eyes of the general public. In our work, we attempt to address this challenge by determining how well a constructed neural network can predict the prices of products on Amazon based on their stock images.

When looking at past research, one study examines the general effect images have on pricing. The study centers around two models: a tailored VGG- 16 convolutional neural network (CNN) and “hedonic regression” (Feuerriegel & Naumzik, 2020). The CNN is adjusted to include a continuous output to track image sentiment, while the regression model uses Ordinary Least Squares as a baseline to quantify the contributions of associated image and text data to the listed prices. When comparing image-based content and text-based content and their effects on real estate listing prices, Feuerriegel & Naumzik (2020) found an increase of one standard deviation change in image sentiment caused a price increase around 10 percent more than an equivalent increase in text sentiment. Furthermore, this study pinpoints a way for a connection between images and price prediction to be tracked.

Implementing a similar model architecture, another study discerns which model predicts prices best. With a set array of models to test, Chen, et al. (2018) observes which model predicts the prices of bicycles and cars based on the images they are fed. After collecting data sets consisting of bicycle and car images and their associated prices, the researchers set baselines for price regression and price segment classification and subsequently train multiple CNNs to compare performance. The models tested consist of: pretrained VGG-16 and MobileNet models (with adjusted outputs for regression and classification) and their own novel deep learning models, called PriceNet-Reg and PriceNet-Class. One interesting finding is that PriceNet-Reg yielded better performance than, “transfer learning, as well as linear regression baselines,” when modeling price regression (Chen, et al., 2018). Considering their novel

architecture contained a different number of convolutional and dense layers, this study provides insight into how a CNN may be further constructed to improve price prediction.

The market percentage of third-party sellers has maintained an upward incline from 26 percent in the second quarter of 2007 to 60 percent in the second quarter of 2023 (Coppola, 2024). Thus, our objective is to construct a neural network model which builds upon past research and acts as a catalyst for the further enhancement of the pricing experience of third-party sellers. To do this, we created a structured convolutional neural network which was designed to take the normalized images and produce a price based off of the training data. The success of the model and the design we used to create our output will be discussed further into the paper.

Data Gathering, Cleaning, and Preprocessing

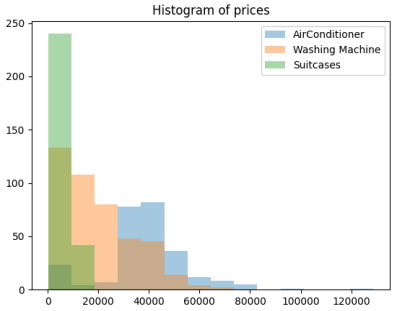
The data gathering process was fairly simple as it only involved pulling data from kaggle, but data cleaning and preprocessing involved many more steps. Data merging did not end up being an issue as we ended up using datasets of singular types of Amazon products and spreadsheets with all of them combined, all formatted the same way, which were both included in the kaggle dataset. The data cleaning involved a few steps, the first was cleaning the price column formatting as it still included commas and symbols which made it a character vector rather than numeric. This was trivially completed with some pandas string splicing. The second step involved using the photo urls in the spreadsheet to query the given photos. The photos were not included in the dataset specifically but were linked, unfortunately some of the links were no longer valid and led to error messages when pulled so a try-except format was used. This process also ended up being the most time consuming as each image had to be pulled, checked, and then added on to the rest of the image stack. The images also were not the same size so it became necessary to pad them to square shapes and then resize to a standard format of 320x320. The images were all low resolution so this did not affect the quality of the images greatly. As a last step we regularized the images and the price in order to provide for better analysis.

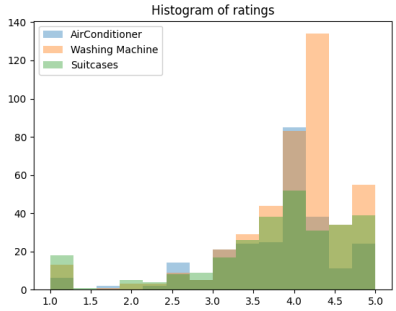
In order to prepare for neural network analysis, the data was also pre processed through a set of image transformers. These conformers conducted the following image augmentations: horizontal and vertical flipping, rotations of various degrees, color jittering, perspective changes, and resizing / cropping within the frames. This was

specifically when preparing the training set/data loader for two purposes. The first purpose was to increase the number of images the model was being trained on which would improve its accuracy by giving it more information to handle. The second purpose was to train the model on manipulated images that were harder to interpret, and thus better for the training process, and then testing it on “easier” images in order to give the model a chance to show the more nuanced patterns it found. This step allowed for an improved model accuracy for these two reasons and was an integral part of the data preprocessing.

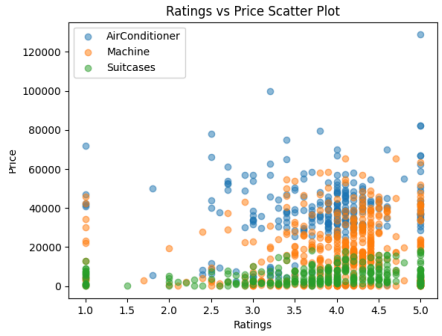
Exploratory Data Analysis

In conducting initial exploratory data analysis, histograms were plotted to examine the distribution characteristics of prices and ratings across three product categories: air conditioners, washing machines, and suitcases. The price distributions for washing machines and suitcases exhibited significant right-skewness, whereas air conditioners displayed a distribution that more closely approximated a





normal curve. This observation wass critical for the selection of hyperparameters in our predictive modeling which will be explained in the following sections. Additionally, the analysis of product ratings revealed a concentration in the range of 4 to 5, with a skew towards the lower end of this range and a noticeable peak at the rating of 1. Despite our primary model input being images, an exploration into the potential



correlation between price and ratings was also undertaken. The scatter plot indicated a modest correlation, noting that higher-priced items tend to cluster within the higher ratings bracket of 4 to 5. Building on the dataset insights, we intend to develop a predictive model that can accurately estimate the price of a product while emulating the observed price distribution patterns.

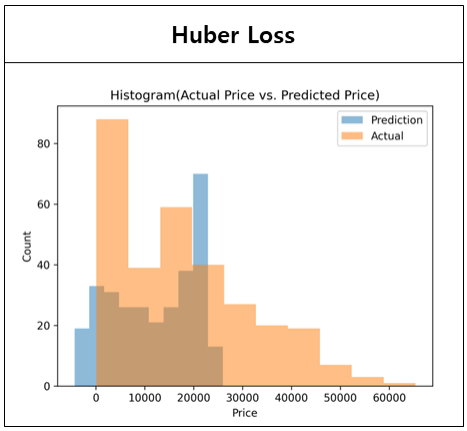
Statistical Methods & Analysis

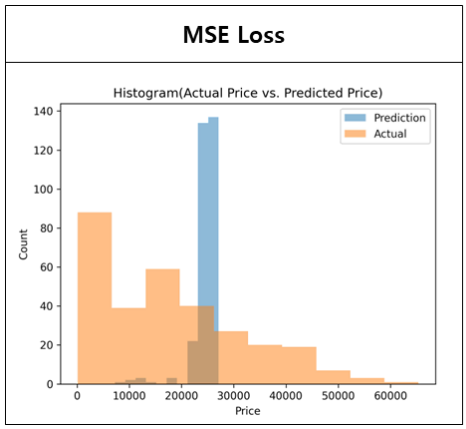
Our approach to solving the problem involved developing a bespoke deep learning architecture, tailored specifically for the task of price prediction based on images. To identify the optimal architecture and hyperparameters, we trained our model on both individual and multiple categories. Given the nature of the input data—images—we incorporated Convolutional Neural Network (CNN) layers. Distinctively, our research did not rely on pre-trained models; instead, we constructed our model from the ground up. This approach necessitated a substantial amount of experimentation and iterative trial and error. The extensive process was critical in developing a model that effectively meets the specific requirements of our task. The final architecture of our model is detailed in the table below.

Layer	Size	Number Filter/Unit
Conv2d	(5,5)	32
Batch Normalization	-	32
ReLu	-	-
MaxPool	(3,3)	-
Conv2d	(5,5)	64
Batch Normalization	-	64
ReLu	-	-
MaxPool	(3,3)	-
Conv2d	(5,5)	64
Batch Normalization	-	64
ReLu	-	-
MaxPool	(3,3)	-
Conv2d	(5,5)	128
Batch Normalization	-	128
ReLu	-	-
MaxPool	(2,2)	-
Dropout	0.3	-
Dense(Feed Forward)	-	1024
ReLu	-	-
Dropout	0.3	-
Dense(Feed Forward)	-	512
ReLu	-	-
Dropout	0.3	-
Dense(Feed Forward)	-	256
ReLu	-	-
Dense(Feed Forward)	-	1

Our model architecture includes four convolutional blocks and three linear blocks. This configuration was chosen to maximize efficiency within the constraints of limited computational resources. A key decision in the design process was determining the appropriate size of the input layer to the feed-forward network. Initially, we hypothesized that a greater number of kernels in the convolutional layers and fewer, smaller max pooling layers would minimize information loss, leading to more accurate predictions. This approach resulted in a feed-forward network with over a million input nodes, which proved to be too computationally taxing. To address this, we explored various configurations, experimenting with different numbers of kernels and pooling sizes. Our findings indicated that a higher number of kernels combined with larger pooling layers yielded the most effective results in terms of balancing computational efficiency and model performance. Furthermore, to mitigate the risk of overfitting, we integrated regularization techniques within each block of the architecture. Specifically, we employed batch normalization in the convolutional blocks and dropout in the linear blocks. These regularization strategies are critical for enhancing the generalizability of the model across diverse datasets.

In developing our model, we prioritized the tuning of hyperparameters, particularly the choice of loss function. Given the right-skewed distribution of our price data, we selected the Huber loss over the traditional Mean Squared Error (MSE). The Huber loss is advantageous because it minimizes the influence of larger values typical of MSE by combining the qualities of MSE for small errors and Mean Absolute Error for large errors. This choice enhances robustness and accuracy across various price levels, particularly in mitigating the impact of outliers, making it ideal for our skewed dataset.



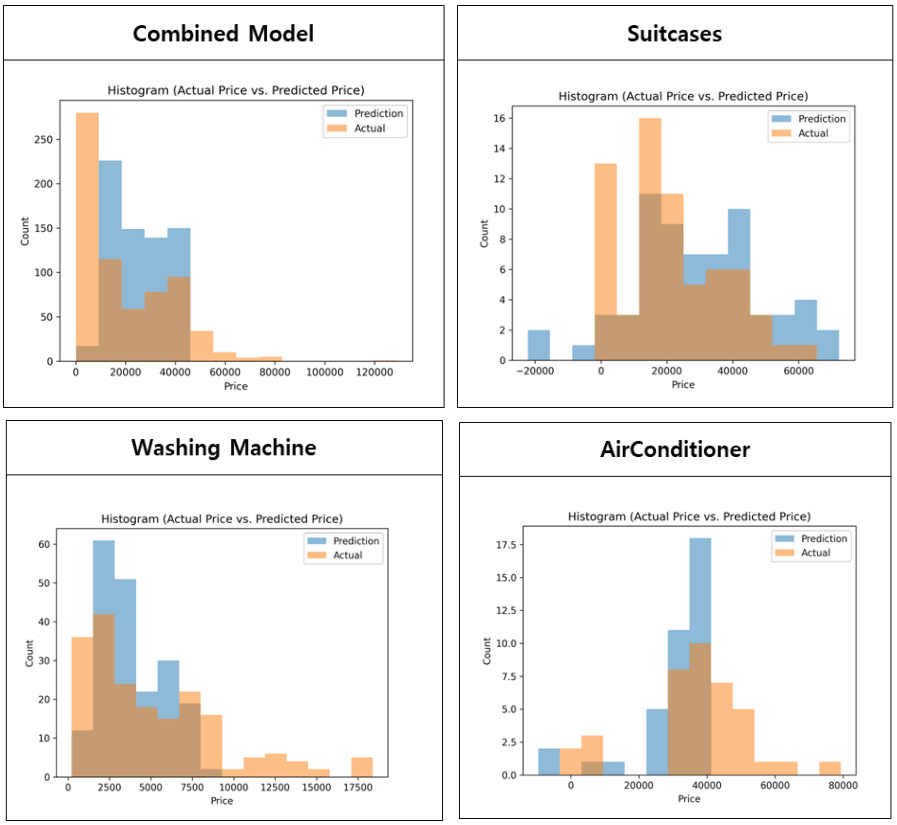


Results

Loss Value and Price Distribution

Our results indicate that a relatively simple architecture was effective in replicating the price distribution across various categories, including air conditioners, washing machines, and suitcases. Additionally, a composite model incorporating all three categories demonstrated the capability of this approach. Despite selecting the Huber loss over the MSE to minimize the influence of outliers, the histograms reveal that the models struggled to accurately predict the lower price values, indicating a residual bias towards higher prices. We found that a learning rate of 0.0005 was optimal for standardizing price values, ensuring consistent and effective model training. The AdamW optimizer was employed to further enhance performance, chosen for its ability to adjust the learning rates based on a running average of recent gradients and squared gradients. This setup proved instrumental in refining the model's ability to capture a broad spectrum of price data, though adjustments are still required to better address the lower end of the price range.

	Combined	Suitcases	Air Conditioner	Washing Machine
Loss	0.23	0.35	0.32	0.30



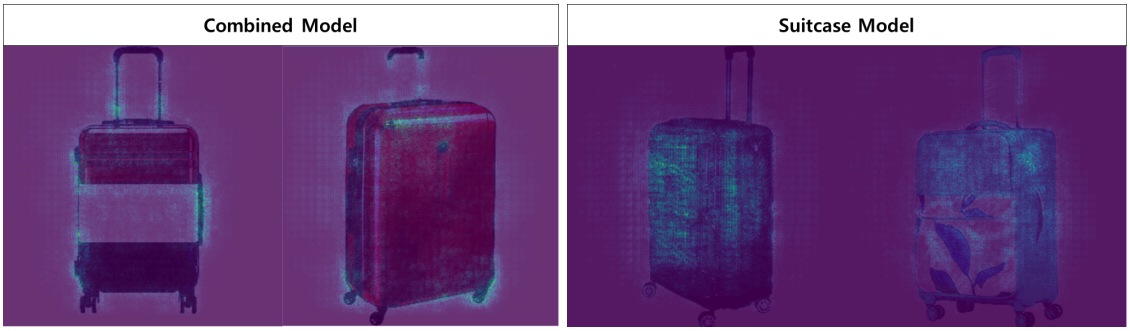
Sanity Check

While our model demonstrates satisfactory accuracy, it is crucial to conduct a thorough sanity check to ensure that it correctly focuses on relevant aspects of the images for price prediction. To this end, we will employ two visualization techniques: saliency maps and class activation maps (CAMs). Saliency maps will help us identify which parts of an image most significantly influence the model's output by highlighting the most responsive pixels. Class activation maps, on the other hand, offer deeper insight by revealing the regions of the image that are most indicative of a particular class according to the model. In our case, since there is only one output node, it would be interpreted as which regions influence the price the most. These techniques are crucial for verifying the internal

mechanics of our model, ensuring that its recognition capabilities are aligned with logical and relevant image features rather than false correlations.

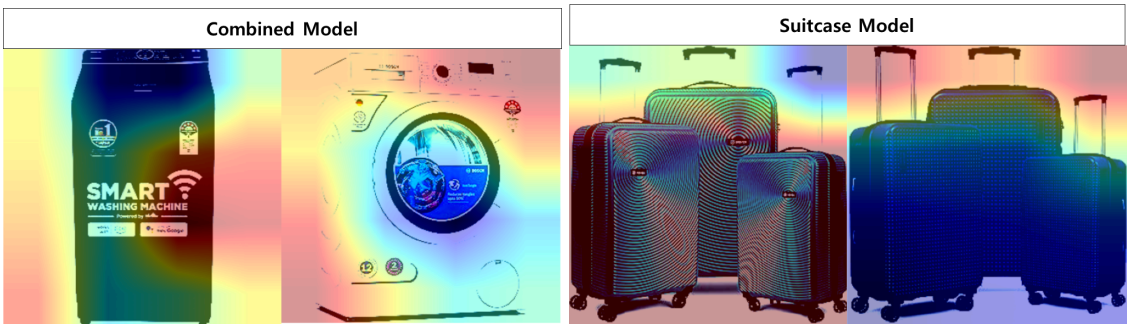
Saliency Maps

Saliency maps are used to compute the gradients of each pixel with respect to the loss function, and the maximum gradient from the RGB channels is displayed on a 2D map to highlight regions crucial for price prediction. Analyzing these maps reveals clear differences between models: those trained exclusively on washing machines show a uniform focus across the object, indicating detailed analysis. In contrast, combined-category models emphasize edges, suggesting that edge detection helps categorize objects which suggests that a critical step for accurately predicting prices relied on item categorization such as washing machines and suitcases. This edge-focused strategy reflects the model's adaptation to the significant price variations between different types of products.



Class Activation Maps

Class Activation Maps (CAMs) offer another visualization method for determining the focus areas of an algorithm in price prediction. These maps highlight broader regions rather than individual pixels by implementing a hook on the last convolutional layer, where gradients are computed relative to the loss function. CAM outputs, such as those from models trained solely on suitcases, identify focused features like handles and main bodies, while models for appliances like washing machines highlight functional areas. Although CAMs provide a more general view of significant regions than saliency maps, their interpretation tends to be less direct and more subjective.



Discussion

The model was able to replicate the price of given appliances through a convolutional neural network. We are satisfied with the performance of the models and find that the construction seems to align with where a human may look for quality, in logos and details around the edges. It was also an interesting result that the combined model with various types of appliances was more successful than single appliance options; this implies that the concept of cost can be generalized between products and in fact it may be more successful while doing that. The alternative is that having more data to train on, the model improved its predictions. The saliency maps confirm this as well as provide insight to some core differences between various products of the same type. The use of class activation maps was not nearly as enlightening but did show some trend towards useful information. The models lack of ability to detect outliers is a limitation that could be addressed with more data or more computational ability but is expected given the complexity of the problem.

Next steps in this research include expanding the efficacy of the model, running the training on a larger dataset, running the training on higher resolution images, and adding more categories in an attempt to make a generalized price prediction model. To comment further on the latter point, having a generalized price detection model would be an incredibly innovative application of machine learning which could very easily improve pre-existing third party sales websites. A general model trained on images from one of those applications would be able to give users a chance to have a predicted price before putting their object up for sale or before comparing it to other prices.

Conclusion and Future Implications

In summary, the experiment aimed at using image data to estimate e-commerce product pricing has yielded encouraging outcomes which could revolutionize third party online markets. We have created a strong basis for reliable model training by preprocessing the Amazon picture dataset using methods like padding, scaling, and normalization, and then further improving it through image augmentation. Through experimentation with different CNN architectures and hyperparameters, we have been able to determine the best combinations for this particular application. The constructed model accurately identifies important image elements across many categories and shows a significant capacity to forecast product prices, displaying price distributions that are similar to the real data. This shows that image-based price prediction might be a useful tool for e-commerce platforms, possibly resulting in more dynamic and responsive pricing strategies with the correct preprocessing and model modification.

Hence, the use of images to predict prices will have a wide range of future impact which could potentially be revolutionary in various domains, especially third party e-commerce. Model efficiency is increased by exploiting transfer learning architectures like ResNet, VGG, or EfficientNet, which enable models to quickly adapt to new domains without requiring a lot of retraining. Training datasets can be greatly enhanced by using Generative Adversarial Networks (GANs) for data augmentation, which increases model resilience, and can find the unique markers for better price prediction. By combining textual, numerical, and visual data, multi-modal modeling can offer a more thorough analysis and improve accuracy. These methods could be applied not only to e-commerce but also to real estate, travel, and agricultural price forecasting. Moreover, adding non-static images—like video data—may be able to capture dynamic variables and provide even more detailed information into pricing tactics.

References

Brown, R. (2023, November 6). *Amazon 1P vs. 3P: Pros & Cons Brands Need To Know*. Pattern. https://pattern.com/blog/amazon-1p-vs-3p-pros-and-cons/#:~:text=in%20your%20browser.-,What%20is%201P%20vs%203P%3F,and%20more%20responsibility%20for%20logistics.

Chen, S., Chou, E., & Yang, R. R. (2018). The Price is Right: Predicting Prices with Product Images. Retrieved April 22, 2024, from https://cs229.stanford.edu/proj2017/final-reports/5237321.pdf

Coppola, D. (2024, February 8). *Amazon: Third-party seller share 2023*. Statista. https://www.statista.com/statistics/259782/third-party-seller-share-of-amazon-platform/

Naumzik, C., & Feuerriegel, S. (2020). One picture is worth a thousand words? The pricing power of images in e-commerce. Proceedings of The Web Conference 2020. https://dl.acm.org/doi/pdf/10.1145/3366423.3380086