MethodsX 13 (2024) 102791

Contents lists available at ScienceDirect

MethodsX

journal homepage: www.elsevier.com/locate/methodsx

Deep neural network-based prediction of tsunami wave attenuation by mangrove forests

Didit Adytia[a,*], Dede Tarwidi[a,b], Deni Saepudin[a], Semeidi Husrin[c], Abdul Rahman Mohd Kasim[d], Mohd Fakhizan Romlie[e], Dafrizal Samsudin[f]

[a] School of Computing, Telkom University, Jalan Telekomunikasi No. 1 Terusan Buah Batu, Bandung 40257, Indonesia
[b] Industrial and Financial Mathematics Research Group, Faculty of Mathematics and Natural Sciences, Institut Teknologi Bandung, Jalan Ganesha No. 10, Bandung 40132, Indonesia
[c] Research Center for Geological Disasters, BRIN, Indonesia
[d] Centre for Mathematical Sciences, Universiti Malaysia Pahang, Lebuhraya Tun Razak, Gambang 26300, Pahang, Malaysia
[e] Department of Electrical and Electronic Engineering, Universiti Teknologi PETRONAS, Seri Iskandar 32610, Perak, Malaysia
[f] Program Studi Ilmu Komunikasi, Fakultas Ilmu Komunikasi, Universitas Islam Riau, Jalan Kaharuddin Nst No.113, Simpang Tiga, Kec. Bukit Raya, Kota Pekanbaru, Riau, Indonesia

ARTICLE INFO

Method name:
DNN for Wave Attenuation Prediction

Keywords:
Tsunami
Deep neural network
Wave attenuation
Hyperparameter optimization
Adam optimizer

ABSTRACT

The goal of this research is to develop a model employing deep neural networks (DNNs) to predict the effectiveness of mangrove forests in attenuating the impact of tsunami waves. The dataset for the DNN model is obtained by simulating tsunami wave attenuation using the Boussinesq model with a staggered grid approximation. The Boussinesq model for wave attenuation is validated using laboratory experiments exhibiting a mean absolute error (MAE) ranging from 0.003 to 0.01. We employ over 40,000 data points generated from the Boussinesq numerical simulations to train the DNN. Efforts are made to optimize hyperparameters and determine the neural network architecture to attain optimal performance during the training process. The prediction results of the DNN model exhibit a coefficient of determination (R^2) of 0.99560, an MAE of 0.00118, a root mean squared error (RMSE) of 0.00151, and a mean absolute percentage error (MAPE) of 3 %. When comparing the DNN model with three alternative machine learning models— support vector regression (SVR), multiple linear regression (MLR), and extreme gradient boosting (XGBoost)— the performance of DNN is superior to that of SVR and MLR, but it is similar to XGBoost.

- High-accuracy DNN models require hyperparameter optimization and neural network architecture selection.
- The error of DNN models in predicting the attenuation of tsunami waves by mangrove forests is less than 3 %.
- DNN can serve as an alternate predictive model to empirical formulas or classical numerical models.

* Corresponding author.
E-mail address: adytia@telkomuniversity.ac.id (D. Adytia).

https://doi.org/10.1016/j.mex.2024.102791
Received 27 March 2024; Accepted 1 June 2024
Available online 11 June 2024

Specifications table

Subject area:	Engineering
More specific subject area:	Coastal Engineering
Name of your method:	DNN for Wave Attenuation Prediction
Name and reference of original method:	Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." *Advances in neural information processing systems* 25 (2012).
Resource availability:	The script in Python is available on GitHub: https://github.com/diditadytia/DNN-based-prediction-of-tsunami-wave-attenuation-by-mangrove-forests.git

Background

Tsunamis, representing catastrophic marine events with the potential for profound damage to regions in their trajectory, arise from diverse triggers such as earthquakes, submarine volcanoes, submarine landslides, and meteorite impacts [1]. In addition to causing significant damage to the areas they traverse, tsunamis pose a formidable threat to human lives, particularly for coastal residents. Addressing these challenges requires approaches to mitigate the impact of tsunamis. Local authorities can implement mitigative measures such as constructing breakwaters and sea walls. However, these strategies entail significant costs and carry the risk of adverse impacts on the natural environment [2,3]. As part of coastal ecosystems, Mangrove forests present viable alternatives for wave attenuation, given their natural capacity to act as barriers that reduce the energy of tsunami-induced waves [2,4]. Additionally, mangrove forests contribute to preserving the natural habitat in coastal areas [3]. Nevertheless, additional research is required to thoroughly examine the efficacy of mangrove forests in wave attenuation, particularly in the context of reducing tsunami-induced wave energy.

Evaluating the efficacy of mangrove forests in attenuating tsunami waves can be approached through empirical and numerical models. Wu and Cox [5] performed a physical model experiment to explore the impact of wave steepness and relative water depth on the attenuation of waves by emergent vegetation. Mendez and Losada [6] utilized an empirical model to assess the propagation of both breaking and nonbreaking waves over fields of vegetation. Huang et al. [7] conducted laboratory experiments and used the Boussinesq numerical model to examine how solitary waves interact with emergent, rigid vegetation. Maza et al. [8] researched the interactions between tsunami waves and mangrove forests, employing a three-dimensional numerical methodology. Adytia et al. [2] utilized a non-dispersive wave model, a numerical approach, to investigate the dissipation of solitary waves attributed to mangrove forests. Other studies on wave attenuation by vegetation using numerical models were also conducted by Abdolali et al. [9], Lee et al. [10] and Limura and Tanaka [11].

However, both empirical and numerical models have limitations that restrict their efficacy when employed. Empirical models require high costs, while numerical models require high computation time. Recent advancements in computer science and AI technology provide a groundbreaking avenue for developing models to address coastal challenges through machine learning. Implementing this approach not only reduces computational expenses but also accelerates forecasting, leading to a methodology with improved efficiency and accuracy. Using machine learning combined with hydraulic expressions, Kim et al. [12] studied wave attenuation in coral reefs. The prediction of the attenuation of solitary waves by emerging vegetation was investigated by Gong et al. [4] using genetic algorithm and artificial neural networks. Other studies that used machine learning to solve marine problems were conducted by Yao et al. [1] and Dharmawan et al. [13].

In this paper, a deep learning model, namely a Deep Neural Network, is developed to predict tsunami wave attenuation by mangrove forests. To construct the Deep Neural Network, we generate simulation data from a wave model, Staggered grid Variational Boussinesq (SVB), validated with the experimental data. This paper marks progress beyond the study of Malvin et al. [3], which focused on employing neural network models to analyze the dissipation of waves caused by mangrove forests.

Method details

Wave model

We use numerical simulations employing the two-profile Variational Boussinesq Model (VBM) wave model proposed in [14] to generate training datasets for deep learning models intended to predict tsunami wave attenuation. The wave model is given by

$$\partial_t \eta = -\partial_x(hu) - \partial_x\left(\beta^{(1)}\partial_x\psi^{(1)}\right) - \partial_x\left(\beta^{(2)}\partial_x\psi^{(2)}\right) \tag{1}$$

$$\partial_t u = -g\partial_x\eta - u\partial_x u - R_B, \tag{2}$$

where $\eta(x,t)$ is free surface elevation, $u(x,t)$ is horizontal velocity, g is gravity acceleration, $h = \eta + d$ is total water depth, $d(x,t)$ is bathymetry, $R_B = C_f\, u|u|/h$ is the bottom dissipation term, $C_f = n^2 g/h^{1/3}$ is the coefficient of bottom dissipation, and n is Manning's coefficient. The horizontal dependent function $\psi^{(m)}$ is obtained by solving the following elliptic system:

$$-\partial_x\left(\begin{bmatrix}\alpha^{(11)} & \alpha^{(12)}\\ \alpha^{(21)} & \alpha^{(22)}\end{bmatrix}\begin{bmatrix}\partial_x\psi^{(1)}\\ \partial_x\psi^{(2)}\end{bmatrix}\right) + \partial_x\left(\begin{bmatrix}\gamma^{(11)} & \gamma^{(12)}\\ \gamma^{(21)} & \gamma^{(22)}\end{bmatrix}\begin{bmatrix}\psi^{(1)}\\ \psi^{(2)}\end{bmatrix}\right) = \begin{bmatrix}\partial_x\left(\beta^{(1)}u\right)\\ \partial_x\left(\beta^{(2)}u\right)\end{bmatrix}, \tag{3}$$

where

$$\alpha^{(ij)} = \int_{-d}^{\eta} F^{(i)}F_j\,dz,\ \beta^{(j)} = \int_{-d}^{\eta} F^{(j)}dz,\ \gamma^{(ij)} = \int_{-d}^{\eta}\left(\partial_z F^{(i)}\right)\left(\partial_z F^{(j)}\right)dz. \tag{4}$$

Here, $F^{(m)}$ denotes vertical profile function and is given by

$$\boldsymbol{F}^{(m)}(z;\eta,d) = \frac{\cosh\left(\kappa_m(z+d)\right)}{\cosh\left(\kappa_m h\right)} - 1 \tag{5}$$

The wave number κ_m is determined a priori by the process of kinetic energy minimization.

The wave model in Eqs. (1)-(4) are solved numerically using staggered grid discretization. In the staggered grid implementation, we use two types of grid, i.e., the full-grid, in which parameters such as η, h, and ψ are discretized in x_i, with $x \in [0,\ L]$, whereas the other grid called the half-grid, in which parameter such as u is discretized in $x_{i+1/2}$, with $i = 1,\ 2,\ \ldots, N$. By substituting these discretizations, Eqs. (1)-(4) are transformed into system of matrix. Details of numerical implementation procedures can be found in [14] and [15]. The numerical scheme is called SVB (Staggered grid Variational Boussinessq). The SVB numerical scheme has been extensively employed in various wave modeling applications, as described in [15–17]. By using the SVB numerical scheme, we conduct a numerical simulation of tsunami wave attenuation.

Validation of the wave model with laboratory experiments

Several standard tests were used to prove the SVB wave model [14]. These included wave run-up, shock-induced dam break, propagation of dispersive and nonlinear broadband waves on a flat seabed, propagation of regular waves breaking over a submerged trapezoidal bar, propagation of cnoidal waves on a flat beach, and propagation of irregular or random waves on a barred beach. The results of the SVB model showed high agreement with the analytical solutions and measurement data. In this work, the capability of the wave model to simulate the attenuation of tsunami waves by mangrove forests is validated. The validation is conducted using experimental data obtained from [18]. In [18], the experiments were conducted in a twin flume wave tank; which consist of a 1 m wide flume for non-mangrove experiments, and the other 2 m wide flume for mangrove experiments. The scale of the experiment is 1:25 with bathymetry consisting of deep, shallow, and sloping areas. They conducted experiments with many variants of wave influxes such as solitary, regular and irregular waves. To describe real world situations, here the mangroves are parameterized using vertical bars. Details of the physical experiment are desribed in [18]. Fig. 1 illustrates the experimental design for tsunami wave attenuation by mangrove forests.

The physical parameters utilized for the tsunami wave attenuation simulation were as follows: water depth $h_0 = 0.615$ m, beach slope 1/20, forest length $B = 3$ m, and incident wave amplitude 0.2 m. The mangrove forest has a Manning roughness coefficient of 0.13. The computational domain of this simulation is [0, 80 m] with spatial discretization $\Delta x = 0.1$ m and temporal step $\Delta t = 0.02$. The simulation is conducted for a maximum time of 90 s. Fig. 2 displays the simulation results for tsunami wave attenuation, including the maximum temporary amplitude up to $t = 75$ s. The figure shows a decrease in wave amplitude from 0.2 m to 0.13 m, indicating a wave attenuation of 35 %.

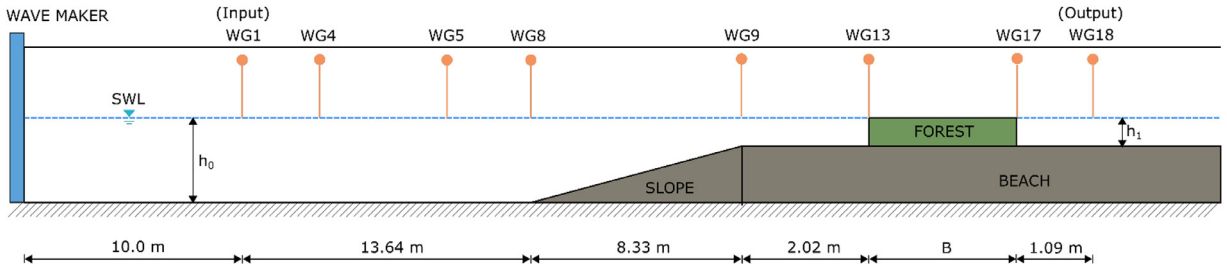

Fig. 1. Experimental set-up of tsunami wave attenuation by mangrove forests.

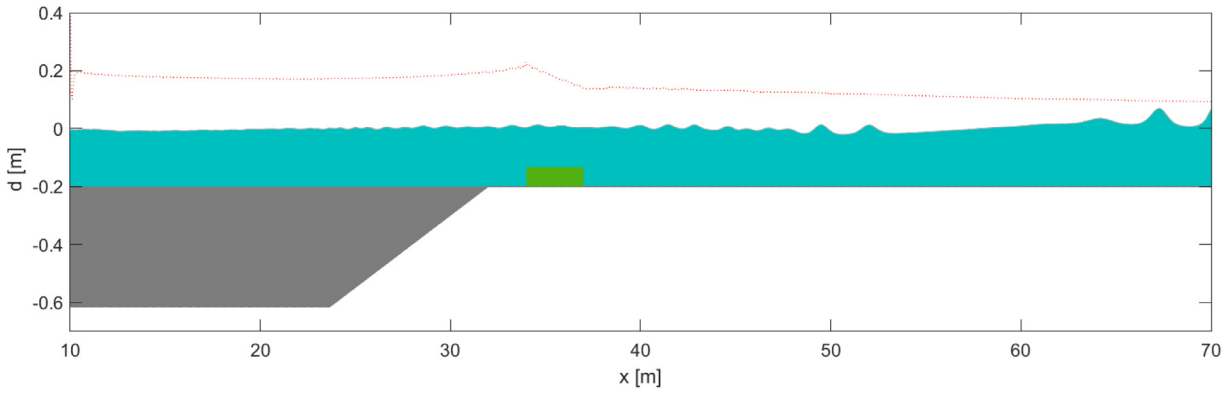

Fig. 2. Simulation results of tsunami wave attenuation with maximum temporary amplitude until $t = 75$ s.

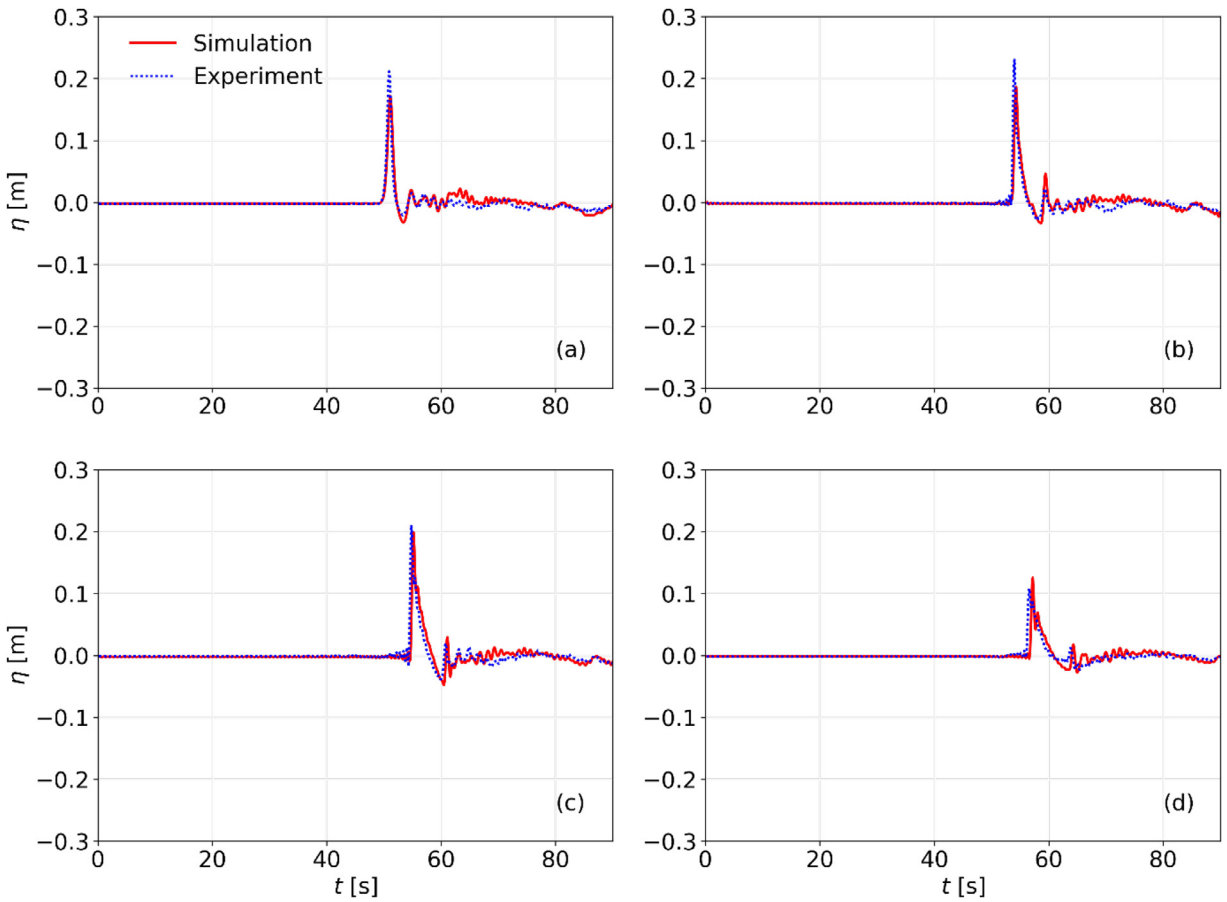

Fig. 3. Comparisons of time series of the free-surface elevation between the SVB [14] and experimental data [18] at several wave gauge locations: (a) WG8, (b) WG9, (c) WG13, (d) WG18.

Table 1
SVB model accuracy at several wave gauge locations.

Wave gauge location	Statistical Measure	
	MAE	RMSE
WG1	0.01110	0.03330
WG8	0.00346	0.00742
WG9	0.00369	0.01033
WG13	0.00491	0.01270
WG18	0.00343	0.00922

Fig. 3 displays a comparison of the time series of the free surface elevation between the numeric and the experimental data at various wave gauge locations. The figure clearly demonstrates a strong agreement between the numerical results and the experimental data. To quantify this comparison, the MAE and RMSE were computed for the results obtained at WG1, WG8, WG9, WG13, and WG18. The error values are provided in Table 1. The MAE and RMSE for SVB range from 0.003 to 0.01 and 0.007 to 0.03, respectively. These findings suggest that the SVB model accurately simulates tsunami wave attenuation by mangrove forests.

Deep neural networks

Deep neural networks (DNNs) are a subset of artificial neural networks (ANNs). The key distinction between Deep Neural Networks (DNNs) and Artificial Neural Networks (ANNs) is the presence of multiple hidden layers between the input and output layers. These hidden layers allow the neural network to apprehend the complex and nonlinear patterns inherent in the dataset. In a DNN, neurons within a layer are interconnected with neurons in neighboring layers. Components of the deep neural network used in this study are shown in Fig. 4. The explanation of each component is as follows.

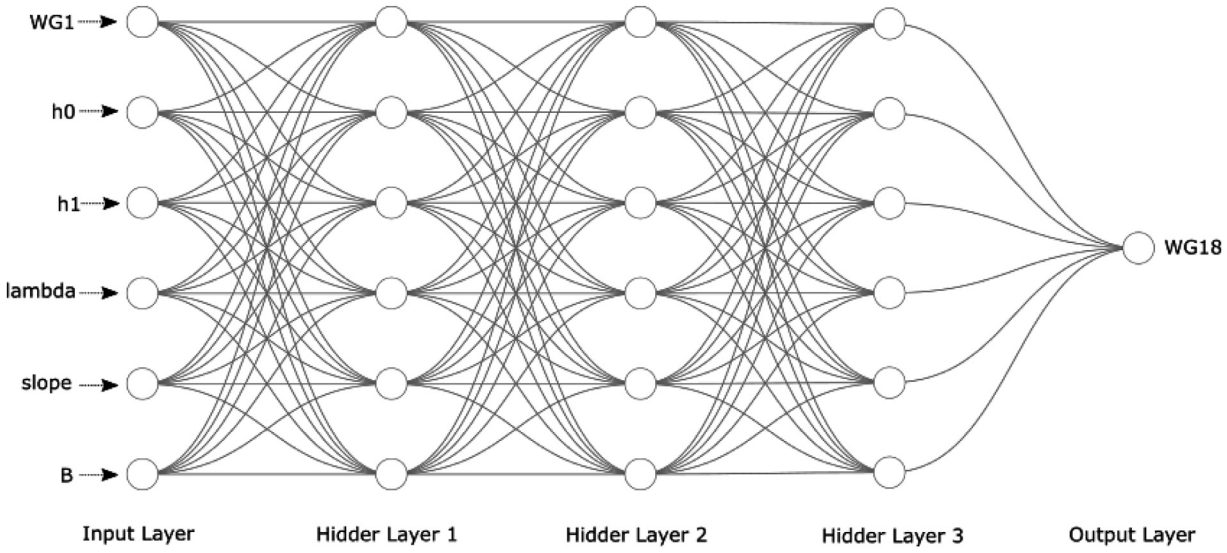

Fig. 4. Deep neural network architecture.

1. **Input Layer:** The input layer comprises neurons that capture the features of the raw input data.
2. **Hidden Layers:** Hidden layers are intermediary layers positioned between the input layer and the output layer. Deep learning involves the use of multiple hidden layers, each of which contains a varying number of neurons.
3. **Weights and Biases:** The connection between neurons in adjacent layers is established through a weight, which represents the strength of the connection. Every neuron possesses a corresponding bias that enables it to detect patterns that may be unrelated to the input data.
4. **Activation Functions:** Inputs received by the neurons in each layer undergo multiplication with their corresponding weights and subsequent summation. The sum is subsequently fed into an activation function to generate the modified output. Widely used activation functions include sigmoid, tanh, and ReLU, which stand for rectified linear units.
5. **Output Layer:** The output layer represents the ultimate result produced by the neural network. In regression models, the output layer typically comprises a single neuron.

The deep neural network consists of several processes, which are outlined below.

1. **Feedforward Propagation:** Feedforward propagation is the process of feeding input data into a neural network so that it produces a prediction or output. The mathematical formulation of feedforward propagation for an individual neuron inside a layer can be represented as follows:

$$z = \sum_{i=1}^{n} w_i \cdot x_i + b, \tag{6}$$

$$a = \sigma(z), \tag{7}$$

where w_i is weights, x_i is input variables, b is the bias term, σ is the activation function. Here, a is activation, which will be the input for the next layer.

2. **Backpropagation:** Backpropagation is a learning process used to train the DNN model. By means of this process, the model can adjust its weights and biases to minimize the discrepancy between the predicted output (the result of the forward propagation) and the actual value.
3. **Loss function:** A loss function is used to determine the magnitude of the predicted output and the actual value.
4. **Training:** The neural network is trained through an iterative process using optimization algorithms, such as stochastic gradient descent (SGD) [19]. One variant of the SGD algorithm is Adaptive Moment Estimation (Adam), while the other is Root Mean Square Propagation (RMSprop). One of the hyperparameters that needs to be optimized related to the SGD algorithm is the learning rate.
5. **Evaluation:** Once the network has completed its training process, its generalization capability is assessed by evaluating its performance using a separate test dataset. Further elaboration on DNN is available in reference [20].

DNN method for predicting tsunami wave attenuation

The DNN method and its application in predicting the attenuation of tsunami waves will be elaborated upon in this section. With the exception of a few phases that require particular attention, the methodology employed for predicting other challenges

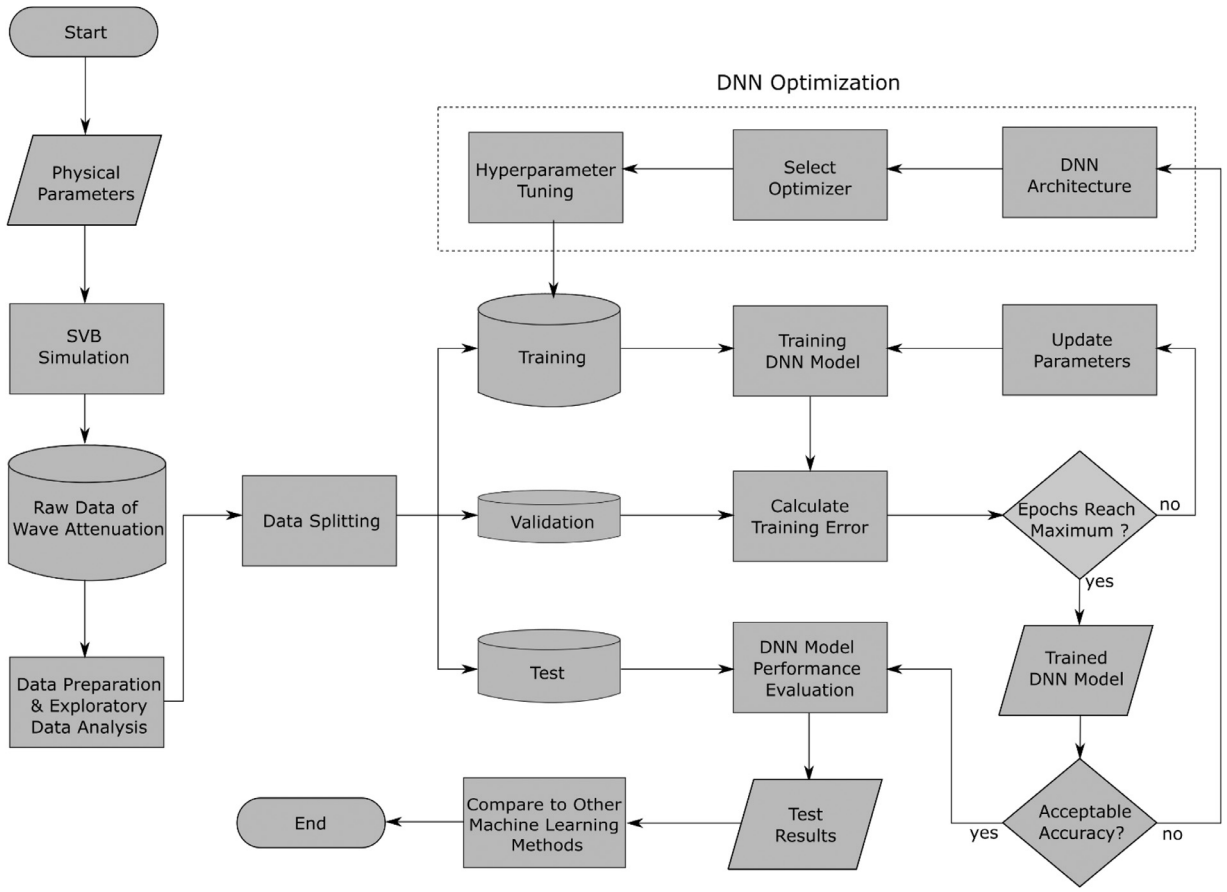

Fig. 5. Deep neural network modeling process for predicting tsunami wave attenuation.

is substantially identical for the most part. The complete methodology of the deep neural network for predicting tsunami wave attenuation is described in Fig. 5.

The step-by-step process can be explained as follows:

1. Using the physical parameters from laboratory experiments, we perform numerical simulations using the staggered variational Boussinesq (SVB) wave model.
2. The simulation is conducted by varying the physical parameters, including the amplitude (WG1) and length (lambda) of the incident wave, water depth (h_0), water depth within the forests (h_1), beach slope, and forest length (B). The output of this simulation is the wave amplitude after passing through the forests (WG18).
3. The result of this simulation is raw data that will be utilized as the training data for the deep neural network. The raw data of this study consists of 43,032 entries with 7 features.
4. Exploratory data analysis involves utilizing data visualization techniques to observe and understand the patterns and characteristics of the data. Data preparation is conducted to eliminate outliers and incomplete data.
5. The dataset is subsequently split into three distinct groups: training data, validation data, and test data. The ratio of training data to test data is 70 % to 30 %. Furthermore, 10 % of the training data is specifically allocated for the validation test.
6. Before starting the DNN model training, we must first determine the DNN architecture, select the optimizer, and perform hyperparameter tuning.
7. By employing the training data, the deep neural network (DNN) model is trained iteratively until the maximum epoch is reached.
8. By utilizing validation data, we compute the mean absolute error of the DNN model. The computed error must be consistent with the error from the training data, or else overfitting may occur.
9. Upon reaching the maximum epoch, the trained DNN model is obtained.
10. We reassess the training outcomes to determine if they meet our criteria. If the level of accuracy remains insufficient, we will proceed with another round of DNN optimization.
11. Once the trained DNN model attains a satisfactory level of accuracy, we proceed to evaluate its performance using test data. At this point, we employ statistical metrics to quantify the performance of the model.
12. The predicted results of the DNN model are compared to those of alternative machine learning models.

Two main simulation tools are used in this research, i.e., the simulation tool for the numerical wave simulation and for performing machine learning modelling. For the numerical wave simulation, we implemented the SVB model using a staggered grid scheme in Python language, with details of the SVB's numerical scheme can be found in [14]. For the machine learning modelling by using DNN, we use well-established KERAS package, which is part of Tensorflow. Both simulation tools are run on a PC with an i9 intel processor 11th generation, with 16 GB RAM and Nvidia GPU GeForce RTX3060. For validation of the numerical wave simulation using the SVB model, we use the physical experimental wave data that are taken from [18].

Method validation

In this section, the deep neural network model to predict tsunami wave attenuation by mangrove forests is validated and discussed. However, we conducted exploratory data analysis and parameter optimization beforehand. To quantify the performance of the DNN model in predicting tsunami wave attenuation, we employ the following metrics: coefficient of determination (R^2), absolute error (MAE), mean absolute percentage error (MAPE), and root mean squared error (RMSE).

Exploratory data analysis

We will conduct exploratory data analysis as a preliminary step in constructing a deep learning model. Exploratory Data Analysis (EDA) is undertaken to extract descriptive information and identify patterns from a dataset using visual representation. The SVB wave model generated 43,032 data used for the deep learning process in this work. The dataset has the following features: WG1 represents the incident wave amplitude measured at wave gauge location 1. h_0 refers to the initial water depth. In contrast, the water depth in the mangrove forest is represented by h_1. Lambda denotes the wavelength of the incident wave, and slope indicates the beach slope. B represents the length of the forests. Lastly, WG18 represents the amplitude of the wave after it passes through the mangrove forest, as measured at wave gauge location 18. The variable to be predicted is WG18. Fig. 6 displays the statistical distribution of these variables.

The statistical description of the tsunami wave attenuation dataset is presented in Table 2. The table indicates that the incident wave has an amplitude ranging from 0.04 to 0.24 m and a wavelength between 2.1 and 3 m. The water depth h_0 is visible between the values of 0.4 and 0.7 m. The beach slope within the dataset varies from 1/60 to 1/14. The forest length achieved effectively ranges from 1.2 to 14.3 m, resulting in an average wave height of 0.04 m at WG18.

Fig. 7 shows the effect of input variables such as the amplitude of the incident wave, water depth, slope, and length of forest on the wave amplitude at WG18, which is the wave after passing through the mangrove forest. It can be seen that when the incident wave increases by 0.05, the wave height at WG18 increases by about 0.02. In addition, the water depth h_0 positively affects the wave height after passing through the forest. The deeper the water, the higher the wave height. For a beach slope smaller than 1/20, the amplitude increases when the beach is steeper, while when the slope is greater than 1/20, the amplitude decreases when the beach is steeper. The length of the forest significantly affects the height of the wave after passing through the forest. The longer the forest, the shorter the amplitude of the tsunami wave.

Fig. 8 describes the correlation heat map among variables in the dataset. In the figure, darker colors indicate a stronger correlation, while lighter colors indicate a weaker correlation. A negative correlation coefficient means an opposite relationship between two variables. With a correlation coefficient of 0.65, the input variable forest length strongly influences the wave height after passing through the forests. Furthermore, the water depth h_1 is the second strongest after the forest's length to influence the wave attenuation. Amplitude and incident wavelength, with correlation coefficients of 0.44 and 0.45, respectively, moderately influence the wave amplitude at WG18. The beach slope correlates most weakly with wave attenuation due to its negative influence on small slopes and positive influence on bigger slopes.

Deep neural network optimization and implementation

In this study, we implement a deep neural network algorithm by addressing the optimization method to obtain accurate prediction results. The optimization of deep neural networks involves four main sections: data preprocessing, architectural design, optimizer selection, and hyperparameter tuning [21]. We select two specific areas for optimization: architecture and hyperparameters. The parameters to be optimized are listed in Table 3. The optimization involves selecting the number of layers in a neural network design,

Table 2
Statistical description of tsunami wave attenuation dataset.

parameters	mean	std	min	25 %	50 %	75 %	max
WG1	0.149171	0.06732	0.040927	0.066395	0.164919	0.209439	0.239227
h_0	0.551829	0.086985	0.418571	0.464695	0.555817	0.634688	0.698436
h_1	0.201939	0.05959	0.116764	0.148338	0.190108	0.256050	0.294119
Lambda	2.635640	0.226209	2.125437	2.457077	2.666356	2.825603	3.050858
Slope	0.042004	0.011392	0.016315	0.034585	0.041096	0.049222	0.069525
B	7.572313	3.790363	1.203618	4.453153	7.175663	10.90263	14.29066
WG18	0.045310	0.022906	0.008191	0.027411	0.040469	0.059382	0.107361

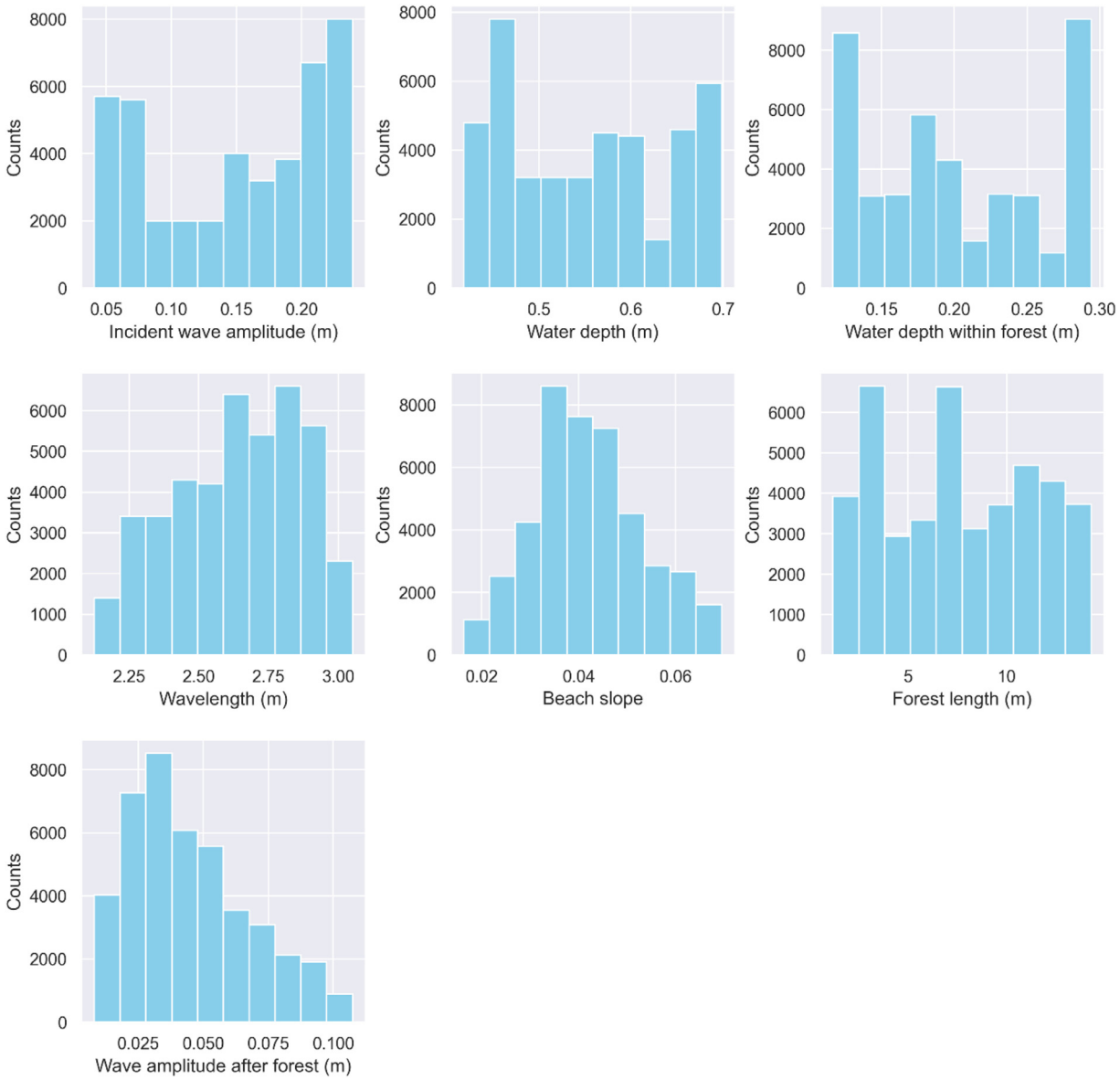

Fig. 6. Data distributions of the input and output variables.

Table 3
Parameters for DNN optimization.

Parameters	Section	Search range	Data type	Optimal value
Number of hidden layers	Architecture	[2,3]	Integer	3
Number of neurons	Architecture	[6,16]	Integer	10
Learning rates	Hyperparameter	$[10^{-9},10^{-1}]$	Real	10^{-3}
Number of epochs	Hyperparameter	[50,300]	Integer	80

which can be either two or three, and determining the number of neurons in each hidden layer, which is within the range of 6 to 16. The DNN architecture, depicted in Fig. 4, consists of three hidden layers, each encompassing six neurons. We will examine the two main hyperparameters in DNN optimization: learning rates and number of epochs. The search intervals for learning rates and the number of epochs are $[10^{-9},10^{-1}]$ and [50,300] respectively. DNN optimization involves selecting from several kinds of optimizer alternatives. Commonly employed optimizers in deep learning encompass SGD, Adagrad, RMSProp, and Adam. The SGD optimizer is straightforward to implement, but it lacks the ability to dynamically adjust the learning rate during the training process.

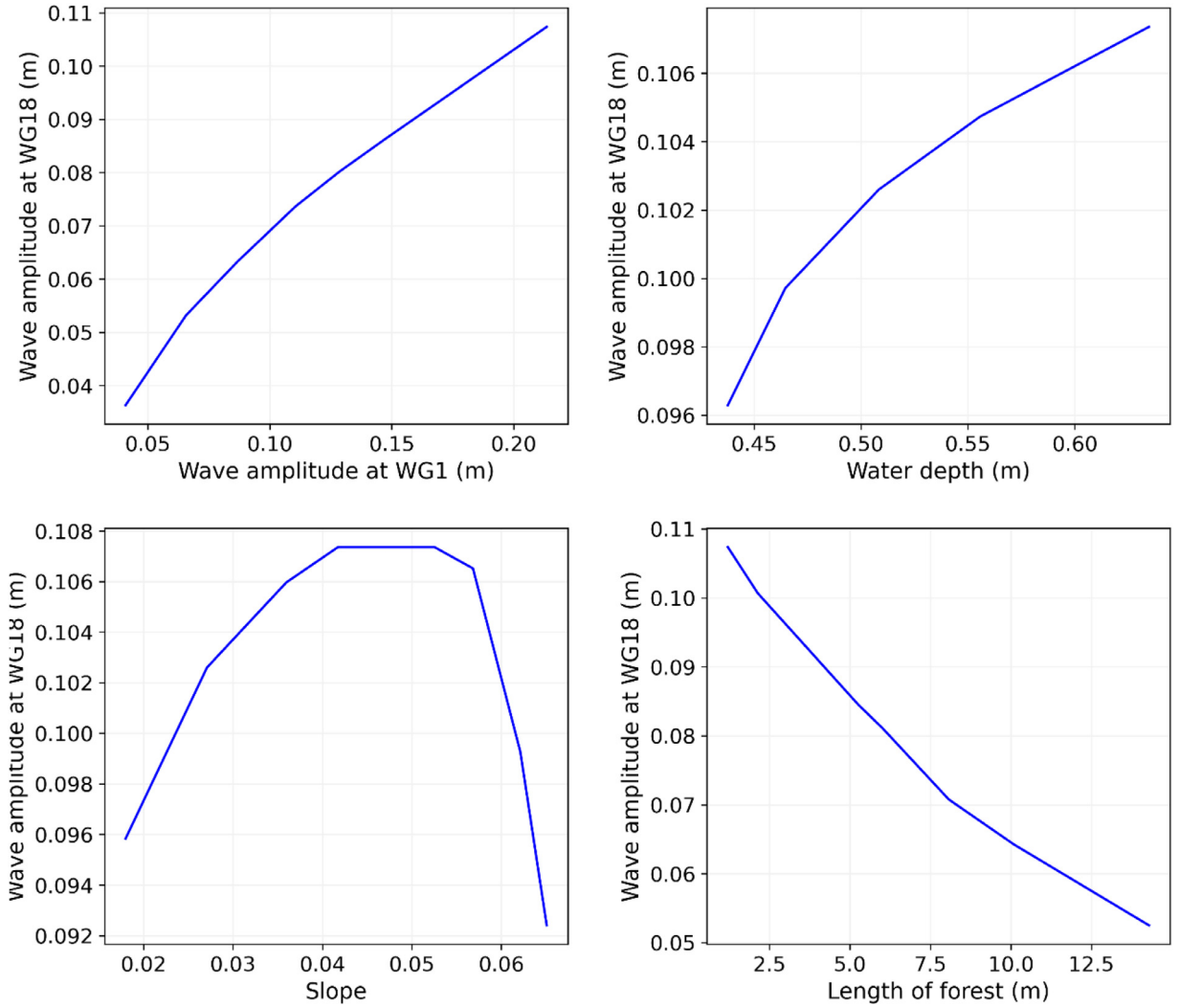

Fig. 7. Effect of several input variables on the wave amplitude after passing the forest.

The adaptability of the learning rate in Adagrad can lead to premature convergence. RMSProp enhances the Adagrad algorithm by incorporating a moving average of the squared gradient. RMSProp is more resilient for all categories of neural networks. In addition, the Adam optimizer integrates the advantages of both RMSProp and momentum optimization. In our case, we utilize the Adaptive Moment Estimation (Adam) optimizer.

The parameters listed in Table 3 were optimized using the grid search approach. Fig. 9 displays the mean absolute error (MAE) of the DNN model as it relates to the number of neurons for both two and three hidden layers. The model with three hidden layers generally exhibits a lower MAE compared to the model with two hidden layers. Further, the MAE for three hidden layers is achieved with ten neurons. The variation of MAE against learning rate is shown in Fig. 10. It can be seen that learning rates of 0.001 and 0.01 give the smallest MAE, but we chose a learning rate of 0.001 in DNN implementation. Furthermore, the variation of MAE against the number of epochs is shown in Fig. 11. The optimization findings indicate that the minimum mean absolute error (MAE) is achieved when the epoch value is set to 80. Any epoch value larger than 80 does not lead to a further reduction in the MAE. Table 3 also displays the optimal parameter values obtained from the grid search optimization.

The DNN method was executed using the optimal parameters specified in Table 3. The dataset, comprising 43,032 entries and 7 features, is divided into two distinct categories: 70 % of the data is designated for training purposes, while the remaining 30 % is labeled for testing the DNN model. Tables 4 and 5 display the first 10 data from the training and test data, respectively. The Rectified Linear Unit (ReLU) was employed as the activation function in the DNN algorithm. The remaining parameters use default values from the deep learning library.

Fig. 8. Correlation heat map among variables.

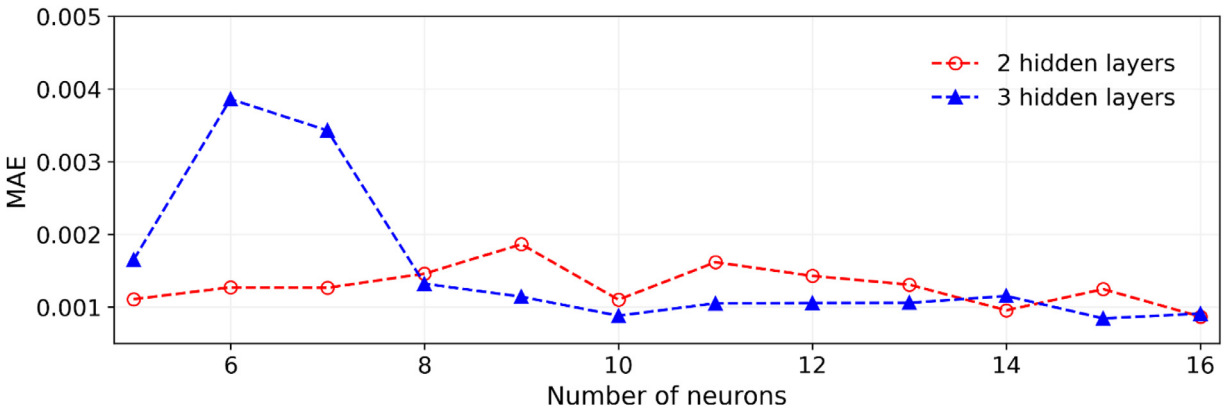

Fig. 9. Variation in MAE with the number of neurons for two and three hidden layers using test data.

Table 4
Training data for DNN.

WG1	h_0	h_1	Lambda	Slope	B	WG18
0.204239	0.685815	0.280011	2.980033	0.048716	8.544877	0.065217
0.209439	0.592699	0.291433	2.837336	0.036166	11.65861	0.055711
0.128536	0.463655	0.145795	2.428328	0.038158	2.815606	0.056856
0.066395	0.562259	0.148338	2.487232	0.04969	7.175663	0.026206
0.065714	0.671338	0.294119	2.691886	0.045284	7.729143	0.040270
0.128536	0.555817	0.230816	2.605072	0.039016	6.897647	0.052249
0.209439	0.671338	0.292978	2.966591	0.045421	2.134128	0.099076
0.155042	0.429896	0.167544	2.423917	0.031495	3.158108	0.062216
0.228410	0.429896	0.280011	2.599157	0.017993	11.86824	0.050122
0.186753	0.661472	0.291433	2.906958	0.044422	1.240857	0.095644

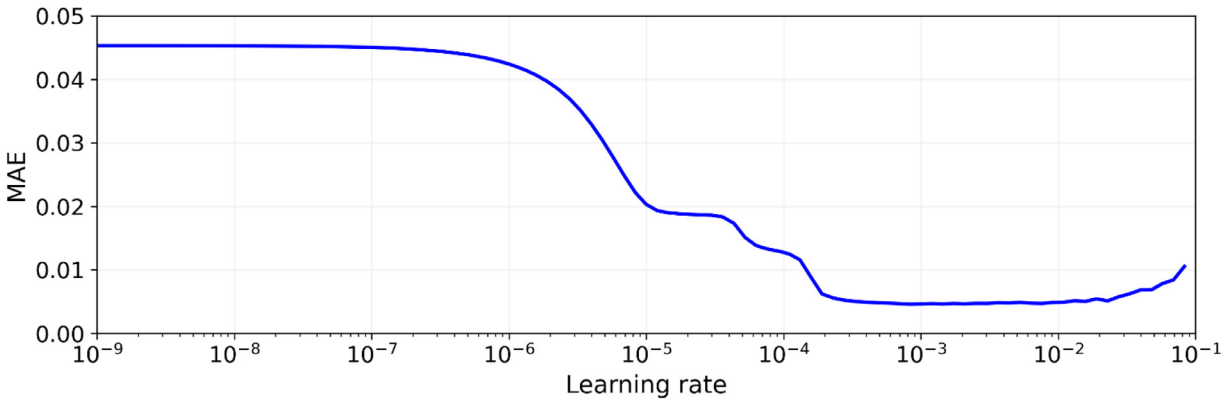

Fig. 10. Variation in MAE with learning rate.

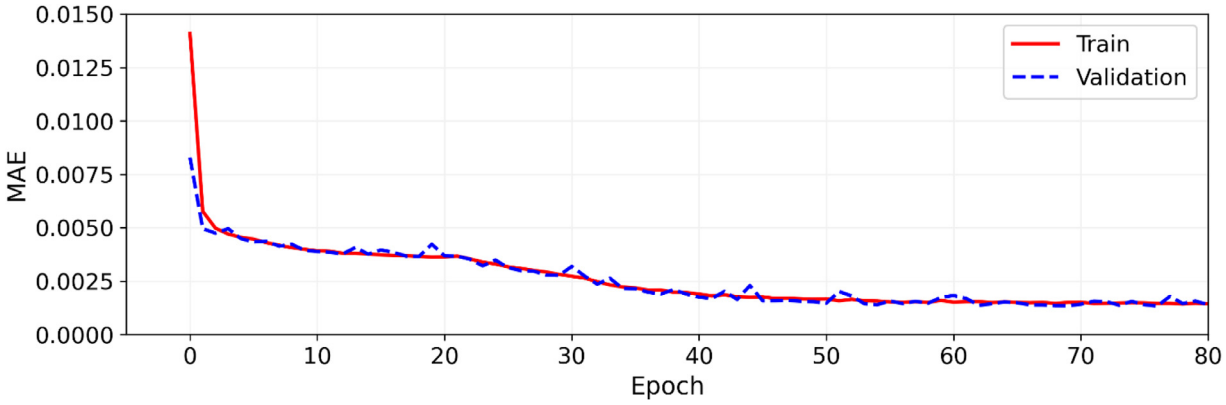

Fig. 11. Variation in the MAE with the number of epochs.

Table 5
Test data for DNN.

WG1	h_0	h_1	Lambda	Slope	B	WG18
0.065714	0.643236	0.184352	2.640315	0.055088	10.81116	0.025160
0.066395	0.470172	0.197851	2.299286	0.032692	3.442609	0.044431
0.155042	0.429896	0.180782	2.423917	0.029906	11.86824	0.026066
0.204239	0.540180	0.119291	2.737177	0.050527	7.686781	0.023294
0.239227	0.453972	0.24963	2.666356	0.024531	12.90569	0.041641
0.205163	0.573560	0.237843	2.796294	0.040302	8.063806	0.059273
0.128536	0.585647	0.152594	2.659947	0.051987	5.751071	0.039770
0.086956	0.562259	0.197851	2.530174	0.043747	9.999896	0.031145
0.086956	0.698436	0.180782	2.780512	0.062143	6.000906	0.040604
0.239227	0.508365	0.152594	2.758616	0.042710	2.815606	0.067828

Deep neural network validation

The evaluation of the deep neural network model in predicting wave attenuation is conducted by utilizing validation and test datasets. The comparison between the predicted and observed results is shown in Fig. 12. By employing the validation data, it was found that the DNN model exhibits an excellent degree of accuracy in its predicting capabilities (MAE = 0.00113, R^2 = 0.99595). The obtained results closely match the level of accuracy that the DNN model attained while using the training data. The efficacy of the DNN model was evaluated using a separate set of test data, which was not utilized during the model training process. As shown in Fig. 12(c), the DNN model has a coefficient of determination of 0.99560 and an error of 0.00118. When all the SVB-generated data is used, the accuracy of the model remains consistent with the evaluation results obtained from the validation and test data, as illustrated in Fig. 12(d). The findings of this research provide evidence supporting the enhanced predictive capabilities of the deep neural network model in estimating wave amplitude after passing through forests.

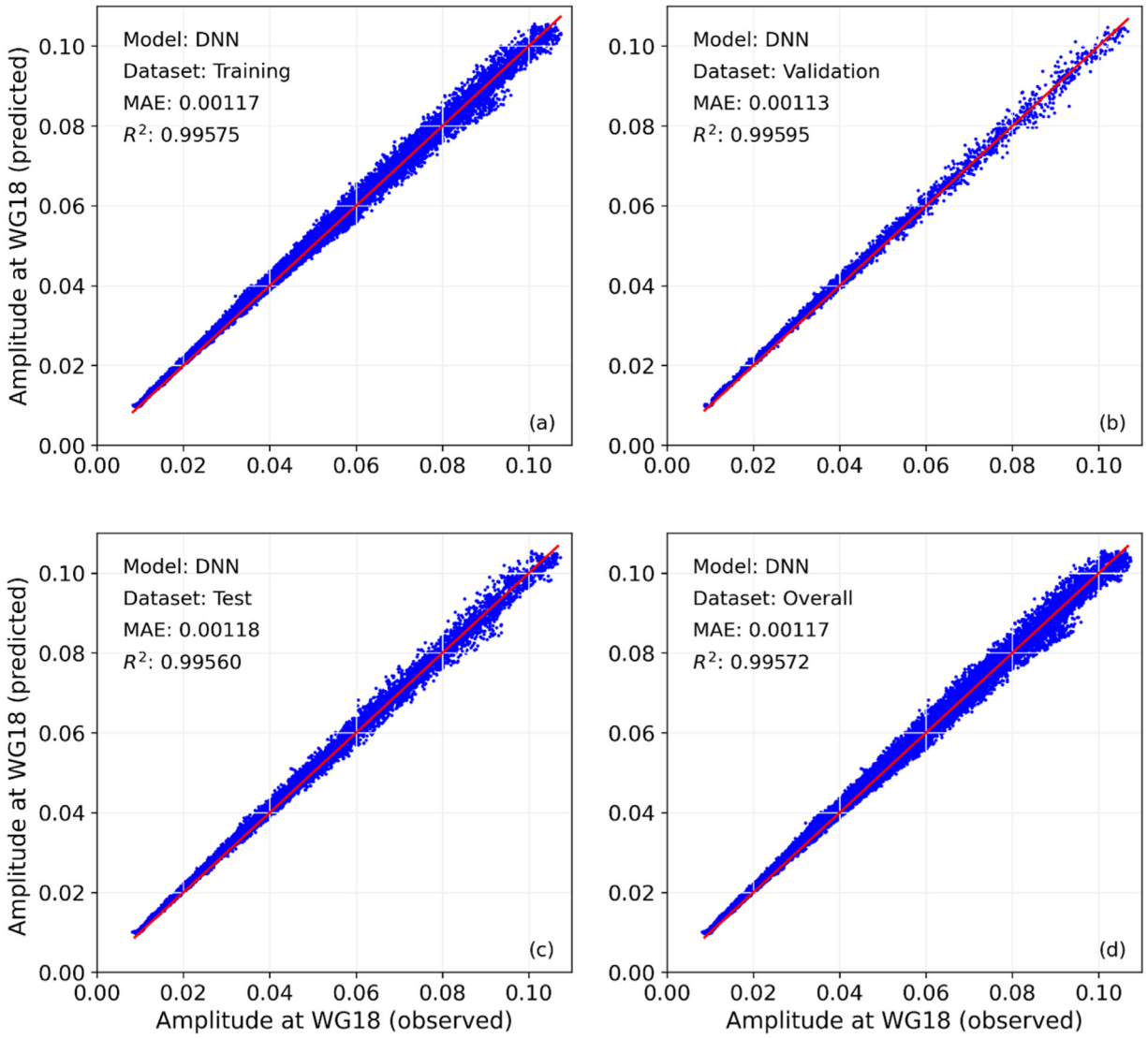

Fig. 12. Comparison of amplitude at WG18 between observed and predicted: (a) training data, (b) validation data, (c) test data, (d) overall data.

The predictions made by the DNN model and those made by alternative machine learning models— multiple linear regression (MLR), support vector regression (SVR), and extreme gradient boosting (XGBoost)—are then compared. MLR is a widely recognized statistical technique that describes the association between several independent variables and one dependent variable. The SVR algorithm is a machine learning technique employed for regression analysis; it is an expansion of the Support Vector Machine (SVM) approach, primarily utilized for classification tasks. XGBoost is an optimization method based on decision trees, integrates regularization terms, and utilizes advanced optimization techniques [22].

The training and test data for these three alternative models are similar to that of the DNN model. Fig. 13 shows the comparison of prediction results among the DNN model and the other three machine learning models. The corresponding quantitative results can be found in Table 6. As can be observed, the DNN model has the lowest MAE (0.00118) and RMSE (0.00151) among the four models. The DNN achieves exceptional performance as evidenced by its MAPE of 3 % and coefficient of determination of 0.99560. With a MAPE of 17 %, the MLR model is clearly inferior to the DNN model. In addition, this finding provides further evidence that the feature-response relationship is nonlinear. The MAPE for the SVR model was 12 %, and its R^2 was 0.94625; this improved over the MLR model but remained inferior to the DNN model. The MAPE for the XGBoost model is 2.8 %, and its determination coefficient is 0.99499. In addition, its MAE and RMSE values are 0.00118 and 0.00161, respectively. XGBoost produces predictions that are comparable to those of the DNN model, as indicated by these metrics.

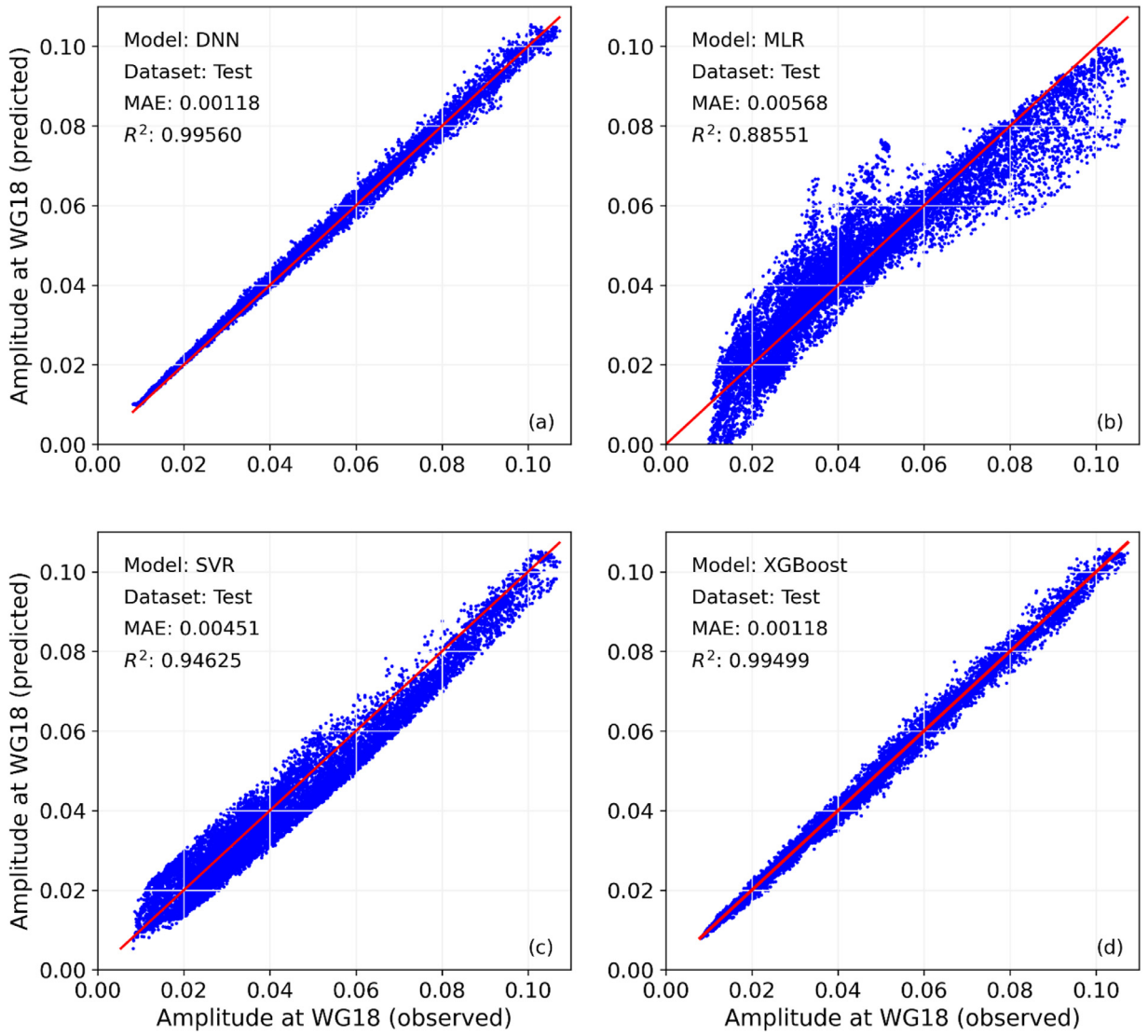

Fig. 13. Comparison of wave attenuation for deep neural networks and other machine learning techniques between the observed and predicted values.

Table 6
Comparative analysis of R^2, MAE, RMSE, and MAPE metrics across DNN, MLR, SVR, and XGBoost models.

Machine Learning Method	Statistical Measure			
	R^2	MAE	RMSE	MAPE
Deep neural network (DNN)	0.99560	0.00118	0.00151	0.03098
Multiple linear regression (MLR)	0.88551	0.00568	0.00770	0.17260
Support vector regression (SVR)	0.94625	0.00451	0.00528	0.12758
Extreme gradient boosting (XGBoost)	0.99499	0.00118	0.00161	0.02853

Conclusion

To predict the attenuation of tsunami waves by mangrove forests, we have developed a deep neural network (DNN) model as an alternative to the comparatively inefficient wave model calculation. The deep learning model was constructed using a dataset generated from the Staggered Variational Boussinesq (SVB) wave model. The SVB model was validated using experimental data exhibiting a mean absolute error (MAE) ranging from 0.003 to 0.01. When constructing the deep neural network model, hyperparameter optimization and neural network architecture selection were considered to achieve optimal performance. The DNN's performance

was assessed using validation and test data. The DNN model yielded an MAE of 0.00118 and a coefficient of determination (R^2) of 0.99560. An evaluation is conducted to compare the predictions produced by the DNN model with those of three alternative machine learning models: extreme gradient boosting (XGBoost), support vector regression (SVR), and multiple linear regression (MLR). The DNN model exhibits the least MAE of 0.00118 and a root mean squared error (RMSE) of 0.00151 among the four models. With a mean absolute percentage error (MAPE) of 3 % and a coefficient of determination of 0.99560, the DNN model demonstrates its outstanding performance. The MAE of XGBoost is 0.00118, which means that it makes predictions comparable to the DNN model.

The use of DNN as a regression model has certain drawbacks. Firstly, the DNN model requires high data volume for accurate regression. Secondly, due to the complexity of hyperparameter optimization in DNN models, only a few parameters can be optimized. In addition, DNNs involve computationally expensive processes, particularly when applied to large datasets and complex architectures. Future studies in wave attenuation prediction using DNN models should focus on devising algorithms to enhance the efficiency of the training process and investigating the integration of DNN algorithms with gradient boosting methods.

Limitations

Not applicable

Ethics statements

The present study did not involve human subjects, animal experiments, or data collected from social media platforms.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Didit Adytia: Supervision, Conceptualization, Methodology, Writing – review & editing. **Dede Tarwidi:** Conceptualization, Methodology, Validation, Visualization, Formal analysis, Writing – original draft. **Deni Saepudin:** Methodology, Writing – review & editing. **Semeidi Husrin:** Conceptualization, Writing – review & editing. **Abdul Rahman Mohd Kasim:** Conceptualization, Writing – review & editing. **Mohd Fakhizan Romlie:** Writing – review & editing. **Dafrizal Samsudin:** Writing – review & editing.

Data availability

The authors do not have permission to share data.

Acknowledgments

This work was supported by the Telkom University International Research Grant. We are grateful for their generous support. Data from the laboratory experiment is from the DFG-TAPFOR Project, LWI, TU Braunschweig, Germany.

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.mex.2024.102791.

References

[1] Y. Yao, X. Yang, S.H. Lai, R.J. Chin, Predicting Tsunami-like solitary wave run-up over fringing reefs using the multi-layer perceptron neural network, Nat. Hazards 107 (1) (2021) 601–616 May, doi:10.1007/s11069-021-04597-w.
[2] D. Adytia, S. Husrin, A.L. Latifah, Dissipation of solitary wave due to mangrove forest: a numerical study by using non-dispersive wave model, ILMU Kelaut. Indones. J. Mar. Sci. 24 (1) (2019) 41–50 Feb., doi:10.14710/ik.ijms.24.1.41-50.
[3] N. Malvin, S.R. Pudjaprasetya, D. Adytia, Neural Network modelling on wave dissipation due to mangrove forest, in: 2020 International Conference on Data Science and Its Applications (ICODSA), 2020, pp. 1–7, doi:10.1109/ICoDSA50139.2020.9212826. Aug..
[4] S. Gong, J. Chen, C. Jiang, S. Xu, F. He, Z. Wu, Prediction of solitary wave attenuation by emergent vegetation using genetic programming and artificial neural networks, Ocean Eng 234 (2021) 109250 Aug., doi:10.1016/j.oceaneng.2021.109250.
[5] W.C. Wu, D.T. Cox, Effects of wave steepness and relative water depth on wave attenuation by emergent vegetation, Estuar. Coast. Shelf Sci. 164 (2015) 443–450 Oct., doi:10.1016/j.ecss.2015.08.009.
[6] F.J. Mendez, I.J. Losada, An empirical model to estimate the propagation of random breaking and nonbreaking waves over vegetation fields, Coast. Eng. 51 (2) (2004) 103–118 Apr., doi:10.1016/j.coastaleng.2003.11.003.
[7] Z. Huang, Y. Yao, S.Y. Sim, Y. Yao, Interaction of solitary waves with emergent, rigid vegetation, Ocean Eng 38 (10) (2011) 1080–1088 Jul., doi:10.1016/j.oceaneng.2011.03.003.
[8] M. Maza, J.L. Lara, I.J. Losada, Tsunami wave interaction with mangrove forests: a 3-D numerical approach, Coast. Eng. 98 (2015) 33–54 Apr., doi:10.1016/j.coastaleng.2015.01.002.
[9] A. Abdolali, Wave attenuation by vegetation: model implementation and validation study, Front. Built Environ. 8 (2022) Accessed: Jan. 22, 2024. [Online]. Available:, doi:10.3389/fbuil.2022.891612.

[10] W.D. Lee, D.T. Cox, D.S. Hur, Numerical model study on the wave and current control by coastal vegetation, J. Coast. Res. 79 (sp1) (2017) 219–223 May, doi:10.2112/SI79-045.1.
[11] K. Iimura, N. Tanaka, Numerical simulation estimating effects of tree density distribution in coastal forest on tsunami mitigation, Ocean Eng. 54 (2012) 223–232 Nov., doi:10.1016/j.oceaneng.2012.07.025.
[12] T. Kim, Y. Kwon, J. Lee, E. Lee, S. Kwon, Wave attenuation prediction of artificial coral reef using machine-learning integrated with hydraulic experiment, Ocean Eng 248 (2022) 110324 Mar., doi:10.1016/j.oceaneng.2021.110324.
[13] W. Dharmawan, M. Diana, B. Tuntari, I.M. Astawa, S. Rahardjo, H. Nambo, Tsunami tide prediction in shallow water using recurrent neural networks: model implementation in the Indonesia Tsunami Early Warning System, J. Reliab. Intell. Environ. (2023) Nov., doi:10.1007/s40860-023-00214-8.
[14] D. Adytia, S.R. Pudjaprasetya, D. Tarwidi, Modeling of wave run-up by using staggered grid scheme implementation in 1D Boussinesq model, Comput. Geosci. 23 (4) (2019) 793–811 Aug., doi:10.1007/s10596-019-9821-5.
[15] D. Adytia, S.R. Pudjaprasetya, Numerical simulation of wave runup and overtopping for short and long waves using staggered grid variational boussinesq, J. Earthquake Tsunami 14 (05) (2020), doi:10.1142/S1793431120400059.
[16] D. Adytia, D. Tarwidi, S.A. Kifli, S.R. Pudjaprasetya, Staggered grid implementation of 1D Boussinesq model for simulating dispersive wave, Journal of Physics: Conference Series, 971, 2018, doi:10.1088/1742-6596/971/1/012020.
[17] D. Tarwidi, D. Adytia, Parallelization of elliptic solver for solving 1D Boussinesq model, J. Phys.: Conf. Series 971 (1) (2018), doi:10.1088/1742-6596/971/1/012036.
[18] S. Husrin, A. Strusińska, H. Oumeraci, Experimental study on tsunami attenuation by mangrove forest, Earth Planets Space 64 (10) (2012) 15 Oct., doi:10.5047/eps.2011.11.008.
[19] Y. LeCun, L. Bottou, G.B. Orr, K.R. Müller, Efficient BackProp, in: G.B. Orr, K.R. Müller (Eds.), Neural Networks: Tricks of the Trade, Springer, Berlin, Heidelberg, 1998, pp. 9–50, doi:10.1007/3-540-49430-8_2. Eds.Lecture Notes in Computer Science.
[20] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) Art. 7553May, doi:10.1038/nature14539.
[21] C.H. Chen, J.P. Lai, Y.M. Chang, C.J. Lai, P.F. Pai, A study of optimization in deep neural networks for regression, Electronics (Basel) 12 (14) (2023) Art. 14, Jan., doi:10.3390/electronics12143071.
[22] D. Tarwidi, S.R. Pudjaprasetya, D. Adytia, M. Apri, An optimized XGBoost-based machine learning method for predicting wave run-up on a sloping beach, MethodsX 10 (2023) 102119 Jan., doi:10.1016/j.mex.2023.102119.