Artificial Intelligence Final Project Report

Industrial Engineering

201801361 Lee Min Jae

1. Abstract

This report is an artificial intelligence final project that predicts the power generation of three different wind turbines, which aims to preprocess data over time (hour, date, month), analyze data, and create deep learning models.

2. Introduction

This project was carried out to make the power system more economically available through wind power generation prediction. The input value consists of wind_energy_1, wind_energy_2, wind_energy_3, and date and time, and the output value is written and output by using a deep neural network, with predictions for wind_energy_1, wind_energy_2, and wind_energy_3.

3. Dataset and Features

The data processing process involves spliting the date column of the dataset into months, days, and hours, dropping the date, time, and ID columns to create the dataset, and copying the newly created dataset into train_features. Then, the average and std values of wind_energy_1,2,3 and month, days, and hours of the dataset are calculated, and normalized using the normalizer function.

After that, Create the deep learning model.

The deep learning model makes the input value receive three (months, days, and hours), and the hidden layer is set in the order of relu, tanh, and tanh, respectively, and the number of nodes of each hidden layer is set to 30.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 normalization_1 (Normalizat  (None, 3)                3
 ion)

 dense (Dense)               (None, 30)                120

 dense_1 (Dense)             (None, 30)                930

 dense_2 (Dense)             (None, 30)                930

 dense_3 (Dense)             (None, 3)                 93

=================================================================
Total params: 2,076
Trainable params: 2,073
Non-trainable params: 3
_________________________________________________________________
```

The optimizer uses Adam, beta_1 and beta_2 use the default values of 0,9 and 0.99, and amsgrad is not applied. For a total of three lessons, the learning rate is learned twice using the default value of 0.001, and in the last lesson, the learning rate value is adjusted to be adjusted to 0.0001.

The validation ratio is set to 0.15, and epoch is set to 100.

The callback function uses early-stopping to observe and callback 10 times based on the value of val_accuracy.

4. Methods

The reference document found that it was most appropriate to use relu for the first hidden layer, so the first hidden layer used relu. In addition, through empirical data, the second and third hidden layers could use tanh to increase the acuity value. In addition, the learning rate was used to find the minimum value of the function by using a relatively large value for the first learning, and the minimum value was adjusted by changing the learning rate value to a low value for the last learning. MSE (mean squared error) was used as a function to find the loss value. The reason was that our scoring method in kaggle was root mean squared error (RMSE), and intuitive accuracy could be seen by adding metrics.

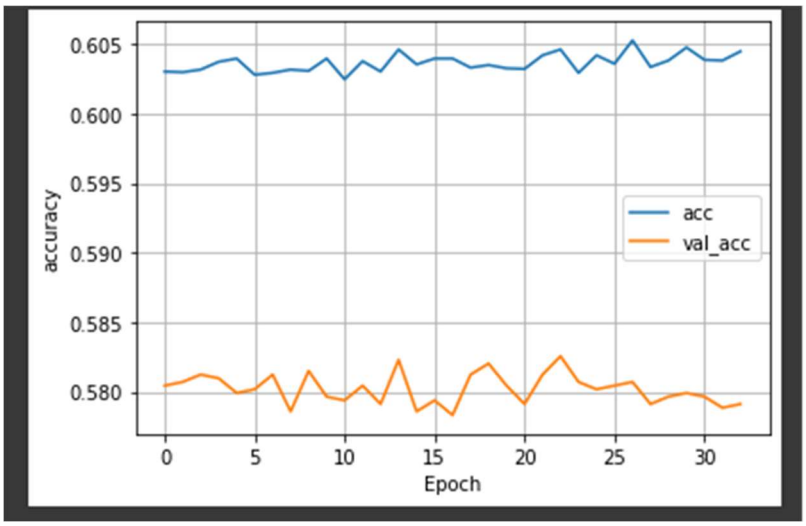
In applying the model, the epochs value was set to 100. The reason is that the early_stopping callback function is used, so if the model is suitable, the learning is automatically terminated. Therefore, the epochs value was set leisurely so that it did not interfere with learning. In addition, the ratio of validation_split was set to 0.15, because the ratio of validation was lowered based on empirical data after setting the default value to 0.2, and the optimal value was determined to be 0.15

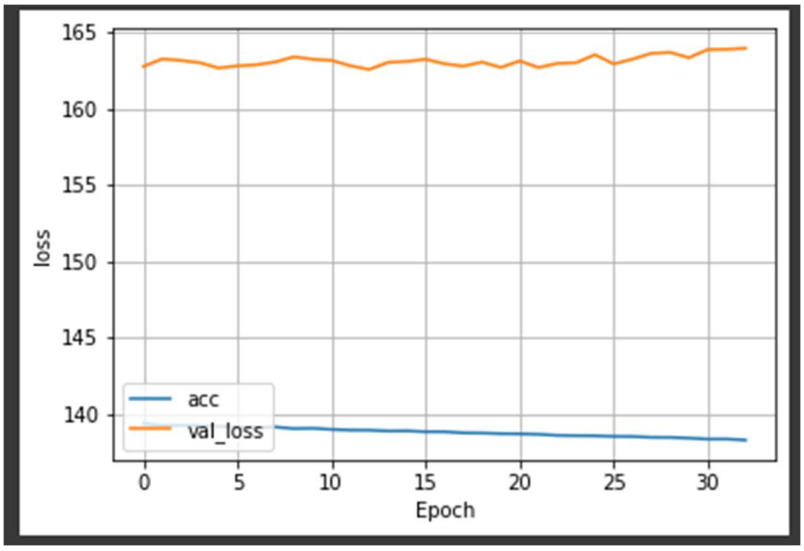
5. Experiments

The callback function is not used when applying the first deep learning model.When learning, the overfitting phenomenon continued to occur while learning until the epochs reached 100. The early stop method was used to prevent overfitting, resulting in remarkably good results.

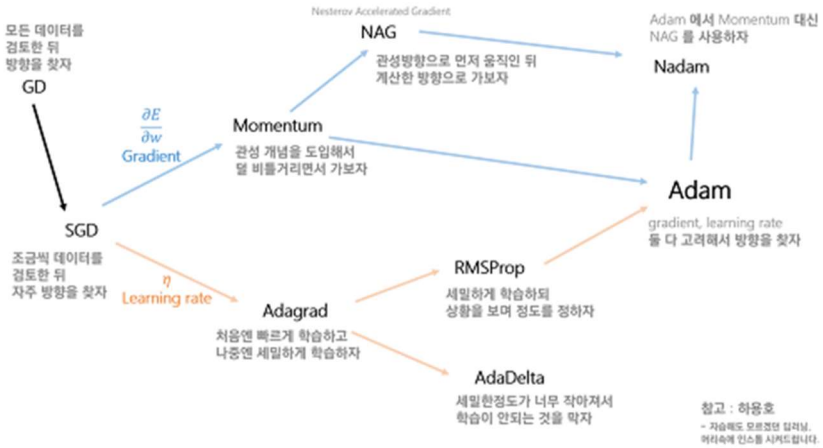
The average of accretion was about 0.604, and the average of val_accuracy was about 0.581.The average of the loss value was 138 and the average of the val_loss value was 163.

The reason why the graphs are linear is that they represent data after the third learning, so the epoch value is small and standardized





The reason for using Adam was that in the bibliography (https://gomguard.tistory.com/187)), the optimizer found that the most optimal development of the optimizer resulted in Adam. Adam's advantage was to find directionality by considering both gradient and learning rate.



In addition, when learning was taught, the score was not good at first if the results were submitted after learning once, but when learning was learned twice or three times, the model was best applied.

For this reason, the learning rate was used twice to find the minimum point with the minimum value, using the default value of 0.001, and the learning rate was reduced to 0.0001 to reduce the error as much as possible at the minimum point (prevent overfitting).

6. Conclusion

The most important point of artificial intelligence is that the number of layers, the number of nodes, and the combination of functions are important when setting the model, rather than which function is used. I think the importance of a combination of one function and not a good function emerges. The algorithm is thought to find the most optimal combination of these processes. If there is a competition like this, I want to participate. Finally, I am grateful to the professor for giving me a good lecture and giving me the best assignment that fits the content of the lecture.