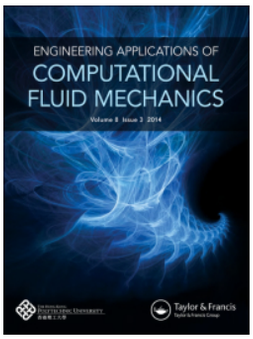

Taylor & Francis
Taylor & Francis Group

Engineering Applications of Computational Fluid Mechanics

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tcfm20

Accurate discharge coefficient prediction of streamlined weirs by coupling linear regression and deep convolutional gated recurrent unit

Weibin Chen, Danial Sharifrazi, Guoxi Liang, Shahab S. Band, Kwok Wing Chau & Amir Mosavi

To cite this article: Weibin Chen, Danial Sharifrazi, Guoxi Liang, Shahab S. Band, Kwok Wing Chau & Amir Mosavi (2022) Accurate discharge coefficient prediction of streamlined weirs by coupling linear regression and deep convolutional gated recurrent unit, Engineering Applications of Computational Fluid Mechanics, 16:1, 965-976, DOI: 10.1080/19942060.2022.2053786

To link to this article: https://doi.org/10.1080/19942060.2022.2053786

Published online: 11 Apr 2022.

ENGINEERING APPLICATIONS OF COMPUTATIONAL FLUID MECHANICS
2022, VOL. 16, NO. 1, 965–976
https://doi.org/10.1080/19942060.2022.2053786

Taylor & Francis
Taylor & Francis Group

Accurate discharge coefficient prediction of streamlined weirs by coupling linear regression and deep convolutional gated recurrent unit

Weibin Chen[a], Danial Sharifrazi[b], Guoxi Liang[c], Shahab S. Band[d], Kwok Wing Chau [e] and Amir Mosavi [f,g,h]

[a]College of Computer Science and Artificial Intelligence, Wenzhou University, Wenzhou, People's Republic of China; [b]Department of Computer Engineering, Shiraz Branch, Islamic Azad University, Shiraz, Iran; [c]Department of Artificial Intelligence, Wenzhou Polytechnic, Wenzhou, People's Republic of China; [d]Future Technology Research Center, National Yunlin University of Science and Technology, Yunlin, Taiwan; [e]Department of Civil and Environmental Engineering, Hong Kong Polytechnic University, Hong Kong, People's Republic of China; [f]Institute of Information Society, University of Public Service, Budapest, Hungary; [h]John von Neumann Faculty of Informatics, Obuda University, Budapest, Hungary; [g]Institute of Information Engineering, Automation and Mathematics, Slovak University of Technology in Bratislava, Bratislava, Slovakia

ABSTRACT

Streamlined weirs, which are a nature-inspired type of weir, have gained tremendous attention among hydraulic engineers, mainly owing to their established performance with high discharge coefficients. Computational fluid dynamics (CFD) is considered as a robust tool to predict the discharge coefficient. To bypass the computational cost of CFD-based assessment, the present study proposes data-driven modeling techniques, as an alternative to CFD simulation, to predict the discharge coefficient based on an experimental dataset. To this end, after splitting the dataset using a *k*-fold cross-validation technique, the performance assessment of classical and hybrid machine learning–deep learning (ML-DL) algorithms is undertaken. Among ML techniques, linear regression (LR), random forest (RF), support vector machine (SVM), *k*-nearest neighbor (KNN) and decision tree (DT) algorithms are studied. In the context of DL, long short-term memory (LSTM), convolutional neural network (CNN) and gated recurrent unit (GRU), and their hybrid forms, such as LSTM-GRU, CNN-LSTM and CNN-GRU techniques, are compared using different error metrics. It is found that the proposed three-layer hierarchical DL algorithm, consisting of a convolutional layer coupled with two subsequent GRU levels, which is also hybridized with the LR method (i.e. LR-CGRU), leads to lower error metrics. This paper paves the way for data-driven modeling of streamlined weirs.

ARTICLE HISTORY
Received 27 April 2021
Accepted 10 March 2022

KEYWORDS
Streamlined weirs; discharge prediction; deep learning; machine learning; deep convolutional neural network; gated recurrent unit

1. Introduction

Weirs are among the most useful and common hydraulic structures, being used in various applications, such as irrigation networks, sewage networks and water supply systems (Abdollahi et al., 2017). According to the crest type, the main weir groups are classified into sharp-, broad- and short-crested weirs. Circular-crested, overflow (ogee) and streamlined weirs are special kinds of short-crested weirs (Bagheri & Kabiri-Samani, 2020a). Streamlined weirs, which are a nature-inspired type of weir, have gained tremendous attention among hydraulic engineers owing to their well-known performance with high discharge coefficient, overflow stability behavior and minimized fluctuation in free water surface. The general shape of a streamlined weir, which is designed according to airfoils, was originally derived from the topology of a bird's wings. The importance of streamlined weirs, purported to be the most state-of-the-art form of weir, is well documented in the hydraulic engineering field (Bagheri & Kabiri-Samani, 2020a, 2020b; Rao & Rao, 1973). However, owing to the complex geometry of streamlined weirs in their design, this kind of weir received less attention from practitioners. The estimation of the discharge coefficient of weirs is an important subject since many experimental and/or numerical studies have been undertaken in different types of weir (Arvanaghi et al., 2014; Arvanaghi & Oskuei, 2013; Borghei et al., 1999; Johnson, 2000; Mahtabi & Arvanaghi, 2018; Qu et al., 2009; Rady, 2011; Tullis, 2011). For the past two decades, computational fluid dynamics (CFD) has drawn tremendous attention from both academia and industry to model problems involving fluid domains and their corresponding boundary conditions and interactions. OpenFOAM software, an open-source toolbox, is widely used in high-fidelity computational models owing to its incorporation of a vast variety of solvers compatible with different ranges of fluid flow. Although CFD-based performance assessment of fluid-flow phenomena leads to

CONTACT Guoxi Liang guoxiliang@wzpt.edu.cn; Shahab S. Band shamshirbands@yuntech.edu.tw; Amir Mosavi amir.mosavi@uni-obuda.hu

reliable results, it suffers from computationally demanding procedures and a requirement for profound academic knowledge in the field of fluid mechanics (Bagheri & Kabiri-Samani, 2020a, 2020b).

Data-driven modeling offers a framework to assess a model as a black box. Hence, it is possible to analyze a broader range of models and systems, irrespective of the nature of the problem. In particular, machine learning–deep learning (ML-DL) modeling is an active field of research in engineering fields such as structural and earthquake engineering (Abasi et al., 2021; Barkhordari & Es-haghi, 2021; Barkhordari & Tehranizadeh, 2021; Esteghamati & Flint, 2021; Hariri-Ardebili & Salazar, 2020; Pourkamali-Anaraki et al., 2020; Soraghi & Huang, 2021) and biomedical engineering. Other applications of ML-DL techniques can also be found (Aswin et al., 2018; Athira et al., 2018; Selvin et al., 2017; Vinayakumar et al., 2017).

Different machine learning (ML) and surrogate modeling algorithms have been applied in various hydraulic engineering problems, such as dams, sedimentation and spillways (Amini et al., 2021a, 2021b; Bhattacharya et al., 2007; Hariri-Ardebili et al., 2021; Roushangar et al., 2014; Torres-Rua et al., 2012). It is recognized that an empirical relationship for the discharge coefficient based on experimental or hydraulic models faces some limitations regarding hydraulic and geometric parameters (Ebtehaj et al., 2018). The main motivation of the present study is to bypass the computational cost of discharge coefficient prediction using a CFD framework by investigating the potential capability of hybrid ML-DL algorithms as an alternative to CFD-based simulations. The comparison between the CFD-based discharge coefficient and the proposed data-driven techniques is graphically illustrated in Figure 1. which was inspired by Bagheri and Kabiri-Samani (2020a, 2020b). The data-driven modeling part of Figure 1 will be discussed comprehensively in Sections 4 and 5.

The incorporation of various geometric and hydraulic parameters affecting the hydraulic operations of weirs requires the application of an accurate model to determine their discharge coefficients. In this context, proposing an accurate technique for the estimation of discharge coefficient is a challenging task.

In this work, a group of 12 classical and hybrid ML-DL algorithms is employed to predict the discharge coefficient of streamlined weirs based on an experimental dataset. In the following text, Section 2 describes literature related to different uses of ML-DL techniques in weirs. Section 3 explains the data employed in this study. Section 4 describes the ML-DL algorithms, including the proposed method. Section 5 illustrates the results obtained by different data-driven techniques. Finally, in Section 6, conclusions are presented and directions for future work are outlined.

2. Related works

The determination of the discharge coefficient of weirs is the most important factor in the design of these hydraulic structures. Several studies have been performed using

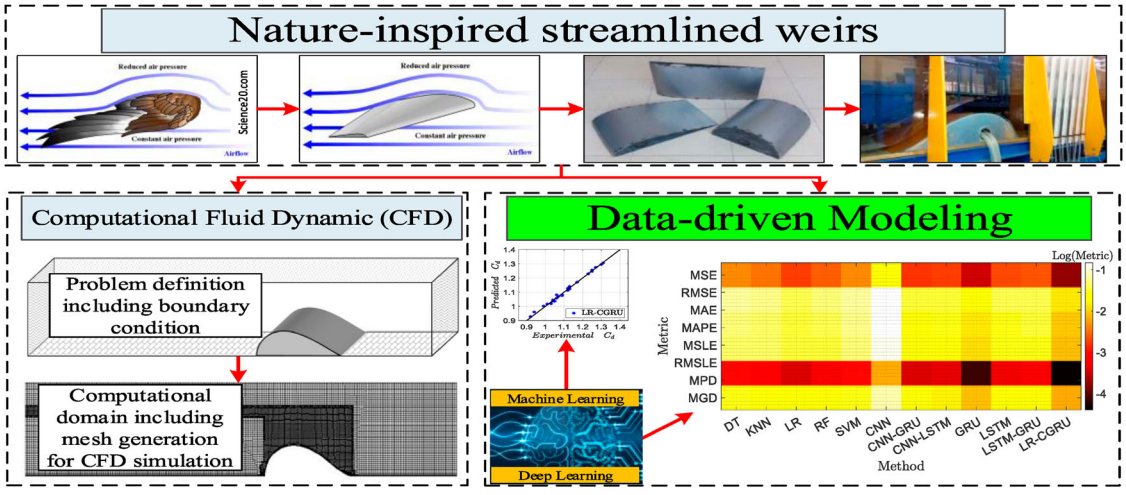

Figure 1. Data-driven discharge coefficient estimation of streamlined weirs as an alternative to the computational fluid dynamics (CFD)-based procedure. MSE = mean squared error; RMSE = root mean squared error; MAE = mean absolute error; MAPE = mean absolute percentage error; MSLE = mean squared logarithmic error; RMSLE = root mean squared logarithmic error; MPD = mean Poisson deviance; MGD = mean gamma deviance; DT = decision tree; KNN = *k*-nearest neighbor; LR = linear regression; RF = random forest; SVM = support vector machine; CNN = convolutional neural network; GRU = gated recurrent unit; LSTM = long short-term memory.

Table 1. Previous works on discharge coefficient estimation of weirs using different soft computing techniques.

Weir configuration	Soft computing techniques	Reference
Sharp-crested weir	FFNN, RBNN	Bilhan et al. (2010)
Triangular labyrinth side weirs	ANN	Emiroglu et al. (2011)
Broad-crested weir	GP, ANN	Salmasi et al. (2013)
Triangular labyrinth side weirs	MLP, RBNN	Zaji and Bonakdari (2014)
Side weirs	MLP	Parsaie and Haghiabi (2015)
Trapezoidal and rectangular side weirs	SVM and GA (SVM-GA), GEP	Roushangar et al. (2016)
Two-cycle labyrinth weirs	ANFIS, MNLR	Aydin and Kayisli (2016)
Side weirs	SVM	Azamathulla et al. (2016)
Triangular labyrinth weirs	SVR, SVR-FA, RSM, PCA	Karami et al. (2017)
Triangular labyrinth weirs	MLP-NN, RBNN, SVM	Parsaie and Haghiabi (2017)
Rectangular side weirs	ANFIS	Ebtehaj et al. (2018)
Labyrinth weirs	ANFIS, MLP-NN	Haghiabi et al. (2018)
Piano key weir	MLP, MLP-FA, MLP-PSO, MLP-GA, MLP-MFO, ANFIS, ANFIS-FA, ANFIS-PSO, ANFIS-GA, ANFIS-MFO	Zounemat-Kermani et al. (2019)
Trapezoidal labyrinth weirs	MLP-NN, RBNN, SVM	Norouzi et al. (2019)
Labyrinth weirs	ANFIS, ANFIS-FFA	Shafiei et al. (2020)
Skew side weir	MLR, GEP	Mohammed and Sharifi (2020)
Sharp-crested weirs	ANN, SVM, ELM	Li et al. (2021)
Triangular labyrinth weirs	ANFIS, ANFIS-PSO, ANFIS-FA, SVR, SVR-FA, MLP, MLP-FA, RBNN	Mahmoud et al. (2021)

Note: FFNN = feed forward neural network; RBNN = radial basis neural network; ANN = artificial neural network; GP = genetic programming; MLP = multi-layer perceptron neural network; SVM = support vector machine; GA = genetic algorithm; GEP = gene expression programming; ABFIS = adaptive neuro-fuzzy inference system; MNLR = multiple nonlinear regression; SVR = support vector regression; FA = firefly algorithm; RSM = response surface methodology; PCA = principal component analysis; PSO = particle swarm optimization; MFO = moth-flame optimization; FFA = neuro-fuzzy-firefly; MLR = multiple linear regression; ELM = extreme learning machine.

various ML-DL algorithms to predict the discharge coefficient. Some of the state-of-the-art ML-DL techniques related to the estimation of the discharge coefficient are presented in Table 1, considering different weir configurations. One may note that none of the existing studies investigated the potential capability of ML-DL techniques for streamlined weirs, which reflects the main motivation of the present study.

3. Data description

The flow rate Q over a short-crest weir is computed based on continuity and Bernoulli's equations, as expressed in Equation (1):

$$Q = \frac{2}{3} C_d B \sqrt{\frac{2}{3} g} H_1^{3/2} \tag{1}$$

where C_d is the weir discharge coefficient; B is the weir width; $H_1 = h_1 + h_v$ is the total head; h_1 is the upstream head over the crest; h_v is the upstream velocity head and is equal to $v^2/2g$; v is the approach velocity; and g is the acceleration due to gravity.

In this research, an experimental dataset for 120 models of streamlined weirs, which are designed based on the principle of the Joukowsky transform function, is used (Bagheri & Kabiri-Samani, 2020a). The model is graphically illustrated in Figure 2 and the related hydraulic parameters are shown in Table 2, which is adapted from (Bagheri & Kabiri-Samani, 2020a). The data consist of two groups, namely with and without a base block under streamlined weirs. In models without a base block, parameter β is considered equal to zero. Table 2 shows nine parameters, which are considered as model inputs in the proposed method. The discharge coefficient is the model output.

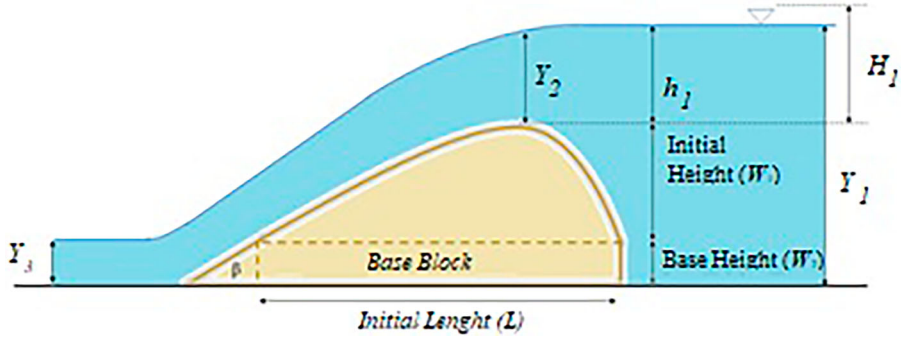

Figure 2. Schematic view of a streamlined weir.

Table 2. Input parameters for estimating the discharge coefficient.

Input parameter	Description
λ	Relative eccentricity
β	Angle between the downstream slope of weirs, fixed and horizontal axis
L	Initial length of the streamlined weir
W	Total weir height
Q	Flow discharge
Y_1	Upstream water depth
Y_2	Water depth at the weir crest
Y_3	Downstream flow depth
h_1	Upstream flow depth on the weir crest

4. Methods

The studied ML-DL methods are introduced in Section 4.1. Details of the implemented methods and parameters are also stated. The proposed method is introduced in detail in Section 4.2. All data-driven techniques are implemented in Python programming language. In this research, 'sklearn' and 'keras' packages by 'tensorflow' backend are used for program development. A GPU GFORCE GTX950 with 16 GB RAM DDR4 is used as the implementation hardware.

4.1. Machine learning–deep learning algorithms

With the development of ML-DL methods, various ML-DL-based models have been introduced and have received extended attention (see Table 1). In the present study, five classical ML techniques are applied to estimate the discharge coefficient. The performance assessment of the support vector machine (SVM), random forest (RF), linear regression (LR), *k*-nearest neighbor (KNN) and decision tree (DT) algorithms is undertaken via error metrics. Among these ML techniques, the candidate with the highest accuracy is considered as the accepted ML technique in the present study. All model parameters of classical ML techniques are summarized in Table 3. Since the applied ML techniques are well documented in the literature, readers are referred to Sammut and Webb (2011) for a detailed discussion on the mentioned classical ML techniques.

As mentioned in Section 1, the main objective of this study is to propose an accurate data-driven technique to estimate the discharge coefficient. Accordingly, we assess the capability of six classical and hybrid deep learning (DL) techniques in comparison to a three-layer hierarchical DL technique for possible adaptive implementation with a successful ML technique in a state-of-the-art hydraulic engineering application. Deep neural networks (DNNs) are created from artificial neural networks (ANNs). ANNs usually contain few (shallow) layers, whereas DNNs contain more hidden (deep) layers. With more layers, DNNs are capable of learning big data (Wang et al., 2019). DL is a method that predicts results through several layers, with each layer containing the weights of a neural network (Zhao et al., 2019). As a result, it can be said that DL is a special kind of neural network that involves more layers. Within this framework, increasing the number of layers in DL has led to better outcomes than simple ANNs. In the context of DL, long short-term memory (LSTM) (Hochreiter & Schmidhuber, 1997), convolutional neural network (CNN) (LeCun et al., 1995) and gated recurrent unit (GRU) (Cho et al., 2014), and their hybrid forms, such as LSTM-GRU, CNN-LSTM and CNN-GRU techniques, are analyzed by different error metrics. In the following, DL techniques are introduced briefly, and a detailed discussion on the proposed algorithm is provided in Section 4.2. As a variant of the recurrent neural network (RNN), LSTM has a long-term memory function that is suitable for processing important events with long intervals and delays in time series. Therefore, the neural network structure, which is primarily composed of LSTM units with memory functions, can make decisions based on the previous states to adapt to various running scenarios (Guo et al., 2021). LSTM has been widely used in issues related to sequential data, such as natural language processing (NLP), voice recognition and time-series analysis (Sezer & Ozbayoglu, 2018).

Table 3. Parameter values of machine learning (ML) algorithms.

SVM	RF	KNN	DT
Kernel = RBF	n_estimators = 100	n_neighbors = 5	Criterion = MSE
Degree = 3	Criterion = MSE	Weights = Uniform	Splitter = Best
Gamma = Scale	min_samples_split = 2	Algorithm = Auto	min_samples_split = 2
Coef0 = 0.0	min_samples_leaf = 1	Leaf size = 30	min_samples_leaf = 1
Shrinking = True	min_weight_fraction_leaf = 0.	p = 2	min_weight_fraction_leaf = 0.
Cache size = 200	Max features = Auto	Metric = Minkowski	
Epsilon = 0.1	Bootstrap = True		
Tol = 1e-3			
C = 1.0			

Note: SVM = support vector machine; RF = random forest; KNN = *k*-nearest neighbor; DT = decision tree; RBF = radial basis function; MSE = mean squared error.

The original idea for the CNN was initially modeled on mammalian vision. This type of network is able to achieve results similar to humans in some cases and even stronger than human vision in other cases. A CNN is made up of a number of convolutional layers. From the combination of these layers of convolution, a DNN is formed. CNN has been widely used and has achieved brilliant results in image processing, image classification and computer vision (Sammut & Webb, 2011).

Similarly to LSTM, GRU is a variant of RNN. In general, two main layers are implemented in GRU. It first determines how the previous information should be passed along to the future. Next, it determines how much of the past information must be discarded in the second layer. GRU leads to better performance for smaller and less frequent datasets in comparison to LSTM (Gruber & Jockisch, 2020). Model parameters of these classical DL techniques are summarized in Table 4.

Hybrid DL techniques are constructed by coupling classical DL algorithms. In this context, LSTM-GRU is developed by two LSTM layers and one GRU layer, in which the number of neurons of LSTM and GRU layers is assumed to be 50. The other remaining parameters are identical to LSTM and GRU parameters. In the CNN-LSTM approach, one convolutional layer and two LSTM layers are applied, while the remaining parameters are obtained from classical DL. The same implementation is assumed for CNN-GRU, where one convolutional layer and two GRU layers are mixed. All the remaining parameters of the proposed LR-CGRU method, which is a three-layer hierarchical DL algorithm consisting of a convolutional layer coupled with two subsequent GRU levels, hybridized with LR, are assumed to be equal to those of the LR, CNN and GRU algorithms.

Table 4. Parameter values of classical deep learning (DL) algorithms.

LSTM	CNN	GRU
Layers: 3 LSTM layers	Layers: 3 convolutional layers	Layers: 3 GRU layers
Number of neurons: 50	Number of filters: 64	Number of neurons: 50
Number of epochs: 200	Number of epochs: 200	Number of epochs: 200
Activation for all layers (except the last): ReLU	Activation for all layers (except the last): ReLU	Activation for all layers (except the last): ReLU
Loss function: MSE	Loss function: MSE	Loss function: MSE
Optimizer: Adam	Optimizer: Adam	Optimizer: Adam
beta1 of optimizer: 0.9	beta1 of optimizer: 0.9	beta1 of optimizer: 0.9
beta2 of optimizer: 0.999	beta2 of optimizer: 0.999	beta2 of optimizer: 0.999
Learning rate: 0.001	Learning rate: 0.001	Learning rate: 0.001
	Size of kernels: 3∗3	

Note: LSTM = long short-term memory; CNN = convolutional neural network; GRU = gated recurrent unit; ReLU = rectified linear activation function; MSE = mean squared error.

4.2. Proposed method (LR-CGRU)

The dataset is split into the 'training' and 'testing' groups to generate meta-inputs for the proposed algorithm. A successful out-of-sampling technique for this purpose is the k-fold cross-validation (CV) technique. In this context, by transforming the whole dataset into k mutually exclusive and collectively exhaustive subsets, only one set is used for testing and the remaining $(k-1)$ subgroup is incorporated in the training procedure. In addition, the initial weight assignment of ML-DL algorithms is commonly performed by a random configuration. Hence, the k-fold CV technique can lead to unbiased assessment. In the ML-DL algorithm proposed in the present study, $k = 5$ is used for the CV tool. According to Razavi-Far et al. (2019), the predictive models are trained in a 'one-step-ahead' configuration.

A three-layer hierarchical DL algorithm consisting of a convolutional layer coupled with two GRU levels is introduced as the final DL algorithm, which is also hybridized by the LR method as the ML technique owing to its lower CV errors (a detailed explanation of the error metrics and their obtained values for ML-DL algorithms will be given in Section 5). Accordingly, LR-CGRU is the combination of LR, CNN and GRU, and uses a convolutional layer as the first layer and two GRU layers in the subsequent DL phase. A graphical representation of the proposed algorithm is shown in Figure 3.

The proposed model is trained five times owing to the use of five-fold CV technique. In the five-fold CV technique, the model is trained with 80% of the dataset and tested on the remaining 20%. Accordingly, we have five predicted datasets for both ML and DL algorithms, in

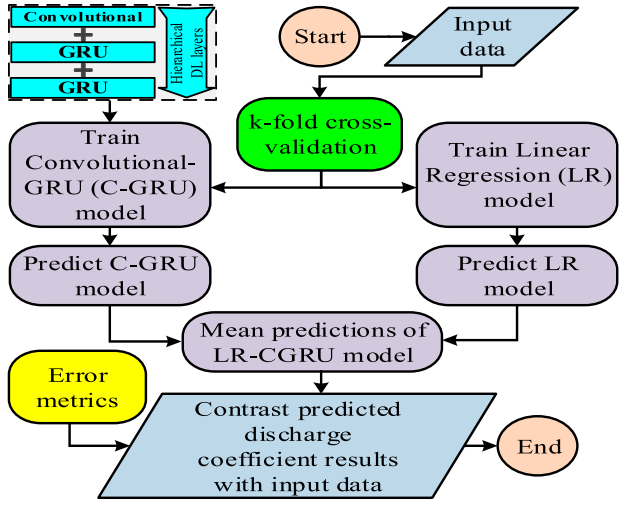

Figure 3. Flowchart of the proposed method: LR-CGRU, consisting of a machine learning (ML) algorithm [i.e. linear regression (LR)] coupled with a three-layer hierarchical deep learning (DL) technique [i.e. convolutional gated recurrent unit (C-GRU)].

which the computed data are averaged for both ML and DL methods.

5. Results and discussion

5.1. Verification of the proposed algorithm

In this section, in the first stage, the predicted results of all ML methods, including SVM, RF, LR, KNN and DT, are compared with the experimental results, which are graphically shown in Figure 4(a)–(e). It can be observed that the LR and RF methods provide better results than the other ML techniques in terms of the YY plot.

An ML-DL model can be evaluated in a complicated manner. The dataset is usually split into training and testing sets. Then, the model performance is evaluated based on an error metric to specify the precision of the model. However, this technique is not reliable enough as the computed accuracy for one test set may be very different from another one. To cope with this problem, *k*-fold CV is performed. As mentioned in Section 4.2, the five-fold CV technique is considered for all applied ML-DL algorithms. In detail, in the first iteration, the first fold is used to test the ML-DL model and the rest of the data are considered as the training set. In the next iteration, the second fold is used as the testing set and the rest of data are employed as a training set. This procedure continues until five folds have been used.

To assess the performance of each ML-DL method, eight error metrics, namely mean squared error (MSE), root mean squared error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), mean squared logarithmic error (MSLE), root mean squared logarithmic error (RMSLE), mean Poisson deviance (MPD) and mean gamma deviance (MGD), are employed. These error metrics are introduced in Equations (2)–(9), respectively:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3}$$

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n}|y_i - \hat{y}_i|}{n} \tag{4}$$

$$\mathrm{MAPE} = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{5}$$

$$\mathrm{MSLE} = \frac{1}{n}\sum_{i=1}^{n}(\log(y_i) - \log(\hat{y}_i))^2 \tag{6}$$

$$\mathrm{RMSLE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\log(y_i) - \log(\hat{y}_i))^2} \tag{7}$$

$$\mathrm{MPD} = \frac{1}{n}\sum_{i=0}^{n-1} 2\left(y_i \log\left(\frac{y_i}{\hat{y}_i}\right) + \hat{y}_i - y_i\right) \tag{8}$$

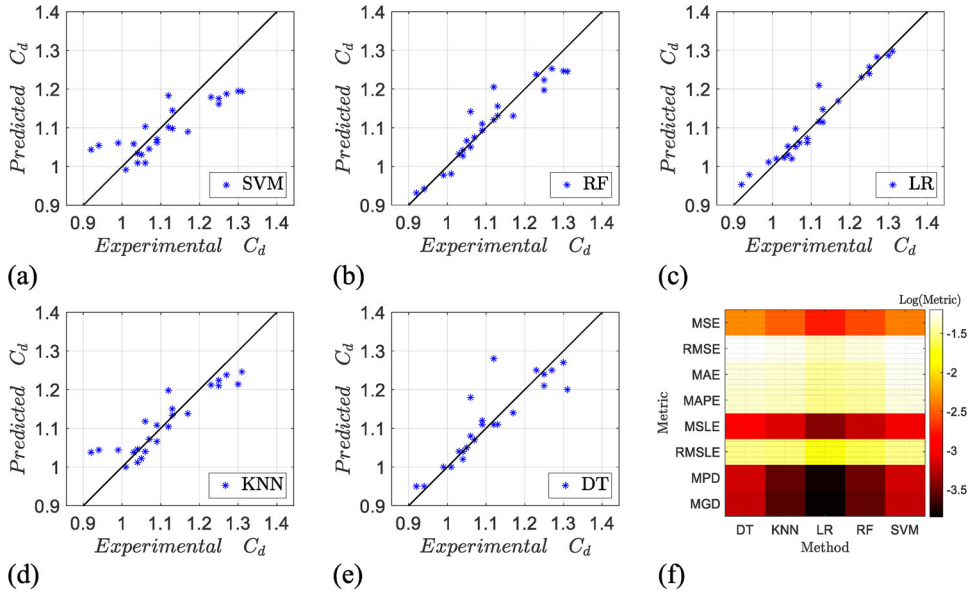

Figure 4. Comparison between the experimental data and machine learning (ML) methods: (a)–(e) predicted discharge coefficient vs experimental dataset; (f) results of six error metrics for all ML algorithms. SVM = support vector machine; RF = random forest; LR = linear regression; KNN = *k*-nearest neighbor; DT = decision tree; MSE = mean squared error; RMSE = root mean squared error; MAE = mean absolute error; MAPE = mean absolute percentage error; MSLE = mean squared logarithmic error; RMSLE = root mean squared logarithmic error; MPD = mean Poisson deviance; MGD = mean gamma deviance.

$$\mathrm{MGD} = \frac{1}{n}\sum_{i=0}^{n-1} 2\left(\log\left(\frac{\hat{y}_i}{y_i}\right) + \frac{y_i}{\hat{y}_i} - 1\right) \qquad (9)$$

where y_i is the real (i.e. experimental) dataset and $\hat{y}_i$ is the predicted outputs. Figure 4(f) shows the logarithmic values of the applied performance metrics for each ML method. According to Figure 4(f), linear regression, which has the darkest color of the methods, is considered the most successful ML technique in the present study.

In the next stage, classical DL methods (namely, LSTM, GRU and CNN) and their variants (namely CNN-LSTM, GRU, LSTM and LSTM-GRU) are applied to predict the discharge coefficient of streamlined weirs. The predicted outputs by the mentioned DL algorithms versus the experimental dataset are demonstrated in YY plots in Figure 5.

As it can be seen in the second row of Figure 5, all hybrid DL algorithms outperform the classical ones. However, to provide a robust conclusion, the eight error metrics in Figure 4(f) are applied again, and the logarithmic values of the error metrics are shown in Figure 6(a). To demonstrate the potential capability of the proposed methods, the error metrics of the LR-CGRU algorithm are also plotted in the last column of Figure 6(a). In general, it can be concluded that all hybrid algorithms considering both ML and DL, which are plotted in Figures 4(f) and 6(a), respectively, provide low error metrics. LR-CGRU not only leads to lower error considering all eight metrics, but also provides considerably lower metrics in MSE, MSLE, MPD and MGD. Moreover, the YY plot for the proposed method is shown in Figure 6(b), which

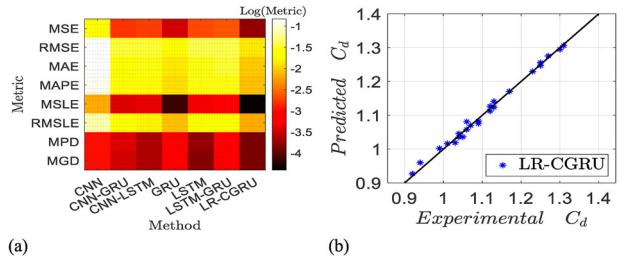

Figure 6. Three-layer hierarchical deep learning (DL) algorithm consisting of a convolutional layer coupled with two subsequent gated recurrent unit (GRU) levels, hybridized with linear regression (LR) method (LR-CGRU): (a) error metrics for all DL algorithms in conjunction with the LR-CGRU method; (b) YY plot for the proposed method. MSE = mean squared error; RMSE = root mean squared error; MAE = mean absolute error; MAPE = mean absolute percentage error; MSLE = mean squared logarithmic error; RMSLE = root mean squared logarithmic error; MPD = mean Poisson deviance; MGD = mean gamma deviance; CNN = convolutional neural network; LSTM = long short-term memory.

highlights the superiority of the LR-CGRU method. The computational cost regarding the training time of all ML-DL algorithms is presented in the Appendix. As expected, there is a sharp distinction between the computational costs of ML and DL algorithms. However, LR-CGRU provides an acceptable computational complexity compared to other classical and hybrid DL algorithms.

5.2. Comparison with previous works

Finally, the data-driven outputs are compared with those of previous related works. Bagheri and Kabiri-Samani

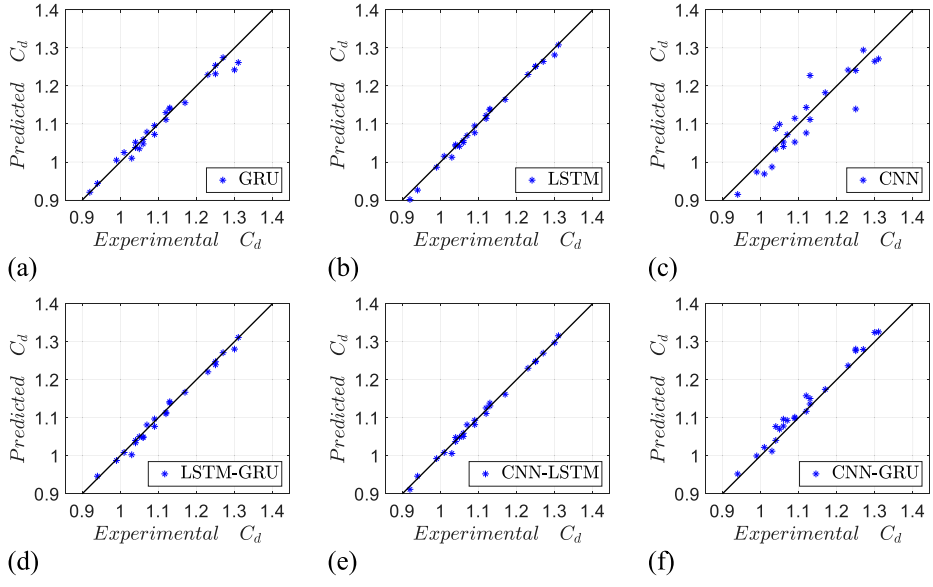

Figure 5. Comparison between the experimental dataset and derived outputs by the applied classical and hybrid deep learning (DL) methods. GRU = gated recurrent unit; LSTM = long short-term memory; CNN = convolutional neural network.

(2020a) proposed an algebraic equation to compute the streamlined discharge coefficient (C_d) using dimensional analysis and a curve-fitting tool in MATLAB, as follows:

$$C_d = 1.4\lambda^{0.05}\left[\frac{h_1}{L}\frac{h_1}{W}\right]^{0.1} \tag{10}$$

Carollo and Ferro (2021) proposed a relationship between discharge Q and upstream water level h_1, based on the experimental results of Bagheri and Kabiri-Samani (2020a), as shown in Equation (11):

$$A = a\left(\frac{h_1}{W}\right) = \frac{Q^{2/3}}{g^{1/3}b^{2/3}W} \tag{11}$$

Based on Equations (10) and (11), the coefficient a was:

$$a = \frac{2}{3}{C_d}^{2/3} \tag{12}$$

In Carollo and Ferro (2021), according to dimensional analysis and self-similarity theory, the stage-discharge relationship was obtained as:

$$A = 0.8546\left(\frac{h_1}{W}\right)^{1.1243}\left(\frac{L}{W}\right)^{-0.1012}\left(\frac{W_1}{W}\right)^{0.0412} \tag{13}$$

By combining Equations (11) and (12):

$$A = \frac{2}{3}{C_d}^{2/3}\frac{h_1}{W} \tag{14}$$

By substituting Equation (13) into Equation (14):

$$\frac{2}{3}{C_d}^{2/3}\frac{h_1}{W} = 0.8546\left(\frac{h_1}{W}\right)^{1.1243}\left(\frac{L}{W}\right)^{-0.1012} \times\left(\frac{W_1}{W}\right)^{0.0412} \tag{15}$$

In the last step, the discharge coefficient was obtained as:

$$C_d = \left[\left[\frac{3}{2}\frac{W}{h_1}\right]\left[0.8546\left(\frac{h_1}{W}\right)^{1.1243}\left(\frac{L}{W}\right)^{-0.1012} \times\left(\frac{W_1}{W}\right)^{0.0412}\right]\right]^{3/2} \tag{16}$$

In Figure 7, the results from equations proposed by Bagheri and Kabiri-Samani (2020a) (i.e. Equation 10) and Carollo and Ferro (2021) (i.e. Equation 16) are compared with those obtained by the proposed LR-CGRU algorithm. As can be seen, the proposed data-driven technique provides more accurate outputs than the algebraic expressions introduced by Bagheri and Kabiri-Samani (2020a) and Carollo and Ferro (2021), which highlights the superiority of ML-DL-driven techniques for the prediction of the discharge coefficient.

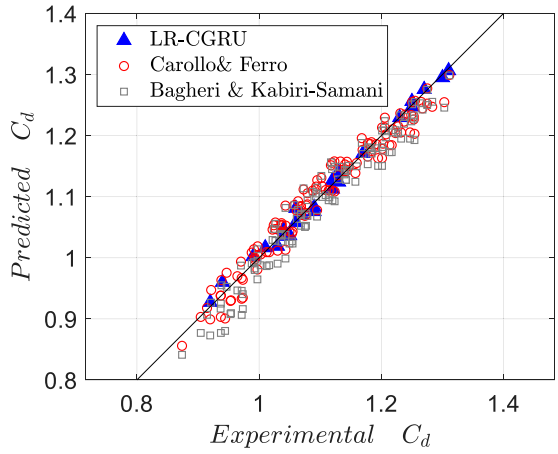

Figure 7. Comparison of outputs from the-layer hierarchical deep learning (DL) algorithm consisting of a convolutional layer coupled with two subsequent gated recurrent unit (GRU) levels, hybridized with linear regression (LR) method (LR-CGRU) (blue triangles) with previous works (Carollo & Ferro, 2021; Bagheri & Kabiri-Samani, 2020a) in a YY plot.

6. Conclusion and future works

This paper aims to predict the discharge coefficient of streamlined weirs, which are known as a state-of-the-art type of weir. As an alternative to the CFD procedure to predict discharge coefficient of this nature-inspired type of weir, the potential superiority of ML-DL algorithms is investigated. Five classical ML techniques, namely LR, RF, SVM, KNN and DT, are applied. In addition, among the DL algorithms, LSTM, CNN and GRU, and their hybrid forms (i.e. LSTM-GRU, CNN-LSTM and CNN-GRU) are compared by eight different error metrics.

To enhance the accuracy, a three-layer hierarchical DL algorithm consisting of a convolutional layer coupled with two subsequent GRU levels, which is also hybridized by the linear regression method (i.e. LR-CGRU), is proposed. In general, hybrid deep data-driven algorithms provide more accurate results than the classical ones. Furthermore, it is clearly demonstrated that the LR-CGRU technique outperforms 11 other ML-DL algorithms.

Finally, the superiority of the proposed data-driven technique is demonstrated by a comparative analysis between previously introduced algebraic expressions to predict the discharge coefficient. The results indicate that the LR-CGRU algorithm can act as an alternative tool to forecast the discharge coefficient of streamlined weirs accurately, which paves the way for data-driven modeling of streamlined weirs. Although the capabilities of 12 ML-DL algorithms are investigated to predict the discharge coefficient, there is still a need for future studies to enhance both the accuracy and the efficiency of the estimation. Furthermore, the application of the

proposed ML-DL algorithm in probabilistic risk assessment (Abyani et al., 2019; Amini et al., 2021a, 2021b; Kia et al., 2021; Zarrin et al., 2020) of streamlined weirs can be investigated in future works. Moreover, the proposed methodology could be used in other applications and scientific fields, including heat transfer, CFD, hydrofoil design and thermal imaging (e.g. Glowacz, 2021a, 2021b; Glowacz et al., 2021; Bahman et al., 2020; Bahman & Kabiri-Samani, 2021; Kabiri-Samani, 2018), by predicting the essential output variables.

Disclosure statement

No potential conflict of interest was reported by the authors.

ORCID

Kwok Wing Chau http://orcid.org/0000-0001-6457-161X
Amir Mosavi http://orcid.org/0000-0003-4842-0613

References

Abasi, A., Harsij, V., & Soraghi, A. (2021). Damage detection of 3D structures using nearest neighbor search method. *Earthquake Engineering and Engineering Vibration*, *20*(3), 705–725. https://doi.org/10.1007/s11803-021-2048-1

Abdollahi, A., Kabiri-Samani, A., Asghari, K., Atoof, H., & Bagheri, S. (2017). Numerical modeling of flow field around the labyrinth side-weirs in the presence of guide vanes. *ISH Journal of Hydraulic Engineering*, *23*(1), 71–79. https://doi.org/10.1080/09715010.2016.1239555

Abyani, M., Bahaari, M. R., Zarrin, M., & Nasseri, M. (2019). Effects of sample size of ground motions on seismic fragility analysis of offshore jacket platforms using genetic algorithm. *Ocean Engineering*, *189*, 106326. https://doi.org/10.1016/j.oceaneng.2019.106326

Amini, A., Abdollahi, A., Hariri-Ardebili, M. A., & Lall, U. (2021a). Copula-based reliability and sensitivity analysis of aging dams: Adaptive Kriging and polynomial chaos Kriging methods. *Applied Soft Computing*, *107524*(9), 133–144. https://doi.org/10.1016/j.asoc.2021.107524

Amini, A., Kia, M., & Bayat, M. (2021b). Seismic vulnerability macrozonation map of SMRFs located in Tehran via reliability framework. *Structural Engineering and Mechanics*, *78*(3), 351–368. https://doi.org/10.12989/sem.2021.78.3.351

Arvanaghi, H., Naderi, V., Azimi, V., & Salmasi, F. (2014). Determination of discharge coefficient in inclined rectangular sharp-crested weirs using experimental and numerical simulation. *International Journal of Current Research*, *2*(3), 401–406.

Arvanaghi, H., & Oskuei, N. N. (2013). Sharp-crested weir discharge coefficient. *Journal of Civil Engineering and Urbanism*, *3*(3), 87–91.

Aswin, S., Geetha, P., & Vinayakumar, R. (2018). Deep learning models for the prediction of rainfall. *2018 International Conference on Communication and Signal Processing (ICCSP)*, 0657–0661. https://doi.org/10.1109/ICCSP.2018.8523829

Athira, V., Geetha, P., Vinayakumar, R., & Soman, K. P. (2018). Deepairnet: Applying recurrent networks for air quality prediction. *Procedia Computer Science*, *132*, 1394–1403. https://doi.org/10.1016/j.procs.2018.05.068

Aydin, M. C., & Kayisli, K. (2016). Prediction of discharge capacity over two-cycle labyrinth side weir using ANFIS. *Journal of Irrigation and Drainage Engineering*, *142*(5), 06016001. https://doi.org/10.1061/(ASCE)IR.1943-4774.0001006

Azamathulla, H. M., Haghiabi, A. H., & Parsaie, A. (2016). Prediction of side weir discharge coefficient by support vector machine technique. *Water Supply*, *16*(4), 1002–1016. https://doi.org/10.2166/ws.2016.014

Bagheri, S., & Kabiri-Samani, A. (2020a). Overflow characteristics of streamlined weirs based on model experimentation. *Flow Measurement and Instrumentation*, *71*(9), 101720–101730. https://doi.org/10.1016/j.flowmeasinst.2020.101720

Bagheri, S., & Kabiri-Samani, A. (2020b). Simulation of free surface flow over the streamlined weirs. *Flow Measurement and Instrumentation*, *71*, 101680. https://doi.org/10.1016/j.flowmeasinst.2019.101680

Bahman, E., & Kabiri-Samani, A. (2021). Experimental investigation of flow characteristics over asymmetric Joukowsky hydrofoil weirs for free and submerged flow. *Flow Measurement and Instrumentation*, *79*(9), 101938–101944. https://doi.org/10.1016/j.flowmeasinst.2021.101938.

Bahman, E., Kabiri-Samani, A., & Moghim, M. N. (2020). Discharge coefficient of hydrofoil weirs based on potential flow theory around a symmetric Joukowsky hydrofoil. *Journal of Hydraulic Research*, *58*(6), 899–909. https://doi.org/10.1080/00221686.2019.1671519

Barkhordari, M. S., & Es-haghi, M. S. (2021). Straightforward prediction for responses of the concrete shear wall buildings subject to ground motions using machine learning algorithms. *International Journal of Engineering*, *34*(7), 1586–1601.

Barkhordari, M. S., & Tehranizadeh, M. (2021). Response estimation of reinforced concrete shear walls using artificial neural network and simulated annealing algorithm. *Structures*, *34*, 1155–1168. https://doi.org/10.1016/j.istruc.2021.08.053

Bhattacharya, B., Price, R. K., & Solomatine, D. P. (2007). Machine learning approach to modeling sediment transport. *Journal of Hydraulic Engineering*, *133*(4), 440–450. https://doi.org/10.1061/(ASCE)0733-9429(2007)133:4(440)

Bilhan, O., Emiroglu, M. E., & Kisi, O. (2010). Application of two different neural network techniques to lateral outflow over rectangular side weirs located on a straight channel. *Advances in Engineering Software*, *41*(6), 831–837. https://doi.org/10.1016/j.advengsoft.2010.03.001

Borghei, S. M., Jalili, M. R., & Ghodsian, M. (1999). Discharge coefficient for sharp-crested side weir in subcritical flow. *Journal of Hydraulic Engineering*, *125*(10), 1051–1056. https://doi.org/10.1061/(ASCE)0733-9429(1999)125:10(1051)

Carollo, F. G., & Ferro, V. (2021). Comments on "overflow characteristics of streamlined weirs based on model experimentation" by Bagheri S. and Kabiri-Samani A. *Flow Measurement and Instrumentation*, *78*, 101908. https://doi.org/10.1016/j.flowmeasinst.2021.101908

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014).

Learning phrase representations using RNN encoder-decoder for statistical machine translation. ArXiv Preprint ArXiv:1406.1078.

Ebtehaj, I., Bonakdari, H., & Gharabaghi, B. (2018). Development of more accurate discharge coefficient prediction equations for rectangular side weirs using adaptive neuro-fuzzy inference system and generalized group method of data handling. *Measurement*, *116*, 473–482. https://doi.org/10.1016/j.measurement.2017.11.023

Emiroglu, M. E., Bilhan, O., & Kisi, O. (2011). Neural networks for estimation of discharge capacity of triangular labyrinth side-weir located on a straight channel. *Expert Systems with Applications*, *38*(1), 867–874. https://doi.org/10.1016/j.eswa.2010.07.058

Esteghamati, M. Z., & Flint, M. M. (2021). Developing data-driven surrogate models for holistic performance-based assessment of mid-rise RC frame buildings at early design. *Engineering Structures*, *245*, 112971. https://doi.org/10.1016/j.engstruct.2021.112971

Glowacz, A. (2021a). Thermographic fault diagnosis of ventilation in BLDC motors. *Sensors*, *21*(21), 7245. https://doi.org/10.3390/s21217245

Glowacz, A. (2021b). Ventilation diagnosis of angle grinder using thermal imaging. *Sensors*, *21*(8), 2853. https://doi.org/10.3390/s21082853

Glowacz, A., Tadeusiewicz, R., Legutko, S., Caesarendra, W., Irfan, M., Liu, H., Brumercik, F., Gutten, M., Sulowicz, M., Daviu, J. A. A., Sarkodie-Gyan, T., Fracz, P., Kumar, A., & Xiang, J. (2021). Fault diagnosis of angle grinders and electric impact drills using acoustic signals. *Applied Acoustics*, *179*, 108070. https://doi.org/10.1016/j.apacoust.2021.108070

Gruber, N., & Jockisch, A. (2020). Are GRU cells more specific and LSTM cells more sensitive in motive classification of text? *Frontiers in Artificial Intelligence*, *3*, 40. https://doi.org/10.3389/frai.2020.00040

Guo, N., Li, C., Gao, T., Liu, G., Li, Y., & Wang, D. (2021). A fusion method of local path planning for mobile robots based on LSTM neural network and reinforcement learning. *Mathematical Problems in Engineering*, *2021*(34), 276–290. https://doi.org/10.1155/2021/5524232

Haghiabi, A. H., Parsaie, A., & Ememgholizadeh, S. (2018). Prediction of discharge coefficient of triangular labyrinth weirs using adaptive neuro fuzzy inference system. *Alexandria Engineering Journal*, *57*(3), 1773–1782. https://doi.org/10.1016/j.aej.2017.05.005

Hariri-Ardebili, M. A., Mahdavi, G., Abdollahi, A., & Amini, A. (2021). An RF-PCE hybrid surrogate model for sensitivity analysis of dams. *Water*, *13*(3), 302. https://doi.org/10.3390/w13030302

Hariri-Ardebili, M. A., & Salazar, F. (2020). Engaging soft computing in material and modeling uncertainty quantification of dam engineering problems. *Soft Computing*, *24*(15), 11583–11604. https://doi.org/10.1007/s00500-019-04623-x

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Johnson, M. C. (2000). Discharge coefficient analysis for flat-topped and sharp-crested weirs. *Irrigation Science*, *19*(3), 133–137. https://doi.org/10.1007/s002719900009

Kabiri-Samani, A. R. (2018). Hydraulic characteristics of flow over the streamlined weirs. *Modares Civil Engineering Journal*, *17*(6), 29–42.

Karami, H., Karimi, S., Rahmanimanesh, M., & Farzin, S. (2017). Predicting discharge coefficient of triangular labyrinth weir using support vector regression, support vector regression-firefly, response surface methodology and principal component analysis. *Flow Measurement and Instrumentation*, *55*, 75–81. https://doi.org/10.1016/j.flowmeasinst.2016.11.010

Kia, M., Amini, A., Bayat, M., & Ziehl, P. (2021). Probabilistic seismic demand analysis of structures using reliability approaches. *Journal of Earthquake and Tsunami*, *15*(03), 2150011. https://doi.org/10.1142/S1793431121500111

LeCun, Y., & Bengio, Y. (1995). Convolutional networks for images, speech, and time series. *The Handbook of Brain Theory and Neural Networks*, *3361*(10).

Li, S., Yang, J., & Ansell, A. (2021). Discharge prediction for rectangular sharp-crested weirs by machine learning techniques. *Flow Measurement and Instrumentation*, *79*, 101931. https://doi.org/10.1016/j.flowmeasinst.2021.101931

Mahmoud, A., Yuan, X., Kheimi, M., & Yuan, Y. (2021). Interpolation accuracy of hybrid soft computing techniques in estimating discharge capacity of triangular labyrinth weir. *IEEE Access*, *9*, 6769–6785. https://doi.org/10.1109/ACCESS.2021.3049223

Mahtabi, G., & Arvanaghi, H. (2018). Experimental and numerical analysis of flow over a rectangular full-width sharp-crested weir. *Water Science and Engineering*, *11*(1), 75–80. https://doi.org/10.1016/j.wse.2018.03.004

Mohammed, A. Y., & Sharifi, A. (2020). Gene expression programming (GEP) to predict coefficient of discharge for oblique side weir. *Applied Water Science*, *10*(6), 1–9. https://doi.org/10.1007/s13201-020-01211-5

Norouzi, R., Daneshfaraz, R., & Ghaderi, A. (2019). Investigation of discharge coefficient of trapezoidal labyrinth weirs using artificial neural networks and support vector machines. *Applied Water Science*, *9*(7), 1–10. https://doi.org/10.1007/s13201-019-1026-5

Parsaie, A., & Haghiabi, A. (2015). The effect of predicting discharge coefficient by neural network on increasing the numerical modeling accuracy of flow over side weir. *Water Resources Management*, *29*(4), 973–985. https://doi.org/10.1007/s11269-014-0827-4

Parsaie, A., & Haghiabi, A. H. (2017). Support vector machine to predict the discharge coefficient of sharp crested w-planform weirs. *AUT Journal of Civil Engineering*, *1*(2), 195–204. https://doi.org/10.22060/CEEJ.2017.13005.5309

Pourkamali-Anaraki, F., Hariri-Ardebili, M. A., & Morawiec, L. (2020). Kernel ridge regression using importance sampling with application to seismic response prediction. *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 511–518. https://doi.org/10.1109/ICMLA51294.2020.00086

Qu, J., Ramamurthy, A. S., Tadayon, R., & Chen, Z. (2009). Numerical simulation of sharp-crested weir flows. *Canadian Journal of Civil Engineering*, *36*(9), 1530–1534. https://doi.org/10.1139/L09-067

Rady, R. (2011). 2D-3D modeling of flow over sharp-crested weirs. *Journal of Applied Sciences Research*, *7*(12), 2495–2505.

Rao, C. R., & Rao, C. R. (1973). *Linear statistical inference and its applications* (Vol. 2, pp. 263–270). Wiley.

Razavi-Far, R., Chakrabarti, S., Saif, M., & Zio, E. (2019). An integrated imputation-prediction scheme for prognostics of battery data with missing observations. *Expert Systems with Applications*, *115*, 709–723. https://doi.org/10.1016/j.eswa.2018.08.033

Roushangar, K., Akhgar, S., Salmasi, F., & Shiri, J. (2014). Modeling energy dissipation over stepped spillways using machine learning approaches. *Journal of Hydrology*, *508*, 254–265. https://doi.org/10.1016/j.jhydrol.2013.10.053

Roushangar, K., Khoshkanar, R., & Shiri, J. (2016). Predicting trapezoidal and rectangular side weirs discharge coefficient using machine learning methods. *ISH Journal of Hydraulic Engineering*, *22*(3), 254–261. https://doi.org/10.1080/09715010.2016.1177740

Salmasi, F., Yıldırım, G., Masoodi, A., & Parsamehr, P. (2013). Predicting discharge coefficient of compound broad-crested weir by using genetic programming (GP) and artificial neural network (ANN) techniques. *Arabian Journal of Geosciences*, *6*(7), 2709–2717. https://doi.org/10.1007/s12517-012-0540-7

Sammut, C., & Webb, G. I. (2011). *Encyclopedia of machine learning*. Springer Science & Business Media.

Selvin, S., Vinayakumar, R., Gopalakrishnan, E. A., Menon, V. K., & Soman, K. P. (2017). Stock price prediction using LSTM, RNN and CNN-sliding window model. *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 1643–1647. https://doi.org/10.1109/ICACCI.2017.8126078

Sezer, O. B., & Ozbayoglu, A. M. (2018). Algorithmic financial trading with deep convolutional neural networks: Time series to image conversion approach. *Applied Soft Computing*, *70*, 525–538. https://doi.org/10.1016/j.asoc.2018.04.024

Shafiei, S., Najarchi, M., & Shabanlou, S. (2020). A novel approach using CFD and neuro-fuzzy-firefly algorithm in predicting labyrinth weir discharge coefficient. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, *42*(1), 1–19. https://doi.org/10.1007/s40430-019-2109-9

Soraghi, A., & Huang, Q. (2021). Probabilistic prediction model for RC bond failure mode. *Engineering Structures*, *233*, 111944. https://doi.org/10.1016/j.engstruct.2021.111944

Torres-Rua, A. F., Ticlavilca, A. M., Walker, W. R., & McKee, M. (2012). Machine learning approaches for error correction of hydraulic simulation models for canal flow schemes. *Journal of Irrigation and Drainage Engineering*, *138*(11), 999–1010. https://doi.org/10.1061/(ASCE)IR.1943-4774.0000489

Tullis, B. P. (2011). Behavior of submerged ogee crest weir discharge coefficients. *Journal of Irrigation and Drainage Engineering*, *137*(10), 677–681. https://doi.org/10.1061/(ASCE)IR.1943-4774.0000330

Vinayakumar, R., Soman, K. P., & Poornachandran, P. (2017). Applying deep learning approaches for network traffic prediction. *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2353–2358. https://doi.org/10.1109/ICACCI.2017.8126198

Wang, J., Chen, Y., Hao, S., Peng, X., & Hu, L. (2019). Deep learning for sensor-based activity recognition: A survey. *Pattern Recognition Letters*, *119*, 3–11. https://doi.org/10.1016/j.patrec.2018.02.010

Zaji, A. H., & Bonakdari, H. (2014). Performance evaluation of two different neural network and particle swarm optimization methods for prediction of discharge capacity of modified triangular side weirs. *Flow Measurement and Instrumentation*, *40*, 149–156. https://doi.org/10.1016/j.flowmeasinst.2014.10.002

Zarrin, M., Abyani, M., & Asgarian, B. (2020). A statistical study on lognormal central tendency estimation in probabilistic seismic assessments. *Structure and Infrastructure Engineering*, *16*(5), 803–819. https://doi.org/10.1080/15732479.2019.1668813

Zhao, R., Yan, R., Chen, Z., Mao, K., Wang, P., & Gao, R. X. (2019). Deep learning and its applications to machine health monitoring. *Mechanical Systems and Signal Processing*, *115*, 213–237. https://doi.org/10.1016/j.ymssp.2018.05.050

Zounemat-Kermani, M., Kermani, S. G., Kiyaninejad, M., & Kisi, O. (2019). Evaluating the application of data-driven intelligent methods to estimate discharge over triangular arced labyrinth weir. *Flow Measurement and Instrumentation*, *68*, 101573. https://doi.org/10.1016/j.flowmeasinst.2019.101573

Appendix

Table A1. Computational cost of training time for all 12 machine learning–deep learning (ML-DL) algorithms.

LR	RF	SVM	KNN	DT	
0:00:00.003218	0:00:00.119285	0:00:00.000996	0:00:00.000630	0:00:00.000409	
LSTM	CNN	LSTM-GRU	CNN-LSTM	CNN-GRU	LR-CGRU
0:00:46.185944	0:00:04.668465	0:00:46.714708	0:00:29.236926	0:00:29.725064	0:00:29.728282

Note: LR = linear regression; RF = random forest; SVM = support vector machine; KNN = *k*-nearest neighbor; DT = decision tree; LSTM = long short-term memory; CNN = convolutional neural network; GRU = gated recurrent unit.