Prediction Model using 1D Convolution Neural Network

Srushti Wadekar
Computer Science
Lakehead University
swadekar@lakeheadu.ca

***Abstract*—This paper consists of the detailed and elaborated steps required to generate the regression model for the purpose of prediction. Regression analysis is a computational modeling technique that explores the correlation between target variable and predictor variable.This paper focuses on the Convolution based neural network method to implement the regression model. Convolution neural network is a type of deep learning technique which employs the mathematical technique of convolution. Convolution neural network model will be used to predict the feature value.**

***Index Terms*—Regression, Convolution, Model, Predict**

I. Introduction

Predictive modelling is a method used to anticipate outcomes by using data and statistics in data models. Such equations can be used to predict anything from weather forecasts and viewing figures to technological developments and corporate income. Machine learning differentiates from predictive modelling and is defined as the use of statistical techniques to make predictive modelling possible for the computer. Machine learning is widely used interchangeably for practical applications. Predictive modelling is beneficial because it gives detailed input into all problems and makes it possible for users to make predictions. It is important to get a glimpse into future events or outcomes that contradict the underlying assumptions in order to retain a competitive advantage. In particular, Machine Learning strives mainly to the the errors of a model or, to the degree of explanability, to allow the most accurate predictions. In machine learning, algorithms from a wide range of different areas including statistics are borrowed, re-used for these purposes.

A Nonlinear regression model approach will be taken into consideration for prediction the feature value of dataset. Nonlinear regression is a regression in which the dependent or criterion features are trained as a non-linear function of model parameters and one or more independent features. This paper consists the approach of implementing a predictive model on California housing dataset. Convolution neural network is used for generating the prediction model to predict the median house value of California houses.

II. Literature Review

In paper [1], convolution neural network based on structured prediction approach has been implemented by Avijit Dasgupta and Sonam Singh. It explains convolution neural network as a special type of neural network where neurons are arranged in a 3-dimensional grid (width, height and depth). A 3D input volumes are taken by every layer of a CNN and converted into 3D output volumes. There are four major layer forms of CNN architectures: Convolutionary Layer, Pooling Layer, Upsampling and Totally Connected Layer. CNN infrastructure is constructed by building these layers. With the convolution process, each convolution layer transforms input representation. A pooling layer simply samples the input character maps spatially while the sampling layer does exactly the opposite.

Paper [2] explains the details of the proposed approach of the multitask CNN model. Considering there are N attributes, the model suggests that, starting from the given image, every CNN will learn an attribute. Each layer of the CNN will generate some feature and will pass it to the next layer. This feature generation fetched from the all the layers of CNN will be passed to the joint MTL loss layer. Multitask layer (MTL) are used when the tasks are under sampled. This weight parameters passed to this layer will be further decomposed into two matrixes which are the latent task matrix and the combination matrix. The latent task matrix is a shared matrix which acts as a fully connected layer between all CNN models. The combination matrix comprises of the specific information obtained from each CNN model. This structure is then iteratively trained until convergence. Paper [2] also declare that the structure proposed by Krizhevsky is being adopted here. This contains 5 convolutions, followed by 2 fully connected layers and finally the softmax and the loss. In addition, some pooling, normalization, and ReLU are applied between some of these layers. Some studies have concluded that including the shared fully connected layers will increase the time taken to train the network, although other work have also mentioned that the performance will drop drastically if these layers were not included. As this model requires more than one CNN the MTL can substitute this fully connected layers.

Paper [3] portrays that CNN are used on known grid like topology. The data to be processed can be 1D like time series data and 2D like image data. This neural network is called as Convolutional Neural Networks because it uses convolution instead of instead of general matrix multiplication in at least one of its layers. RNN and LSTM architectures are used to identify any long-term dependencies in the data by using the three models. Both the training data and the test data were normalized and after obtaining the predicted output, error was calculated using available true labels

III. Proposed Model

Artificial neural networks (ANN) compose of simple elements called neurons, which can make simple judgments in the area of mathematics. The neurons can together evaluate complicated issues, imitate almost any feature, including very complex, and provide detailed answers. There are three levels of neurons on a superficial neural network: an input layer, a hidden layer, and an output layer. A Deep Neural Network (DNN) includes more than one hidden layer that increases model flexibility and enhances predictive power dramatically. A neural network can perform as a regressive model. The network takes various input parameters which are depended variables to find the product with weights. The product is passed through an activation function to generate a regressive model. In this paper, the prediction model is generated using a convolution neural network.

A CNN is a multi-layer neural network used for images processing for image recognition, segmentation or object detection. CNNs work to reduce an item to its key characteristics, and to classify the type using the cumulative likelihood of defined characteristics. The value of CNNs is that they require fewer hyperparameters and less testing relative to other classified algorithms. A classification layer is in essence a logistic regression classifier. Given a fixed-dimensional input from the lower layer, the classification layer affine transforms it followed by a relu activation function. Convolution neural networks usually consists of three layers i.e. convolution layer, pooling layer and fully connected layer.

The first layer in the proposed model starts with a convolution neural network layer which takes 8 input features. The kernel size and stride rate defined in the layer is set to 1. Kernel size is the filter size which is used as a scanning window. In the convolution layers, an input is analyzed by a set of filters that output a feature map. This output is then sent to a pooling layer, which reduces the size of the feature map. This helps reduce the processing time by condensing the map to it's most essential information. The output if CNN layer is passed further to the batch normalization layer. Batch normalization is used to increase the performance, accuracy and speed of a model. The significance of a batch normalization layer is to decrease the unwanted shifts within the hidden layers to which will eventually fasten the training process for a more better results.

To achieve a non-linear solution, Relu activation function is applied to the batch normalized output of the convolution layer. This enables the generalisation or modification of the model with various data and the separating between the output. Relu layer also helped in vanishing the gradient problem. The batch normalized output is passed to the max pooling layer in the training model. The purpose of a pooling layer is to gradually reduce representation spatial size to limit parameters and network computation. Pooling layer operates on each feature map independently. The pooling layer summarizes the features by considering the feature map. It operates upon each feature map separately to create a new set of the same number of pooled feature maps. Convolution and pooling procedures are repeated several times and depending on the network, the amount of iterations is then sent to the fully connected (FC) layers. Such FC layers flatten the maps together and evaluate the possibilities of each function, together with the other ones, until the best classification is calculated. A flatten layer is used to convert the format of the convolution layer outputs into a linear form that can be passed to a fully connected layer (Linear). Fig. 1. shows the diagrammatic representation of the trained model.

This modelling allows CNNs to learn the position and scale of features by making them especially good at the regression of hierarchical or spatial data and the extraction of unlabelled features.

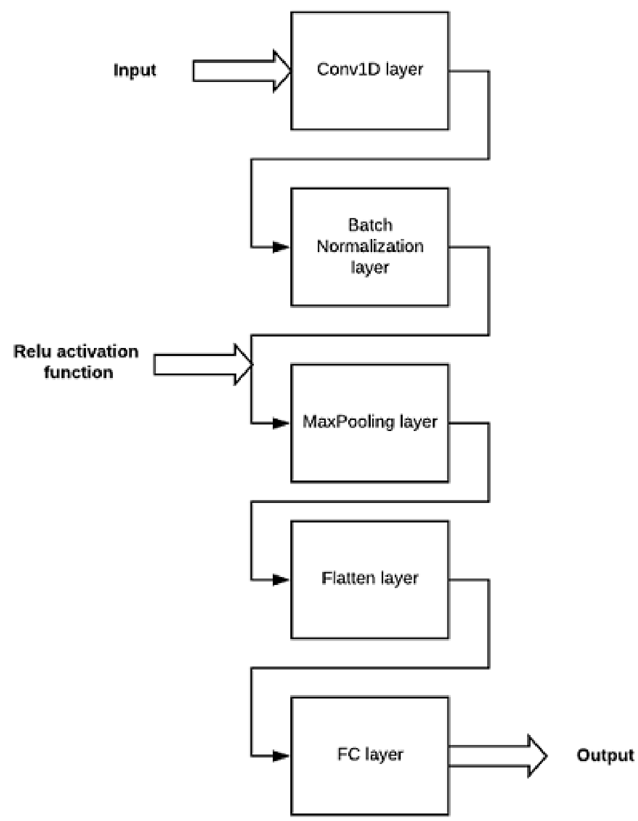

Fig. 1. CNN model flow

IV. Experimental Analysis

The training model was implemented on the California housing dataset to predict the median house value of every houses in the dataset. The dataset had features such as longitude, latitude, age, no of rooms, population, income and no of bedrooms. Fig.2 shows the graphical representation of all feature values of the California dataset for the initial rows.

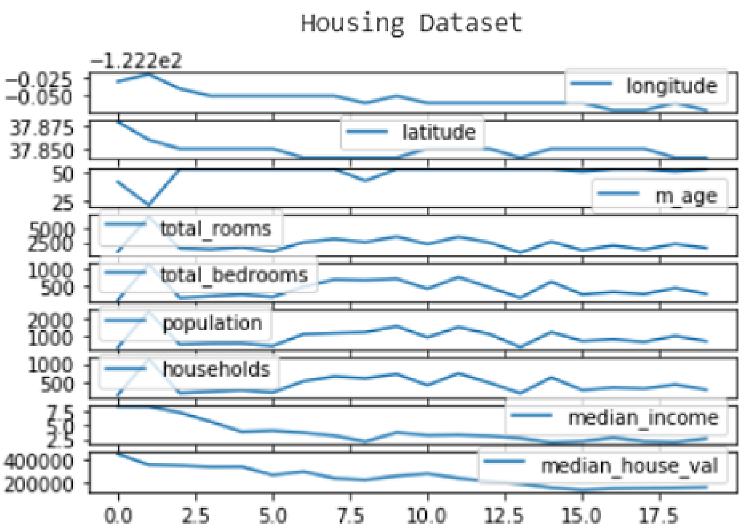

Fig. 2. Sub-plots for first 10 samples

The regression model was trained using one-dimensional convolution neural network. For a better model performance, two convolution layers were added to the model. It was observed that performance of the model was increased by addition of convolution layers. The following parameters impacted significantly while building the model. An additional step of batch normalization was added after convolution layer to fasten the model training process. It was observed that applying batch normalization on the output neurons from convolution layer increased the R2score which lead to a better model performance. Hence, batch normalization was applied on both the convolution layers in the model. Finally, for the pooling layer, MaxPooling and AvgPooling layer was included in the model. MaxPooling was taken into final consideration due to it's high performance. Thus, a model with two convolution layers by applying batch normalization and a MaxPooling layer was considered as a final model.

A. Training parameters

There were several training parameters such as batch size, kernel size, no of epochs, learning rate, padding and stride rate that were considered while building the prediction model. Batch size of 64 and 128 was considering during training the model where batch size of 64 trained the model in a more accurate way. Number of epochs were increased from 300 to 500 for training the model. Inference time increased by considering 500 epochs but lead to a more accurate model. Hence, R2score of the model was increased by considering greater number of epochs. Learning rate was increased from 0.0001 to 0.01 which caused the issue of gradient descent and generated a under-fitted model. Thus, learning rate of 0.0001 suited the best for the model. Also, by considering stride rate as 2, the R2score of the model was decreased to 0.37 from 0.54.

B. Activation function

Activation function was mainly added to the output of convolution layer to maintain the non-linearity in model. As the model is a non-linear regressive model, Relu activation function was used between the layers. Softmax and Sigmoid activation functions were also considering while training the model but did not perform better compared to Relu activation function.

C. Loss function

Loss function for a model is necessary to evaluate the performance and back propagate to adjust the parameters to generate better performance. Various optimizer were used in the loss function to optimize the model at every step. Optimizers such as SGD, AdamW, Adamax, Adam, Adadelta and Adargrad were taken into consideration in the loss functionality. Following was the performance for various optimizers.

1) *AdamW:* R2score as 0.51 and L1 Loss as 64,000
2) *Adamax:* R2score as 0.56 and L1 Loss as 58,000
3) *Adadelta:* R2score as 0.40 and L1 Loss as 71,000
4) *Adargrad:* R2score as 0.41 and L1 Loss as 71,000
5) *SGD:* R2score as -0.063 and L1 Loss as 96,000
6) *Adam:* R2score as 0.64 and L1 Loss as 51,000

These results lead in selection of Adam as an optimizer while training the model.

V. Conclusion

In this paper, for the California housing dataset, CNN is found to be the best suited regression model. It uses the information given at a particular instant for analysis. Another reason for CNN performing better is due to the fact that CNN does not depend on historic data for prediction. As a result, the model can better estimate the dynamic changes in the training phase which leads to a better prediction accuracy. Thus, on analysis and experimentation, a two-layered convolution neural network by application of batch normalization and a MaxPooling layer resulted in a maximum accuracy of 62 percent.

VI. Appendix

The dataset used to train the model and the source code of trained model is present in the given Github link. Link: https://github.com/Srushti10/NLP

References

[1] Avijit Dasgupta, and Sonam Singh, "A Fully Convolution Neural Network based Structured Prediction Approach towards the Retinal Vessel Segmentation,"IEEE 14th International Symposium on Biomedical Imaging, 2017.

[2] Abrar H. Abdulnabi,Jiwen Lu,Kui Jia, and Gang Wang, "Multi-task CNN Model for Attribute Prediction,"IEEE Transactions of Multimedia, November 2015.

[3] E. A. Gopalakrishnan,Vijay Krishna Menon, and Soman Kp, "Stock Price Prediction using LSTM,RNN AND CNN Sliding Window Model,"Procedia Computer Science, January 2018.

[4] Lei Ren,YAQIANG SUN,HAO WANG, and LIN ZHANG, "Prediction of Bearing Remaining Useful Life With Deep Convolution Neural Network,"IEEE special section on Cyber-Physical-Social Computing and Networking, February 2018.