International Journal of Mechanical Engineering

A Comparative Study on the Prediction Accuracy of Wide and Deep models in Deep Learning

Min-Ho Cho
Professor, Computer Engineering, Jungwon University, 85, Munmu-ro, Goesan-gun, Chungbuk, 28024, Republic of Korea

Abstract.
BACKGROUND/OBJECTIVES: When creating a deep learning model, there are many factors to consider. Among them, the deep model and the wide model should be considered at the initial stage of model construction. However, there is still a lot of debate about which model is effective to use.
METHODS/STATISTICAL ANALYSIS: In this study, we intend to create and test criteria to determine which model to use for analyzing the given data. For this, we will build and compare deep learning models using California housing data, which is widely used for model evaluation. The models developed for the research are deep model, wide model and deep & wide model.
FINDINGS: A new evaluation criterion was introduced for the evaluation of the three developed models. The new criterion is to convert the difference between the predicted value and the actual value of the model into an absolute value and add them. This criterion can be used to judge the prediction accuracy of the three models developed through this criterion.
IMPROVEMENTS/APPLICATIONS: The procedure and criteria of this study can be used as criteria for determining in advance which model is best to use in deep learning for given data. In particular, it will be very helpful in determining the model type that should be determined in the early stages of a deep learning model.

Keywords: Deep learning, wide model, artificial intelligence, wide model. model evaluation

1. INTRODUCTION

Artificial intelligence has made great strides since the concept of deep learning targeting labeled data was announced in 2010[1-4]. In particular, it is widely used in image processing, language recognition, and translation[5-7]. Recently, it has been used in web recommendation systems and in the field of biology, and this trend continues to spread[8-12]. But, applying deep learning in practice is a process that requires many considerations, time, and various possibilities in mind. In particular, determining whether to use a deep model or a wide model at the stage of constructing a deep learning model has a great influence on the accuracy of the model[4,8]. In this study, the criteria for determining a deep learning model for the data to be analyzed were set. This criterion shows the difference between the predicted value of the developed model and the actual value in terms of the total data. The classification criteria introduced are based on the premise that only part of the data is performed before full-scale analysis is performed. To demonstrate, three deep learning models were developed and applied for California housing data. There are three developed deep learning models: deep model, wide model, and deep & wide model. All models were developed using Keras and TensorFlow and performed in Google's Colab. The criteria introduced in this study can be applied directly in practice, and will be of great help in determining the initial model.

2. TEST DATA ENVIRONMENT

The data used to measure model-specific predictive accuracy of deep learning is the “California Housing Prices”. It have total 20,640 record about California housing. Each record has 9 items, “longitude”, “latitude”, “housingMedi anAge”, “totalRooms”, “totalBedromms”, “population”, “household”, “medianIncome”, “medianHouseValue”. All data type is float64. More detailed information about the data is summarized in Table 1. Using the given data and model, we will predict the value of “medianHouseValue”. The accuracy of the model can be evaluated by summing and comparing the difference between the predicted data and the actual data. Through such evaluation, it is possible to determine a model of deep learning that is suitable for the given data.

Detailed information on the data can be found here : http://kaggle.com/camnugent/california-housing-prices.

Table 1: California housing data structure and example

Data Item	mean of value	range of value	data type
longitude	longitude value of house (The higher the value, the loc of house is further west)	range of longitude : -180 ~ 180 Min of data : -124.3, Max of data : -114.3	float64
latitude	latitude value of house (The higher the value, the loc of house is further north)	range of latitude : -90 ~ 90 Min of data : 32.5, Max of data : 42.5	float64
housingMedianAge	Median age of houses in the block (The lower the value, the newer the building)	Min of data : 1.0, Max of data : 52.0	float64
totalRooms	Total room number of houses in the block	Min of data : 2.0, Max of data : 37937.0	float64
totalBedrooms	Total bedrooms number of houses in the block	Min of data : 1.0, Max of data : 6445.0	float64
population	Total number of residents in the block	Min of data : 3.0, Max of data : 35682.0	float64
households	Total number of households in the block	Min of data : 1.0, Max of data : 6082.0	float64
medianIncome	Median income of households in the block(unit: 10,000 doller)	Min of data : 0.5, Max of data : 15.0	float64
medianHouseValue	Median home sales price in the block	Min of data: 14999.0 Max of data : 5000001.0	

-122.230000, 37.880000, 41.000000, 880.000000, 129.000000, 322.000000, 126.000000, 8.325200, 452600.000000
-122.220000, 37.860000, 21.000000, 7099.000000, 1106.000000, 2401.000000, 1138.000000, 8.301400, 358500.000000
-122.240000, 37.850000, 52.000000, 1467.000000, 190.000000, 496.000000, 177.000000, 7.257400, 352100.000000
-122.250000, 37.850000, 52.000000, 1274.000000, 235.000000, 558.000000, 219.000000, 5.643100, 341300.000000
-122.250000, 37.850000, 52.000000, 1627.000000, 280.000000, 565.000000, 259.000000, 3.846200, 342200.000000
-122.250000, 37.850000, 52.000000, 919.000000, 213.000000, 413.000000, 193.000000, 4.036800, 269700.000000

3. DEEP LEARNING MODEL FOR EVALUATION

There are many factors to consider when developing models and making predictions using deep learning. In particular, there are numerous considerations such as the depth of the model, the format of the input data, the type of optimizer used, the format of the input and the format of the output. However, this study attempts to determine the type of the deep learning model before any considerations. Therefore, this study constructs a test model to use a limited hidden layer and a limited input and output format. Before building a full-fledged model, it is a preliminary work, the construction of the model was also carried out in a simple form using Keras and TensorFlow. However, Various models can be considered, three models were prepared for evaluation: a deep model, a wide model, and a deep & wide model. By extracting a portion of the data and applying it to a given model to check the results, you can check the model suitable for the given data in advance. Naturally, the same data was applied to the three models, and the three models were also configured in the same form in the hidden layer or the input/output layer. Therefore, we would like to examine by applying the same data/model to the following three models. If you want to determine the type of model using this study in practice, you can use the given model.

Analysis of a given data requires many factors to be considered. For example, the number of hidden layers, the number of hidden layers, neurons, learning rate, optimizer, batch size, activation function, and number of repetitions may be considered. Among these, there are various types of optimizers such as SGD, Adgrad, RMSprop, Adam, Nadam, AdaMax. All of these techniques are important, but they should be considered step by step while constructing the model.

However, this study focuses on the process of determining the appropriate model type for the given data before considering all of these factors. Therefore, it is the key to determine which type of model is suitable by applying the three models shown in Figure 1 to the target data. Detailed descriptions of the three models are as follows.

“Deep Model” : This is a typical deep learning model. It has two hidden layer and data is input through on input layer.

“Deep&Wide Model” : It has the same structure as the “Deep Model”. However, it is a structure in which the input of “Input Layer” is added as a separate input to the “Link Layer”. It is a model that adds input to the existing deep learning model.

“Wide Model” : Unlike general deep learning models, It is a model that expands the hidden layer horizontally and provides the same input to each. The Link Layer plays the role of receiving and controlling the output of the expanded hidden layer. This model is an example of a non-sequential neural network. This structure was introduced in Cheng’s thesis in 2016[3], and is currently being used in many fields.

In practice, when the deep learning model is used to determine the type of the model before concrete implementation, other types of models may be considered in addition to the above three.

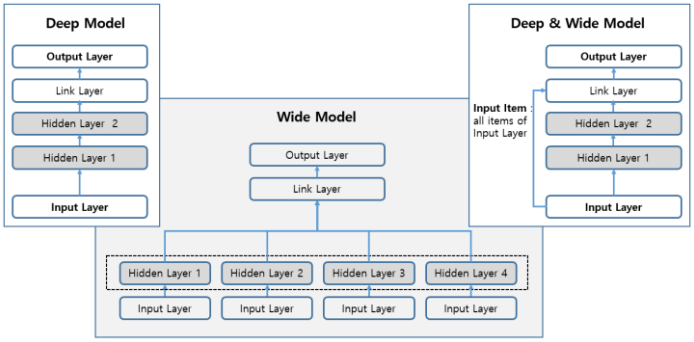


Figure 1. Deep, Wide, Deep&Wide model summary

The model shown in Figure 1 is a sample model for evaluating whether it is better to select a wide or deep model for a given data. Various models may exist, but given the general environment, using the model given in Figure 1 will suffice to

determine the type of model suitable for the data. The reason for suggesting the Deep&Wide model in Figure 1 is that the hybrid type is often appropriate when determining the shape of the model. For the exact shape of the model presented in Figure 1, the appearance of the model is summarized in Figures 2 and 3 using Keras' model view function. Figure 2 shows the structure of the wide model, and Figure 3 shows the structure of the Deep&Wide model. The structures shown in Figures 2 and 3 are tailored to the data used for this verification. Therefore, if you want to use the results of this study to determine the shape of the model in practice, you will have to maintain the overall structure and adjust the values of related variables.

Layer (type)	Output Shape	Param #	Connected to
input_2 (InputLayer)	[(None, 8)]	0	
dense_7 (Dense)	(None, 30)	270	input_2[0][0]
dense_8 (Dense)	(None, 30)	270	input_2[0][0]
dense_9 (Dense)	(None, 30)	270	input_2[0][0]
dense_10 (Dense)	(None, 30)	270	input_2[0][0]
concatenate_1 (Concatenate)	(None, 120)	0	dense_7[0][0] dense_8[0][0] dense_9[0][0] dense_10[0][0]
dense_11 (Dense)	(None, 1)	121	concatenate_1[0][0]

Total params: 1,201
Trainable params: 1,201
Non-trainable params: 0

Figure 2. Wide model summary

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, 8)]	0	
dense (Dense)	(None, 30)	270	input_1[0][0]
dense_1 (Dense)	(None, 30)	930	dense[0][0]
concatenate (Concatenate)	(None, 38)	0	input_1[0][0] dense_1[0][0]
dense_2 (Dense)	(None, 1)	39	concatenate[0][0]

Total params: 1,239
Trainable params: 1,239
Non-trainable params: 0

Figure 3. Deep&Wide model summary

This study is to apply a simple model in advance to determine the type of model suitable for the data. To this end, three models to be applied were presented separately. Therefore, in keeping with the idea of this study, the given model can be freely changed by the user. For example, you can separate the input data of the model in a specific environment and apply it to the deep & wide model. An example of this is presented in Figure 4. The model presented in Figure 4 is also a case that can be considered when determining a deep learning model suitable for data. However, in this paper, only the three models presented in Figure 3 were considered when considering the shape of the given data.

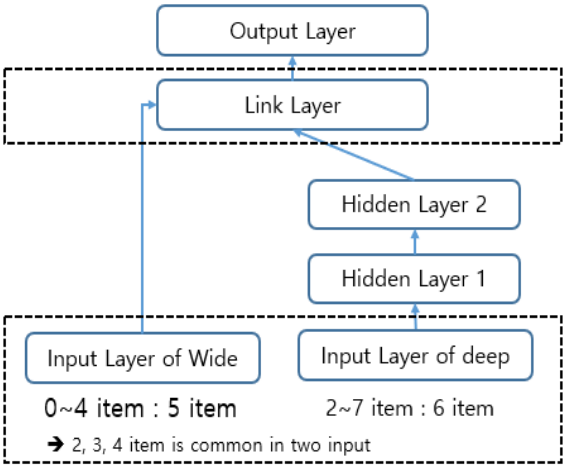
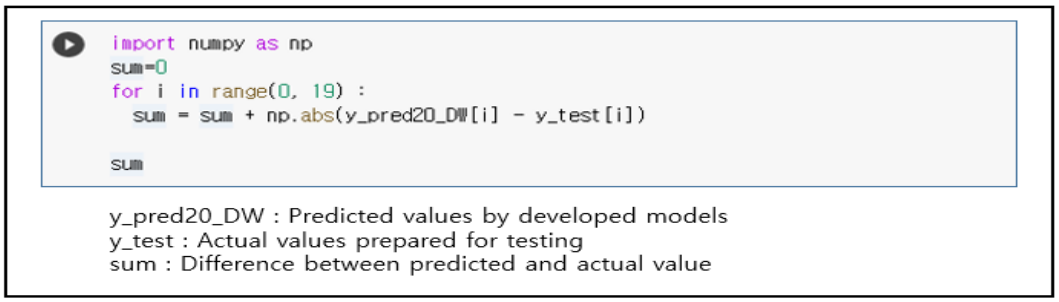


Figure 4. Additional Model

4. EVALUATION CRITERIA

Evaluation of deep learning models is performed in a variety of ways. In most cases, the model uses the evaluate function to determine the final correct answer rate and loss value. loss shows the difference between the predicted value and the actual value, the smaller the better. In the case of a classification model, the model is evaluated using Precision, Recall, F1 score, and Accuracy based on the confusion matrix. In addition there are methods such as ROC (Receiver Operating Characteristics), AUC (Area Under the ROC Cuve), Multi-class ROC.

This study is to find out the type of model suitable for the given data. Therefore, among the existing evaluation methods, a form similar to the easiest and intuitive loss method was used. The core idea of the algorithm is to judge that the small difference between the predicted value and the actual value has performed an accurate prediction. In addition, in order to identify the exact difference, the absolute value was taken for the difference value. Refer to Figure 5 for details of the algorithm made in Python.

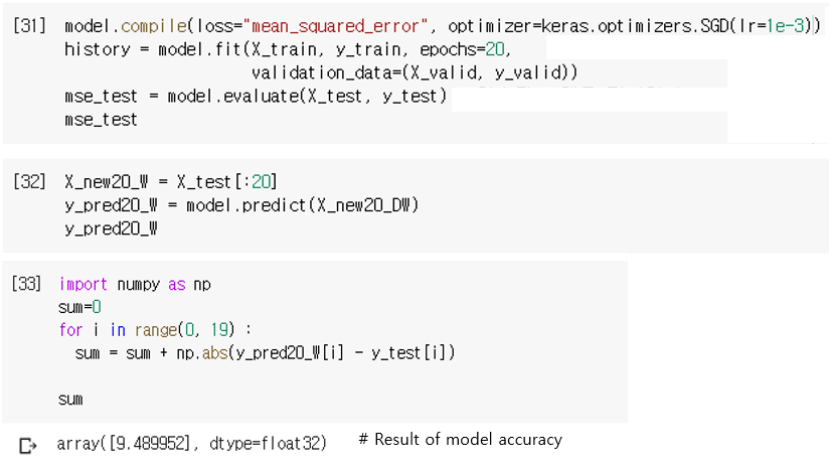


```
import numpy as np
sum=0
for i in range(0, 19) :
  sum = sum + np.abs(y_pred20_DW[i] - y_test[i])

sum
```

y_pred20_DW : Predicted values by developed models
y_test : Actual values prepared for testing
sum : Difference between predicted and actual value

Figure 5. Evaluation algorithm made in Python



```
[31] model.compile(loss="mean_squared_error", optimizer=keras.optimizers.SGD(lr=1e-3))
     history = model.fit(X_train, y_train, epochs=20,
                         validation_data=(X_valid, y_valid))
     mse_test = model.evaluate(X_test, y_test)
     mse_test

[32] X_new20_W = X_test[:20]
     y_pred20_W = model.predict(X_new20_DW)
     y_pred20_W

[33] import numpy as np
     sum=0
     for i in range(0, 19) :
       sum = sum + np.abs(y_pred20_W[i] - y_test[i])

     sum

     array([9.489952], dtype=float32)    # Result of model accuracy
```

Figure 6. Example of applying the new criteria to the model in Figure 1

Figure 6 shows the actual application process of the evaluation criteria presented in Figure 5. The configuration provided in Figure 2 is modeled using Keras and TensorFlow, and then compiled. The compiled model is trained using the training data separated in advance. A prediction is performed based on the learned model, and a process of adding the difference between the predicted value and the actual value is performed.

At this time, it is not necessary to perform the process of Figure 6 for all data. Our purpose is to decide whether to construct a deep learning model in a deep or wide configuration. Therefore, we just need to perform the process of extracting some of the data to be used and applying it to the three models presented in Figure 1. This process determines the form of a deep learning model suitable for the data to be analyzed through pre-analysis. And this process helps to determine the shape of the deep learning model, making it useful for many procedures and time required for the development of the deep learning model later.

5. CONCLUSION

The three models in Figure 1 were implemented using Keras and Tensorflow for easy use. The implemented model in Figure 1 was trained and tested using “California Housing” data. The training and test data were organized separately. The test results for each model are summarized in Table 2. As for the test results, the Wide Model scored the best in the accuracy of the prediction. And “Deep&Wide Model” showed almost the same level of accuracy as the “Wide Model” in the accuracy of prediction. The commonly used “Deep Model” showed inferior results in prediction accuracy compared to the other two models. Based on the results summarized in Table 2, the deep learning model based on the California housing data can be expected to show better results for the wide model than the deep model.

Therefore, when constructing a model related to deep learning later, it is desirable to construct the model by considering various variables based on the wide model. Of course, this is the case based on California housing data.

Table 2: Test Result by each model by using California housing data

	Deep Model	Deep & Wide Model	Wide Model
Sum Value	10.395	9.896	9.489

*The smaller the sum value, the more accurate it is, so we can find that Wide Model is the most accurate.

Through this study, a new factor to be considered when building a deep learning model was discovered. When creating a deep learning model, we need to consider new things in addition to data set separation and randomization, data set size and adjustment, selection of function including relu, and depth of hidden layer. It is a model type, like deep or wide. Through this study, we can be confirmed that the selection of deep or wide in deep learning modes is as important as other factors. Creating a deep learning model based on given data is difficult and takes a long time. In particular, optimization of the developed model is an important step in determining the value of a deep learning model. However, before all these processes are performed, it is also important to determine what kind of deep learning model suitable for the given data. A well-chosen model reduces the optimization process and provides a basis for better results. In that respect, there are good reasons for practitioners to consider the standards and methods proposed in this study.

6. REFERENCES

1. LeCun Y, Bengio Y, Hinton G. Deep Learning. Nature. 2015 May; 521: 436-444.
2. Salakhutdinov R, Hinton G. An Efficient Learning Procedure for Deep Boltzmann Machines. Neuro Computation. 2012 April; 24:1967-2006.
3. Schmidhuber J. Deep learning in neural networks. Neural and Evolutionary Computing. 2015 Jan; 61:85-117.
4. Zhang ML,Zhou ZH. A Review on Multi-Label Learning Algorithms. IEEE Transactions on Knowledge and Data Engineering. 2014 Aug; 26(8):1819-1836.
5. Krizhevsky A, Sutskever J, Hinton G. Imagenet classification with deep convolution neural networks. Proceedings of the 25 international Conference on Neural Information Processing System,Lake Tahoe. 2012 Dec ; 1:1097-1105. DOI:10.1145/3065386.
6. Yun YG, Mun SH, Shim SJ, Park SH. Recognition of License Plate Number for Web Camera Input using Deep Learning Technology. Journal of Next-generation Convergence technology Association. 2020 Dec :4(6):565-572.
7. Dahl GE, Yu D, Deng L, Acero A. Context-dependent pre-trained deep neural networks for large vocabulary speech recognition. IEEE Transactions on Audio, Speech and Language Processing. 2012 Jan; 20(1):32-42.
8. Zhang S, Yao L, Sun A, Tay Y, Hallal A. Deep Learning based Recommanded System :A Survery and New Perspectives. ACM Computing Surveys. 2018. July; 1(1):1-35.
9. Bordes A, Glorot X, Weston J, Bengio Y. Joint Learning of Words and Meaning Representations for Open-Text Semantic Parsing. Proceedings of the AISTATS. 2012 Jan; 22:127-135.
10. Guo Y, Liu Y, Oelemans A, Lao S, Wu SM, Lew M. Deep Learning for visual understanding : A review. Neurocomputing. 2016 April; 187 :27-48.
11. Bengio Y, Lecun Y. Scaling learning algorithm towards AI. MIT Press. 2007
    Dreiseitl S, Lucila O. Logistic regression and artificial neural network classification nodels :a methodology review. Journal of Biomedical Informatics. 2012 Oct; 35(5-6): 352-359.