Concrete mix design modelling based on variation of hidden layer and neuron of ANN for virtual learning development

S Santosa1,*, S Suroso1, M B Utomo1, M Martono1, M Mawardi1 and Y P Santosa2

1 Civil Engineering, Politeknik Negeri Semarang, Semarang, Indonesia
2 Institute of Informatik Universität Bonn, Bonn, Germany

*stefanus.santosa@polines.ac.id

Abstract. Artificial Neural Network (ANN) is a Machine Learning (ML) algorithm which learn by itself and organize its thinking to solve problems. Although the learning process involves many hidden layers (Deep Learning) this algorithm still has weaknesses when faced with a high noise data. Concrete mixture design data has a high enough noise caused by many unidentified / unmeasurable aspects. Information needs about the compressive strength of early age concrete (under 28 days) are often needed while the construction process is still ongoing. H2O’s Deep Learning type of ANN has been tried to predict the compressive strength of early age concrete, but the results are less than optimal. This study aims to improve the H2O’s-ANN prediction model using Bagging to reduce the influence of noise and overfitting. The lowest RMSE that are able to be achieved in this research with Bagging is 6.385 while it is 6.674 without Bagging. This result proves that Bagging is significant to reduce the Deep Learning error rate for predicting the compressive strength of early age concrete. Future work this model, as a new innovation in machine learning in civil engineering vocational education subject, can be used to various concrete mixture design and learning about concrete. Because this concrete mix design model is digital, it can be used for virtual learning. Currently the author is conducting research to create a virtual concrete mix design learning model.

1. Introduction

Concrete is a mixture of cement, water and aggregates (coarse and fine), which mixed by some formula [1,2]. This mixture can be mixed with various chemical components and mineral admixtures to improve the quality of the concrete, the self-compacting [2-4], the ductility [5], and the durability.

Concrete compressive strength is measured on 28 days, but to control the quality of the concrete, the early measurements is needed [6]. Prediction of the compressive strength of concrete at the age of under 28 days is required when concrete is still under construction. Construction failures often occur because of overloading. This is because often the construction manager sets the loading without proper calculation due to the lack of information of the strength of the early age concrete. If the information about the strength of concrete at the age of under 28 days is easily obtainable, the construction manager can calculate the safety barrier of the load.

Variety of the mixture and the characteristic of the concrete produce many effects [7], for instance the same ingredients can produce different strength when the mixture design is different [3,8], including different self-compacting, ductility, durability [9], delivery, and curing. This shows that concrete design problem is very complex, thus many complex mixture designs were proposed. As a result, concrete compressive strength prediction becomes inaccurate.
In the recent development, several researchers used Computational Engineering to design concrete mixture, either for slump or compressive strength, by using linear regression, multiple linear regression, and non-linear regression [10-12]. Despite that, the results often not satisfying, conventional statistical approaches are often not a choice anymore. Evolutionary model and deep learning also be used to improve the results. Nikoo created The Model of Prediction of Concrete Compressive Strength by Evolutionary Artificial Neural Networks which used Genetic Algorithm (GA) to find the optimal weight parameters for ANN [13]. Deng et al. used deep learning Convolution Neural Network (CNN) to recycled concrete compressive strength prediction. The results show that CNN has lower error than backpropagation neural network (BPNN) and support vector machine (SVM) [14]. Even though this is the case, CNN is used only when the positional information of a certain feature in the data is important such as in image processing or computer vision case rather than concrete mixture which the data is transactional.

Several other types of DNNs are popular as well, such as Multilayer Perceptron (MLP) and Recurrent Neural Networks (RNNs). However, for transactional data (tabular data), MLP works better than CNN or RNN. While CNN relies on parameter sharing for calculating grid or continuous signal data and RNN are strong in sequential data. For the purpose of this research, MLP implementation of H2O’s deep learning will be used. The computational model obtained will be developed for virtual learning specifically about concrete mix designs

2. Research methods
This research will use H2O Deep Learning library to create a model. ReLU is chosen to be the activation function for the model and will be tested later in regards to the performance. The research method is described in Figure 1.

2.1. Data collection
A secondary data was collected from UCI Data Repository created by I-Cheng Yeh, Department of Information Management, Chung-Hua University (Republic of China). The data consists of concrete mix design with its concrete compressive strength on different age. These concrete mix designs were tested with destructive test to record the concrete compressive strength values. The composition of the mix, which is cement, blast furnace slag, fly ash, water, superplasticizer, coarse aggregate, fine aggregate, and age (3, 7, 14, 28 days), were varied to obtain various results.

![Figure 1. Research method.](image)

2.2. Experiment
Before the experiment process were done, the data was split into training and test data. This training data then later on will be split again into train set and validation set by using cross validation by splitting the data into k subset of data. k-1 subset was used for training and the remaining subset was used for validation. For numerical stability, the data then normalized based on the training data. The data was split into 0.6 and 0.4 ratio for training and test data respectively.
Firstly, several parameters were defined while weights and biases are randomly generated. The parameters are number of hidden layers, number of hidden neurons, epoch, L1, L2, epsilon, and rho. Secondly, the model was trained with the train set to calculate the prediction. Then, this prediction will be used to calculate the loss by using root mean-squared error (RMSE). Afterward, the loss is back propagated to optimize the bias and weight. To check whether overfitting is occurred or not, the validation set is used to test the current parameters.

To improve the model over noisy data, Bagging was also used. The model was replicated b times and each model was trained using a randomly selected data with replacement. This stage is also called bootstrapping. Afterwards, the result of each model was aggregated.

2.3. Evaluation
After the training process was done, the test data was used to measure the accuracy of the model. In this step, the effect of Bagging was tested by training two model with and without Bagging then the test accuracies were compared.

3. Results and discussion
Before trying to find the best parameters for the prediction model, the data is split into 60/40 for training and testing respectively. Afterwards, the parameters are set to be the default value from H2O library. The summary of the parameters is described in Table 1.

| Parameter   | Value  |
|-------------|--------|
| Epoch       | 20     |
| L1          | $1.0 \times 10^{-5}$ |
| L2          | 0      |
| epsilon ($\epsilon$) | $1.0 \times 10^{-8}$ |
| rho ($\rho$) | 0.99   |

Firstly, the effect of the ReLU activation function over several architecture settings based on H2O’s deep learning ANN architecture and H2O’s deep learning ANN with Bagging architecture will be tested. The name of this architecture will be written as ANN and the others ANN+Bagging. The number of hidden layers’ parameters are 2, 4, 6, 9, 12, and 15 with number of neurons for each layer 50. In order to pick the number of hidden layers, the model was tested with 50 neurons and then the hidden layer numbers were tested individually then compared. The results are shown in Figure 2 and Figure 3 below.

**Figure 2.** ANN hidden layer test results (lower is better).

**Figure 3.** ANN+Bagging hidden layer test results (lower is better).
From the results, the lowest RMSE achieved by 2 hidden layers on ANN architecture with or without Bagging. The testing RMSE are 7.195 using ANN and 6.843 using ANN architecture with Bagging. The worst performance for both architectures is when the hidden layer is 15. This shows that deep architecture does not always guarantee lower error.

The second experiment is to test the optimal number of neurons in the hidden layer. Following the previous results, the number of hidden layers for ReLU activation function is 2. Then, the hidden neurons are 10, 30, 50, 80, and 100. The results are shown in Figure 4 for ANN architecture without Bagging and Figure 5 ANN architecture with Bagging.

Architectural design experiments based on different numbers of neurons have different error patterns than previous experiments. Architecture with too little or too many neurons shows a high error rate. The lowest RMSE by ANN architecture without Bagging is 7.015 for the number of neurons 30 and 80 (Figure 4). The architecture of ANN with Bagging has the lowest error level for the number of neurons = 30, with RMSE 6.760 (Figure 5). Therefore, the experiment is continued by designing an architecture based on 30 neurons with the number of hidden layers be varied. The experiment was carried out using 2, 4, 6, 9, 12, and 15 hidden layers.
The third experiment shows the results of RMSE has regular patterns. The lowest RMSE by ANN architecture without Bagging is 7,204 (Figure 6) and ANN with Bagging has the lowest error level RMSE 6,818 (Figure 7). Both experiments show that the best architecture is indicated by 2 hidden layers and 30 neurons. These results are achieved after both architectures have been trained and validated.

The next experiment is to test both architectures with data that has not been used for training and validation before. Testing uses 40% of the overall data. The test results are shown in Figure 8 and Figure 9.

The testing results of ANN architecture without Bagging and ANN architecture with Bagging show that both architectures have a good performance based on validation test result. The level of error shows that the test results are lower than the validation test. Figure 8 shows that ANN has lower test results than validation tests except for architectures with 12 hidden layers 30 neurons. The best architecture is achieved by 2 hidden layers 30 neurons arrangement with RMSE 6,674.

As shown in Figure 9 none of the test results ANN with Bagging has same or higher RMSE than the validation test. The highest error occurs in 12 hidden layers and 30 neurons architecture with RMSE 7,215. This proves that the depth of learning in the deep learning architecture as indicated by the number of hidden layers does not always guarantee better results. The lowest RMSE of ANN with Bagging architecture is 6,385 with 4 hidden layers 30 neurons arrangement.

As part of research on intelligent concrete mix design models, this paper aims to prove the role of Bagging for improving ANN performance, especially H2O’s deep learning. Several architectural design experiments have been done by varying the number of hidden layers 2, 4, 6, 9, 12, 15 and the number of neurons 8, 10, 30, 50, 80, 100 with the ReLU activation function for data processing. The data was split into 0.6 and 0.4 ratio for training and testing data respectively. This training data was split again into train set and validation set using Cross Validation by splitting the data into k subset of data. k-1 subset was used for training and the remaining subset was used for validation. For numerical stability, the data then normalized based on the training data. From various experiments that have been done shows that an error rate test on architecture settings based on H2O’s deep learning ANN and H2O’s deep learning ANN with Bagging are as shown in Figure 10 below.
Figure 10. RMSE H2O’s deep learning ANN vs. H2O’s deep learning ANN with Bagging.

The experimental results show that 6 architectures based on H2O’s deep learning ANN have higher errors than the 6 architectures based on H2O’s deep learning ANN with Bagging. The best architecture is achieved by 4 hidden layers 30 neurons ANN with Bagging with RMSE value of 6,385 versus 6,674 by 2 hidden layers 30 neurons without Bagging. These results have proven the hypothesis of this study that Bagging has been able to reduce the error rate of the H2O’s deep learning ANN algorithm to make predictions on a concrete mix design model.

Bagging is used due to the possibility of the noisy data. In this study Bagging has the ability to solve problems caused by concrete mix design data noise. Bagging is short for bootstrap aggregating which its purpose is to increase the stability and avoid overfitting by reducing variance. Bagging has the ability to solve problems caused by data noise. The role of Bagging has been proven in all architectures with lower RMSE values than without Bagging.

4. Conclusion

Based on various experiments that have been carried out, it can be concluded that:

- Artificial Neural Network (ANN), especially H2O’s deep learning for predicting early age concrete mix design accuracy can be improved by using Bagging. As shown in the results above that the RMSE is lower when Bagging is used in the training process.
- The lowest RMSE that are able to be achieved in this research with Bagging is 6,385 achieved by 4 hidden layers 30 neurons architecture while it is 6,674 achieved by 2 hidden layers 30 neurons architecture without Bagging.
- The deepness of an Artificial Neural Network architecture not always guarantee optimum results. The model with 4 hidden layers is able to perform better than the model with higher number of hidden layers.

5. Future works

This model, as a new innovation in machine learning in civil engineering vocational education subject, can be used to various concrete mixture design and learning about concrete. The various composition of mixture design can be handled with care because this model learns from unseen condition.

Because this concrete mix design model is digital, it can be used for virtual learning. Virtual learning is very important to provide opportunities for students to explore without involving high costs due to material needs. Currently the author is conducting research to create a virtual concrete mix design learning model.

Acknowledgment

The authors would like to thank the Politeknik Negeri Semarang (Grant No.: SP DIPA-042.0.1.2.400997 / 2019) for the financial support.
References

[1] Chopra P, Sharma R and Kumar M 2016 Prediction of Compressive Strength of Concrete Using Artificial Neural Network and Genetic Programming Advances in Materials Science and Engineering 1–10

[2] Tangchirapat W, Rattanashotinunt C, Burananasing R and Jaturapitakkul C 2013 Influence of Fly Ash on Slump Loss and Strength of Concrete Fully Incorporating Recycled Concrete Aggregates Journal of Materials in Civil Engineering 25(2) 243–251

[3] Najimi M, Sobhani J and Pourkhorshidi A R 2012 A comprehensive study on no-slump concrete: From laboratory towards manufactory Construction and Building Materials 30 529–536

[4] Hüsken G and Brouwers H J H 2012 On the early-age behavior of zero-slump concrete Cement and Concrete Research 42(3) 501–510

[5] Wendne R, Vorel J, Smith J, Hoover C G, Bažant Z P and Cusatis G 2015 Characterization of concrete failure behavior: a comprehensive experimental database for the calibration and validation of concrete models Mater Struct 48(11) 3603–3626

[6] Ray N, Dedi P and Rizky D 2016 Correlation Coefficient Study of High Quality Concrete Compressive Strength Based on Age & Shape of SNI 03- 2847-2002 Standard Test Objects (Studi Angka Koefisien Korelasi Kuat Tekan Mutu Tinggi Berdasarkan Umur & Bentuk Benda Uji Standar SNI 03-2847-2002) 1(1)

[7] Kurt M, Kotan T, Gül M S, Gül R and Aydin A C 2016 The Effect of Blast Furnace Slag on the Self-compactability of Pumice Aggregate Lightweight Concrete Sadhana 41(2) 253–264

[8] Riahi S and Nazari A 2011 Compressive strength and abrasion resistance of concrete containing SiO2 and CuO nanoparticles in different curing media Sci. China Technol. Sci. 54(9) 2349–2357

[9] Hui C and Zhenyu L 2013 Research on the Experiment of Self-Leveling Concrete with Large Content of Fly Ash in 2013 Fifth International Conference on Measuring Technology and Mechatronics Automation 934–936

[10] Jain A, Jha S and Misra S 2008 Modeling and Analysis of Concrete Slump Using Artificial Neural Networks Journal of Materials in Civil Engineering - J MATER CIVIL ENG 20

[11] Yeh I C 2007 Modeling slump flow of concrete using second-order regressions and artificial neural networks Cement and Concrete Composites 29(6) 474–480

[12] Yeh I C, Lien C, Peng C H and Lien L C 2010 Modeling concrete strength using genetic operation trees 2010 International Conference on Machine Learning and Cybernetics 3 1572–1576

[13] Nikoo M, Sadowski L and Torabian F 2015 Prediction of Concrete Compressive Strength by Evolutionary Artificial Neural Networks Advances in Materials Science and Engineering

[14] Deng F, He Y, Zhou S, Yu Y, Cheng H and Wu X 2018 Compressive strength prediction of recycled concrete based on deep learning Construction and Building Materials 175 562–569