International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-2, July 2019

Mixed Data through Multiple Input for Price Prediction with Multilayer Perception and Mini VGG Net

Satyasangram Sahoo, Prem Kumar. B, R. Lakshmi

Abstract**:** ***The multi-input with mixed data modality of the model is based on three folded structure. The first input model is structured by Convolution Network that accepts the images related to the house. The implementation of the network is miniVGGNet design. The network is tested among, which gives a better outcome. The output valued is concatenated eventually with numerical value entry of the same set which is trained and processed by multi-layer perceptron for review the house price of the building. The linear activation is helped to evaluate the predicted value of price after equal dimension merging of convolutional and multi-layer perceptron network.***

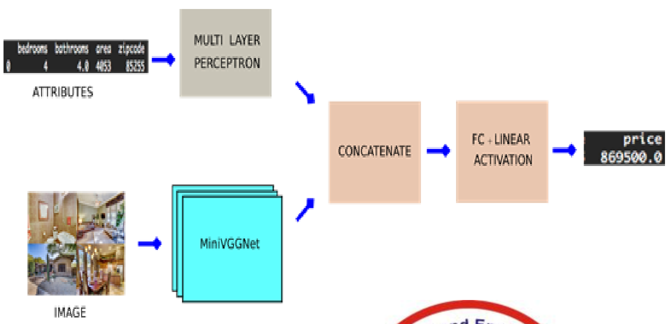
Index Terms**:** ***Multi-Layer Perceptron, MiniVGGNet , Mixed Data , Multi-Input , House attributes ..***

I. INTRODUCTION

Mixed data comprise numerical, categorical, and image can be processed with the aid of various machine learning models. In mixed data [4] numeric valued data or categorical data is used for regression analysis and image data for classification, which in combination help in accurate prediction. Combining mixed data in a single model is much more challenging due to the heterogeneous nature of data. This data heterogeneity warrants different preprocessing steps for an end-to-end single network execution. A complex neural structure is required for accepting the multi-input and multi-output model, which is possible with the help of Keras[8]. Ahmed et al. used the visual and textual feature in their dataset to determine house pricing in various cities [1] for price estimation. The same dataset has been utilized for this article, albeit with a different approach. Lim et al. used a neural network for house price prediction [6] whereas Limsombunchai et.al. Differentiates hedonic price model with Artificial Neural Network model [7]. The deep neural network is immensely beneficial in evaluating the house pricing according to its features and geographical location. The convolutional neural network is very efficient to work on image data while the categorical or numerical dataset is best handled by simple multilayer perceptron [3] with the help of Keras, as Keras is capable of handling multi-input to predict the outcome. Multilayer Perceptron is a feedforward artificial neural network by connecting multiple layers, which update neuron values through backpropagation [3]. Convolutional Neural Network (CNN) is a neural network of layers made up of neurons with learnable weight and bias. The processing of the multi-channeled image is convoluted to a learning pattern which is capable of differentiating data for evaluation. On the other hand, a Multi-Layer Perceptron (MLP) is a logistic regression model which is capable of projecting the input data into space [3] where it becomes linearly separable. Keras is a functional API, which is used for processing high-level neural network to solve the regression problem [5]. The database contained images of the house that covered interior decoration with a full frontal image along with its attributes, which determines the pricing of the structure [4]. The characteristics are fused in a CSV file format which is later proceeded along with the image file of the house. Different real estate companies use different pricing strategies for their construction. House pricing is also influenced by many attributes. Some of the key attributes that affect the price are the number of bedrooms, number of bathrooms, total area of the house, and location of the house.

II. METHOD

The method for analysis is three-fold, image, attributes, followed by concatenation. The image processing section is archived by the convolutional neural network. Multilayer Perceptron handles the numerical or categorical part. In the final section, both the results are combined to give final predicted output with the help of a neural network.



Revised Manuscript Received on 30 July 2019.

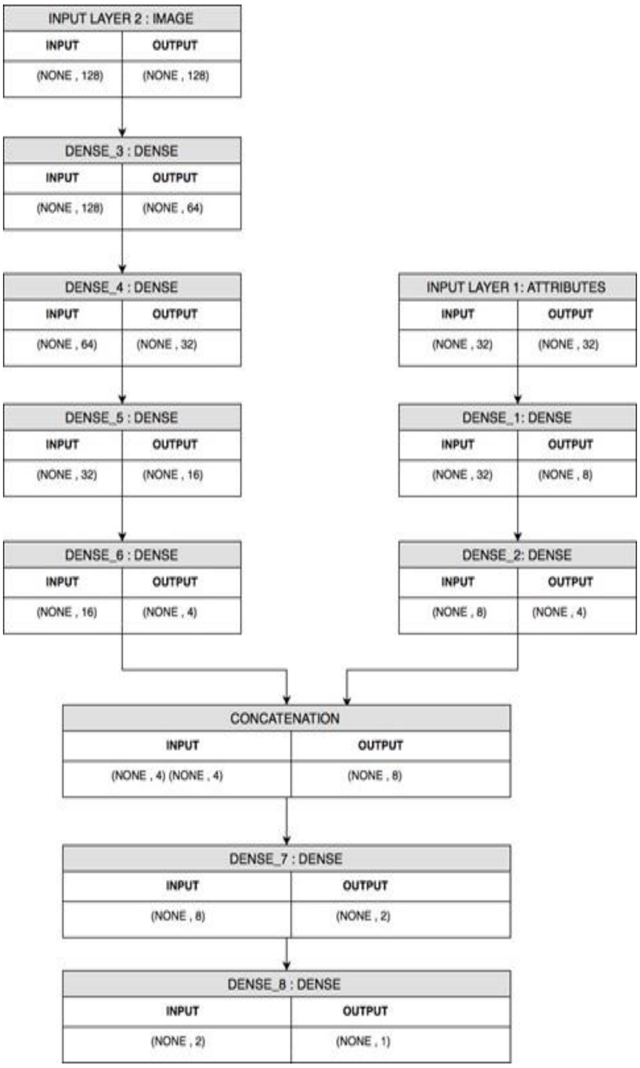
***** Correspondence Author

Satyasangram Sahoo*,Research Scholar, Department of Computer Science , Pondicherry University , Pondicherry,India.

Prem Kumar Borugadda, Research Scholar, Department of Computer Science , Pondicherry University , Pondicherry,India..

Dr R Lakshmi, Assistant Professor, Dept. of Computer Science , Pondicherry university , Pondicherry ,India.

Individual images of the house are insufficient for preprocessing, hence all the images of the house are tiled together into a single montage image which should be arranged correspondingly for all elements of the set [5]. Image set of a unique dataset comprises of a bathroom, a bedroom, a kitchen, and a frontal view of the apartment. These individual images are tiled together as one image set to derive a single value per data. CNN is a good predictive classifier, but in order to use for regression, it should fall under a range of values. All the tiled image data are handled by various convolutional neural networks, while the image set is divided by 255.0, which is the maximum image value, so it is scaled into the range between 0 and 1. Each image is converted into a size of 32×32. The tiled image output has a pre-configured dimension of 64×64, exactly four images can be tiled accordingly to be in the same position for all the dataset.



The neural network section is processed by three different forms of neural network for this experiment. Out of these forms, MiniVGGnet has shown the best outcome for this small dataset network parallelly with Multi-layer Perceptron on the other end. VGGnet is generally applied for substantial scale image recognition [2]. MiniVGG is the smaller version of the VGGnet, which is comparable to any image size. The ShallowNet and LeNet have a structure like CONV => ReLU => POOL, but the MiniVGGNet enters into a little depth from previous by stacking CONV => ReLU before entering single POOL layer. Before entering into MiniVGGNet, the image size is converted from 128 dimension vector into 64 dimension vector.

Algorithm 1: Algorithm for Multiple Input Mixed Data Processing

```
Input A ← House Attribute File
Input B ← Image
Resize : Image(32,32)← Input B;
Function: Append
initialization : Input Image[i];
Where i,j ← 0;
N = No of image/4;
while j ≤ (N-1) do
    while i ≤ 3 do
        Output Image(64,64) ← Input Image[i];
        Image ← Output Image / 255.0;
    end
    return Image;
end
Max Price ← Max of Input A[Price];
House Attribute ← Input A[Price] / Max Price ;
MLP.Output←MLP(House Attribute);
CNN.Output← MiniVGGNet(Image);
Output ← Concatenate( MLP.OUTPUT, CNN.OUTPUT);
```

The miniVGGnet is applied in a channel last fashion manner of three-phase convolution.Each convolution has three different types of filter i.e. 16,32 and 64 where each filter has the same size of 3×3 in loop. The result is fed into an activation layer of ReLU followed by the Batch normalization layer. The max pool layer is applied for reducing the spatial dimension of the input size .The pool size is used two times of size 2×2 to diminish the input size from 64 to 32 and 32 to 16 . The Structure of miniVGGNet along with variable output image size of layers is represented as [5]:

Layer Type	Output Size	Filter Size / Stride
Raw Input image	$128 \times 128 \times 3$	
Input Image	$64 \times 64 \times 3$	
Conv2D	$64 \times 64 \times K$	3×3, K = (16,32,64)
Activation	$64 \times 64 \times K$	
BN	$64 \times 64 \times K$	
Conv2D	$64 \times 64 \times K$	
Activation	$64 \times 64 \times K$	
BN	$64 \times 64 \times K$	
POOL	$32 \times 32 \times K$	2×2
Dropout	$32 \times 32 \times K$	

Retrieval Number: B3597078219/2019©BEIESP
DOI: 10.35940/ijrte.B3597.078219
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Layer Type	Output Size	Filter Size / Stride
Conv2D	$32 \times 32 \times K$	3×3, K = (16,32,64)
Activation	$32 \times 32 \times K$	
BN	$32 \times 32 \times K$	
POOL	$16 \times 16 \times K$	2×2
Dropout	$16 \times 16 \times K$	
FC	16	
Activation	16	
BN	16	
Dropout	16	
FC	4	

The miniVGGnet is applied in a channel last fashion manner of three-phase convolution. Each convolution has three different types of filter i.e. 16, 32 and 64 where each filter has the same size of 3×3 in loop. The result is fed into an activation layer of ReLU followed by the Batch normalization layer. The max pool layer is applied for reducing the spatial dimension of the input size .The pool size is used two times of size 2×2 to diminish the input size from 64 to 32 and 32 to 16 . In Multilayer Perceptron, the perceptron is also min-max scaled, which is ranged between 0 to 1. A fully connected dense layer of 8 dimensions and followed by fully connected 4-dimension hidden layer are activated by ReLU activation, whereas linear activation is opted for the final optimal regression output layer.

All the dataset is ranged between 0 to 1, which leads to better converging for the training process. In the final stage, the 4-dimension vector of the convolutional neural network is combined with 4-dimension of multilayer perceptron to create an 8-dimension vector of regressive structure. The structure is processed for an 8-4-1 output with the architecture of Neural Networks MLP => ReLU => Linear. The final linear layer is performed to give output prediction in the numerical form with the help of Keras, which is good at handling multi-input with mixed data.

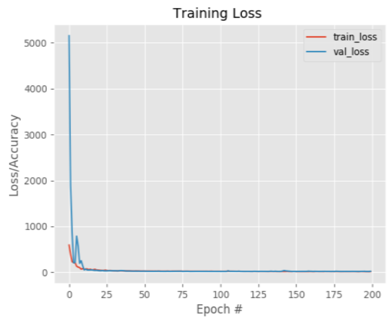


IMAGE 1:Result of Training loss of LeNet

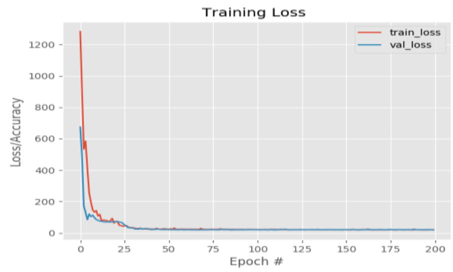


IMAGE 2:Result of Training loss of MiniVGGNet

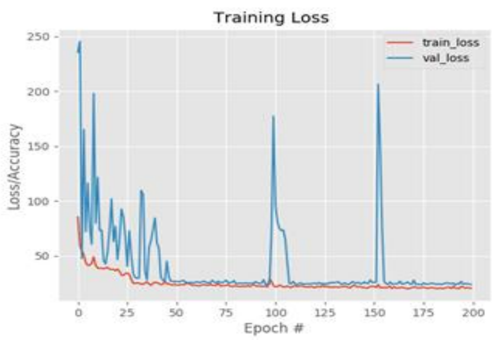


IMAGE 3:Result of Training loss of ShallowNet

III. RESULT AND DISCUSSION

Image set is processed via different set of neural architecture to improvise outcome of the total architecture as well as to predict the best result among them. The architectures are LeNet, SallowNet and MiniVGGNet. LeNet [9] has the architecture of two series of CONV => ACTIVATION =>POOL where here the activation is taken as ReLU. The architecture is summarised as:

INPUT => CONV => ReLU => POOL => CONV => ReLU => POOL =>FC => ReLU=> FC

LeNet showed the result of 76.12% accuracy. The mean error is resulted 23.88% where the standard error is fallen to 29.21%. Image 1 shows training and value loss per number of epochs.

Networks	Mean error	Standard error
LeNet	23.88	29.21
ShallowNet	21.81	24.20
MiniVGGNet	16.79	15.59

ShallowNet is the smallest architecture than the rest of other implementation. It consists of one series of CONV => ACTIVATION => POOL . So the simple architecture can be defined as: INPUT => CONV => ReLU => POOL. In the case of Shallownet shown in image 3, the accuracy rate is 78.19% with the mean error of 21.81% where the standarderror is 24.20%. The normal CNN architect with CONV => ReLU => BN => POOL which is already used by Adrian [5] has the mean error of 22.41% and the standard error of 20.11%. Overall the MiniVGGnet has more accuracy of 83.21% with the minimum error rate among all these architecture 16.79% mean error and 15.59% standard error as in image 2. The experiment was done by the c4.4xlarge instance of Amazon AWS which has the configuration of 16 CPUs and 30 GB of RAM. The result may vary up to +/- 2 scales with different instance and configuration.

IV. CONCLUSION

Multi-Layer Perceptron is well suited for regression prediction than Convolution Neural Network trained for image classification as a standalone method. The result of this study goes on to show that the output is optimized by concatenation of the above networks. There exists a bigger scope for future research which can focus on improving Convolution Neural network with fine-tuning of transfer learning. This method of concatenating MLP and CNN specifically MiniVGGnet might be implemented across domains which consist of mixed data format.

REFERENCES

1. Ahmed, E., & Moustafa, M. (2016). House price estimation from visual and textual features. *arXiv preprint arXiv:1609.08399*.
2. Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*
3. Pardo, C., Rodríguez, J. J., García-Osorio, C., & Maudes, J. (2010, June). An empirical study of multilayer perceptron ensembles for regression tasks. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems* (pp. 106-115). Springer, Berlin, Heidelberg.
4. H.Ahmed, E. (2016). Houses dataset. https://github. com/emanhamed/Houses- dataset.
5. Adrian Rosebrock, Keras: Multiple Inputs and Mixed Data , https://www.pyimagesearch.com/2019/02/04/keras-multiple-inputs-and-mixed-data/ accessed on 23 June
6. Lim, W. T., Wang, L., Wang, Y., & Chang, Q. (2016, August). Housing price prediction using neural networks. In 2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD) (pp. 518-522). IEEE.
7. Limsombunchai, V. (2004, June). House price prediction: hedonic price model vs. artificial neural network. In New Zealand Agricultural and Resource Economics Society Conference (pp. 25-26).
8. Manaswi, N. K. (2018). Understanding and working with Keras. In Deep Learning with Applications Using Python (pp. 31-43). Apress, Berkeley, CA.
9. LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

.AUTHORS PROFILE

Satyasangram Sahoo received Bachelor degree in Information technology management from Ravenshaw University and master degree in information technology from Manipal University.he also awarded master in business administration from ICM. He is now pursuing Ph.D. in computer Science from Pondicherry University.

Prem kumar Borugadda received MTech (CSE) degree from Pondicherry University, Pondicherry, India. Now He is pursing PhD degree in department of computer Science from Pondicherry University. He is having 3 years of teaching experience. His broad research area is in Image Processing and Machine Learning.

Dr. R Lakshmi hold Bachelor of Engineering degree from Bharathiyar University and received M.tech and Ph.D. degree in the department of computer Science from Pondicherry University, Pondicherry, India. Now she is serving as Assistant Professor in Pondicherry University.