ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/343856615

Mixing Deep Visual and Textual Features for Image Regression

Chapter · August 2020

DOI: 10.1007/978-3-030-55180-3_57

CITATIONS
0

READS
74

2 authors, including:

Youshan Zhang
Cornell University

53 PUBLICATIONS **209** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project Geodesic feature learning on Riemannian manifolds View project

Project Automating management of teat tissue condition in dairy cows through machine learning View project

All content following this page was uploaded by Youshan Zhang on 18 May 2021.

The user has requested enhancement of the downloaded file.

Mixing Deep Visual and Textual Features for Image Regression

Yuying Wu[1] and Youshan Zhang[2]

[1] Liaoning University, Econometrics, Shenyang, Liaoning, China
Wuyyyyy@outlook.com
[2] Lehigh University, Computer Science and Engineering, Bethlehem, PA, USA
yoz217@lehigh.edu

Abstract. Deep learning has been widely applied in the regression problem. However, little work addressed both visual and textual features in one unit frame. In this paper, we are the first to consider the deep feature, shallow convolutional neural network (CNN) feature, and textual feature in one unit deep neural network. Specifically, we propose a mixing deep visual and textual features model (MVTs) to combine all three features in one architecture, which enables the model to predict the house price. To train our model, we also collected large scale data from Los Angeles of California state, USA, which contains both visual images and textual attributes of 1000 houses. Extensive experiments show that our model achieves higher performance than state of the art.

Keywords: Deep feature, Textual feature, Convolutional neural network, House price prediction

1 Introduction

In recent years, the real estate market continuously attracts attentions of government and public. The real estate industry gradually becomes an essential pillar of national economy in each country and a necessary part in the development of federal and local economies [1]. Although it has made a great contribution to the growth of economy, it also brings lots of issues, such as, the high-speed rising of housing prices has made many low-income groups in a dilemma since they cannot afford it, and the contradiction between the rich and the poor becomes increasingly prominent. Housing price becomes the focus of various social problems, and it can threaten the sustainable development of society. Moreover, the oscillation of house prices not only affects the value of asset portfolio for most households but also affects the profitability of financial institutions and the surroundings of financial system [2]. The house renovation and construction can also cause difficulty for householder.

The prediction of house price is beneficial for individual investors and they can more intuitively analyze the real estate market. More importantly, it can help the government to regulate the real estate market reasonably and make the house price more reasonable and the development of the real estate market more stable. Therefore, it is important and necessary to predict the house price with high accuracy.

However, due to the low liquidity and heterogeneity of the real estate market from both physical and geographical perspective, and there are many factors that can affect the house price, they bring great challenges of house price prediction. Although previous studies analyzed the relationship among social, economic, demographic changes and characteristic house prices, quantifying such a correlation is not enough to accurately estimate or predict house prices [3, 4]. In addition, most previous work focused on textual features of house price prediction [5, 6]. Although Ahmed et al. 2016 first combined the visual and textual features, they have a limited sample size and less textual attributes [2].

In this paper, we address these challenges within one unit deep neural network frame. Specifically, we address the house price prediction issue through considering both deep visual and textual features with exploring several pre-trained deep neural networks.

Our contributions are three-fold:

- We provide a large scale house price prediction dataset, it combines deep visual and ten textual features. The dataset can be of significant value in future research for the community.
- We are the first to propose one unit deep neural network to consider the deep feature, shallow CNN feature, and textual feature. The designed architecture can output the house price with less mean absolute percentage error.
- We are the first to propose a novel absolute mean discrepancy loss to insure that mean predicted price closes to actual mean house price.

Extensive experiments on competitive benchmark housing datasets show that ResNet50 is the best neural network for feature extraction in house price prediction problem.

2 Related work

House price prediction is a regression problem that aims to output house price given either visual or textual features as input. In the past decade, many methods are proposed for regression problem, such as support vector regression (SVR) [2, 7, 8], time-series forecasting [9, 10], artificial neural network [11, 12], deep neural networks [13–15], etc.

There are also many studies addressed house price prediction problem. Limsombunchai et al. proposed an artificial neural network (ANN) to predict house prices using 200 house information in Christchurch, New Zealand, and they found that their ANN model significantly improved its predictive power than Hedonic price model [11]. Khamis et al. compared the performance between the multiple linear regression (MLR) model and neural network (NN) model for predicting the house price. They concluded that the prediction of the house price in the NN model is closer to the ground truth, comparing with the MLR model. Nevertheless, there are some limitations in their model. Firstly, the used house price is not the actual sale price but the estimated price. Secondly, this study considered only the specific year's information on the houses and ignored the time effect of the house price. Finally, the house price could be affected by some other economic factors that are not included in the estimation [5]. Ahmed et al. collected 535 sample houses in California, USA, which contained both visual images and textual features. They compared the NN and support vector machine methods on the effect of predicting house prices, and they observed that using NN can achieve better results than SVM model. In addition, they found that compared to the individual textual features, the combination of both visual and textual features produced better estimation accuracy. However, they had less training data, which may lead to overfitting; they also used a lower level feature, which is from SURF feature extractor, and deeper neural networks can be applied to extract features [2]. Nguyen identified the four most essential attributes in housing price prediction across three counties that are assessment, comparable houses' sold price, listed price, and the number of bathrooms [16]. Varma et al. considered another seven features that affect house price (area, number of bedrooms, number of bathrooms, type of flooring, lift availability, parking availability, and furnishing condition). Differing from the traditional hedonic house price model, they focused on spatial econometrics and cross-validates the out-of-sample prediction performance of 14 competitive models, which contributes to house price prediction. Their results indicated that a nonlinear model, which considered the spatial heterogeneity and flexible nonlinear relationship between a specific individual or regional characteristics of a house and its price is the best strategy for predicting house prices [17]. Wang et al. designed a multilayer feedforward neural network with a memory resistor, which realized automatic online training. Memristor weights of the ANN can be adjusted by the BP algorithm to build up a regression model simultaneously. The neural network is trained and predicted by using the housing price samples of several cities in Boston, USA, and the predicted results close to the target data [18]. Gao et al. noted that the location of the buildings is the most critical attribute that affects the house price, and they defined and captured a fine-grained location profile powered by a diverse range of location data sources [19].

However, none of these work addressed both visual and textual features in one unit deep neural network. First of all, for the visual feature in image regression, the community focused on feature representation from the hand-crafted features to the extracted features from the deep pre-trained neural networks. Before the rise of the convolution neural network, hand-crafted features were well used (e.g., SURF [2]), and since deep features can substantially improve the performance of domain adaption, they are now widely used in both classification [20, 21] and regression problem [14,15,22]. Secondly, our work include more influential factors of house price. Furthermore, we investigate both deep visual and textual features on house price estimation in one unit frame.

The rest of this paper is organized as follows. Sec. 3 describes our collected house data. Sec. 4 first defines the house price prediction problem and then describes the proposed network architecture and objective function for solving the optimization problem. Sec. 5 reports the experimental results. We further discuss the results in Sec. 6 and conclude in Sec. 7.

3 Los Angeles housing dataset

We collected house images and their associated attributes from www.zillow.com. The data consists of 1,000 sample houses from Los Angeles, California state in the United States. We collected both visual and textual information from houses. For visual features, we saved most representative images of houses, that each house has four different views: frontal, bedroom, bathroom, and kitchen. Fig. 1 shows one house image from four different views.

For textual features, there are nine factors in our collected data (number of bedrooms, number of bathrooms, area, zip code, year build, year renovation, house type, parking space, and sun number). Therefore, we have a total of ten attributes on one house, including the house price. Tab. 1 lists the detailed information of the collected dataset (mode is reported for Zipcode attribute).

We also extract deep visual features from pre-trained models. Fig. 2 shows the extracted feature vectors using pre-trained ResNet50 model. It also presents both visual and textual features of the house.

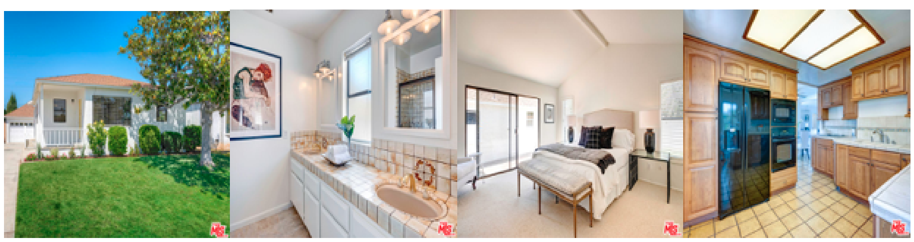
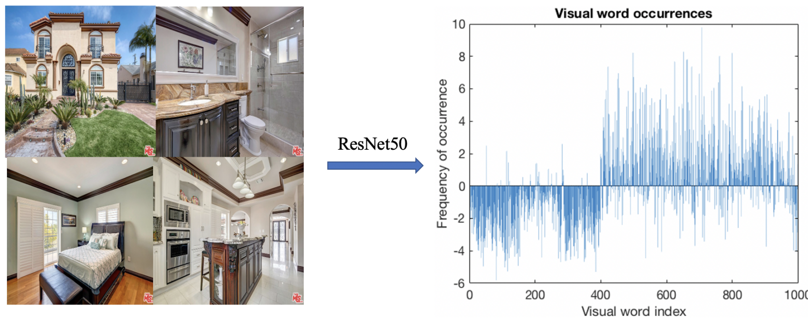


Fig. 1: One sample house image from (www.zillow.com), it is represented by four different views: frontal, bathroom, bedroom, and kitchen.

4 Methods

4.1 Problem and notation

For house price prediction, it is a regression problem. Given house data (including visual images and textual factors) $\mathcal{X}$ with its associated house price $\mathcal{Y}$; our ultimate goal is to predict the house price ($\mathcal{Y}'$) by given $\mathcal{X}$, which closes to the actual house price $\mathcal{Y}$, that is to minimize the difference between the $\mathcal{Y}'$ and $\mathcal{Y}$.



No. Bedroom	No. Bathroom	Area	Zip code	Year build	Year renovation	Type	Parking spaces	Sun Number	Price
5	3	3374	90045	2007	2015	0	3	90.94	1.959M

Fig. 2: Sample features from one house. The top left is the original image, and top right is the extracted feature from the ResNet50 model, which shows the frequency of occurrence of features. In the bottom, we list the ten attributes of house (Area: square ft, and there are five types of the house (0: single family, 1: townhouse, 2: condo, 3: multi-family, 4: apartment), M: million).

Table 1: Some detailed statistics of Los Angeles housing dataset

Details	Minimum	Maximum	Average
No. bedrooms	1	32	3.52
No. bathrooms	1	12	2.9455
Area (sqft)	528	31520	2211.6
Zipcode	90020	90292	90045 (mode)
Year build	1902	2019	1952.4
Parking space	0	8	2.162
Sun number	22.56	95.75	86.5714
House price ($)	0.2295 M	10.9 M	1.6813 M

4.2 Mixing deep visual and textual features model (MVTs)

The architecture of our proposed MVTs model is shown in Fig. 3, we have three modules in the model. First of all, we extract deep feature from a pre-trained ResNet50 model. Second, we design the shallow CNN module to extract the shallow feature from raw images. Third, we add textual feature, that includes the number of bedrooms, the number of bathrooms, area, zipcode, year build, year renovation, house type, parking space, and sun number. We then concatenate all three modules together to form our MVTs model[3].

Our model minimizes the following objective function:

$$\mathcal{L}(\mathcal{X}, \mathcal{Y}) = \arg\min \quad \mathcal{L}_{\mathcal{M}}(\mathcal{X}, \mathcal{Y}) + \alpha\ \mathcal{L}_{\mathcal{A}}(\mathcal{X}, \mathcal{Y}) \tag{1}$$

[3] Dataset is available at https://github.com/heaventian93/House_Price.

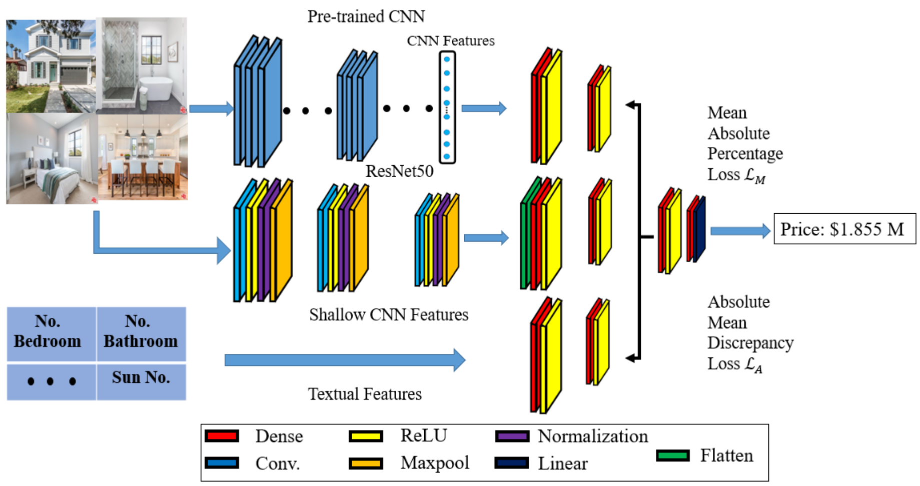


Fig. 3: The architecture of our proposed mixing deep visual and textual features for house price prediction. There are three modules in the model: deep feature module, shallow CNN feature module and textual feature module. In deep feature module, the feature is obtained from the pre-trained ResNet50 neural network. The shallow CNN feature module directly gets features from raw images via three repeated blocks. The textual feature module majorly contains two layers. The model consists of two different loss functions, including mean absolute percentage error loss and absolute mean discrepancy loss. The mean absolute percentage error loss measures mean percentage difference between predicted price and actual house price, and absolute mean discrepancy loss ensures that the mean predicted price approximates the mean house price.

where $\mathcal{L_M}$ is the mean absolute percentage error loss, $\mathcal{L_A}$ is the proposed absolute mean discrepancy loss and α is the balance factor between two loss functions. Specifically ($\bar{\cdot}$ is the mean house price),

$$\mathcal{L_M} = \frac{1}{N}\sum_{i=1}^{N} |\frac{\mathcal{Y}_i - \mathcal{Y}_i'}{\mathcal{Y}_i}|, \quad \mathcal{L_A} = |\frac{\overline{\mathcal{Y}} - \overline{\mathcal{Y}'}}{\overline{\mathcal{Y}}}|. \tag{2}$$

In deep feature module, we followed the protocol from [20,21,23,24], that all features are extracted from the last fully connected layer. Thus the final output of one image becomes one vector 1×1000. Feature extraction is implemented via two steps: (1) rescale the image into different input sizes of pre-trained neural networks; (2) extract the feature from the last fully connected layer. After feature extraction, it follows by two repeated blocks. In each block, it has a dense layer and a “ReLu” activation layer. The units of the dense layer are 1000 and 4, respectively.

In the shallow CNN feature module, it first includes three repeated blocks, and each block has four layers: convolutional 2d layer, “ReLu” activation layer, batch normalization layer and maxpooling layer. In the convolutional 2d layer, the kernel size is (3, 3) in all three blocks and the filter size is 16, 32, 64, respectively. The pool size in maxpooling layer is (2, 2). After a flatten layer, it then followed by the same block as deep feature module, but the units of the dense layer are 16 and 4.

In the last textual feature module, the input includes nine factors of the house price (number of bedrooms, number of bathrooms, area, zipcode, year build, year renovation, house type, parking space, and sun number). It also followed by the same block as deep feature module, but the units of the dense layer are 8 and 4 since it has fewer features than the other two modules.

We then concatenate all three modules together, it majorly consists of four layers: a dense layer with units number of 4, the "ReLu" activation layer, a dense layer with units number of 1 and finally ends with a "linear" activation layer. Therefore, it can output the house price in the last layer.

There are three obvious advantages of the proposed mixing deep visual and textual features (MVTs) model. First of all, we consider the mixing features from both deep visual and textual features. By contacting all three features, our MVTs model can take advantage of all these features and achieve high performance. In addition, we propose a novel loss function: absolute mean discrepancy loss. It effectively measures the mean difference between the predicted house price and real house price. The performance of this loss function is shown in Sec. 6. Furthermore, the designed Dense and Activation block can successfully leverage deep visual and textual features to predict the house price without overfitting problems. If there are more layers added (e.g., repeated Dense, ReLU, Normalization, and Dropout several times), the final error will be first stuck in a global minimum number and then increased.

5 Results

5.1 Experimental setting

Our experiment is based on Keras and runs on top of TensorFlow. In addition, our network is trained on a graphics processor NVIDIA Geforce 1080 Ti equipped with 11Gb of memory on a 16 GB RAM Alienware computer to exploit its computational speed. The network parameters are set to:

1. Batch size: 32
2. α: 10
3. Iterations: 60

We also randomly split our data into training and testing data, and there are 500 samples in both training and testing stages.

5.2 Performance evaluation

Fig. 4 compares the predicted house prices with the actual house prices. We find that the predicted house prices close to the actual house prices, which represents our proposed MVTs is suitable for house price prediction.

Fig. 4: Two examples of house price prediction. The actual price and predicted price are labeled in house images (M: million).

To show the effeteness of our proposed MVTs models on house price prediction, we report both the mean absolute percentage error (MAPE) and the mean square error (MSE) of the testing data (500 samples). MAPE is defined in Eq. 2, and MSE is measured by Eq. 3. Notice that, the smaller of these two metrics, the better the performance of the model.

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y_i})^2, \tag{3}$$

where y_i is the actual house price, and $\hat{y_i}$ is predicted house price.

As shown in Tab. 2, it lists the MAPE and MSE of different settings of our model. We first run each feature individually to predict the house price in testing data and then report the result of combination of all three features. Across all three different features, the textual feature achieves a higher performance than the other two features. Although textual features have fewer features than deep features, it contains more useful information than just simply the exterior and interior of a house such as the location and area. Moreover, deep feature has a better result than shallow CNN feature since the deep neural network is trained with millions of images. Therefore, we can extract a better feature than the shallow CNN feature. However, a combination of all three features achieves the highest performance (lowest MAPE and MSE). We also observe that all four different variant results are better than the previous SVR model [2].

Furthermore, MAPE is a better metric than MSE. For shallow CNN feature and deep feature, the normalized MSE closes to each other, but there is a significant difference between the two MAPE scores. To further validate this observation, we show the effectiveness of two metrics in Tab. 3. we find that the MSE is significantly changed if the data is normalized. However, MAPE still keeps similar as before. As shown in Tab. 3, it lists the number of iterations and time of different data processing. It suggests that normalized the house price leads to less time, smaller number of iterations and higher performance than the unnormalized data. Therefore, our MVTs model will benefit from the normalized house price.

Table 2: House price results prediction

	Deep feature	Textual feature	Shallow feature	MVTs	SVR
MAPE (%)	42.06	20.93	60.77	**18.01**	72.97
MSE	0.00998	0.00546	0.08012	**0.00477**	0.1304

Table 3: Iterations and time of different data processing

	No. Iterations	times (s)	MAPE	MSE
Unnormalized price	300	2405	34.45	8.87908e11
Normalized price	**60**	**535**	**18.45**	**0.003438**

5.3 Our model on Ahmed et al. 2016 housing dataset

To show the applicability of our model, we also test our model using previous dataset. The housing data is from [2], it consists of 535 house images and we use 75% as training and 25% as the testing. Our model is significantly better than the SVR model as shown in Tab. 4.

Table 4: House price prediction results of [2]

	MVTs	SVR
MAPE (%)	**34.29**	56.09
MSE (norm.)	**0.01734**	0.1494

6 Discussion

From above results, our proposed MVTs model is able to predict the house price with lower MAPE and MSE values, which demonstrate the robustness of our model. Also, we compare the effectiveness of two different loss functions in Tab. 5, we find that the performance of single $\mathcal{L}_{\mathcal{A}}$ loss is worse than single $\mathcal{L}_{\mathcal{M}}$ loss since $\mathcal{L}_{\mathcal{A}}$ only measures that mean average of absolute percentage error. However, the combination of these two loss functions achieves the highest performance.

Table 5: House price results with different loss functions

	$\mathcal{L}_{\mathcal{M}}$	$\mathcal{L}_{\mathcal{A}}$	Combination
MAPE (%)	22.10	25.77	**18.01**
MSE	0.00447	0.00589	**0.00477**

Table 6: MAPE and MSE of predicted house price with minimizing $\mathcal{L}_{\mathcal{M}}$ and $\mathcal{L}_{\mathcal{A}}$ using sixteen pre-trained neural networks

Task	MAPE (%)	MSE
SqueezeNet	42.09	0.01005
AlexNet	49.66	0.01095
GoogleNet	51.33	0.00985
ShuffleNet	47.76	0.00943
ResNet18	41.10	0.00884
Vgg16	46.21	0.00869
Vgg19	49.69	0.00942
MobileNetv2	48.18	0.00946
NasnetMobile	48.32	0.00925
ResNet50	**40.58**	**0.00652**
ResNet101	43.92	0.00794
DenseNet201	43.89	0.00830
Inceptionv3	45.76	0.00818
Xception	47.13	0.00882
InceptionresNetv2	49.17	0.00851
NasnetLarge	44.14	0.00869

In addition, we explore how different pre-trained models affected the final results. We examine sixteen well trained models. Specifically, these sixteen neural networks are Squeezenet [25], Alexnet [26], Googlenet [27], Shufflenet [28], Resnet18 [29], Vgg16 [30], Vgg19 [30], Mobilenetv2 [31], Nasnetmobile [32], Resnet50 [29], Resnet101 [29], Densenet201 [33], Inceptionv3 [34], Xception [35], Inceptionresnetv2 [36], Nasnetlarge [32]. As shown in Tab. 6 and Fig. 5, the ResNet50 surprisingly achieves higher performance than the other models. We further explore the relationship between the top-1 accuracy of sixteen pre-trained neural networks and the MAPE value in Fig. 6. However, we find that the correlation score and R^2 value are relatively smaller. The smaller of these values, the correlation is less [15]. Therefore, we cannot observe a significant trend that how to choose the best pre-trained model via top-1 accuracy. Although we do not know the underlying mechanisms, ReseNet50 is useful in the house price prediction problem.

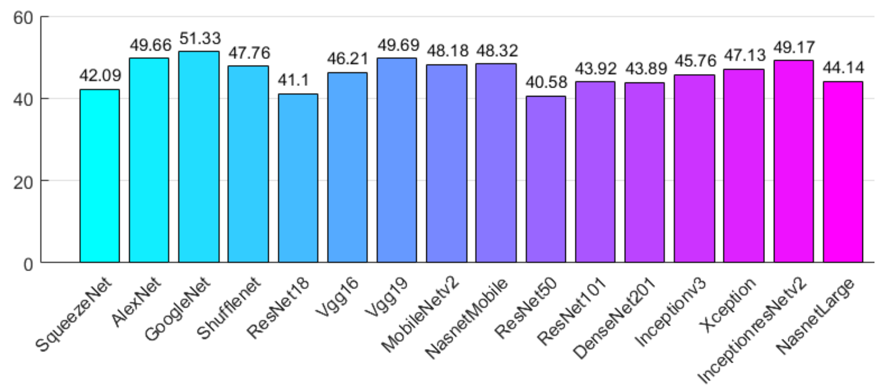


Fig. 5: Bar plot of MAPE values with extracted features from sixteen pre-trained models (x-axis is the order of Top-1 accuracy on ImageNet model).

Tab. 2 shows that the performance of image features (both deep feature and shallow CNN feature) is significantly lower than the single textual feature. What causes this problem? Actually, the price of house can be affected by lots of factors such as the factors in the textual features. Especially, location is one of the most important factors that affects house prices. However, house images come from four different views, and it only considers the exterior and interior of a house. Therefore, we expect the performance of single deep feature and shallow CNN feature is worse than the single textual feature.

We clearly observe several advantages of our proposed mixing deep visual and textual features model. First of all, our model considers three different features: deep feature from the pre-trained model, shallow CNN feature, and textual feature. Secondly, we propose a novel loss function called absolute mean discrepancy loss, and it can jointly reduce the difference between predicted house price and actual house price.

One interesting phenomenon is that we find the performance of our new data is better than the results of the dataset. One reason is that we collected more images (1000 samples) with 500 samples in the training stage. The second reason is that we collected more factors which can affect the house price. We have nine factors in new house data, while there are only four factors in Ahmed et al. 2016 dataset.

However, our model also has some limitations. Firstly, the lowest MAPE score of testing data is higher than 18%, which implied that the predicted house price can be 18% away from the real price. Therefore, our model can still be improved. Secondly, the MAPE scores of both visual features (deep feature and shallow CNN feature) are relatively lower. This further indicates the deep architecture of images needs to be further investigated.

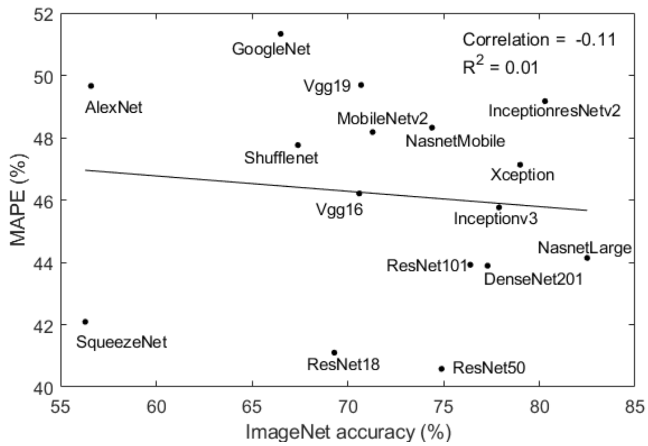


Fig. 6: Correlation between ImageNet accuracy of sixteen pre-trained models and MAPE in the house price prediction.

7 Conclusion

In this paper, we are the first to consider both visual and textual features for house price prediction in one unit architecture. The proposed mixing deep visual and textual features model shows its robustness in house price prediction. Our experiments show that aggregating both visual and textual attributes yielded better prediction results than the deep feature, shallow CNN feature, and textual feature alone. In addition, ResNet50 is the best pre-train model for feature extraction in our dataset.

References

1. Yarui Li and David J Leatham. Forecasting housing prices: dynamic factor model versus lbvar model. Technical report, 2010.
2. Eman Ahmed and Mohamed Moustafa. House price estimation from visual and textual features. *arXiv preprint arXiv:1609.08399*, 2016.
3. John M Quigley. Real estate prices and economic cycles. 2002.
4. Kostas Tsatsaronis and Haibin Zhu. What drives housing price dynamics: cross-country evidence. *BIS Quarterly Review, March*, 2004.
5. Azme Bin Khamis and Nur Khalidah Khalilah Binti Kamarudin. Comparative study on estimate house price using statistical and neural network model. *International Journal of Scientific & Technology Research*, 3(12):126–131, 2014.
6. Aaron Ng and Marc Deisenroth. Machine learning for a london housing price prediction mobile application. In *Imperial College London*. 2015.
7. Chih-Hung Wu, Chi-Hua Li, I-Ching Fang, Chin-Chia Hsu, Wei-Ting Lin, and Chia-Hsiang Wu. Hybrid genetic-based support vector regression with feng shui theory for appraising real estate price. In *2009 First Asian Conference on Intelligent Information and Database Systems*, pages 295–300. IEEE, 2009.

8. Vasilios Plakandaras, Rangan Gupta, Periklis Gogas, and Theophilos Papadimitriou. Forecasting the us real house price index. *Economic Modelling*, 45:259–267, 2015.
9. Ian D Wilson, Stuart D Paris, J Andrew Ware, and David Harrison Jenkins. Residential property price time series forecasting with neural networks. In *Applications and Innovations in Intelligent Systems IX*, pages 17–28. Springer, 2002.
10. Rangan Gupta and Stephen M Miller. The time-series properties of house prices: A case study of the southern california market. *The Journal of Real Estate Finance and Economics*, 44(3):339–361, 2012.
11. Visit Limsombunchai. House price prediction: hedonic price model vs. artificial neural network. In *New Zealand Agricultural and Resource Economics Society Conference*, pages 25–26, 2004.
12. Youshan Zhang, Liangdong Guo, Qi Li, and Junhui Li. Electricity consumption forecasting method based on mpso-bp neural network model. In *2016 4th International Conference on Electrical & Electronics Engineering and Computer Science (ICEEECS 2016)*. Atlantis Press, 2016.
13. Ryo Akita, Akira Yoshihara, Takashi Matsubara, and Kuniaki Uehara. Deep learning for stock prediction using numerical and textual information. In *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, pages 1–6. IEEE, 2016.
14. Youshan Zhang and Qi Li. A regressive convolution neural network and support vector regression model for electricity consumption forecasting. In *Future of Information and Communication Conference*, pages 33–45. Springer, 2019.
15. Youshan Zhang and Brian D Davison. Shapenet: Age-focused landmark shape prediction with regressive cnn. In *2019 International Conference on Content-Based Multimedia Indexing (CBMI)*, pages 1–6. IEEE, 2019.
16. An Nguyen. Housing price prediction. 2018.
17. Ayush Varma, Abhijit Sarma, Sagar Doshi, and Rohini Nair. House price prediction using machine learning and neural networks. In *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, pages 1936–1939. IEEE, 2018.
18. JJ Wang, SG Hu, XT Zhan, Q Luo, Qi Yu, Zhen Liu, Tu Pei Chen, Y Yin, Sumio Hosaka, and Yang Liu. Predicting house price with a memristor-based artificial neural network. *IEEE Access*, 6:16523–16528, 2018.
19. Guangliang Gao, Zhifeng Bao, Jie Cao, A Kai Qin, Timos Sellis, Zhiang Wu, et al. Location-centered house price prediction: A multi-task learning approach. *arXiv preprint arXiv:1901.01774*, 2019.
20. Youshan Zhang and Brian D Davison. Modified distribution alignment for domain adaptation with pre-trainedinception resnet. *arXiv preprint arXiv:1904.02322*, 2019.
21. Youshan Zhang, Sihong Xie, and Brian D Davison. Transductive learning via improved geodesic sampling. In *Proceedings of the 30th British Machine Vision Conference*, 2019.
22. Hongfa Ding, Youliang Tian, Changgen Peng, Youshan Zhang, and Shuwen Xiang. Inference attacks on genomic privacy with an improved hmm and an rcnn model for unrelated individuals. *Information Sciences*, 512:207–218, 2020.
23. Youshan Zhang, Jon-Patrick Allem, Jennifer Beth Unger, and Tess Boley Cruz. Automated identification of hookahs (waterpipes) on instagram: an application in feature extraction using convolutional neural network and support vector machine classification. *Journal of medical Internet research*, 20(11):e10513, 2018.

24. Youshan Zhang and Brian D Davison. Impact of imagenet model selection on domain adaptation. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision Workshops*, pages 173–182, 2020.
25. Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size. *arXiv preprint arXiv:1602.07360*, 2016.
26. Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.
27. Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.
28. Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6848–6856, 2018.
29. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.
30. Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
31. Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4510–4520, 2018.
32. Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V Le. Learning transferable architectures for scalable image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8697–8710, 2018.
33. Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.
34. Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.
35. François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1251–1258, 2017.
36. Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

View publication stats