THE BALTIC JOURNAL
OF ROAD
AND BRIDGE
ENGINEERING
2022/17(3)

ISSN 1822-427X/eISSN 1822-4288
2022 Volume 17 Issue 3: 120–145
https://doi.org/10.7250/bjrbe.2022-17.572

AN IMPROVED NEURAL NETWORK MODEL FOR ENHANCING RUTTING DEPTH PREDICTION

SHUZHAN XU, JUNXIN YANG, CHANGBAI WANG*

School of Civil Engineering and Architecture,
Anhui University of Science and Technology, Huainan, China

Received 21 March 2021; accepted 13 November 2021

Abstract. Rutting is the main distress form of asphalt pavement, and its prediction accuracy is directly related to the reliability of the designed road. This research developed a neural network model to improve the prediction ability about the rutting of a pavement performance criterion and compared it with the multiple linear regression model and the existing neural network model. The neural network model is developed using the Keras module from the TensorFlow package in Python. Two reports generated by the National Cooperative Highway Research Program project 01-37A and the Long-Term Pavement Performance website records have been used as data sources for training the neural network model, which are reliable data preserved after years of monitoring. The input variables include the pavement thickness, service time, average annual daily traffic of trucks and the deformation of the asphalt concrete layer, granular base layer and subgrade layer. This experiment used 440 samples, of which 352 samples (80%) were used for model training and 88 samples (20%) for testing. The training results of the model reveal that the neural network model is significantly better than the multiple linear regression model, and the newly built neural network model performs better than another similar neural network in predictive performance. For the multiple linear

* Corresponding author. E-mail: chbwang@aust.edu.cn

Shuzhan XU (ORCID ID 0000-0002-7250-6962)
Junxin YANG (ORCID ID 0000-0002-6446-1207)
Changbai WANG (ORCID ID 0000-0003-1370-8449)

Copyright © 2022 The Author(s). Published by RTU Press

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

regression model, the correlation coefficient R^2 value between the measured and predicted in the testing set increased from 0.265 to 0.712. In contrast, it promotes from 0.867 to 0.902 for the neural network model.

Keywords: Mechanistic-Empirical Pavement Design Guide (MEPDG), multiple linear regression (MLR), neural network, nonlinear factors, overfitting, rutting prediction.

Introduction

Rutting is the main distress form of asphalt pavement, and it is also a significant criterion to measure the performance of asphalt pavement. The design method originated from the Mechanistic-Empirical Pavement Design Guide (MEPDG), was funded by the National Cooperative Highway Research Program Project (NCRHP) in 1996 and published in 2004, representing the most advanced pavement design guide in the United States. Because the depth of rutting is an essential criterion for measuring pavement performance, the accuracy of its prediction directly affects the absolute reliability of pavement design (Gong et al., 2018), which must be considered carefully. Prediction accuracy weighs heavily on the function of the model form and the data occupied during its calibration. Researchers have invested much time and effort in over 20 performance prediction models in the MEPDG (Schram & Abdelrahman, 2006). However, just as the viewpoint of Pierce & McGovern (2014), only three institutions or organisations indicated that the primary design tool for their pavement is MEPDG. A potential reason is that there are some disadvantages in using multiple linear regression (MLR) models to estimate the occurrence of distress and the service life of the pavement.

Rutting on the pavement comes from the permanent deformations (PD) of each structure layer or sub-layer under vehicle load and the combination of climate, environment and material factors (Ali et al., 2017). Of course, as mentioned above, MEPDG linearly combines the PD of all sensitive layers when calculating the total rutting. But in practice, many factors affect the PD of the pavement, including some that maintain an excellent linear relationship with it and other exceptions. Therefore, incorporating nonlinear functions solves this problem well when considering establishing a suitable pavement model to predict distress and improve accuracy.

Artificial intelligence technology, such as neural networks (NN) and deep learning, have been widely utilised to establish a model for predicting pavement performance. Najafi et al. (2019) applied an artificial neural network model to predict the rate of wet and dry vehicle crashes based on surface friction, traffic level, and speed limit. Esra'a & Abo-Qudais (2018) combined MLR analysis and feed-forward neural

network (FNN) to predict Hot Mixture Asphalt (HMA) creep behaviour with high accuracy in a short period. Bosurgi et al. (2019) employed an artificial neural network to help optimise the evaluation of asphalt pavement structural performance based on a multilayer feed-forward neural network (MFNN). Li et al. (2020) automatically classified pavement cracks by a convolutional neural network (CNN). Neural network performs very well in solving nonlinear problems.

Besides predicting pavement distress, NN similarly shows excellent estimation power in pavement structure performance (Gong et al., 2018). For example, researchers used NN to estimate the structural performance of the pavement based on roughness data collected from modern high-performance survey devices. The results proved a correlation between good precision of roughness and structural performance (Sollazzo et al., 2017). Simpson et al. (1995) and Shafabakhsh et al. (2015) fed several variables that affect rutting into the NN, such as asphalt concrete (AC) thickness, base thickness and porosity, which successfully predicted rutting depth. Compared to the traditional linear regression model, accuracy has been significantly improved. However, there are too many variables to be considered, and the number of neurons and the depth of the network are inappropriate, leading to overfitting problems. On their basis, Gong et al. (2018) improved the input variables and the number of network layers, and solved the problems caused by overfitting to a certain extent. But the network construction failed to carry out the neuron and depth design well, making the model performance considered to improve further.

The primary purpose of this research is to build an NN model to enhance the accuracy of rutting depth prediction through the TensorFlow extension package under the Python programming language. At the same time, the scikit-learn extension package is also used to realise the establishment of the MLR model and variable analysis. The results of the NN model were simultaneously compared to the MLR model and the existing NN model. By analysing and reducing the pavement performance measurement data collected from various states in the United States, two prediction models, MLR-pro and NN-pro, were established, and the models were discussed and analysed.

1. Rutting depth calculation

1.1. Calculation formula

Rutting is a plastic deformation caused by various loads during long-term work that has to consider its rate and accumulation. The

accumulation rate of plastic deformation is measured in the laboratory using repeated load PD triaxial tests for both AC mixtures and unbound materials. The laboratory-derived relationship is then adjusted to match the rut depth measured on the roadway (American Association of State..., 2008). For all AC mixtures, the MEPDG field calibrated form of the laboratory-derived relationship from repeated load PD tests is shown in Equation (1):

$$\begin{aligned} \Delta_{p(\mathrm{AC})} &= \varepsilon_{p(\mathrm{AC})} h_{(\mathrm{AC})} = \beta_{1r} k_z \varepsilon_{r(\mathrm{AC})} 10^{k_{1r}} n^{k_{2r}\beta_{2r}} T^{k_{3r}\beta_{3r}}, \\ k_z &= \left(C_1 + C_2 D\right) 0.328196^D, \\ C_1 &= -0.1039\left(H_{\mathrm{AC}}\right)^2 + 2.4868 H_{\mathrm{AC}} - 17.342, \\ C_2 &= 0.0172\left(H_{\mathrm{AC}}\right)^2 - 1.7331 H_{\mathrm{AC}} + 27.428, \end{aligned} \tag{1}$$

where $\Delta_{p(\mathrm{AC})}$ is the accumulated permanent or plastic vertical deformation in the AC layer/sublayer, mm; $\varepsilon_{p(\mathrm{AC})}$ is the accumulated permanent or plastic axial strain in the AC layer/sublayer, mm./mm; $h_{(\mathrm{AC})}$ is the thickness of the AC layer/sublayer, mm; β_{1r}, β_{2r}, β_{3r} are the local or mixture field calibration constants, and for the global calibration, these constants were all set to 1.0; k_z is the depth confinement factor; k_{1r}, k_{2r}, k_{3r} are the global field calibration parameters (k_{1r} = –3.35412, k_{2r} = 0.47910, k_{3r} = 1.56060); $\varepsilon_{r(\mathrm{AC})}$ is the resilient or elastic strain calculated by the structural response model at the mid-depth of each AC sublayer, mm/mm; n is the number of axle-load repetitions; T is the mix or pavement temperature, °F; D is the depth below the surface, mm; H_{AC} is the total AC thickness, mm.

The Equation (2) shows the field-calibrated mathematical Equation used to calculate plastic vertical deformation within all unbound pavement sublayers and the foundation or embankment soil (American Association of State..., 2008):

$$\Delta_{p(soil)} = \beta_{s_1} k_{s_1} \varepsilon_v h_{soil} \left(\frac{\varepsilon_0}{\varepsilon_r}\right) e^{-\left(\frac{\rho}{n}\right)^{\beta}}, \tag{2}$$

where $\Delta_{p(soil)}$ is the permanent or plastic deformation for the layer/sublayer, mm; k_{s_1} is the global calibration coefficients (2.03 – for granular materials and 1.35 – for fine-grained materials); ε_v is the average vertical resilient or elastic strain in the layer/sublayer and calculated by the structural response model, mm/mm; h_{soil} is the thickness of the unbound layer/sublayer, mm; ε_0 is the intercept determined from laboratory repeated load permanent deformation tests, mm/mm; ε_r is the resilient strain imposed in laboratory test to obtain material properties ε_0, β, and ρ, mm/mm; n is the number of axle-load applications; ε_{s_1} is the local calibration constant for the rutting in the unbound layers (the local calibration constant was set to 1.0 for the global calibration effort).

Calculation parameters β and ρ are determined by Equation (3):

$$\log\beta = -0.61119 - 0.017638(W_c),$$

$$\rho = 10^9 \left(\frac{C_0}{\left(1 - (10^9)^{\beta}\right)} \right)^{\frac{1}{\beta}}, \quad (3)$$

$$C_0 = Ln\left(\frac{a_1 M_r^{b_1}}{a_9 M_r^{b_9}} \right),$$

where W_c is the water content, %; M_r is the resilient modulus of the unbound layer or sublayer, kPa; a_1 and a_9 are the regression constants ($a_1 = 0.15$ and $a_9 = 20.0$); b_1 and b_9 are the regression constants ($b_1 = 0$ and $b_9 = 0$).

Just like the above formula, it was necessary to calibrate some parameters to ensure the best combination. After selecting the appropriate value of k_{2r} and k_{3r} from the Equation (1), use the parameter minimisation function of the Equation (4) to determine the value of k_{s_1} in the Equation (2). The final result of Δ_{total} is obtained after calculating the linear combination utilizing Equation (5) where $\boldsymbol{k_{ij}}$ represents different calibration coefficients matrix; Δ_{layers} is the plastic deformation value matrix of each layer, and both are one-dimensional column vectors.

$$\arg\min \sum_{k_{1r}, k_{2r}, k_{3r}, k_{s1}} \left(\Delta_{\text{total}} - \Delta_{\text{actual}}\right)^2, \quad (4)$$

$$\Delta_{\text{total}} = k_{ij}^T \Delta_{layers}. \quad (5)$$

1.2. Regression metric evaluation on calibration

After going through the building steps of the above model, the next step is to evaluate the performance of the model. The metrics selected this time are the correlation coefficient R^2 and the standard error S_e. R^2 is also called the coefficient of determination, which defines how many of the regression mel predicts future samples (Wang et al., 2021). Besides, it also means that the model explains the variance ratio of the predicted response variable (Haroon & Clustering, 2017). Its value ranges from 0 to 1.0, where 1.0 is the best. S_e reflects the deviation of the predicted value from the observed value. The less the S_e, the better the model fitting performance. Equation (6) and Equation (7) describe how to calculate R^2 and S_e:

$$R^2\left(y,\hat{y}\right)=1-\frac{\sum_{i=0}^{n_{\text{samples}}-1}\left(y_i-\hat{y}_i\right)^2}{\sum_{i=0}^{n_{\text{samples}}-1}\left(y_i-y'\right)^2},\ y'=\frac{1}{n_{\text{samples}}}\sum_{i=0}^{n_{\text{samples}}-1}y_i, \tag{6}$$

$$S_e=\sqrt{\frac{\sum_{i=0}^{n_{\text{samples}}-1}\left(y_i-\hat{y}_i\right)^2}{n_{\text{samples}}}}, \tag{7}$$

where $\hat{y}_i$ is the estimated target output of the ith sample; y_i is the corresponding (correct) target output; n_{samples} is the number of samples.

Table 1 shows the achievements of different organisations or individuals in optimizing linear regression models. As in Table 1, only the R^2 of the American Association of State... (2008), Kaya (2015), Mallela et al. (2009a, 2009b), and Schram & Abdelrahman (2010) exceed 0.5, indicating that most linear models perform poorly. As shown in Table 1, when developing a linear regression model to predict the value, the impact of nonlinear factors on the target value was ignored. The final model was inaccurate, and the R^2 deviated from 1.0. Such a model is difficult to explain the problem, let alone used to predict rutting.

Table 1. A list is collected from predecessors on rutting calibration

Reference	Region	R^2	S_e, mm	Samples
American Association of State...(2008)	National	0.577	2.72	334
Applied Research Associates (2004b)	National	0.399	3.07	387
Sun et al. (2015)	Kansas	0.240	0.51	132
Muthadi & Kim (2008)	North Carolina	0.340	2.82	161
Darter et al. (2014)	Arizona	0.181	3.12	616
Souliman et al. (2010)	Arizona	0.059	24.60	177
Jadoun & Kim (2012)	North Carolina	0.150	3.09	235
Schram & Abdelrahman (2010)	Nebraska	0.550	1.02	398
Smith & Nair (2015)	Virginia	0.237	1.93	236
Kaya (2015)	Iowa	0.630	1.27	299
Darter et al.(2009)	Utah	0.097	3.94	86
Mallela et al. (2009b)	Missouri	0.520	1.30	183
Mallela et al.(2009a)	Ohio	0.630	0.36	101
Bennert & Williams (2009)	Colorado	0.417	3.73	137

2. Artificial neural networks

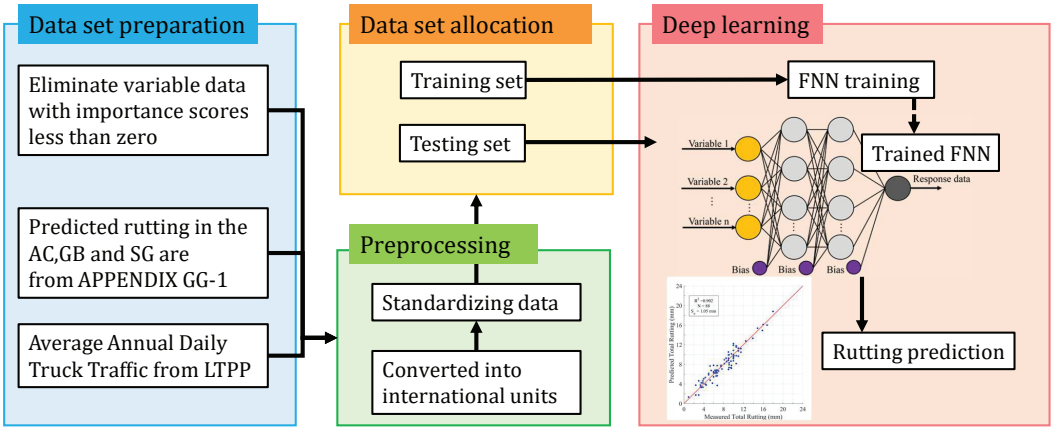
Inspired by biology, the NN is composed of a series of simple units closely related. Each unit has a certain number of real inputs and actual individual output (Dayhoff & DeLeo, 2001; Hopfield, 1988). An essential use of NN is to accept and process complex inputs from sensors and perform adaptive learning. The neural network algorithm simulates a biological neural network and is a pattern-matching algorithm usually used to solve classification and regression problems.

2.1. The framework of the neural network method

This section explains the overall establishment of the NN model for estimating pavement rutting using sample data collected by the NCHRP and the Federal Highway Administration (FHWA) Long-Term Pavement Performance (LTPP). It includes 11 columns of data, such as predicted rutting in the pavement structure layers, time of service, original air void, and resilient modulus. The data used for training is the most original without being modified by algorithms to prevent distress to the practicality of the model.

Figure 1, the NN framework consists of four parts:

1. the NN architecture;
2. data set preparation for preprocessing;
3. preprocessing and data set allocation;
4. rutting reckoning through deep machine learning.



Note: *Applied Research Associates (2004a);
**Applied Research Associates (2004c)

Figure 1. The overall framework of the neural network method

In the first step, the NN architecture is designed to be used by converting pavement performance parameters (i.e., historical record data) into a variable spreadsheet. The second step is to use the actual road data provided in the spreadsheet to generate a sample table. Before calling the corresponding sample training, it is necessary to perform feature standardisation, data cleaning, and feature extraction. The purpose is to hope that the value of each feature is obtained after the data is processed and float within a low range. In the third step, the processed data is divided into data sets, employing one part as the training set for model building and the other as the testing set for model evaluation. In the last step, use the prepared data set to train the NN model and then estimate the rutting depth for the untrained samples.

2.2. Proposed neural network architecture

Figure 2 shows the NN system structure used for rutting estimation, consisting of some hidden input and an output layer. Every layer of communication must add a bias. The input layer is the pavement performance data read from the variable spreadsheet, and then the data is passed to the first hidden layer through feature transformation. The first hidden layer is passed to the second hidden layer through feature transformation, and so on, until all the hidden layers are traversed. This process is also called forward propagation. A regularisation penalty must be introduced during forwarding propagation to prevent the model from overfitting. Finally, the results of the NN are output through the output layer.

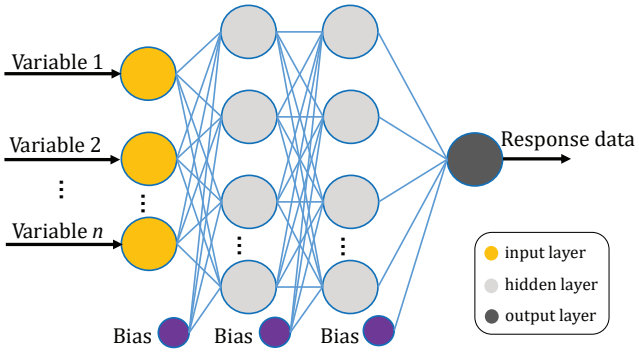


Figure 2. Baseline network architecture

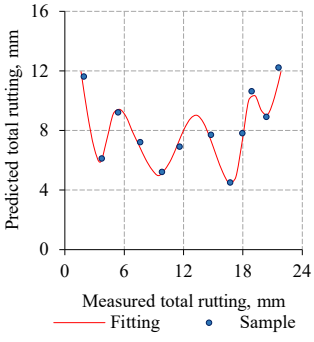


Figure 3. Model overfitting

2.3. Standardisation and regularisation

Since the different attributes of the sample, this time, has different magnitudes, to eliminate the extent of influence, the data needs to be standardised so that the characteristics of the sample are scaled to a specified range. The Equation (8) shows the standardised formula:

$$Z = \frac{X - X_{\text{mean}}}{std(X)}, \tag{8}$$

where Z is the standardised data; X is the original data; X_{mean} is the mean of the original data; $std(X)$ is the standard deviation of the original data.

As shown in Figure 3, overfitting refers to the phenomenon that the model positively fits the data set, causing the model to fail to output high-precision predictions for untrained data. It happens when the features transformed from the original data are fed to the NN, and the network fits the noisy data by remembering these features instead of learning them (Dietterich, 1995; Jeong et al., 2020). Overfitting models have low bias and high variance and lack the generalisation of untrained data sets, resulting in inaccurate predictions (Jeong et al., 2020). Regularisation is to scale a particular sample norm to unit one and select weight parameters with stronger generalisation ability. That is, all tend to be stable. Usually, the L2 regularisation is chosen as in Equation (9), which takes the square sum of the weight parameters and penalises it more strongly.

$$J = J_0 + \alpha w^T w, \tag{9}$$

where J is a regularised loss function; J_0 is the sum of squared error between actual values and prediction; α is the penalty coefficient; w is the weight vector.

2.4. Activation function

In the NN architecture, after inputting the data, the features are transformed ler by layer, and finally, the error is evaluated through the loss function, forward propagation. Next, choose an appropriate optimisation method, load backpropagation, go from back to front, and update the weight parameters layer by layer. If there is a linear transformation among layers, then only one set of weight parameters is needed to represent all the remaining products. This single group of weight parameters is impossible to explain practical principles. On the one hand, the NN must solve linear and nonlinear problems (Hopfield, 1988); on the other hand, it needs to filter the transformed features so that the valuable weight features play a more significant role in the activation function.

Common activation functions include the sigmoid function and rectified linear unit (ReLu) function (Ramachandran et al., 2017). As shown in Figure 4, a derivation is required layer by layer in the backpropagation process. The sigmoid derivative of the function is close to zero when the value is significant. Since backpropagation is carried out layer by layer, if the gradient of a layer is zero, all the network layers behind it are updated restrictedly (Wang et al., 2022), which is also the biggest problem with the sigmoid function (Bengio et al., 1994).

The ReLu function is a continuous piecewise function consisting of two parts. The value of the function on the negative semi-axis of x is all zero, and the function expression on the positive semi-axis of x is $y = x$. The gradient is kept active by preventing gradient saturation and providing better calculations through linear operations. For the input x, the output is zero when it is less than zero, and when it is higher than zero, the output is equal to the input itself. It is also a nonlinear function, but it solves the problem of the disappearance of the gradient, and the calculation is straightforward, speeding up the iteration speed of the network.

2.5. Optimizing with adaptive moment estimation

The predicted value of the regression equation and the actual value of the sub-sample are common unequal, indicating a difference between the actual value of the data and the predicted value, usually called an error term. The objective function, introduced in Equation (10), is considered optimizing to reduce the error value. Equation (10) calculates the extreme value of its partial derivative to obtain the lowest parameter θ. Therefore, the extreme value is found as long as the appropriate learning rate is selected and the gradient is calculated. This process is

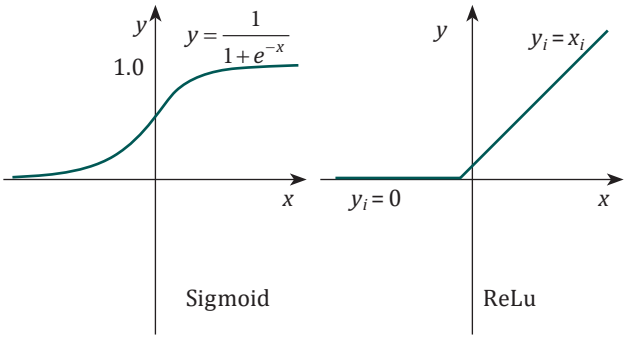


Figure 4. Activation function

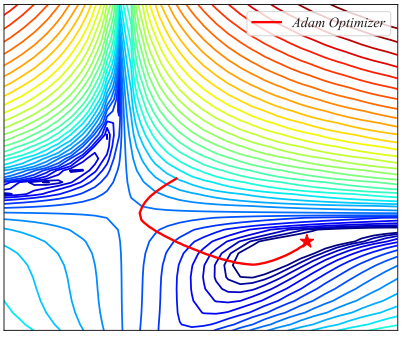


Figure 5. Adam visualisation

also called the gradient descent algorithm. The batch gradient descent algorithm requires consideration of all sample data. Every iteration of the optimisation calculation needs to calculate all the samples in the formula. Still, the iteration speed considers being plodding if the number of samples is huge. Facts have proved that the Stochastic Gradient Descent (SGD) algorithm is an effective optimisation tool and is essential in many successful machine learning cases (such as advanced deep learning technologies) (Deng et al., 2013; Kingma & Ba, 2014). It uses only one sample simultaneously, dramatically improving the iteration speed. However, SGD also has the following problems:

1. difficulty selecting a suitable starting learning rate;
2. the pre-specified adjustment rules limit the learning rate adjustment strategy;
3. the same learning rate is applied to each parameter;
4. the optimisation process of the highly non-convex error function tends to fall into many local sub-optimal solutions or saddle points.

$$J(\theta)=\frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right)-y^{(i)}\right)^2, \tag{10}$$

where m is the number of samples in the entire data set; $h_\theta(x^{(i)})$ is the regression equation; $y^{(i)}$ is the actual value.

For the problems of simple SGD, Duchi et al. (2011) released the adaptive gradient (AdaGrad) optimisation algorithm that adjust different learning rates for each parameter and use lower parameters for frequently changing parameters. The step size is updated, and the sparse parameter is updated with a larger step size. But the main flaw comes from the continuous accumulation of the square of the gradient in the denominator term, and the time step increases. The denominator term becomes larger and larger, which eventually causes the learning rate to shrink to too low to be effectively updated. Root mean square prop (RMSProp) is an algorithm mentioned by Hinton et al. (2012) in their teaching plan that adjusts the learning rate by combining the exponential moving average of the gradient square to deal with the problem of non-convergence caused by the stable (non-stationary) objective function.

Two scholars, Kingma & Ba (2014), proposed the adaptive moment estimation (Adam) optimiser in 2014, as shown in Figure 5, combining the advantages of AdaGrad and RMSProp optimisation algorithms. Based on the AdaGrad and RMSProp algorithms, Adam calculates the update step size by comprehensively considering the first-moment estimate (average of the gradient) and the second-moment estimate (the uncentered variance of the gradient). Adam is considered to be

an optimiser with better default performance in many cases, mainly including the following significant advantages:

1. weakening the influence of parameter update by the gradient transformation;
2. very few adjustments are required to ensure that the hyperparameters have good interpretability;
3. the step size is limited to a specific range (initial learning rate);
4. suitable for scenarios with large amounts of data or parameters;
5. solving the unstable target function problem is excellent;
6. assisting in solving the problem of gradient sparsity or gradient with considerable noise.

3. Data acquisition and input

Plentiful and reliable data sets are essential for a successful supervised learning approach. However, obtaining a large amount of high-quality labelled data is difficult in practice but feeding such data into the neural network is crucial to the training effect of the model. In this NN training, historical monitoring data collected from the LTPP database of a total of 88 sites in 28 states were used to improve the reliability of the model training results. These data are currently available in the two reports issued by NCHRP Project 01-37A, Appendix GG-1 (Applied Research Associates, 2004a) and Appendix EE-1 (Applied Research Associates, 2004c). Other traffic data are accessed directly by visiting the LTPP InfoPave website (https://infopave.fhwa.dot.gov/). Figure 6 shows the source of these data.

The data extracted from Appendix EE-1 (Applied Research Associates, 2004c) and Appendix GG-1 (Applied Research Associates, 2004a) are listed in Table 2 and Note.

Table 3, including ten columns of values of the feature, such as the rutting of AC, granular base (GB) and subgrade (SG) layers, and a column of target values representing the actual rutting depth. The selection and sources of specific variables are listed in Table 4.

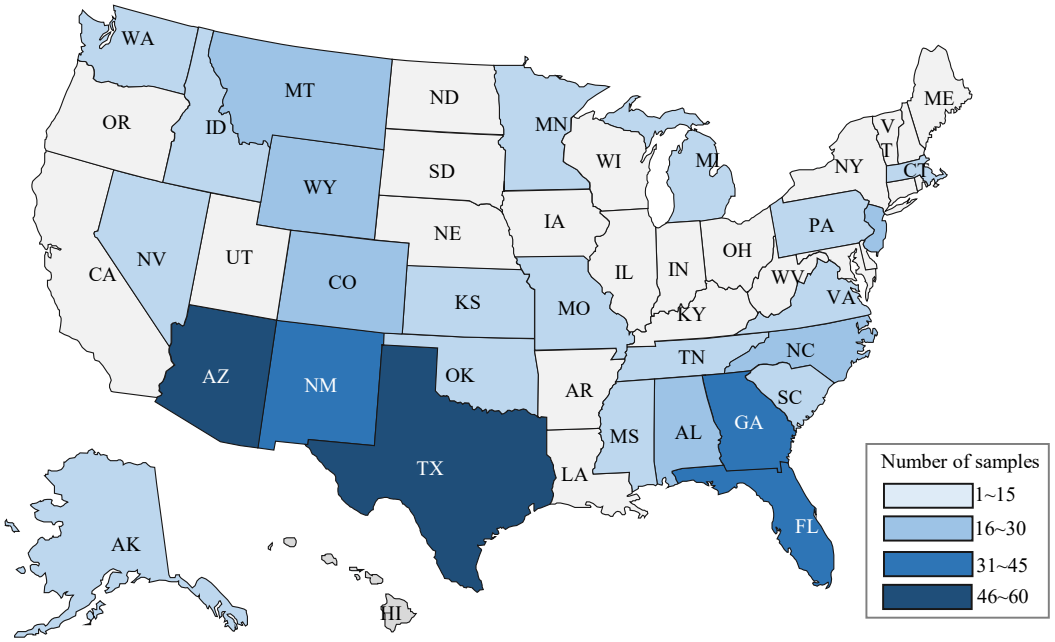


Figure 6. Distribution of data sources

Table 2. Part of the data from Table C-58 in Appendix EE-1*

Section	Month	Time, hours	Predicted							Measured			
			AC	GB	SG	Total	AC, %	GB, %	SG, %	AC	GB	SG	Total
11 001	4/5/1989	103	0.434	0.176	0.12	0.730	59.47	24.13	16.40	0.199	0.081	0.055	0.335
11 001	2/12/1991	125	0.485	0.180	0.122	0.787	61.62	22.82	15.56	0.194	0.072	0.049	0.315
11 001	4/2/1992	139	0.501	0.181	0.124	0.806	62.18	22.46	15.36	0.184	0.066	0.045	0.295
11 019	5/15/1989	32	0.140	0.025	0.135	0.300	46.73	8.26	45.00	0.239	0.042	0.230	0.512
11 019	4/16/1990	43	0.164	0.026	0.140	0.329	49.79	7.78	42.43	0.274	0.043	0.234	0.551
11 019	1/15/1991	52	0.186	0.026	0.143	0.355	52.29	7.40	40.31	0.288	0.041	0.222	0.551
11 019	3/31/1992	66	0.201	0.027	0.146	0.374	53.72	7.11	39.17	0.296	0.039	0.216	0.551
11 019	3/22/1994	90	0.232	0.031	0.151	0.414	56.00	7.43	36.57	0.331	0.044	0.216	0.591
11 019	1/8/1996	112	0.260	0.034	0.155	0.449	57.89	7.52	34.58	0.342	0.044	0.204	0.591
11 019	1/23/1998	136	0.280	0.034	0.158	0.472	59.28	7.23	33.50	0.420	0.051	0.237	0.709

Note: *Applied Research Associates (2004c).

Table 3. Part of the data from Table 4 in Appendix GG-1*

Section	Number of layers	Layer number	Layer type	Representative thickness, mm	Comments
11 001	5	1	AC	40.64	AC surface layer combined with AC layer beneath
		2	AC	40.64	
		3	GB	157.48	
		4	GS	485.14	
		5	SS		BR above 1.524 m (USGS-Regional)
11 019	6	1	AC	27.94	LTPP data shows 0.9″
		2	AC	66.04	
		3	AC	76.20	
		4	GB	139.70	
		5	SS	6400.80	
		6	BR		BR above 6.400 m (Harold Von Quintus Report)

Note: *Applied Research Associates (2004a).

Table 4. Explanation of input variables

Variable	Explanation	Source
AC_pre	Rutting expected with AC	Appendix GG-1*
GB_pre	Rutting expected with GB	Appendix GG-1*
SG_pre	Rutting expected with SG	Appendix GG-1*
Time	Service time	Appendix GG-1*
h_Tot	The total thickness of pavement layers	Appendix EE-1**
h_AC	AC layer thickness	Appendix EE-1**
h_GB	GB layer thickness	Appendix EE-1**
MR_SG	SG layer resilient modulus	Appendix EE-1**
MR_GB	GB layer resilient modulus	Appendix EE-1**
Voids	Original air voids in AC	Appendix EE-1**
$AADT_{truks}$	Average annual daily trucks traffic	LTPP

Note: *Applied Research Associates (2004a); **Applied Research Associates (2004c)

4. Neural network model construction

Four hundred forty data set samples were used to develop the rutting prediction model. For the NN training process, 352 samples were randomly selected (80% of the data). In comparison, for the NN testing processes, the remaining 88 samples (20% of the data) were divided equally (Esra'a & Abo-Qudais, 2018). This neural network is called NN-pro. Because of the limited sample data, it is very wasteful to set up the validation set separately. To fully use all the existing data, the cross-validation method in machine learning has been introduced that divides the training set into multiple parts. After numerous attempts, four-fold cross-validation is finally used, as shown in Figure 7. The process must be divided into four steps when verifying a particular result. In the first step, use the first three as the training set and the last as the validation set to get a result. By analogy, each time uses another copy as the validation set and the rest as the training set. After four steps, four results are obtained, each corresponding to each small part, and the combination contains all the data in the original training set. Then the final four results are averaged to obtain the final model evaluation result. Cross-validation plays a significant role in evaluating the model and making the results more accurate.

When configuring the network, the appropriate hyperparameters, such as the number of layers, the number of neurons in each layer, learning rate, batch and epoch, need to be determined. Different hyperparameters have different effects on the model. Regularisation parameters, the number of layers of the neural network, and the number of neurons mainly affect the classification accuracy of the neural network; learning rate and activation function primarily

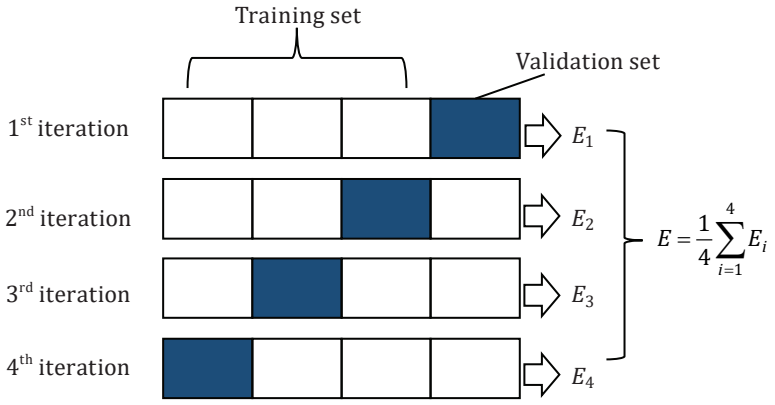


Figure 7. Cross-validation

affect the learning speed (Huang et al., 2004). To build the best model configuration, try to find the optimal number of hidden layers and neurons from one to five, and finally exploit a random search to determine the best model (Bergstra & Bengio, 2012; Huang et al., 2004, 2006; Pao et al., 1994). The specific structure of the NN-pro and the selection of hyperparameters are shown in Table 5.

Figure 8 shows how the error generated during model training decreases with the accumulation of iterations. The results show that the standard error of the network dramatically decreases as the learning process deepens. When the error value stops changing, the entire model training ends. The results show that after about 2000 iterations, the error generated by the model stabilises, and the standard error in the training set is only 1.034, implying that NN performs well in predicting rutting depth.

Table 5. Neural network-pro configuration and hyperparameter selection

Layer	IL	HL1	HL2	HL3	HL4	OL
Neurons	10	100	50	20	10	1
Activation function	ReLu	ReLu	ReLu	ReLu	ReLu	–
Regularisation penalty	–	L2	L2	L2	L2	–
Learning rate	–	0.001	0.001	0.001	0.001	–

Note: IL – input layer; HL – hidden layer; OL – output layer.

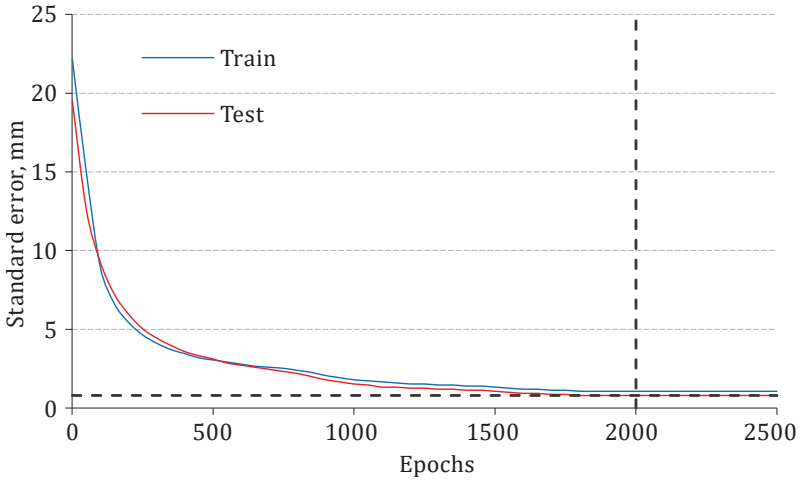
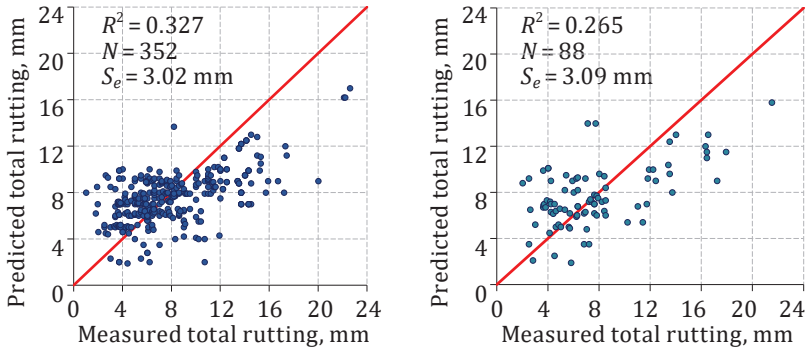


Figure 8. Rutting neural network performance plot

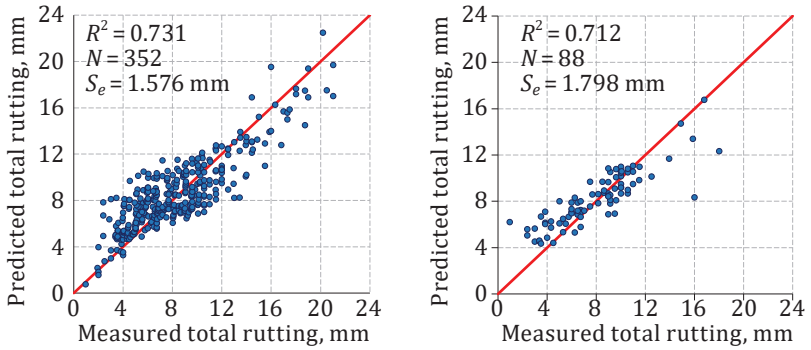
5. Results and analysis

As illustrated in Equation (5), the rutting depth calculation method given by MEPDG is to calculate the PD from the AC layer, GB layer, and SG layer, integrating the deformation of all layers through a linear combination equivalent to fitting ten zero-intercept rutting feature variables to the MLR model (Gong et al., 2018). Consequently, to better evaluate the performance of the NN, it is important to build a multiple linear regression model MLR-pro with the same input variables compared to the two types of models. Of course, the MLR-pro model is also compared to the models MLR20 and NN20 established by Gong et al. (2018) to show the performance of model optimisation and choose R^2 and S_e as evaluation indicators.



a) training **b)** testing

Figure 9. Measured and predicted rutting of MLR20



a) training **b)** testing

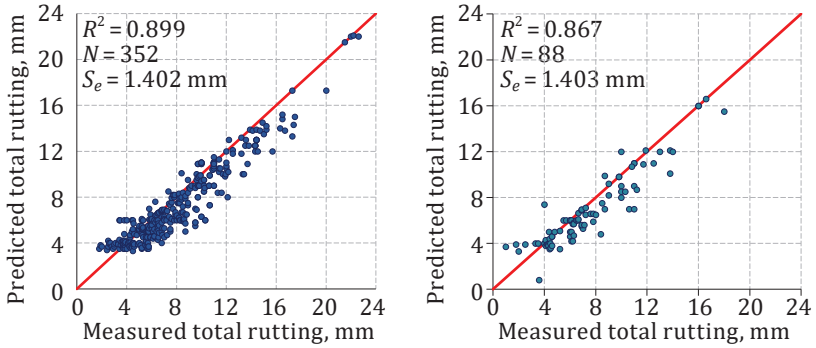
Figure 10. Measured and predicted rutting of MLR-pro

5.1. Multiple linear regression model

Figure 9 and Figure 10 show the actual and predicted rutting from MLR20 and MLR-pro. R^2 and S_e in training set with 352 sample data increased by 123.5% and decreased by 47.8%, respectively. In a total of 88 sample data from the testing set, the R^2 and S_e in the MLR20 model are 0.265 and 3.09 mm, respectively. The MLR-pro showed R^2 equal to 0.712 and S_e equal to 1.798 mm. The results show that compared to MLR20, the R^2 of MLR-pro has increased by 168%, and the S_e has decreased by 41.8%.

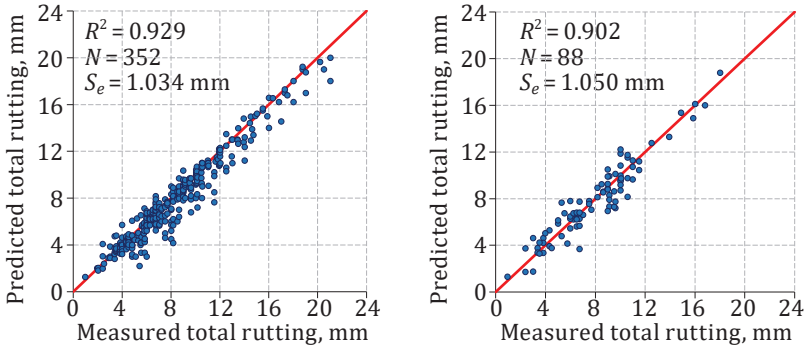
5.2. Neural network model

Figure 11 and Figure 12 show the actual performance of the NN20 and NN-pro models. For the NN20 model, R^2 and S_e in the training set



a) training **b)** testing

Figure 11. Measured and predicted rutting of NN20



a) training **b)** testing

Figure 12. Measured and predicted rutting of NN-pro

are 0.899 and 1.402 mm, respectively, and R^2 and S_e in the testing set are 0.867 and 1.403 mm, respectively. R^2 and S_e in the training set and testing set of the NN-pro model are 0.929 and 1.034 mm, and 0.902 and 1.050 mm, respectively. After testing, the R^2 of NN-pro is 3% and 4% higher than that of NN20 in the training and testing set, and the S_e are reduced by 26% and 25%. Compared to the MLR-pro model, the R^2 increased by 26.7%, and the S_e decreased by 41.6% in the testing set. Similarly, the R^2 has exceeded 0.9 by NN-pro, which means that more than 90% of the samples are generally in line with the prediction of this model, implying that NN-pro has good generalisation ability. The specific data comparison is listed in Table 6.

5.3. Model analysis

Choosing the correct variables to feed the NN is also very important because the NN model is prone to overfitting. When the input features of the NN are too many, the dimension is too large, making the model excellent performance on the training set but unsatisfactory on the testing set. NN-pro optimises the input variables based on NN20, selects ten variables to feed neurons, and combines Adam to optimise the loss function so that the residual of the model is significantly reduced. Although ten input variables are used in this model, their status is unequal, and their impact on the entire model varies in importance. Using the random forest algorithm in ensemble learning makes it easy to get the degree of the determinism of different variables in the model. Figure 13 depicts the importance of different variables to the results. The main influences on the rut depth are the PD of the AC layer

Table 6. Model performance

Model		Number of data sets	R^2	S_e, mm
MLR20	Training	352	0.327	3.020
	Testing	88	0.265	3.090
MLR-pro	Training	352	0.731	1.576
	Testing	88	0.712	1.798
NN20	Training	352	0.899	1.402
	Testing	88	0.867	1.403
NN-pro	Training	352	0.929	1.034
	Testing	88	0.902	1.050

and $AADT_{truks}$ because the surface layer is directly loaded, and the pavement mechanism undergoes cumulative plastic deformation under the repeated rolling action of the vehicle. Deformation of the GB layer plays a more critical role in developing rutting. For this kind of pavement structure composed of loose materials, slip and displacement occur among particles, causing it to lose bearing capacity and trigger distress.

In summary, NN is much better than MLR in terms of rutting model prediction performance and has good generalisation performance. The difference between the models is essentially the selection of different weights and biases, which means that the final result of the model is entirely determined by the weight and bias parameters. Multiple linear regression is a vital algorithm in machine learning, and the predictive ability of the model is also relatively strong. However, linear correlations among variables and nonlinear influencing factors are common in practical applications. The activation function in NN achieves this function. The input-to-loss calculation process is completed through forwarding propagation, optimised parameters through backpropagation, and a suitable weight or bias is obtained. The original perception is that the output of each layer and the input of the previous layer constitute a linear function. Even if the number of NN layers increases, the output is a linear combination of the input. However, suppose an activation function that coordinates nonlinear factors is introduced. In that case, there are both linear connections among neurons and nonlinear connections, so such NN applies to any nonlinear function or model. Figure 14 plots the correlation analysis between the first three variables with more significant influence and the actual value of the rutting. The results show that although the depth of the rutting does increase with the increase of the independent variable, it is hard to explain the relationship among them with a linear relationship, so this is also the necessity of introducing nonlinear factors.

However, the proposed NN model still has some shortcomings. The sample points near the centerline are a little sparse, especially on the test set, limiting the model from reaching a higher level of prediction. The main reason for this problem is insufficient training sample data sets, only 440 groups. In the future, high-performance test equipment will be considered available to collect more accurate road performance data and expand the data set to develop the potential of the model.

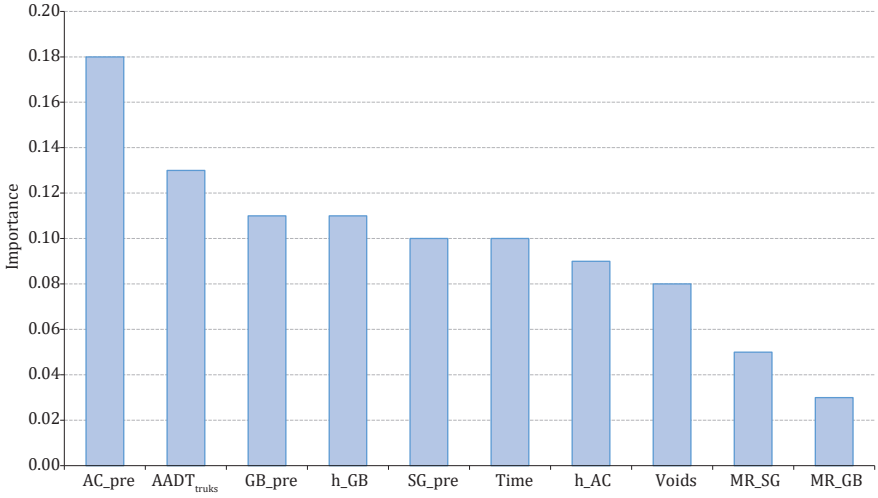


Figure 13. Variable importance

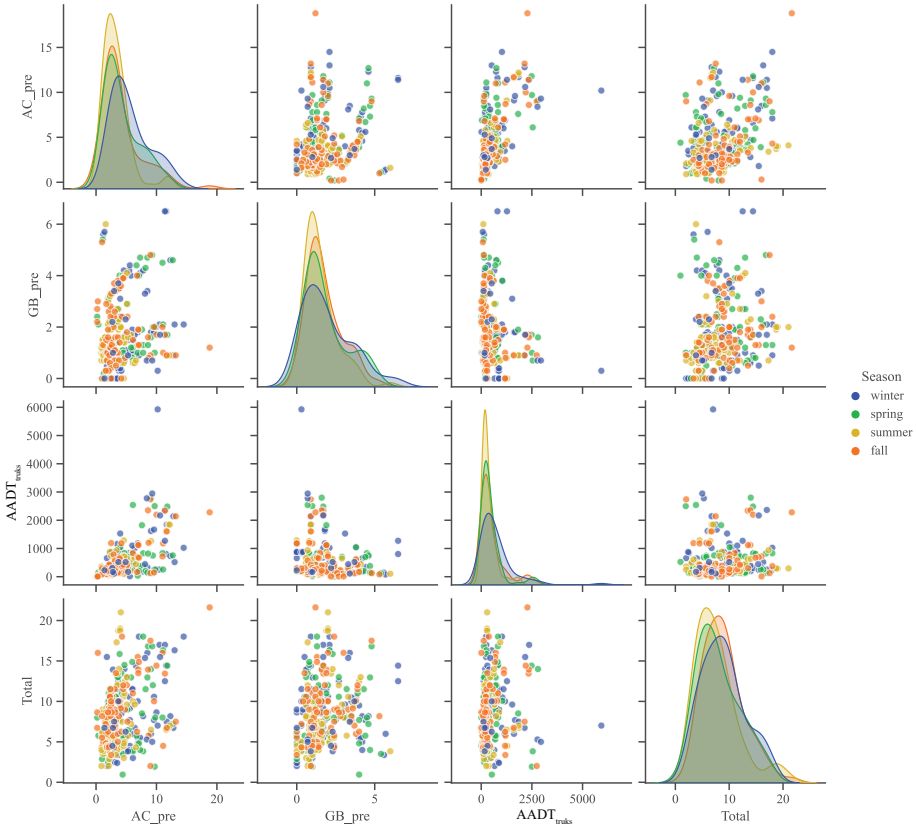


Figure 14. A plot matrix for the variables considered

Conclusions

This paper carefully analyses and discusses the construction of artificial neural networks and how to use a neural network model to estimate the depth of rutting of asphalt pavement structure performance. By considering relevant impact factors, the input variables of the neural network model are controlled to maximise the accuracy and reliability of the final result. Also, the hyperparameters of the neural network model are adjusted in detail to optimise the accuracy of the network model. The collected data comes from the two final reports of the National Cooperative Highway Research Program Project 01-37A and the data downloaded from the Long-Term Pavement Performance InfoPave website. They are based on reliable data accumulated over many observations, making the model very convincing. The following conclusions are drawn through the analysis and discussion of the results.

1. This model is built on continuous data learning through deep learning algorithms in artificial intelligence to calibrate the rutting calculation function locally that accurately predicts the rutting distress of the asphalt pavement. It provides a reference for obtaining pavement distress information and facilitates more specific periodic measurements and road maintenance.

2. When using the multiple linear regression model MLR-pro to predict rut depth, compared to MLR20, in the testing set, the correlation coefficient R^2 of MLR-pro has increased by 168%, and the standard error S_e has decreased by 41.8%. The main reason is that the control of input variables in the model effectively prevents the occurrence of overfitting.

3. Using the same neural network (NN) model, compared to the MLR-pro model, the correlation coefficient R^2 increased by 26.7%, and the standard error S_e decreased by 41.6% in the testing set. By changing different network structures and reducing input variables, in the testing set, the correlation coefficient R^2 of NN-pro is 4% higher than that of NN20, and the standard error S_e is reduced by 25%. The correlation coefficient R^2 of NN-pro has exceeded 0.9 by NN-pro, indicating that NN-pro has good generalisation ability.

4. The correlation analysis among the variables reveals that different variables have different degrees of influence on the model, and some variables show a nonlinear relationship. These results also confirm the previous conjecture that a single linear transmission cannot explain more deeply and comprehensive data.

Overall, the neural network model is better than the multiple linear regression model in all aspects. The reason is that the relationship transmitted by the neural network model in the training process contains linear and nonlinear so that it can fit the model better**.** The

outstanding performance of the neural network model optimises the asphalt pavement design. It promotes the development of precise and intelligent road maintenance that helps improve the method of the Mechanistic-Empirical Pavement Design Guide. In the future, it is recommended to use high-performance test equipment to collect more accurate road performance data and expand the data set to develop the potential of the model.

Acknowledgements

The authors sincerely thank the School of Civil Engineering and Architecture, Anhui University of Science and Technology, for their financial support and technical contribution.

Funding

This work was supported by the 2020 Graduate Innovation Fund Project of Anhui University of Science and Technology #1 under Grant No. 2020CX2021 and the 2021 Graduate Innovation Fund Project of Anhui University of Science and Technology #2 under Grant No. 2021CX2039.

Disclosure statement

Competing financial, professional, or personal interests from other parts are inexistent.

REFERENCES

American Association of State Highway and Transportation Officials (AASHTO) (2008). Mechanistic-empirical pavement design guide: a manual of practice. *AASHTO: Washington, DC, USA*.

Ali, Y., Irfan, M., Ahmed, S., & Ahmed, S. (2017). Permanent deformation prediction of asphalt concrete mixtures–a synthesis to explore a rational approach. *Construction and Building Materials*, *153*, 588-597. https://doi.org/10.1016/j.conbuildmat.2017.07.105

Applied Research Associates (ARA) (2004a). *Guide for Mechanistic-Empirical Design of New and Rehabilitated Pavement Structures, Final Report Appendix GG-1: NCHRP 1-37A*. Transporation Research, Board of the National Academies, Washington D.C.

Applied Research Associates (ARA) (2004b). *Guide for Mechanistic-Empirical Design of New and Rehabilitated Pavement Structures, Final Report NCHRP*

1-37A. Transporation Research Board of the National Academies, Washington D.C.

Applied Research Associates (ARA) (2004c). *Input Data for the Calibration and Validation of the Design Guide for New Constructed Flexible Pavement Sections, Final Report Appendix EE-1: NCHRP 1-37A*. Transporation Research Board of the National Academies, Washington D.C.

Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, *5*(2), 157-166. https://doi.org/10.1109/72.279181

Bennert, T., & Williams, S. G. (2009). Precision of AASHTO TP62-07 for use in mechanistic–empirical pavement design guide for flexible pavements. *Transportation Research Record*, *2127*(1), 115-126. https://doi.org/10.3141/2127-14

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, *13*(2).

Bosurgi, G., Pellegrino, O., & Sollazzo, G. (2019). Optimizing artificial neural networks for the evaluation of asphalt pavement structural performance. *The Baltic Journal of Road and Bridge Engineering*, *14*(1), 58-79. https://doi.org/10.7250/bjrbe.2019-14.433

Darter, M., Titus-Glover, L., & Von Quintus, H. (2009). Draft user's guide for UDOT mechanistic-empirical pavement design guide. *Report No. UT-09.11a, Applied Research Associates, Inc, Champaign, Illinois, USA*.

Darter, M. I., Von Quintus, H., Bhattacharya, B. B., & Mallela, J. (2014). *Calibration and implementation of the AASHTO mechanistic-empirical pavement design guide in Arizona* (No. FHWA-AZ-14-606). Arizona. Dept. of Transportation. Research Center.

Dayhoff, J. E., & DeLeo, J. M. (2001). Artificial neural networks: opening the black box. *Cancer: Interdisciplinary International Journal of the American Cancer Society*, *91*(S8), 1615-1635. https://doi.org/10.1002/1097-0142(20010415)91:8+<1615::AID-CNCR1175>3.0.CO;2-L

Deng, L., Li, J., Huang, J. T., Yao, K., Yu, D., Seide, F., ... & Acero, A. (2013, May). Recent advances in deep learning for speech research at Microsoft. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing* (pp. 8604-8608). IEEE. https://doi.org/10.1109/ICASSP.2013.6639345

Dietterich, T. (1995). Overfitting and undercomputing in machine learning. *ACM Computing Surveys (CSUR)*, *27*(3), 326-327. https://doi.org/10.1145/212094.212114

Duchi, J., Hazan, E., & Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, *12*(7).

Esra'a, I. A., & Abo-Qudais, S. A. (2018). Modeling of creep compliance behavior in asphalt mixes using multiple regression and artificial neural networks. *Construction and Building Materials*, *159*, 635-641. https://doi.org/10.1016/j.conbuildmat.2017.10.132

Gong, H., Sun, Y., Mei, Z., & Huang, B. (2018). Improving accuracy of rutting prediction for mechanistic-empirical pavement design guide with deep neural networks. *Construction and Building Materials*, *190*, 710-718.

https://doi.org/10.1016/j.conbuildmat.2018.09.087

Haroon, D., & Clustering, I. (2017). *Python Machine Learning Case Studies*. Apress Berkeley, CA. https://doi.org/10.1007/978-1-4842-2823-4

Hinton, G., Srivastava, N., & Swersky, K. (2012). Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. *Cited on*, *14*(8), 2.

Hopfield, J. J. (1988). Artificial neural networks. *IEEE Circuits and Devices Magazine*, *4*(5), 3-10. https://doi.org/10.1109/101.8118

Huang, G. B., Zhu, Q. Y., & Siew, C. K. (2004, July). Extreme learning machine: a new learning scheme of feed-forward neural networks. In *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No. 04CH37541)* (Vol. 2, pp. 985-990). Ieee. https://doi.org/10.1109/IJCNN.2004.1380068

Huang, G.-B., Zhu, Q.-Y., & Siew, C.-K. (2006). Extreme learning machine: Theory and applications. *Neurocomputing*, *70*(1), 489-501. https://doi.org/10.1016/j.neucom.2005.12.126

Jadoun, F. M., & Kim, Y. R. (2012). Calibrating Mechanistic–Empirical Pavement Design Guide for North Carolina: Genetic Algorithm and Generalized Reduced Gradient Optimization Methods. *Transportation Research Record*, *2305*(1), 131-140. https://doi.org/10.3141/2305-14

Jeong, J.-H., Jo, H., & Ditzler, G. (2020). Convolutional neural networks for pavement roughness assessment using calibration-free vehicle dynamics. *Computer-Aided Civil and Infrastructure Engineering*, *35*(11), 1209-1229. https://doi.org/10.1111/mice.12546

Kaya, O. (2015). *Investigation Of Aashtoware Pavement ME Design/Darwin-ME TM Performance Prediction Models for Iowa Pavement Analysis and Design* (Doctoral dissertation, Iowa State University).

Kingma, D. P., & Ba, J. (2014). Adam: a method for stochastic optimization. *arXiv preprint arXiv:1412.6980*. https://doi.org/10.48550/arXiv.1412.6980

Li, B., Wang, K. C. P., Zhang, A., Yang, E., & Wang, G. (2020). Automatic classification of pavement crack using deep convolutional neural network. *International Journal of Pavement Engineering*, *21*(4), 457-463. https://doi.org/10.1080/10298436.2018.1485917

Mallela, J., Glover, L. T., Darter, M. I., Von Quintus, H., Gotlif, A., Stanley, M., & Sadasivam, S. (2009a). *Guidelines for Implementing NCHRP 1-37A ME Design Procedures in Ohio: Volume 1 - Summary of Findings, Implementation Plan, and Next Steps* (No. FHWA/OH-2009/9A). Ohio. Dept. of Transportation.

Mallela, J., Titus-Glover, L., Von Quintus, H., Darter, M. I., Stanley, M., & Rao, C. (2009b). *Implementing the AASHTO Mechanistic-Empirical Pavement Design Guide for Missouri. Vol. II, MEDPG Model Validation and Calibration* (No. CM08.01).

Muthadi, N. R., & Kim, Y. R. (2008). Local Calibration of Mechanistic-Empirical Pavement Design Guide for Flexible Pavement Design. *Transportation Research Record*, *2087*(1), 131-141. https://doi.org/10.3141/2087-14

Najafi, S., Flintsch, G. W., & Khaleghian, S. (2019). Pavement friction management – artificial neural network approach. *International Journal of Pavement Engineering*, *20*(2), 125-135. https://doi.org/10.1080/10298436.2016.1264221

Pao, Y.-H., Park, G.-H., & Sobajic, D. J. (1994). Learning and generalization characteristics of the random vector functional-link net. *Neurocomputing*, *6*(2), 163-180. https://doi.org/10.1016/0925-2312(94)90053-1

Pierce, L. M., & McGovern, G. (2014). *Implementation of the AASHTO Mechanistic-Empirical Pavement Design Guide and Software* (No. Project 20-05, Topic 44-06).

Ramachandran, P., Zoph, B., & Le, Q. V. (2017). Searching for activation functions. *arXiv preprint arXiv:1710.05941*. https://doi.org/10.48550/arXiv.1710.05941

Schram, S., & Abdelrahman, M. (2006). Improving prediction accuracy in mechanistic–empirical pavement design guide. *Transportation Research Record*, *1947*(1), 59-68. https://doi.org/10.1177/0361198106194700106

Schram, S. A., & Abdelrahman, M. (2010). Integration of Mechanistic–Empirical Pavement Design Guide distresses with local performance indices. *Transportation Research Record*, *2153*(1), 13-23. https://doi.org/10.3141/2153-02

Shafabakhsh, G. H., Ani, O. J., & Talebsafa, M. (2015). Artificial neural network modeling (ANN) for predicting rutting performance of nano-modified hot-mix asphalt mixtures containing steel slag aggregates. *Construction and Building Materials*, *85*, 136-143. https://doi.org/10.1016/j.conbuildmat.2015.03.060

Simpson, A. L., Daleiden, J. F., & Hadley, W. O. (1995). Rutting analysis from a different perspective. *Transportation Research Record*, *1473*, 9-17.

Smith, B., & Nair, H. (2015). *Development of Local Calibration Factors and Design Criteria Values for Mechanistic-Empirical Pavement Design* (No. FHWA/VCTIR 16-R1). Virginia Center for Transportation Innovation and Research.

Sollazzo, G., Fwa, T. F., & Bosurgi, G. (2017). An ANN model to correlate roughness and structural performance in asphalt pavements. *Construction and Building Materials*, *134*, 684-693. https://doi.org/10.1016/j.conbuildmat.2016.12.186

Souliman, M. I., Mamlouk, M. S., El-Basyouny, M. M., & Zapata, C. E. (2010). Calibration of the AASHTO MEPDG for flexible pavement for arizona conditions. In *Proceedings of the Transportation Research Board 89th Annual Meeting* (Vol. 22, pp. 243-286). Washington, DC, USA: Transportation Research Board.

Sun, X., Han, J., Parsons, R. L., Misra, A., & Thakur, J. K. (2015). *Calibrating the Mechanistic-Empirical Pavement Design Guide for Kansas* (No. KS-14-17). Kansas. Dept. of Transportation. Bureau of Materials & Research.

Wang, C., Xu, S., Liu, J., Yang, J., & Liu, C. (2022). Building an improved artificial neural network model based on deeply optimizing the input variables to enhance rutting prediction. *Construction and Building Materials, 348*, 128658. https://doi.org/10.1016/j.conbuildmat.2022.128658

Wang, C., Xu, S., & Yang, J. (2021). Adaboost Algorithm in Artificial Intelligence for Optimizing the IRI Prediction Accuracy of Asphalt Concrete Pavement. *Sensors, 21*(17), 5682. https://doi.org/10.3390/s21175682