CS230

The Art of predicting Art Auction Price

Henry Nho
Stanford Electrical Engineering PhD Alumni (SCPD)
gohenry@stanford.edu

Haijin Park
Stanford Law School
haijin@stanford.edu

Abstract

In this research, we build a series of deep learning models to predict the art auction price given numeric features and the images of the artworks from the last 10 years. We trained multilayer perception (MLP) model using only numerical data and several CNN models using only images. The best result is achieved when fused model is used based on both image and numerical data.

1 Introduction

Traditionally, the art market has been considered inaccessible for laypeople. Recent technological advances made gigantic amounts of art market data available to the public; however, many still lack the ability to synthesize the data to understand the market and predict art prices. Therefore, people still rely on human experts to estimate the value of artwork, which can sometimes be expensive to attain or subject to the influence of human biases. We would like to provide a deep neural network model using both numerical/textual data and image data to predict hammer prices at art auctions. This tool will potentially contribute to the growth of the art market in several ways: (1) prospective sellers of art will have more access to the pricing of the artwork they own; (2) prospective bidders at an art auction will readily have an independent appraisal of the painting other than the auctioneer's prediction prices; and (3) auction houses can use this tool to predict their profit or loss.

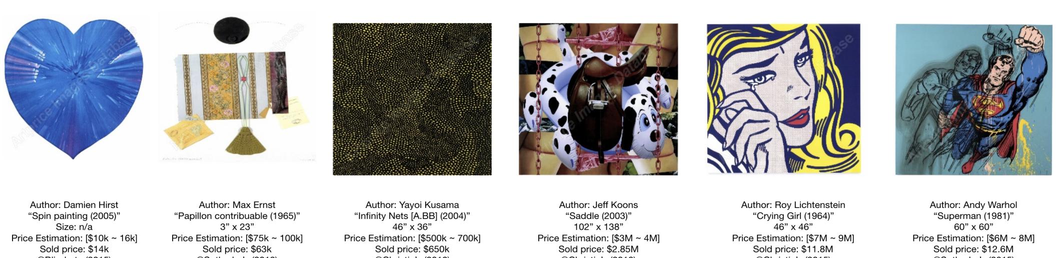

Figure 1: Examples of images and numerical/textual data

2 Related work

While the art industry has shown some interest in applying deep neural networks in predicting art auction prices [1], there has been little previous research in this domain. [2] One of the previous projects in CS230 built a model to predict art auction prices using images of artworks, but could not overcome the high variance problem. [3] Our project will be the first study that experiments with different deep neural network models using textual/numerical data along with image data to provide a tool to predict art auction hammer prices. Although there is little work in art auction,

,

useful insights can be drawn from research in other domains. Recent studies in art classification used transfer learning to improve its performance, by fine-tuning pre-trained convolutional neural networks (CNNs) [4, 5, 6,7]. Similar to our approach, many studies [8, 9, 10, 11] aimed at housing valuation combined image features extracted from CNNs and numerical features to estimate the house price.

3 Data collection and preprocessing

We created a dataset specifically for this project that consists of image and textual/numerical data (See Figure 2). Auction data encompassing a ten-year period (2009-2019) was collected from Artprice.com and Artfacts.net, using a custom-built python crawler. In addition, since the demand and liquidity in the art market significantly grew over the last ten-year period, we also included Forbes billionaire index from each year to contextualize the sales prices in our dataset.

We dropped auction data that contains missing values or has pre-sale estimates or hammer prices lower than \$10,000. This process shrunk the size our dataset from 14,705 examples to 7,755 examples. The output variable, hammer price, is log transformed because it was severely skewed. For the auction houses' pre-sale low and high estimates, we use the logged average of the low and the high estimate. The images were re-sized into two uniform dimensions, 224x224 pixels and 229x229 pixels.

Note that auction houses' estimates are intentionally left out from the textual/numerical features because we intend to provide a tool to predict the hammer price independent from the auction houses' estimates. Neither linear regression model, multilayer perception model, or any of the fused models use auction houses' pre-sale estimates.

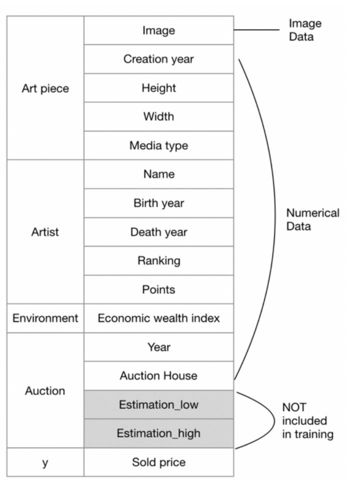

Figure 2: Structure of the features

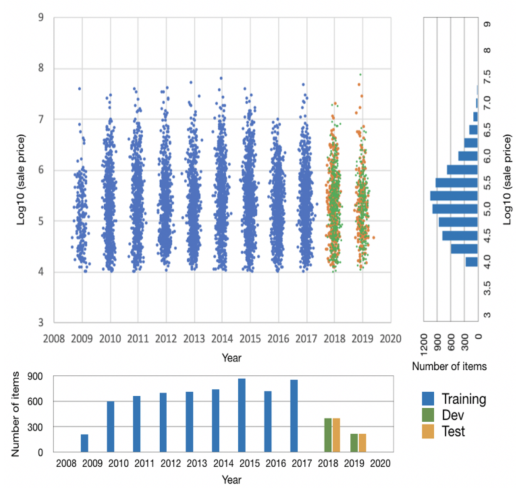

Figure 3: Description of dataset

4 Methods

4.1 Baseline model: Linear regression model using textual/numerical data

Our first baseline is multivariate linear regression model using textual/numerical data (hereafter "Linear"). This model can be understood as a network with no hidden layers, that consists of a single neuron with no nonlinearity which directly outputs the response.

4.2 Multilayer perception model using textual/numerical data

In our multilayer perception model using textual/numerical data (hereafter "MLP") includes 11 hidden layers, with 64 neuron units. We use the rectified linear units (ReLU) function for hidden activations and a linear function for the output activation.

4.3 Convolutional neural network models using image data

We implemented a custom-built convolutional neural network (hereafter "custom CNN") using a mixture of Conv2D, batchNormalization, pooling, and dropout layers. We train the custom CNN model from scratch on our dataset, across various parameters to optimize the performance. We also applied transfer learning to the pre-trained ImageNet image recognition network ResNet50, InceptionV3, and VGG-16, by removing the last softmax and dense layer, fixing the remaining layers.

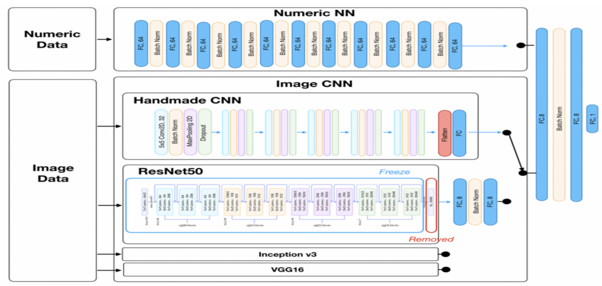

Figure 4: Model architecture

4.4 Fused model using bimodal data

Lastly, we design a fused model using bimodal data of textual/numerical features and images (See Figure 4). We concatenate the output of the last fully connected layer of the MLP model and the CNN models, and feed into the two hidden fully connected layers using ReLU activations followed by the linear activation output neuron (for the fused model using custom CNN, "MLP+CNN", for the fused model using ResNet50, "MLP+ResNet50", for the fused model using InceptionV3, "MLP+InceptionV3", for the fused model using VGG16, "MLP+VGG16").

4.5 Hyperparameter tuning

Hyperparameters were chosen by maximizing R2 on the test subset. We obtain the highest performance with the stochastic gradient descent optimizer ("sgd"). We also use kernel regularization (L2) and activity regularization (L1) for fully connected layers and dropouts for the convolutional layers during training to help reduce overfitting. For parameter selection, we first tune learning rates over a log scaled and then conduct a random search in a narrower range to find the best candidates. We then tune the minibatch size and number of hidden units. The momentum and learning rate decay were set to 0.99 and 0.01 respectively, since they did not seem to meaningfully affect the performance.Figure 5a shows that the performance of the MLP+ResNet50 significantly improve after the hyperparameter tuning. Figure 5b also shows the MLP+ResNet50 using the minibatch size of 128 achieve the highest R2. Table 2 summarizes the test set R2 of the MLP+ResNet50 when tuning the hyperparameters in the order of learning rate, minibatch size, the number of units in fully connected layers after the convolutional layers (num units1), and the number of units in fully connected layers after concatenating the output of the ResNet50 and the output of the MLP (num units2).

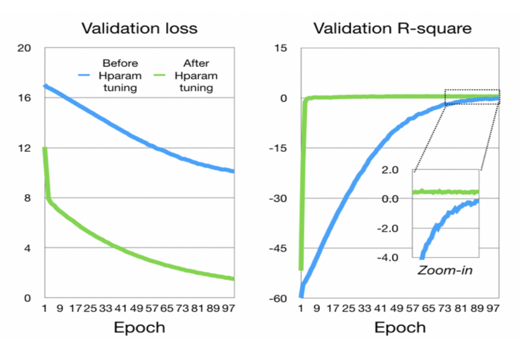

Figure 5a: Improvement of loss and R2 after hyperparameter tuning

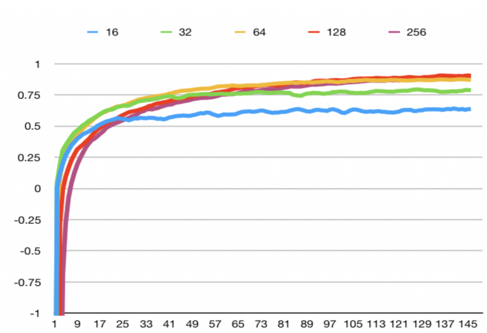

Figure 5b: Validation R^2 over different batchsizes

learning rate	batch size	num units1	num units2	R^2
0.001	128	16	16	**0.4728**
0.01				0.4577
0.1				0.3514
0.001	16	16	16	0.5123
	32			0.5144
	64			0.4964
	128			**0.5423**
	256			0.4352
0.001	128	**8**	16	**0.4658**
		16		0.4386
		32		0.4596
0.001	128	8	**8**	**0.5431**
			16	0.3547
			32	0.4757

Table 1: Hyperparameter tuning results of MLP+ResNet50

Model	RMSE	R^2
Linear	0.3801	0.4151
MLP	0.3495	0.4874
CNN	0.682	-0.15
ResNet50	0.4529	0.1425
InceptionV3	0.511	-0.0356
VGG16	0.4705	0.1129
MLP+ CNN	0.329	0.556
MLP+ ResNet50	0.3189	0.5714
MLP+ InceptionV3	0.336	0.514
MLP+ VGG16	**0.3054**	**0.6177**

Table 2: Results of experiments

4.6 Loss function and Metrics

We use Root Mean Square Error (RMSE) as our loss function and two standard regression metrics to report the efficacy of our proposed method in estimating hammer prices: (1) RMSE and (2) Coefficient of Determination, also known as R2.

5 Results and Discussion

5.1 Results

The Table 2 show the results of the best-performing architecture of each model in RMSE and R2. The MLP model and the fused model outperform the baseline, while the CNN models using only images significantly underperform than the baseline linear regression model. ResNet50 achieves best metrics among models using only images, although its explanatory power is limited. All the models using bimodal data outperform the models using unimodal data of either textual/numerical data or image data. The MLP+VGG16 model achieves the best results in both metrics.

Figure 6 shows the scatter-plots between the actual and the predicted log hammer price for various models, using the best-performing architecture. The result shows clearly that the models using textual/numerical attributes as one of the inputs achieve much higher correlation than the CNN models which use only image data.

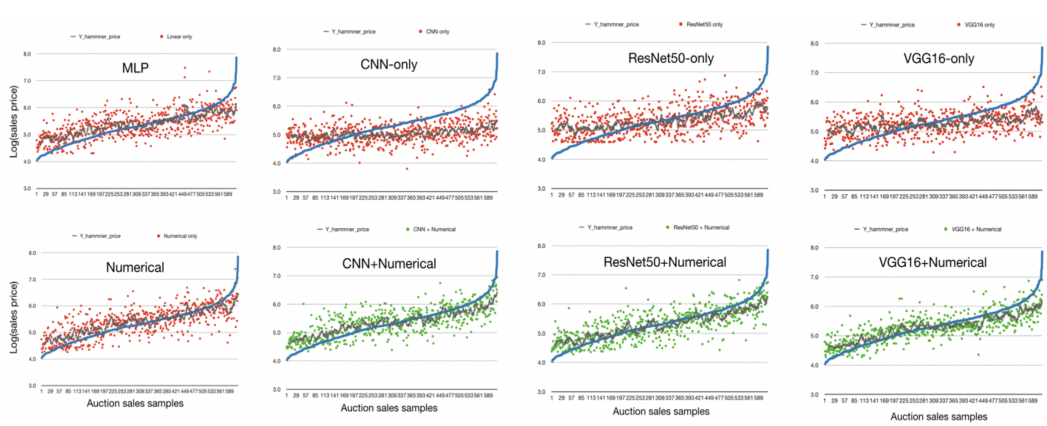

Figure 6: Scatterplot of predicted hammer prices

5.2 Comparing with auction house estimates

As explained earlier, our models' primary aim was to predict art auction hammer prices without considering auction houses' estimates. In addition, we wanted to see what value our model can add after the pre-sale estimations are provided by auction house.

Table 3 shows the results of the regression model with only auction house estimates and adding our valuation models along with the auction house estimate to the regression model. The bottom row in column 2-9 show the increase in R2 relative to the model in column (1) scaled by the hammer price variation unexplained (1-R2) by the benchmark model.

Auction house estimates explain about 93.8The results show that the multilayer perception model and VGG16-transfer learning model and the fused models can help in predicting hammer price outcomes conditional on pre-sale estimates. From the last column, we can see that adding a fused model of MLP+VGG16 would explain 3.2% more of variation of hammer price. Since we used the logged (10) hammer price as our dependent variable, even a slight additional explanatory power might be translated to a large economic value.

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
Automated valuation		Linear	MLP	CNN	ResNet50	VGG16	MLP+ CNN	MLP+ ResNet50	MLP+ VGG16
Auction house	0.9549***	0.9594***	0.9181***	0.9558***	0.9501***	0.9476***	0.9286***	0.9174***	0.9062***
	(0.010)	(0.014)	(0.014)	(0.010)	(0.011)	(0.011)	(0.015)	(0.015)	(0.016)
Automated valuation		-0.0085	0.0581***	-0.0055	0.0143	0.0301	0.0437*	0.0586**	0.0814***
		(0.018)	(0.017)	(0.019)	(0.016)	(0.018)	(0.020)	(0.018)	(0.021)
Constant	0.2571***	0.2789***	0.1382*	0.2796**	0.2068**	0.1369	0.1617*	0.1447*	0.0847
	(0.053)	(0.070)	(0.062)	(0.096)	(0.077)	(0.089)	(0.068)	(0.063)	(0.069)
N	615	615	615	615	615	615	615	615	615
R^2	0.938	0.938	0.940	0.938	0.938	0.939	0.939	0.939	0.940
Increase in R^2 as % of variation unexplained by (1)		0.0%	3.2%	0.0%	0.0%	1.6%	1.6%	1.6%	3.2%

***p<0.001, **p<0.1, *p<0.5

Table 3: Improvement in predictive power over auction house estimates

5.3 Discussion

The results above show that the fused models utilizing bimodal data of textual/numerical data and image data outperforms both the models using only textual/numerical data and the models using only image data. Interestingly, the convolutional neural network models using only images performed relatively poorly to other types of models. This may suggest that, quite ironically, the visual aspect of an artwork alone plays an insignificant role in determining the price of an artwork. One caveat is that our preprocessed image data, which were resized as 224 x 224 pixels, do not include any information about the artworks' width, height, or medium (canvas, paper, oil, etc.). If people appreciate the size, ratio, and medium of the artworks along with its color composition, the image data alone would provide only part of the information about visual appeal of artworks.

6 Conclusion and future work

In this work, we collected an art auction dataset spanning the last 10 years and propose a model which estimates the log hammer price from two separate types of input data: textual/numerical features and images. We find that the model including both types of data performs better than the model without image features. Although the hammer price seems to strongly correlate with the auction house pre-sale estimates, our bimodal models help explain the additional variance of hammer prices.

Additional research is also needed to improve the transfer learning of pretrained convolutional neural networks and to visually explain the convolutional neural network part of the model. For example, with more resources, we could train some of the last convolutional layers of pretrained network or extract feature vector from an earlier convolutional layer in the pretrained networks. Moreover, using visual explanation methods such as Grad-CAM [12] and DeepLift [13] to identify the regions of the image that gets activated for higher or lower hammer prices.

7 Contribution

The authors had equal contribution, working together for data-collection, developing and training models, and preparing for the report, poster, and presentation.

8 Code

The code for our model can be viewed and downloaded from Github using the following link: https://gist.github.com/pakejin/1c801877e744242a9b8e5deca0a1c57e

References

[1] Devin Liu and Doug Woodham, Using AI to Predict Rothko Paintings' Auction Prices, May 7, 2019, Artsy, https://www.artsy.net/article/artsy-editorial-ai-predict-rothko-paintings-auction-prices

[2] Aubry, M., Kräussl, R., Manso, G., Spaenjers, C. (2019). Machines and Masterpieces: Predicting Prices in the Art Auction Market. Available at SSRN 3347175.

[3] Alexander Verge and Ishaan Singal, State-of-the-Art: End to End Deep Learning for Art Appraisal. Spring 2019 CS230.

[4] Tan, W. R., Chan, C. S., Aguirre, H. E., Tanaka, K. (2016). Ceci n'est pas une pipe: A deep convolutional network for fine-art paintings classification. In Image processing (ICIP), 2016 IEEE international conference on (pp. 3703–3707). IEEE. doi:10.1109/ICIP.2016.7533051.

[5] Hentschel, C., Wiradarma, T. P., Sack, H. (2016). Fine tuning CNNs with scarce training data adapting ImageNet to art epoch classification. In Image processing (ICIP), 2016 IEEE international conference on (pp. 3693–3697). IEEE. doi:10.1109/ ICIP.2016.7533049.

[6] Lecoutre, A., Negrevergne, B., Yger, F. (2017). Recognizing art style automatically in painting with deep learning. In Asian conference on machine learning (pp. 327–342).

[7] van Noord, N., Postma, E. (2017). Learning scale-variant and scale-invariant features for deep image classification. Pattern Recognition, 61, 583–592. doi:10. 1016/j.patcog.2016.06.005.

[8] Bessinger Z, Jacobs N (2016) Quantifying curb appeal. In: 2016 IEEE International Conference on Image Processing (ICIP), pp 4388–4392

[9Th] Bency AJ, Rallapalli S, Ganti RK, Srivatsa M, Manjunath BS (2017) Beyond spatial auto-regressive models: Predicting housing prices with satellite imagery. In: 2017 IEEE winter conference on applications of computer vision (WACV), pp 320–329

[10] You Q, Pang R, Cao L, Luo J (2017) Image-based appraisal of real estate properties. IEEE Trans Multimed 19(12):2751–2759

[11] Liu X, Xu Q, Yang J, Thalman J, Yan S, Luo J (2018) Learning multi-instance deep ranking and regression network for visual house appraisal. IEEE Tran Knowl Data En 30(8):1496–1506. https://doi.org/10.1109/tkde.2018.2791611

[12] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2016. Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization. arXiv e-prints, Article arXiv:1610.02391 (Oct 2016), arXiv:1610.02391 pages. arXiv:cs.CV/1610.02391

[13] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. 2017. Learning Important Features Through Propagating Activation Differences. (2017). arXiv:cs.CV/1704.02685