Archives of Computational Methods in Engineering (2023) 30:115–159
https://doi.org/10.1007/s11831-022-09793-w

REVIEW ARTICLE

Check for updates

Artificial Intelligence, Machine Learning, and Deep Learning in Structural Engineering: A Scientometrics Review of Trends and Best Practices

Arash Teymori Gharah Tapeh[1] **· M. Z. Naser**[1,2]

Received: 19 April 2021 / Accepted: 7 July 2022 / Published online: 24 July 2022
© The Author(s) under exclusive licence to International Center for Numerical Methods in Engineering (CIMNE) 2022

Abstract

Artificial Intelligence (AI), machine learning (ML), and deep learning (DL) are emerging techniques capable of delivering elegant and affordable solutions which can surpass those obtained through traditional methods. Despite the recent and rapid advancements in developing next-gen AI-based techniques, we continue to lack a systemic understanding of how AI, ML, and DL can fundamentally be integrated into the structural engineering domain. To advocate for a smooth and expedite the adoption of AI techniques into our *field*, we present a state-of-the-art review that is specifically tailored to structural engineers. This review aims to serve three purposes: (1) introduce the art and science of AI, ML, and DL in terms of its commonly used algorithms and techniques with particular attention to those of high value to this domain, (2) map the current knowledge within this domain through a scientometrics analysis of more than 4000 scholarly works with a focus on those published in the last decade to identify best practices in terms of procedures, performance metrics, and dataset size etc., and (3) review past and recent efforts that applied AI derivatives into the various subfields within structural engineering. Special attention is given to the application of AI, ML, and DL in earthquake, wind, and fire engineering, as well as structural health monitoring, damage detection, and prediction of properties of structural materials as collected from over 200 sources. Finally, a discussion on trends, recommendations, best practices, and advanced topics towards the end of this review.

Abbreviations

AI	Artificial intelligence
ALD	Applied load
ANFIS	Adoptive neuro-fuzzy interface
ANN	Artificial Neural Network
ARI	Arias intensity
ASI	Acceleration spectrum intensity
BA	Bagging technique
BD	Bracketed duration
BFGS	Broyden–Fletcher–Goldfarb–Shanno
BP-ANN	Back propagation-Artificial Neural Network
CFL	Ceiling finish layer
CGB	Powell–Beale conjugate gradient algorithm
CGF	Fletcher–Powell conjugate gradient back propagation
CGP	Polak–Ribiere conjugate gradient back propagation
CSA	Coupled simulated annealing
CVA	Cumulative absolute velocity
DT	Decision tree
EPA	Effective peak acceleration
FFNN	Feed forward neural network
FMCDM	Fuzzy multi-criteria decision analysis
GA	Grid search/genetic algorithm
GANs	Generative adversarial networks
GBRT	Gradient boosting regression tree
GDA	Gradient descent with adaptive linear back propagation gradient
GDM	Gradient descent BP with momentum
GDX	Gradient descent w/momentum and adaptive linear back propagation
GEP	Gene expression programming
GMDH	Group method of data handling

✉ M. Z. Naser
mznaser@clemson.edu
https://www.mznaser.com

Arash Teymori Gharah Tapeh
teymouriarash89@gmail.com; ateymor@g.clemson.edu

1 Glenn Department of Civil Engineering, Clemson University, Clemson, SC, USA

2 AI Research Institute for Science and Engineering (AIRISE), Clemson University, Clemson, SC, USA

GP	Genetic programming (linear-based GP, Cartesian GP, grammatical GP, stack GP)
GSA	Grid search algorithm
HI	Housner intensity
HSSB	High strength steel bolt
IBS	Interfacial bond strength
JTY	Joist type
KNN	K-nearest neighbor
LGP	Linear genetic programming
LM	Levenberg–Marquart (back propagation)
LOOCV	Leave one out cross-validation
LSTM	The long short-term memory
LWLS-SVMR	Locally weighted least squares support vector machines for regression
MCDM	Multi-criteria decision analysis
MCFT	Modified compression field theory
MGGP	Multigene genetic programming
ML	Machine learning
MLS-SVMR	Multi-output least-squares support vector machine for regression
MOE	Module of elasticity
MOR	Module of rupture
OSS	One step secant back propagation
PCA	Principal component analysis
PGA	Peak ground acceleration
PGD	Peak ground displacement
PGV	Peak ground velocity
PP	Predominant period
PRSC	Perfobon rib shear connector
PSO	Particle swarm optimization
RC	Reinforced concrete
RF	Random forest
RP	Resilient back propagation
SCG	Scaled conjugate gradient back propagation
SD	Significant duration
SED	Specific energy density
SVM	Support vector machine
TCC	Thermal conductivity of concrete
TGP	Tree-based Genetic Programming
UD	Uniform Duration

1 Introduction

The rapid rise in computational knowledge and capacity has opened exciting opportunities that can be leveraged to realized new methods for analysis—especially those of data-driven nature to make up for the limitations of mechanics-based approaches [1]. One such opportunity is that provided by Artificial Intelligence (AI); and, by extension, machine learning (ML) and deep learning (DL) [2]. AI-based techniques have been proven successful in parallel fields (such as robotics [3], manufacturing [4], medicine [5], etc.), yet remain to be underutilized by structural engineers [6]. Despite ongoing efforts aimed at adopting AI, ML, and DL into our domain, these efforts are often faced with inertia. Understandably, this inertia is habitually tied to the notion of AI-based methods providing solutions in a blackbox-manner which structural engineers are not familiar with—as opposed to their fluency in transparent methods (e.g., experimental, numerical, and analytical methods).

A deep dive into the above notion showcases that most structural engineers have been part of an experimental, numerical, or analytical program, whether while pursuing their education or during their tenure in practice/field. In a way, structural engineers were exclusively "primed" and then "accustomed" to employing classical methods [7]. Still, in reality, courses on advanced numerical simulations (i.e., finite element (FE) method) or high-order nonlinear mechanics briefly touch upon complex three-dimensional (3D) phenomena. While this may well indeed be a contradiction to the above since most problems tackled by structural engineers are inherently complex and do require means of advanced computations, the same also brings in a pragmatic perspective. In this perspective, it is the "practice" and "continual education" components that enable an engineer from getting familiar with the innings of advanced numerical modeling (or simply FE software [8]). Similarly, the authors believe that with continued practice, a structural engineer can also be adapted to AI, ML, or DL.

From a logistical point of view, structural engineering projects often amount to repetitive checks and steps which have been seamlessly, yet historically intentionally, developed systematically (e.g., building code provisions) by structural engineering authorities (i.e., regulatory committees and building codes). Naturally, structural engineering provisions can be automated and fitted into software for simplicity [9]. Analyzing project-sized models involves high-capacity workstations, access to proper software and license, as well as the availability of expert personnel. Given the complexity of modern projects, a considerable volume of computation and time is needed to pursue proper structural design. This makes realizing and evaluating such designs through traditional methods challenging, and in some instances, impractical (due to cost or time constraints) [10]. For example, structural health monitoring of a typical structure (say a bridge) is associated with a series of sensor networks that continually measure and record valuable information on the status of such a bridge. Analyzing such data in real-time, or near real-time, mechanically or by means of legacy software may not only be infeasible but may hinder fully utilizing the potential of the deployed sensing infrastructure [11, 12].

Hence, it is of merit to this domain to examine modern methods that may bypass some of the aforenoted complications. From this view, AI-based methods have proven

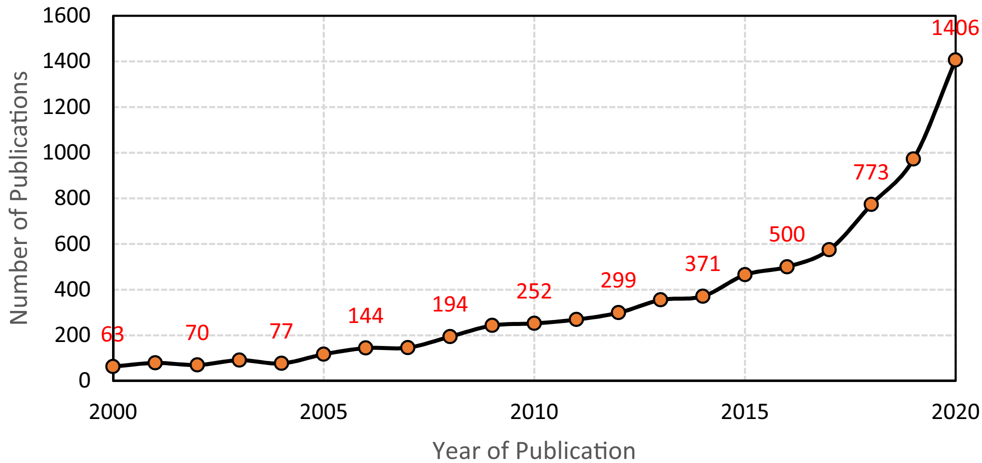


Fig. 1 Publications adopting AI derivatives in structural engineering (2000–2020) [arrived at by searching "artificial intelligence" and "structural engineering" using the Dimensions scholarly database]

effective in creating affordable, scalable, and unique approaches that allow engineers to design, monitor, and assess structures while at the same time overcome many of the above-noted challenges [13, 14]. To set the stage and to have a better understanding of the concepts of AI, ML, and DL, as well as associated technologies to be described during this review, we will first outline key terminologies and concepts herein.

In recent years, AI and ML are often used interchangeably [15]. However, one should be cognizant that ML is considered as a subset of the more pronounced domain of AI [16]. A key distinction is often made between AI and ML. All in, AI is a branch of computer science that attempts to solve complex problems through imitation of biological processes such as cognition and logic [17, 18]. In this pursuit, AI creates programming systems that are capable of performing tasks that often require some degree of cognition (i.e., human intelligence). On the other hand, ML trains methods (or simply algorithms) to perform tasks via automatically recognizing patterns within data as opposed to explicitly programming such algorithms to carry out the aforenoted tasks [17, 19]. A third terminology also exists and is commonly referred to as DL. DL is a special form of ML and capitalizes on training neural networks with deep and fluid architectures (i.e., contains a series of processing layers as will be described in a later section [20]).

An examination of publication trends within this domain during the last two decades shows a continued rise in the number of articles related to AI derivatives (including ML and DL)—see Fig. 1. In fact, the number of publications that utilized AI techniques almost doubled during the last two years. In general, these works revolve around four themes [21–23]:

- Extraction of models based on data retrieval.
- Prediction of structural behavior.
- Derivation of mathematical representations of physical phenomena.
- Examination of visuals from images and videos.

In spirit of this review, it is worth noting that some of the earliest research articles that mention the use of AI in civil or structural engineering date back to the late 1980s and early 1990s [24, 25]. It is also worth noting that one of the first reviews on the use of AI in civil engineering applications was conducted by Adeli [26] over 20 years ago. Since then, a few notable reviews were also undertaken. For example, Salehi and Burgeno [17] recently reviewed AI methods focusing on pattern recognition and classification algorithms. Zhang et al. [27] focused their review on articles related to genetic algorithms in civil engineering. Mirrashidi and Naderpour [28] reviewed the use of AI in concrete structures to explore the behavior of concrete beams, columns, joints, and slabs. Penadés-Plà et al. [29] reviewed the use of AI and ML in decision-making methods applied to bridge design. Aldwaik and Adeli [30] thoroughly examined optimization-based algorithms into 2D and 3D high-rise structures from a cost–benefit perspective.

Noting how: (1) the above notable reviews focused in some form or shape on a specific area or specific AI technique within the civil engineering domain, and (2) the substantial improvements in AI-based methods during the past few years, we dedicate our review to accumulate and summarize recent studies (2010–2021) that successfully explored the use of AI, ML, and DL in structural engineering problems. We hope this review will bridge some of the burning questions often raised by structural engineers, thereby accelerating the adoption of the above emerging technologies into our domain.

This review is structured as follows. A gentle introduction to AI, ML, and DL is introduced in Sect. 2. Section 3 presents commonly used AI-based algorithms of high merit to structural engineers with details. Section 4 outlines the

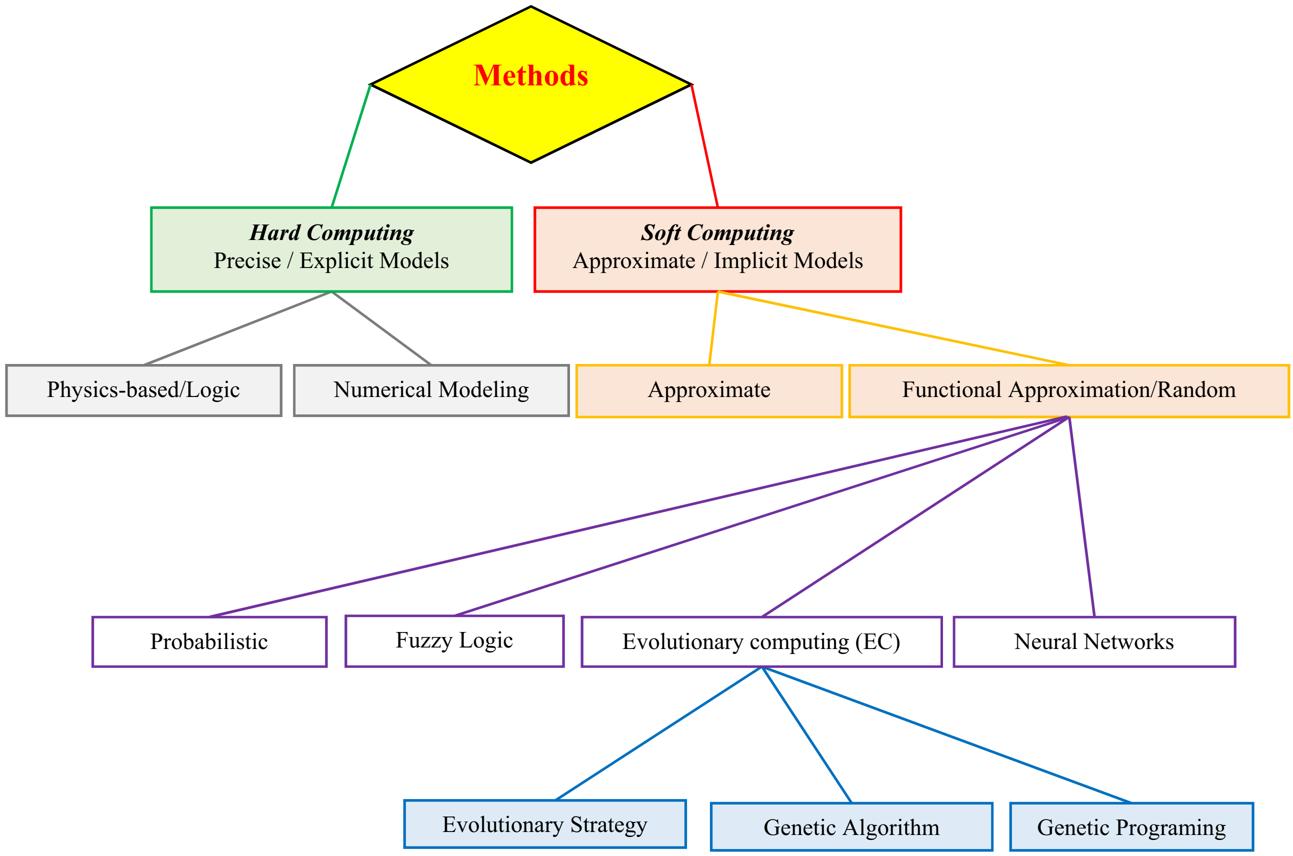


Fig. 2 Computational methodologies

procedures used for collecting and sorting the reviewed literature supplemented with scientometrics statistics and knowledge maps. Section 5 reviews works conducted over the last decade with regard to AI, ML, and DL and their applications in structural engineering, and Sect. 6 collectively analyzes the outcome of such review. Finally, our conclusions are provided in Sect. 7.

2 A Gentle Introduction to General Concepts within AI, ML, and DL

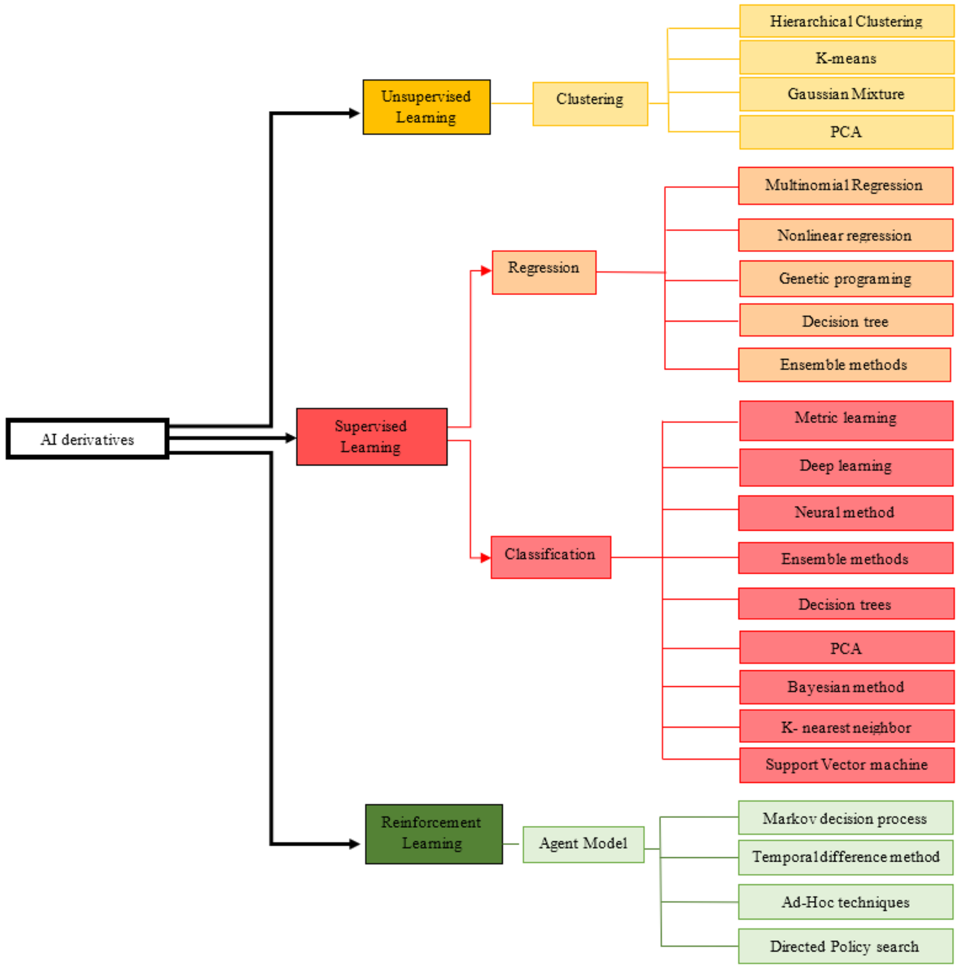
2.1 Big Ideas

Collectively, the nature of computational methods that can be applied to solve a problem can be generally divided into two groups, often referred to as *Soft Computing* and *Hard Computing* (see Fig. 2). Hard computing comprises methods that integrate a high degree of certainty to solve problems [31, 32]. On the contrary, soft computing covers approximate methods that can be used to solve problems via implicit/useable yet not exact solutions [33]. As one can see, there is a common ground between the previously noted definition of AI derivatives with that of soft computing methods.

In parallel, associated terminology that is often applied to describe features of the computational methodologies also exists. In this terminology, methods of computation can be further classified into three separate groups (*whitebox*, *greybox*, and *blackbox*) [10]. Whitebox methods are those that clearly articulate the functional relationship(s) between the variables governing a phenomenon to the outcome (or target/response) an engineer is trying to evaluate. Such methods resemble those of a classical sense (i.e., Hooke's laws) and by extension equation-based models. Greybox methods refer to models wherein the relationship and/or mathematical model the ties variables to observations is partially built on theoretical understanding or prior knowledge while also making use of data driven-like approaches [34]. Finally, blackbox models have complex inner workings that are hard to interpret yet can still convey the outcome of a phenomenon with high accuracy (i.e., a neural network that can correctly predict the axial behavior of a given structural member despite exactly knowing as to how, or why such a network is capable of attaining high predictivity [35]). At this point in time, the majority of AI derivatives may fall under blackbox models [36].

In general, AI derivatives fall under one of three learning methods; *supervised learning*, *unsupervised learning,*

Fig. 3 Learning methods



and *reinforcement learning* (see Fig. 3) [37, 38]. Supervised learning is adopted when both the outcome (target/response) and governing variables of a phenomenon are known (i.e., details of a structural member and its corresponding sectional capacity) [39]. This type of learning can be further grouped under regression (when the target is a quantity) or classification (when the target is a label/class). Unsupervised learning is applied in scenarios where the data is not labeled, and an engineer seeks to learn the inherent structure of such data (e.g., analyze if a signal received from an onsite sensor implies cracking of a structural member, or not). Reinforcement learning refers to algorithms capable of adjusting actions in response to arising conditions (say from an environmental factor). This type of learning is not as commonly used yet in structural engineering as the previous two. Figure 3 outlines the aforenoted three methods of learning, along with some of their corresponding algorithms. The reader is invited to review Sec. 3.0 for a complete discussion on such algorithms.

2.2 Problem Formulation and Data Handling

AI-based methods operate through a model development procedure. In this procedure, a research question or a hypothesis is formed first. Such a question can be, "can we develop an AI model that can predict the deformation history of a steel beam under seismic loading?" Or, "can we develop a DL model that can identify different failure modes by examining imagery of failed reinforced concrete (RC) structures?". In all cases, the structural engineer is to collect observations pertaining to the phenomenon and question on hand. Such observations can be in terms of numeric/tabulated data, or footage, etc. [15]. Given the nature of this introductory section, a discussion covering the use of tabulated data is presented herein.

Tabulated data is organized into a dataset matrix. This matrix can be referred to as the observed data matrix in which the number of rows (r) represents the samples collected (from experiments or simulations), and the number of columns (c) is equal to the characteristics or features of the measured samples. As a result, the dimensions of the matrix can be considered equal to ($r \times c$). If supervised learning is used, then an additional column related to the target (or dependent response) is also available; otherwise, this column is removed (see Eq. 1).

In k-fold cross validation (and its variants), the cleansed dataset is randomly split into two main sets; a testing set and a training set. The training set is further divided into k number of sub-sets. The model is then validated on one of the

$$\text{Response Variable} \begin{cases} Categorial\ for\ Classification \\ or \\ Numerical\ for\ Regression \end{cases}$$
$$\text{Matrix} = \begin{bmatrix} 1 & .. & target \\ 2 & .. & .. \\ r & .. & .. \end{bmatrix} \tag{1}$$

sub-sets and trained using the remaining $k-1$ sub-sets, and this process is repeated k times until each unique group has been used as the validation sub-set [47]. On the other hand, in ratio sampling, the dataset is shuffled and then randomly split into a training sub-set and a testing sub-set (which can also be further split into a validation and a testing sub-sets). Common ratios used in this method vary between 50-80% for training, with the remaining samples used for validation/testing [48–50]. The overarching goal of the above methods is to prevent overfitting, the notion of "lucky guess," as to increase the accuracy and our confidence of the model.

When the training and validation process is completed, the model's performance is further evaluated through performance metrics. Such metrics are mathematical or logical constructs that examine how the model's predictions converge or diverge from real observations. Some of the commonly used metrics in regression and classification structural engineering applications are listed herein, and a more exhaustive review can be found elsewhere [51, 52]. In case the performance of the model is adequate, then the training procedure terminates, and the model is deployed. If not, then the model is to be further fine-tuned until a satisfactory performance is achieved. Such finetuning can involve adopting different training strategies, require additional data/observations, or tuning model hyperparameters [53].

Once the needed observations are collected, these observations are processed through feature selection and feature handling techniques. Engineers may handle data through two techniques: feature selection and feature extraction. In feature selection, features of observations are selected through a procedural analysis that comprises of three methods, *filter* (by filtering essential features via ranking systems [i.e., correlation analysis]), *wrapper* (by using feedback from monitoring the performance of the AI derivative model [i.e., in terms of the obtained accuracy when examined by systematically adding or removing features], or *embedded methods* [e.g., algorithms with intrinsic capabilities to select features such as LASSO] [40].

In lieu of the above, a phenomenon can be governed by a large number of features with several dimensions. In this event, it could be of merit to reduce the space of such features to eliminate redundant features or those with minimal influence - thereby accelerating the AI analysis [41]. This can be taken care of via adopting feature extraction techniques that rely on applying feature reduction (or dimensionality) reduction methods such as principal component analysis (PCA) [42].

2.3 Model Development

Now that the dataset is ready for an AI analysis, an AI technique (or algorithm) is to be selected. At this stage, it is up to the designer to select a technique, or perhaps a combination of techniques. It is common for an engineer to prefer one technique that s/he is familiar with (in a similar manner to preference when selecting a FE software) [43]. Our review also indicates that in some works, researchers tend to utilize a series of techniques in an individual manner [44], or group (ensemble) manner [45], or competitive manner (where algorithms compete to attain high-performance metrics) [46]. In all cases, the selected AI derivative is to be trained and then validated. There are several ways to "train" an AI model. Two of the most widely used methods are referred to as *k-fold cross validation* and *ratio sampling*.

$$Mean\,Absolute\,Error\,(MAE) = \frac{\sum_{i=1}^{n} |E_i|}{n} \tag{2}$$

Measures the difference between two continuous variables

$$Root\,Mean\,Squared\,Error\,(RMSE) = \sqrt{\frac{\sum_{i=1}^{n} E_i^{\,2}}{n}} \tag{3}$$

Measures the square root of the average of squared errors

$$Coefficient\,of\,Determination\,(R^2) = 1 - \sum_{i=1}^{n} (P_i - A_i)^2 / \sum_{i=1}^{n} (A_i - A_{mean})^2 \tag{4}$$

Measures the goodness of fit of a mode

$$Area\ under\ the\ Receiver\ Operating\ Characteristic\ (ROC)\ curve\ (AUC) = \sum_{i=1}^{N-1} \frac{1}{2}\left(FP_{i+1} - FP_i\right)\left(TP_{i+1} - TP_i\right) \tag{5}$$

Measures the two-dimensional area underneath the entire ROC curve

$$Log\ Loss\ Error\ (LLE) = -\sum_{c=1}^{M} A_i logP \tag{6}$$

Measures the where the prediction input is a probability value

$$Sensitivity = \frac{TP}{TP + FN} \tag{7}$$

Measures the proportion of actual positives that are correctly identified as positives.

$$Specificity = \frac{TN}{TN + FP} \tag{8}$$

Measures the proportion of actual negatives that are correctly identified negatives.

$$Precision = \frac{TP}{TP + FP} \tag{9}$$

The proportions of positive observations that are true positives.

$$Negative\ Predictive\ Value\ (NPV) = \frac{TN}{TN + FN} \tag{10}$$

The proportions of negative observations that are true positives.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

Evaluates the ratio of the number of correct predictions to the total number of samples. where, E: Error = Actual (A) − predicted (P), n: number of observations, TP (denotes true positives), TN (denotes true negatives), FP (denotes false positives), and FN (denotes false negatives), M: number of classes, c: class label, y: binary indicator (0 or 1) if c is the correct classification for a given observation.

3 Overview to AI, ML, and DL Algorithms

This section presents a general discussion on commonly used AI techniques/algorithms with high merit to structural engineers.

3.1 Principal Component Analysis (PCA)

As mentioned earlier, PCA is a dimension reduction algorithm [54]. This technique adopts a linear static method to convert multidimensional inputs to a more leaner feature space data that still contains most of the information in the original dataset. PCA identifies patterns in a dataset and then distill the observations down to their most important features so that the dataset is simplified without losing valuable attributes. PCA creates new variables by transforming the original observations to a new set of variables (dimensions) using eigenvectors and eigenvalues calculated from a covariance matrix of the original variables [55]. PCA algorithm can be readily found at online repositories such as [56]. Similarly, PCA can be obtained considering a $[x_{ij}]$ dataset such that $(i = 1, 2, \ldots m)$ and$(j = 1, 2 \ldots k)$, where m and k are equal to observation dimension and number of observations, respectively. First, we calculate the mean ($\widetilde{x_j}$) and the standard deviation (s_j) for data (j_{th}) column which is equal to:

$$\widetilde{x_j} = \frac{1}{m}\sum_{i=1}^{m} x_{ij} \tag{12}$$

$$s_j = \sqrt{\frac{\sum_{i=1}^{m}\left(x_{ij} - \overline{x}j\right)^2}{m}} \tag{13}$$

Then, we obtain $[\widetilde{x}]$ from the $[x]$ transformation matrix. The normalized elements $\widetilde{x_{ij}}$ is obtained as follows.

$$\widetilde{x_{ij}} = \frac{x_{ij} - \overline{x}j}{s_j} \tag{14}$$

Then the covariance matrix $[c]$ can be calculated as $[c] = \left[\widetilde{x}\right]^T\left[\widetilde{x}\right]/m - 1$ and finally, the stated principal components will be obtained in the form of $[c]\,\{p_i\} = \lambda_i\,\{p_i\}$, where λ_i is eigenvalue i^{th} and $\{p_i\}$ is its related vector.

3.2 Support Vector Machine (SVM)

The Support Vector Machine (SVM) algorithm can predict regression and classification data [57]. In SVM, operators aim to identify a line or a boundary (referred to as a hyperplane or decision boundary) in an n-dimensional space (where n represents the number of features) that can be used to classify data into separate classes. Such a plane needs to maximize the distance between data points of each class to allow for confident classification [58]. The "support vectors" refer to data points in close proximity to the hyperplane, and hence these can significantly influence the position and orientation of the plane (see Fig. 4). SVM uses a special form of mathematical functions defined as kernels (k). A kernel

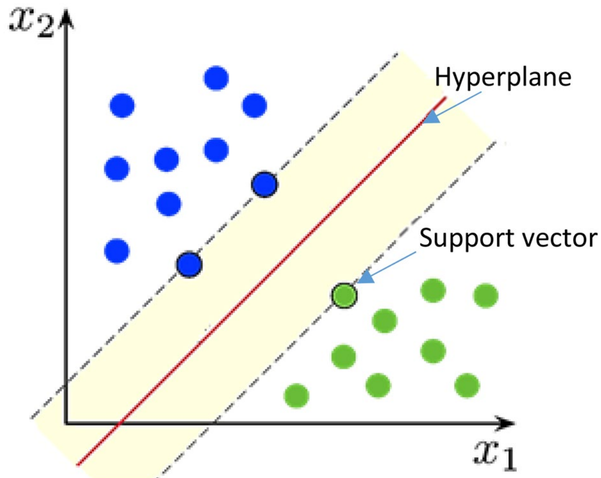


Fig. 4 Illustration of SVM

function transforms inputs into the required form. Typical kernel (k) functions for classification are as follows:

$$\text{Linear} = k\left(x_i.y_i\right) = {x_i}^T x_j \tag{15}$$

$$\text{Polynomial}: k\left(x_i.y_i\right) = \left(\gamma {x_i}^T x_j + r\right)^d . \gamma > 0 \tag{16}$$

$$\text{Radial basis function (RBF)}: k\left(x_i.y_i\right) = exp(-\gamma {x_i}^T x_j \| {x_i}^T x_j \|^d) . \gamma > 0 \tag{17}$$

$$\text{Sigmoid}: k\left(x_i.y_i\right) = tanh(\gamma {x_i}^T x_j + r) \tag{18}$$

While SVM was initially designed for classification, this technique has been since revised to accommodate regression as well [59]. When used for regression, the SVM algorithm employs an intense loss function to maintain the maximum margin. The linear model of the intense loss function is as follows:

$$L^{\in}(x.y.f) = |y - f(x)|_{\in} = \left\{ \begin{array}{cc} 0 & if\,|y - f(x)| < \in \\ |y - f(x)| - \in & otherwise \end{array} \right. \tag{19}$$

To fit a model of the form:

$$f(x) = \sum_{i=1}^{n} c_i k(x.x_i) \tag{20}$$

where c_i refers to as a choice of coefficient and $k(x.x_i)$ is the Guassian kernel function.

Similar to the classification version, the regression SVM also requires the use of kernel functions, such as:

$$\text{Linear}: k\left(x.x_i\right) = x_i x \tag{21}$$

$$\text{Polynomial kernel function}: k\left(x.x_i\right) = \left(x_i(x+1)^d\right) \tag{22}$$

$$\text{Radial Basis Function}: k\left(x.x_i\right) = exp\left[-\frac{(x_i - x)(x_i - x)}{2\sigma^2}\right] \tag{23}$$

$$\text{Sigmoid Kernel Function}: k\left(x.x_i\right) = \tanh\left(x_i(x+1)\right) \tag{24}$$

wherex_i,x are the training and test patterns, respectively, and (σ, d) are global basic function and vector dimension, respectively. SVMs codes for classification and regression can be readily found at [60].

3.3 Decision Tree (DT)

The decision tree (DT) algorithm can be used for regression or classification problems [61]. DT resembles the structure of a tree and can also accommodate various types of inputs, including those of nominal, alphabetical, and numerical nature. A variant of DT is the *CART* algorithm (abbreviated for classification and regression tree). DT is a simple decision-making algorithm whose main feature is to minimize the amount of Gini impurity (g, a measure of how often a randomly chosen data point would be incorrectly labeled if it was randomly labeled per the distribution of the subset). For instance, the value of Gini impurity for a node, T, equals [38]:

$$g(t) = \sum_{j \neq i} p(j|t)p(i|t) \tag{25}$$

where, i and j are target category, $p(j|t) = \frac{p(j.t)}{p(t)}$, $p(jt) = \frac{\pi(j).N_j(t)}{N_j}$,$(t) = \sum_j p(j.t)$, $P(j)$= prior probability for category$j$, $N_j(t)$= number of records in category of j of node$t$, and N_j = number of records of category j in the root node.

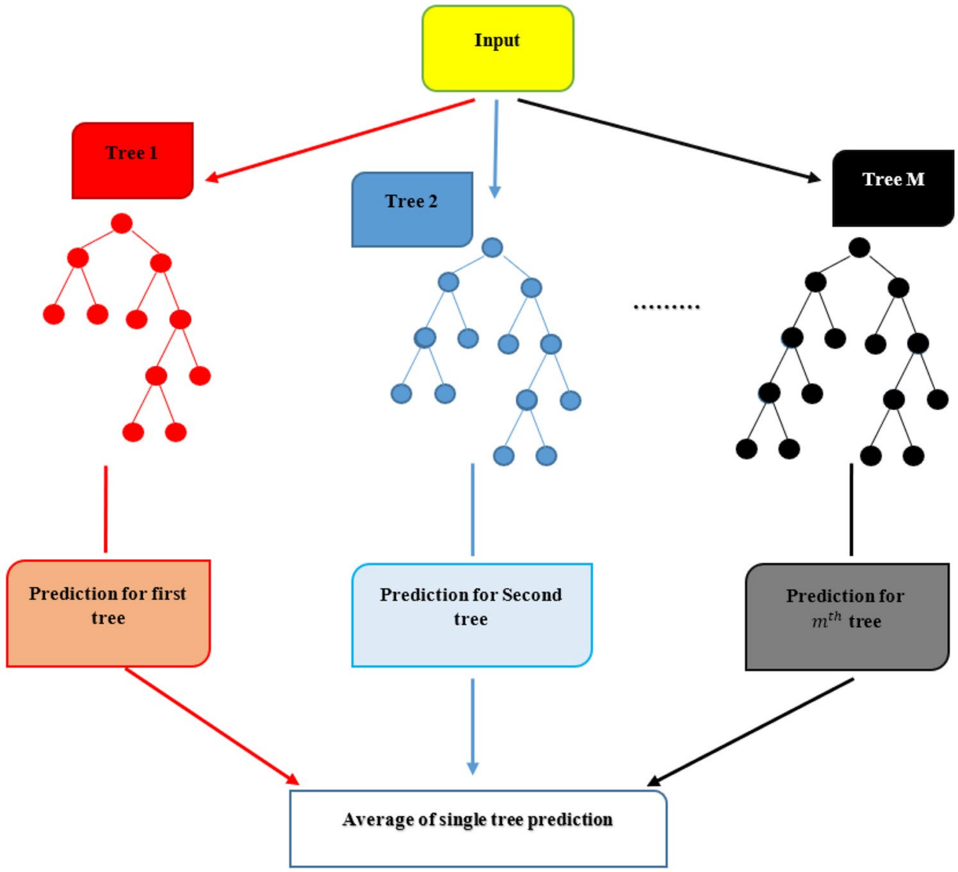
3.4 Random Forest (RF)

The random forest (RF) is an ensemble algorithm that utilizes weaker algorithms (i.e., DT) by repetition over a number of times and together as a single group to realize improved performance. In RF, the algorithm tries to predict the data from the base tree (for regression problems) or tries to predict the data using the most votes from the base tree [62] (see Fig. 5). For brevity, this ensemble can be found online at [63] and calculated using:

$$f(x) = \sum_{m=1}^{M} \frac{1}{m} f_m(x) \tag{26}$$

where, f_m is considered as a m^{th} tree.

Fig. 5 Typical layout of RF algorithm



3.5 Extreme Gradient Boosted Trees (ExGBT)

The ExGBT algorithm re-samples the collected data points into a tree-like format, where each tree sees a bootstrap sample of the database in each iteration [64]. ExGBT fits each successive tree to previous residual errors obtained from previous trees, thereby focusing on the observations that are most difficult to predict to improve prediction accuracy, as shown below. ExGBT can be found at [65, 66].

$$Y = \sum_{k=1}^{M} f_k(x_i), f_k \in F = \{f_x = w_{q(x)}, q : R^p \to T, w \in R^T\} \tag{27}$$

where M is additive functions, T is the number of leaves in the tree, w is a leaf weights vector, w_i is a score on i-th leaf, and $q(x)$ represents the structure of each tree that maps an observation to the corresponding leaf index [67].

3.6 K-Nearest Neighbor (KNN)

The K-nearest neighbor (KNN) algorithm utilizes a distance metric, d, to find K data points near the case data. For this purpose, KNN can use two metrics to determine the length between data points, namely, Euclidean distance and Manhattan distance. For example, if we consider two points,$i = (y_{i1}, y_{i2}, \dots y_{in})$ and $j = (y_{j1}, y_{j2}, \dots y_{jn})$ with n-numeric attributes, respectively, their distance quals to:

$$\text{Euclide and}\,(i.j) = \sqrt{(y_{1i} - y_{1j})^2 + (y_{2i} - y_{2j})^2 + \cdots + (y_{ni} - y_{nj})^2} \tag{28}$$

$$\text{Manhatt and}\,(i.j) = |y_{1i} - y_{1j}| + |y_{2i} - y_{2j}| + \cdots + |y_{ni} - y_{nj}| \tag{29}$$

If satisfied, then these equations are subject to the following four conditions: 1) $d(i.j) \gg 0$, 2) $d(i.j)$=0, 3) $d(i.j)$=$d(j.i)$ and 4) $d(i.j) \ll d(i.k) + d(k.j)$.

Also, for the positive number, K, and the observed data, x, the number of data close to x is equal to the conditional probability for x in class K as estimated by [68], and can be found online at [69]:

$$P_k(X) = P_r(Y = k.X = x) = \frac{1}{k} \sum_{i \in N_k} I(y_i = k) \tag{30}$$

3.7 Genetic Algorithm (GA)

Genetic Algorithm (GA) is a population-based metaheuristic algorithm that can be used to solve optimization problems

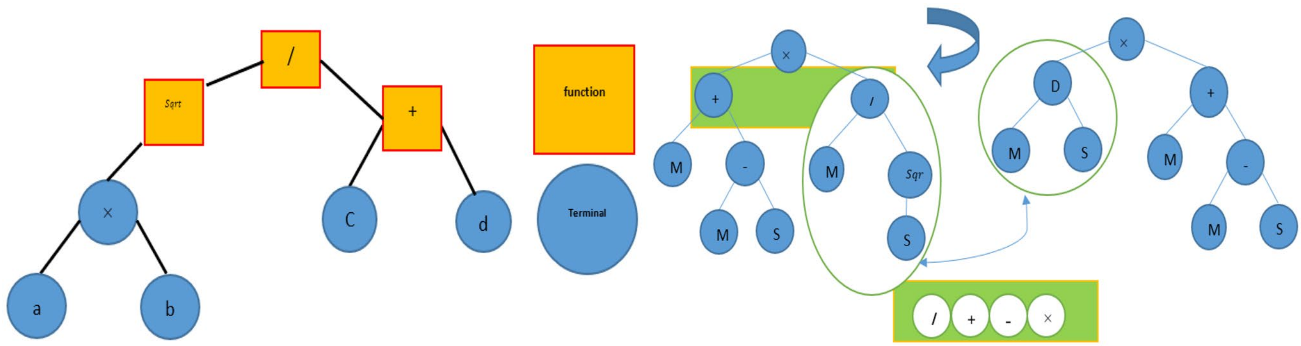


Fig. 6 Schematic of GA (left figure demonstrates expression of $\frac{\sqrt{a\times b}}{c+d}$, right figure demonstrates mutation process)

[70]. GA imitates the Darwinian evolutionary theory and the principle of survival of the fittest, where individuals in the population begin to reproduce and undergo genetic mutations in their structure to pass it into future generations [71]. In this analysis, a randomly selected population is formed. This population comprises features and mathematical symbols (i.e., exp, log, $\times$, $+$, etc.) to form terminals and functions [72]. A GA model has a tree-like structure whose leaves are made up of numbers and variables and whose branches contain functions (see Fig. 6). As long as the mathematical construct is not obtained, the GA continues to process. Once the best model has been selected, evolutionary operations (i.e., mutation and cross-over) take place to enhance the created model and attain satisfactory fitness.

3.8 Genetic Programming (GP)

Genetic programming (GP) is a modern variant of GA. GP can be further grouped under *linear genetic programming (LGP)* and *genetic expression programming (GEP)* [73, 74]. Unlike GA, which is made of strings, GP, on the other hand, creates computer programs in the form of a tree-like structure [27]. A more thorough discussion on GA and GP types can be found elsewhere [75, 76].

3.9 Artificial Neural Networks (ANN)

Artificial neural networks (ANN) imitate the cognitive capability of the human brain to solve complex problems. First introduced in the 1940s [77], ANNs can broadly be grouped under two categories; *feedforward neural networks (FFNN)* and *recurrent neural networks (RNN)*. Typical ANNs are shallow and consist of three main layers; an input layer, a hidden layer, and an output layer. The first layer is visible and receives the input data, and then this data is weighted and transferred to the next hidden layer by a series of connections called "synaptic weights." The nodes, which resemble human neurons, reside in this layer and process that input data and then transfer the processed data to the output layer [22]. This process is called feed-forward and continues until the selected performance metric is satisfied [78]. Other approaches to feeding an ANN also exist, such as multilayer perceptron network, carpenter network, Hopfield network, and back-propagation. The aforenoted architecture can be extended to a deep neural network with several hidden layers (see Fig. 7). The steps associated with a typical ANN analysis include [79] and a readily developed ANN can be found at [80]:

- Receiving input variables $(x_1, x_2, x_3 \ldots, x_n)$
- Summation of input data and assign them weight $(h_i = \sum_{i=1}^{m} w_{ji}x_i + b_j)$
- Applying an activation function such as:

  o linear function: $f_n = a \bullet n + b$
  p Hyperbolic function: $f_n = \tanh(n)$
  q Logarithmic function: $f_n = \frac{1}{1+e} - n$

- Performing error propagation so that it meets the predefined metric error(s).

3.10 Convolutional Neural Networks (CNN)

The CNN algorithm is an extended ANN and is commonly referred to as DL, and has been heavily deployed to examine images and footage. The most crucial part of CNN is the configuration of the hidden layers that form the essential computational parts of this algorithm. Operations on the data are performed using different layers: convolutional layer, max-pooling layer, full connected layer, and soft-max layer [81]. The reader may refer to [82] and [83] for examples. Typical architecture from CNN can be seen in Fig. 8.

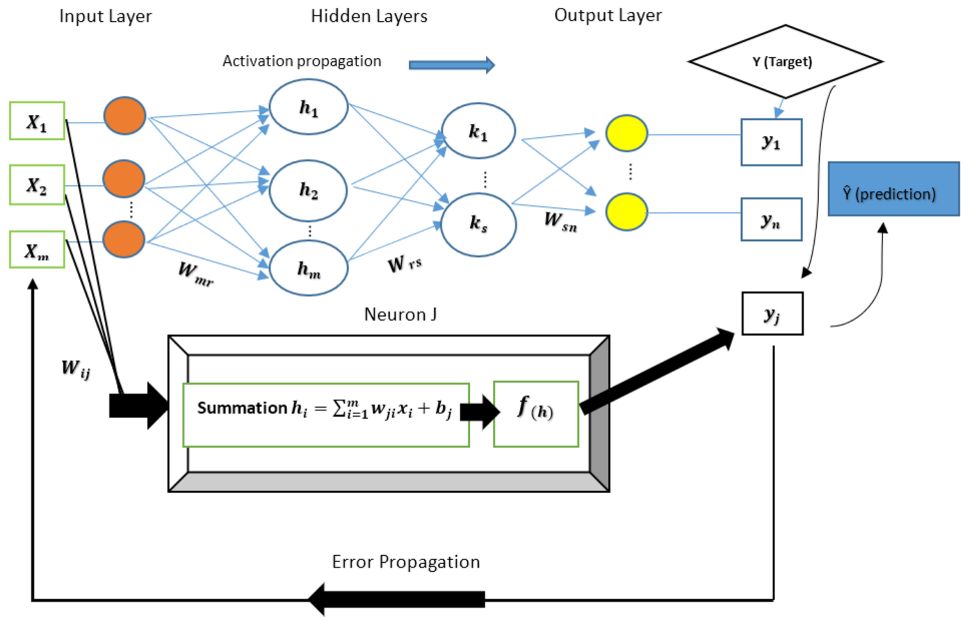


Fig. 7 Typical layout of an ANN

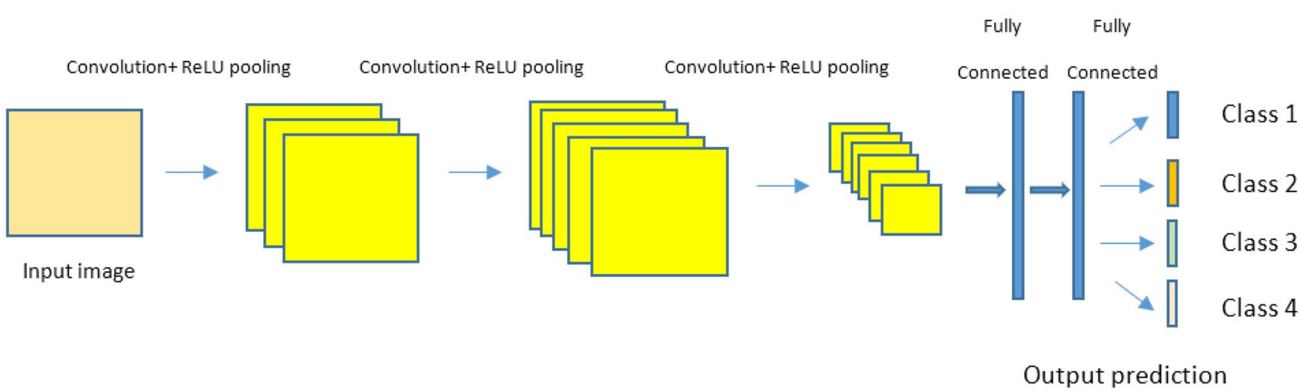


Fig. 8 Layout of CNN

4 Research Methodology

This section outlines the scientometrics analysis methodology followed to map the knowledge domain of structural engineering to classify works that adopted AI, ML, and DL.

4.1 Scientometrics Analysis

Our scientometrics analysis adopted scholarly databases, wherein information was collected and analyzed for items such as *keywords*, *year of publication*, *institutions*, and *authors*. This mapping was carried out through the VOSviewer software [84] to help visualize the collected observations. We followed a three-stage approach (i.e., research and classification, analysis, and *discussion and review*) as shown in Fig. 9 as inspired by the work of Cioffi et al. [85].

Each of the aforenoted stages is further described herein. The reader is to note that a timeframe between 2011 and 2020 was maintained throughout this analysis.

4.1.1 First Stage: Research and Classification

The first stage initiates this scientometrics analysis and comprises three steps, namely, *identification*, *screening for exclusion*, and *inclusion*. The identification step starts by exploring the *Dimensions* database [86, 87] which is a partly free scholarly database launched by Digital Science in January 2018 and is considered one of the world's largest linked research information dataset covering over 117 million publications. The Dimensions database was thoroughly examined by Thelwall [87] and shows similar results to that obtained by traditional databases such as Scopus and the Web of Science (with a range of 92–97%), and hence is

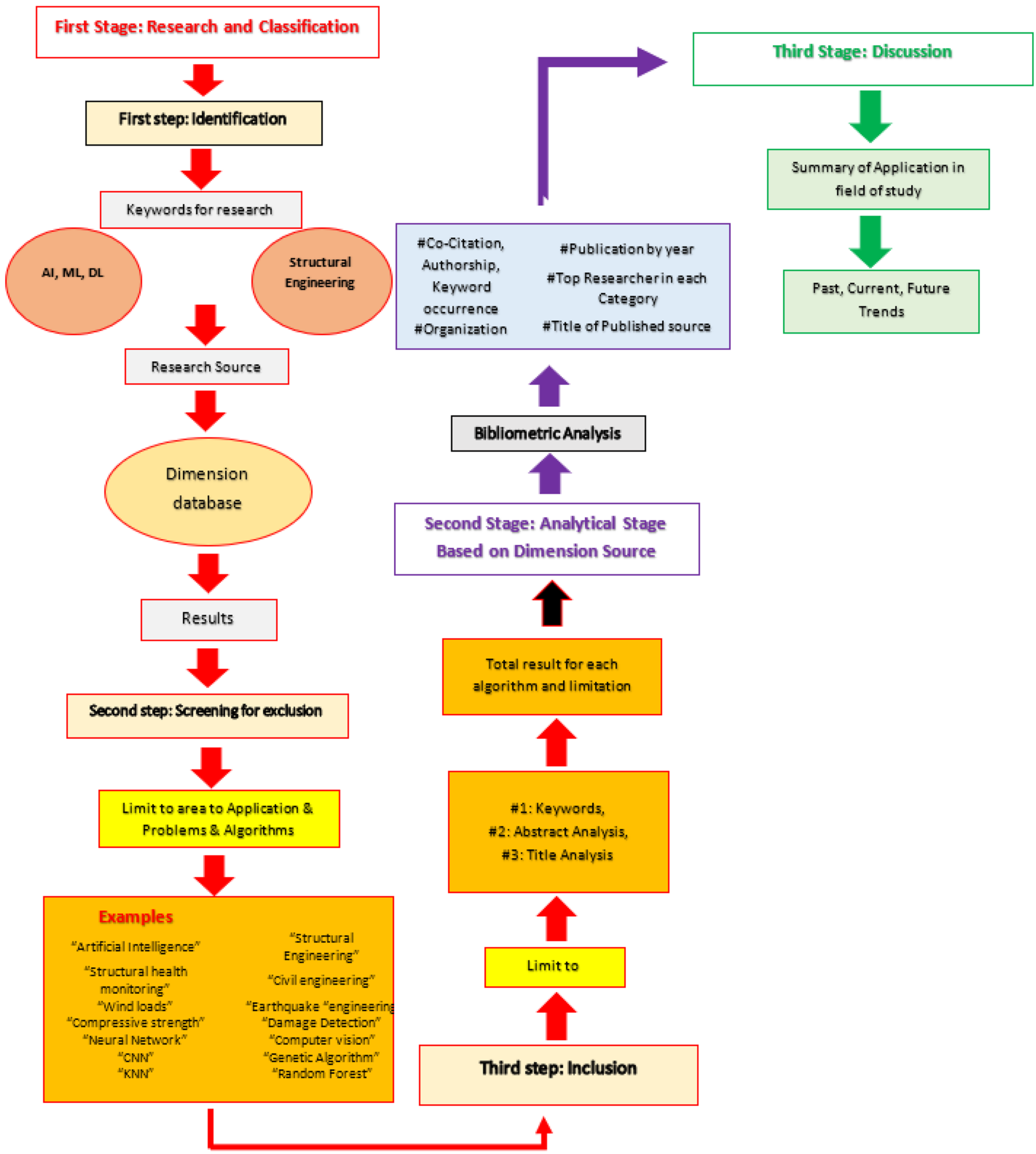


Fig. 9 Example of the adopted methodology adopted in this review for a search using Dimensions scholarly databases

selected herein. This database was investigated for possible *"keywords"*[1] related to this review. Some of the keywords used herein include "artificial intelligence and structural engineering", "machine learning and structural design", "deep learning and earthquake engineering," etc. The result of this step returns documents in the form of research articles, edited book chapters, conference proceedings, monographs, and preprints. Overall, 6048 documents were found

[1] This survey primarily favored search via *"keywords"* and confined this search to the las decades – future efforts can apply other filters such as search by *"document title"*, *"document abstract"* etc. or for a different time span.

Fig. 10 Number of articles published per year between 2011 and 2020 [SE: Structural Engineering]

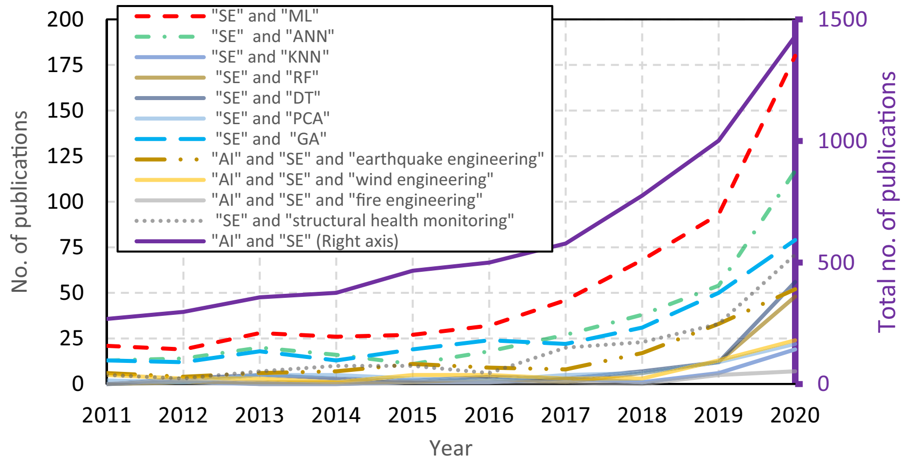
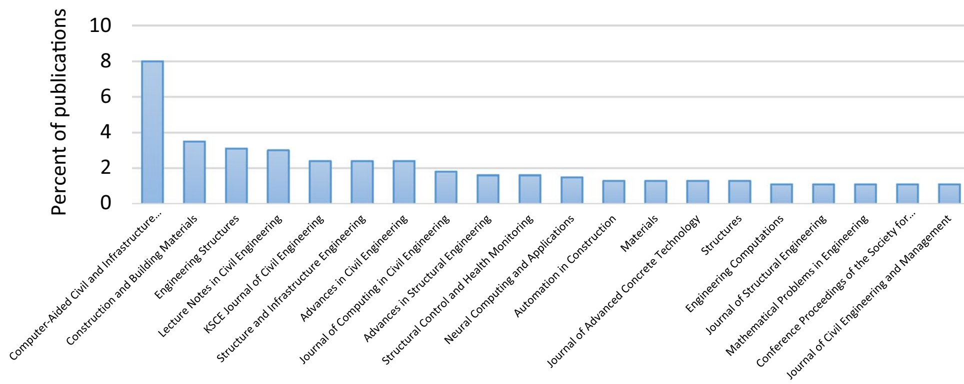


Fig. 11 Frequently publishing journals



to mention AI methods in the context of structural engineering. The collected documents were then screened for exclusion as part of the second step. In this step, unrelated documents such as those belonging to other domains were excluded from further examination. Finally, the remaining documents were individually examined through additional filtering to group those of common nature (i.e., algorithm-based documents, sub-field [i.e., seismic engineering, wind engineering, etc.] related document).

4.1.2 Second Stage: Analysis

In this stage of scientometrics analysis, the grouped documents are further analyzed using a number of works within each group, frequently publishing journals, and then a series of visualization maps were developed.

- Number of articles by year: Figure 10 shows the number of articles per year between 2011 and 2020 for AI, ML, and DL-related structural engineering works. As one can see, the most used algorithm is ANN with 115 documents, and the lowest was for KNN with 18 documents. Most of the works were applied into structural applications within seismic, wind, and fire engineering, respectively.
- Frequently publishing Journals: This review identifies twenty journals with the most published articles on AI, ML, and DL in structural engineering within the aforenoted timeframe. The top-ranking journal with the highest number of published articles is the Computer-Aided Civil and Infrastructure Engineering journal, with about 8% of all published works. This journal was followed by Construction and Building Materials and Engineering Structures, as can be seen in Fig. 11.
- Mapping the knowledge: The collected bibliometric information was also augmented using VOS-viewer software. Such maps outline the relationship between existing publications based on the relationship between journals, institutions, and types of problems. For a start, applied keywords noted algorithms and appropriate problems in which algorithms are most used are searched. Then we analyze the collected data using their title and abstract. A typical map showcasing highly publishing journals and institutions that adopted ML in structural engineering is presented in Fig. 12.

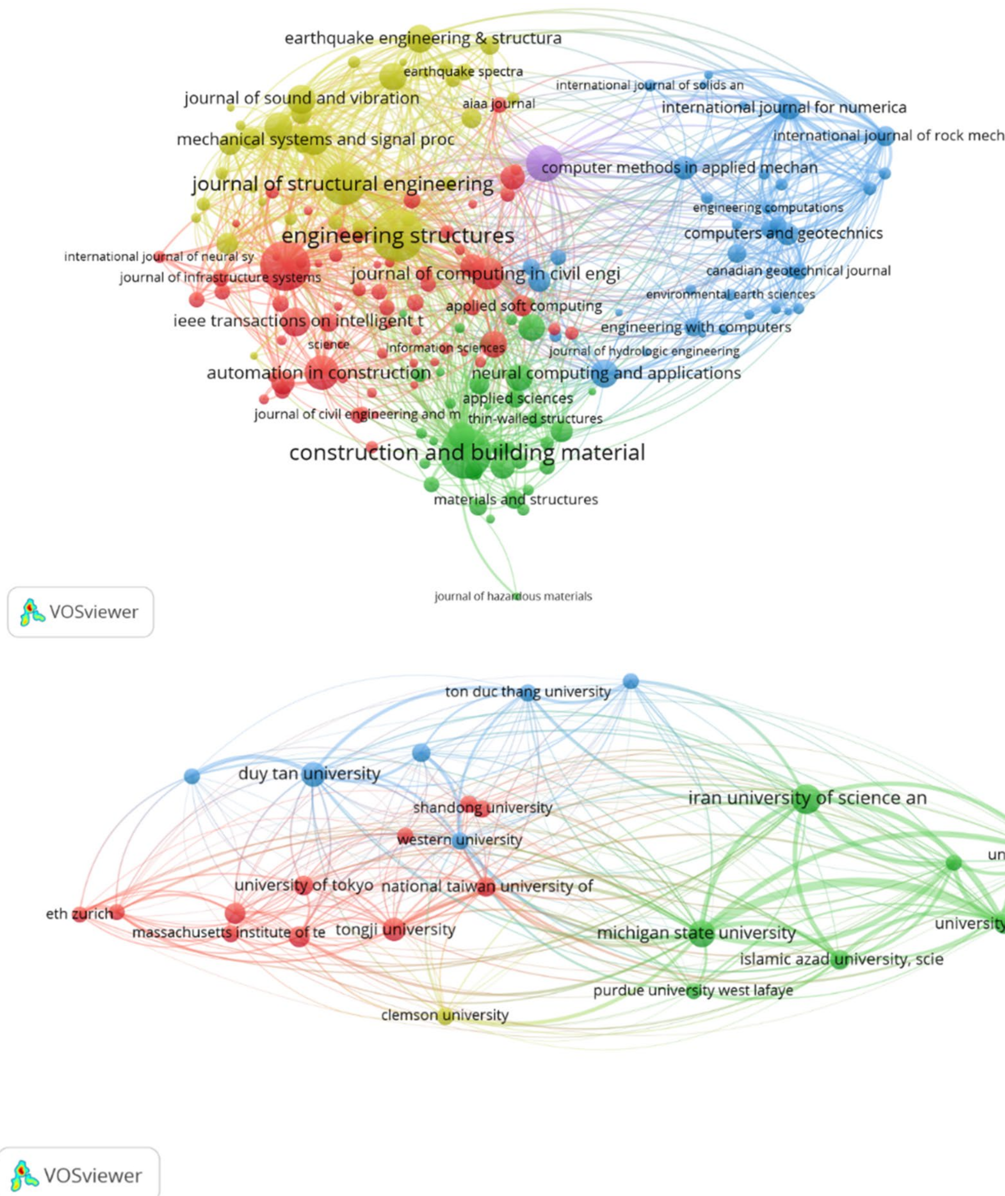


Fig. 12 Map of highly publishing journals (top) and institutions (bottom) with publications on AI, ML, and DL in structural engineering

4.1.3 Third Stage: Discussion and Review

Our scientometrics analysis identifies three main structural engineering problems that were addressed using AI derivatives. These problems are *modeling*, *simulation*, and *optimization*. Modeling problems were found to have the highest quantity in which structural engineers applied some form of AI to model the response of construction materials or structures, and most notably with regard to predicting mechanical properties or behavior of structural members. In simulation problems, our analysis shows that the large majority of studies that fall under this group is that where engineers aim to simulate damage or in structural health monitoring of structures (with the lion share belonging to seismic effects). Finally, the optimization problems tackled in this domain were primarily applied to optimize structural layout or concrete material mixtures.

In addition to the above, it is a good practice to examine the evolution with this research area to analyze its current, past, and future directions. Most importantly, topic variation is used to monitor research methods, resulting from which widely used methods and problem-solving related to them can be identified and used. In this effort, Fig. 13 maps *keyword* re-occurrences in the reviewed works. As one can see, keyword re-occurrences can be divided into four different clusters, where each of the nodes (or bubbles) represents

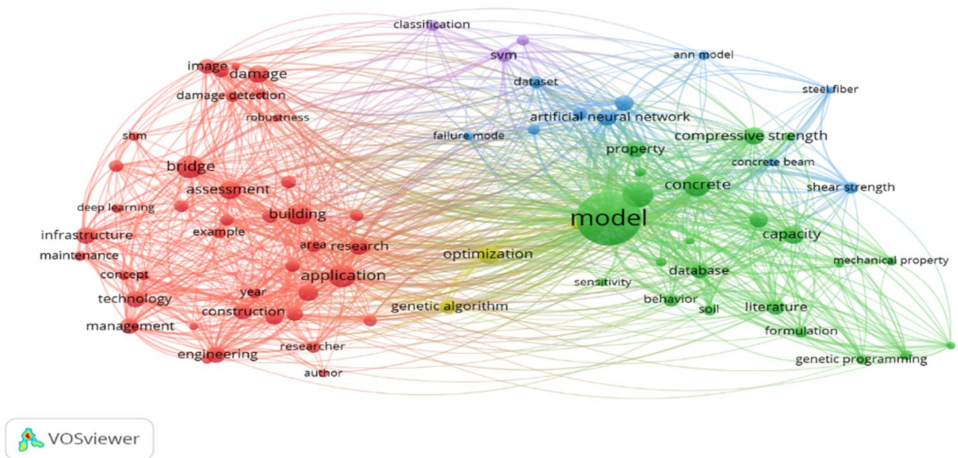


Fig. 13 Keyword frequency occurrences of works examined herein

a keyword, each of the colors represents a specific cluster, and the size of these nodes indicates the occurrence frequency of the examined keyword (i.e., the larger the size of these nodes, the larger the value). Additionally, the links that attach these nodes indicate the connection between them, and the smaller the distance between them, the greater the relationship is.

The scientometrics analysis of the above clusters shows that "*Model*" has the most repetitions with 1002 occurrences representing *modeling* problems in the green cluster. The closest node in this cluster to the "*Model*" node is the "*Prediction*" problem domain with 231 occurrences, which indicates that most problems are related to prediction. This was followed by "*Concrete*" with 183 occurrences and "*Properties*", both of which were in close proximity to the blue cluster (which is dominated by "ANN" as the most frequently used algorithm herein). In comparison, the largest node in the yellow cluster is "*Optimization,*" which has 120 occurrences, followed closely by genetic algorithms. Finally, the red cluster comprises the majority of the left nodes with keywords such as "*Building*", "*Bridge*", "*Construction*", and "*Infrastructure*"; all of which implies how the red cluster is tied to all of the other clusters (e.g., "*Models*" were used to explore solutions for "*Infrastructure*" via different algorithms.

In addition, a dedicated bibliographical coupling analysis of the reviewed works is also conducted with regards to the country of origin and primary institutions of research.

- Bibliographic coupling analysis based on Countries of origin: Figure 14 shows the extend of countries where AI-based methods in structural engineering applications have been produced. As can be seen, the four countries, namely, United States, China, Iran, and United Kingdom, dominate this figure.
- Bibliographic couple analysis based on institutions; Overall, works were gathered from 446 organizations. Figure 15 illustrates visual relationships between key institutions in works that adopted AI-based methods into structural engineering applications. As one can see, new hubs are identified as induvial institutions such as Clemson University.

4.1.4 Third Stage: Discussion

In the third and final stage of this scientometrics analysis, all complied works are further reviewed to explore specific details with regard to the used AI derivative and techniques, as well as structural engineering problems. This discussion is presented in Sec. 5.0 for structural materials, applications in earthquake, wind, and fire engineering, together with structural health monitoring, damage detection, structural connections, and various structures/structural elements.

5 A Review of Recent Structural Engineering Literature with a Focus on AI, ML, and DL

This section articulates recent works that adopted AI, ML, and DL into structural engineering problems.

5.1 Structural Materials

The branch of structural materials is seen to receive much attention from the reviewed works herein. A prime example that was identified by this review is that related to utilizing AI-based methods to explore properties of concrete materials (i.e., compressive strength of concrete variants such as high-performance concrete (HPC), self-consolidating concrete (SCC), fiber-reinforced concrete (FRC) to name a few). Of all reviewed works in this branch, 64% covered compressive strength property, followed by shear strength

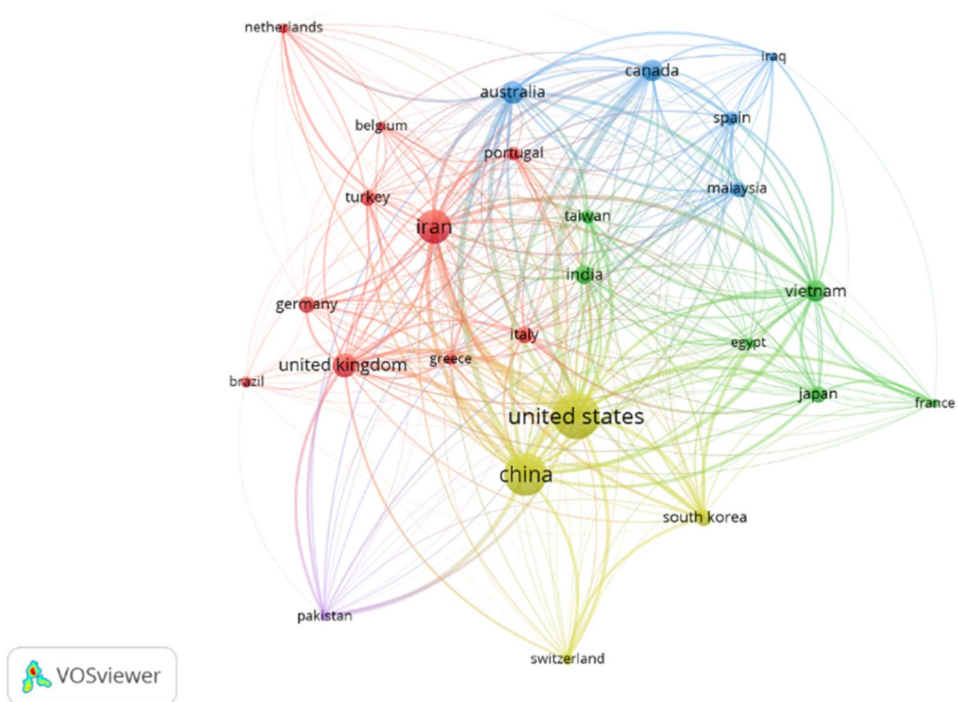


Fig. 14 Publications as per the country of origin

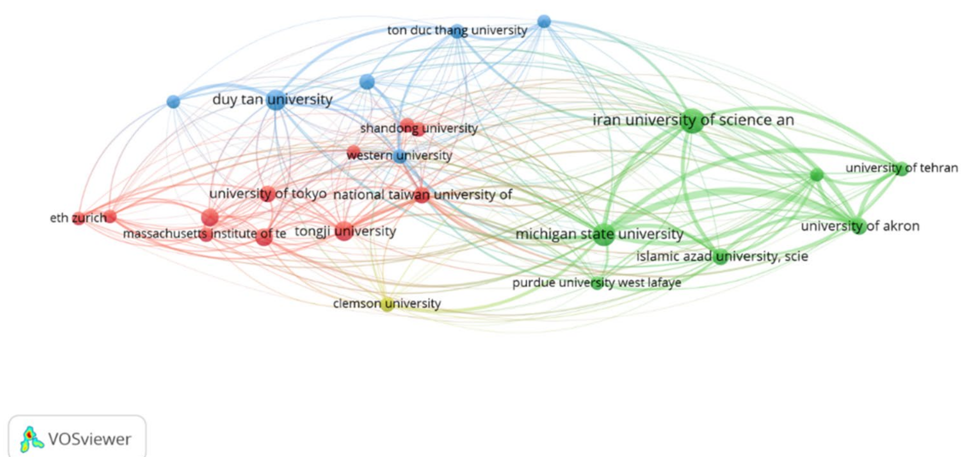


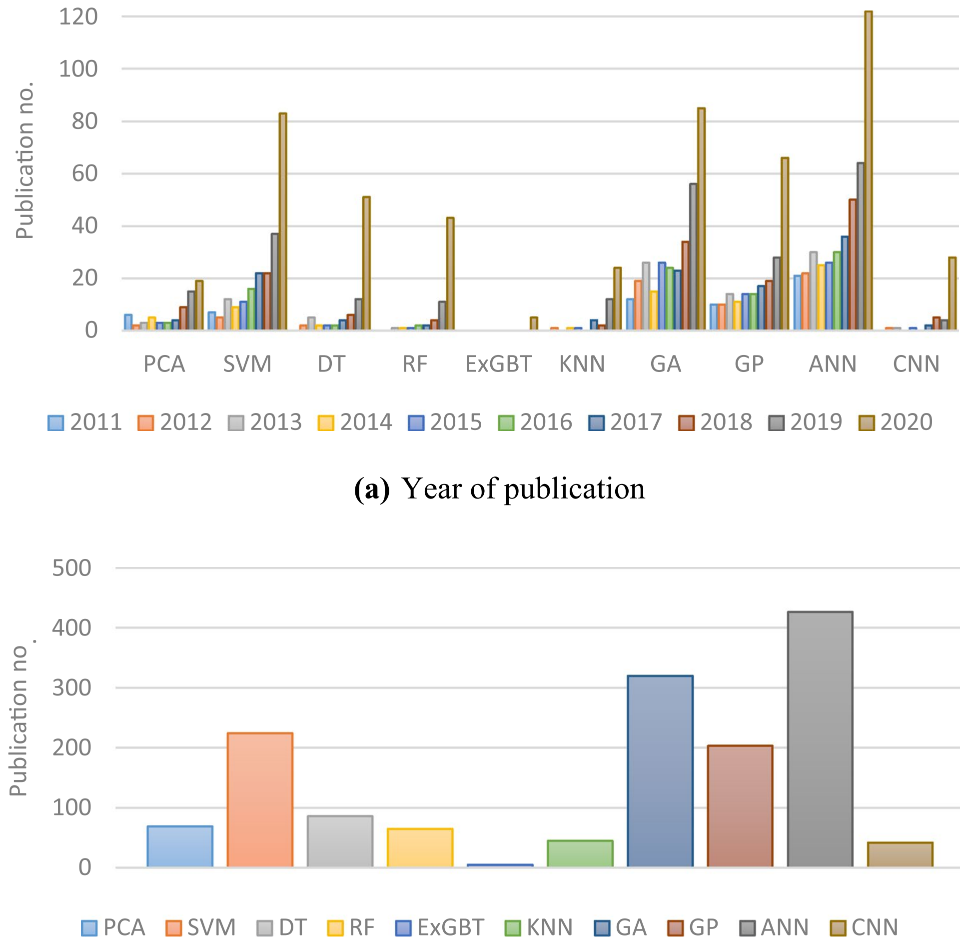
Fig. 15 Publications as per institution of origin

(18%), elastic modulus (9%), and shear modulus (9%), and only a comparatively small portion covered other structural materials such as metals and composites [88, 89]. In parallel, the most frequently used algorithms as ranked by appearance are ANN-based, GA-Based, and SVM-based (see Fig. 16).

Some of the most notable works are described herein, and the remainder is summarized in Table 1. Marani and Nehdi [90] applied a series of tree-like algorithms to predict the compressive strength of concretes with phase change materials (PCM) and concluded that ML approaches could attain a successful accuracy ranging between 93 and 97%. These researchers also noted that the gradient boosting tree (GBT) algorithm achieved the highest performance. Given a sample size of about 154 points used in their work, the same researchers recommended extending their dataset to better identify the features that affect PCM concretes with more confidence. In another work, Javed et al. [91] applied GEP to 65 data points collected from 21 studies to explore the compressive strength property of bagasse ash-based concrete (BABC). In this study, these researchers noted that cement content is the most critical parameter in concrete strength.

In a recent work, Nguyen et al. [92] developed two forms of deep neural networks with high-order neurons (i.e., conventional artificial neural network (C-ANN) and second-order artificial neural network (SO-ANN)) for the prediction of foamed concrete strength. Nguyen et al. [92] examined 177 concrete mixtures and reported that density, followed by the water-to-cement and sand-to-cement ratios, were the

Fig. 16 Details on AI-based methods often used in structural materials (ANN, GA, and SVM rank highest)



(a) Year of publication

(b) Algorithm frequency

three most important features to correctly predicting the compressive strength of foamed concrete. Jalal et al. [93] applied nonlinear multi-variable regression (NMVR), adaptive neuro-fuzzy inference system (ANFIS), ANN, GP, and SVM, to 72 data points to examine the properties of rubber concrete composite containing silica fume (SF) and zeolite (ZE). Sultana et al. [94] analyzed the compressive strength of jute fiber reinforced concrete composition using different algorithms and noted the SVM algorithm's superiority in predicting compressive strength over ANN. Castelli et al. [95] compared the geometric semantic genetic opera (GSGO) algorithm to GA and noted high prediction capability in evaluating the strength of high-performance concrete. Yaseen et al. [96] compared extreme learning machine (ELM) and support vector regression (SVR) to predict the compressive strength of lightweight foamed concrete. In total, these researchers tested 91 data points and showed the higher predictive capability of the ELM algorithm.

Within the concrete property realm, Ben Seghier et al. [97] examined hybrid ANNs (such as multilayer perceptron (MLP) and the radial basis function neural network) and GEP to evaluate the bond strength of corroded steel reinforcement using 218 data points. These researchers reported accuracy in terms of 96%. Gorphade et al. [98] combined GA and ANN to predict the workability and strength of high-performance concrete by examining 324 data points. Naseri et al. [99] applied a series of algorithms (e.g., water cycle algorithm (WSA), soccer league competition (SLA) algorithm, GA, ANN, and SVM) to design sustainable concrete mixtures with compressive strength, embodied CO2 emission, and energy and resource consumptions as objective functions. They also perform a sensitivity analysis and found the most influential parameters in their models to be compressive strength, coarse aggregate, water, and fine aggregates. Huang et al. [100] examined 269 data points via support vector regression (SVR) and firefly (FF) algorithm to also optimize concrete mixtures. Other works applied various ML algorithms to study the influence of concrete mixtures or properties [101–105], and other structural materials used in construction [106–110]. Additional works are also summarized in Table 1.

Table 1 Summary of works on AI-based methods tackling structural materials and properties

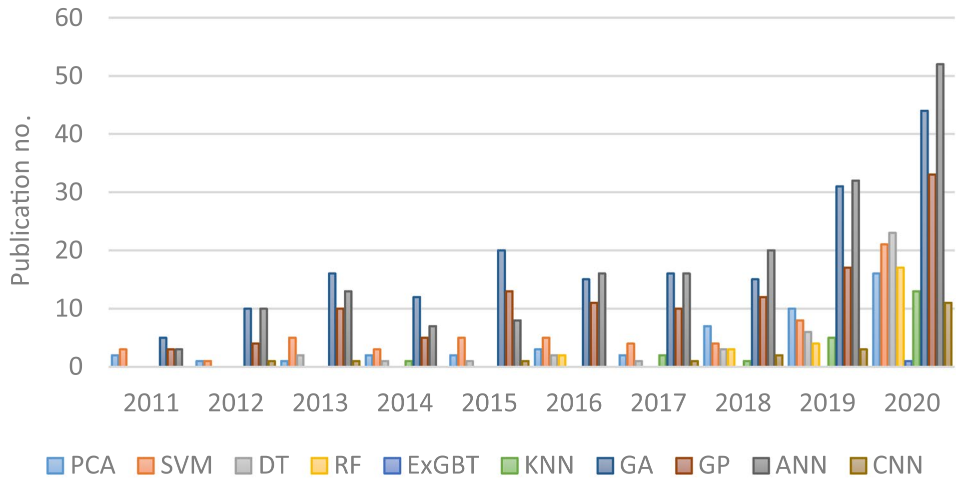
Study	Dataset	Objective	Algorithm(s)	Training procedure (%)	Performance metrics
Bal and Buyle-Bodin [111]	233	Prediction of creep in concrete	ANN	Training (70%) Testing (15%) Validation (15%)	MSE
Ince [112]	464	Characterizing failure of concrete structures	ANN	Training (55%) Testing (45%) Validation (–)	MSE
Marani and Nehdi [90]	154	Explore the compressive strength of cementitious composites incorporating phase change material microcapsules	RFR, Extra trees, ExGBT	fivefold cross-validation	R^2, MSE, RMSE, MAE, RMSE
Aslam et al. [74]	357	Predicting mechanical behavior of high strength concrete	GEP	Training (70%) Testing (15%) Validation (15%)	R^2, RMSE, MAE, RRMSE
Ahmad et al. [113]	915	Assessment of RC designs	ANN	Training (60%) Testing (20%) Validation (20%)	MSE, MAE, R^2
Huang and Burton [62]	114	Classifying in-plane mode of failure	LR, SVM, DT, RF, Adaptive Boosting (AB), MLP	Training (70%) Testing (30%) Validation (–) fivefold cross-validation	Accuracy
Javed et al. [91]	65	Predicting the compressive strength of sugarcane bagasse ash concrete	GA, GEP	NA	Nash Sutcliff efficiency (NSE), R^2, RMSE
Kaloop et al. [114]	1030	Extract the optimum inputs that use to design the HPC	Multivariate adaptive regression splines model (MARS) with gradient tree boosting machine (GBM)	Training (70%) Testing (30%) Validation (–)	R, normalized percentage root mean square error (NRMSE), MAE, ratio of RMSE to the standard deviation (RSR), coefficient of persistence (*cp*), and degree of index (d)
Okazaki et al. [102]	265	Model cracking of concrete	MLR, SVM, DT, ANN, Gaussian process regression (GPR)	fivefold cross validation	RMSE
Ben Chaabene et al. [105]	484	Identify failure mode of steel fiber reinforced beams	Atom search optimization (ASO), ANN, SVM, DT, KNN	Training (75%) Testing (25%) Validation (–)	MAE, RMSE, R, Modified agreement index (d′)
Feng et al. [115]	254	Failure mode classification of RC columns	CART, SVM, ANN, RF, AdaBoost	Training (80%) Testing (20%) Validation (–) tenfold cross validation	Precision, Recall, Accuracy
Nguyen et al. [92]	Dataset 1: 177 Dataset 2: 1133	Compressive strength prediction	DNN	Training (70%) Testing (15%) Validation (15%)	R, RMSE, MAE, RRMSE, relative MAE (RMAE)
Huang et al. [100]	299 for compressive strength & 269 for flexural strength	Predicting the optimum mixture design of steel fiber reinforced beams	SVR, firefly algorithm (FA)	Training (70%) Testing (30%) Validation (–) tenfold cross-validation	MAPE, RMSE, R, MAE

Table 1 (continued)

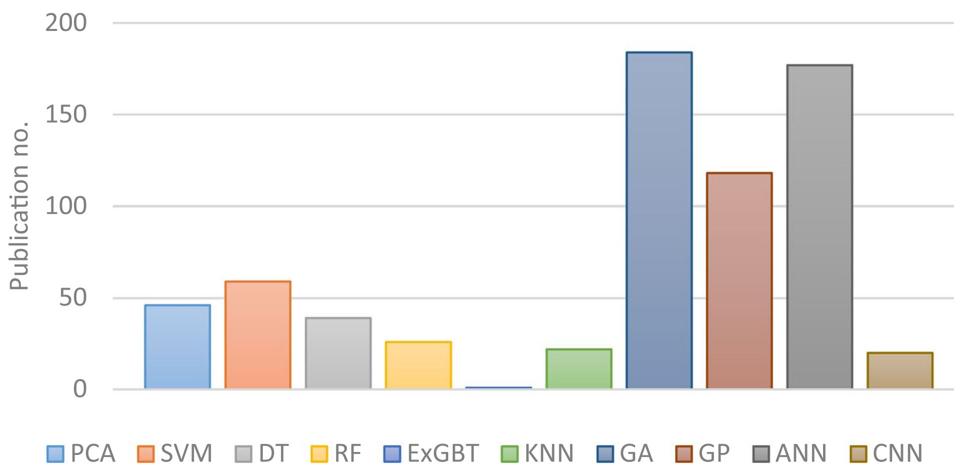
Study	Dataset	Objective	Algorithm(s)	Training procedure (%)	Performance metrics
Jalal et al. [93]	72	Predicting the compressive strength of the rubberized cement composites	ANN, GEP, ANFIS	Training (80%) Testing (10%) Validation (10%)	MAPE, RMSE, R^2
Golafshani and Ashour [101]	413	Predicting the elastic modulus of SCC	Biogeographical-based programming (BBP), artificial bee colony programming (ABCP)	Training (80%) Testing (20%) Validation (–)	MAE, RMSE, MAPE, R^2, Objective function
Sultana et al. [94]	13	Prediction of the mechanical properties of jute fiber reinforced concrete	ANN, SVR, Response Surface Methodology (RSM)	Training (70%) Testing (15%) Validation (15%) fivefold cross validation	R, residual, relative error (RE), MAE, RMSE, and fractional bias (FB)
Chou et al. [116]	1700	Predicting the compressive strength of high-performance concrete	MLP, SVM, CART	tenfold cross-validation	MAE. RMSE, MAPE, synthesis index (SI) based on the above three statistical measures
Ben Seghier et al. [97]	218	Predicting the ultimate bond strength	ANN, GEP	Training (80%) Testing (20%) Validation (–)	Standard deviation (SD), MSE, RMSE, absolute percent relative error (APRE) and average absolute per- cent relative error (AAPRE)
Duong et al. [117]	150	Predicting columns behavior	ANN, Balancing Composite Motion Optimization (BCMO)	Training (80%) Testing (20%) Validation (–)	MAE, R^2
Naseri et al. [99]	232	Investigating mixture design of sustainable concretes	Water cycle algorithm (WCA), soccer league competition (SLC), GA, ANN, SVM	Training (75%) Testing (25%) Validation (–)	MAE, RMSE, R, MSE, R^2
Gorphade et al. [98]	324	Predicting properties of high-performance concretes	GA, ANN	Training (80%) Testing (15%) Validation (5%)	RMSE
Yan et al. [118]	77	Predicting fracture parameters	ANN	Training (70%) Testing (15%) Validation (15%)	MSE
Golafshani et al. [119]	179	Predicting the bond strength of steel bars	ANN, fuzzy logic (FL)	Training (70%) Testing (15%) Validation (15%)	MSE, MAPE, R, RMSE, R^2
Naik and Kute [120]	118	Predicting the shear strength of high-strength steel fiber-reinforced concrete deep beams	ANN	Training (80%) Testing (10%) Validation (10%)	Residual sum of squares
Hoang et al. [121]	218	Predicting bond strength of corroded steel reinforcement	LSSVR, Differential flower pollination (DFP)	Training (90%) Testing (10%) Validation (–)	MAPE, RMSE, R^2

Table 1 (continued)

Study	Dataset	Objective	Algorithm(s)	Training procedure (%)	Performance metrics
Akin and Abejide [122]	NA	Determining effective parameters in concrete strength	GEP	Training (–) Testing (–) Validation (–)	MSE, RMSE, R^2
Salami et al. [104]	80	Extracting the Corrosion initiation time of embedded steel	RF, LR, KNN, ANN, SVR	Training (67/70/80/85%) Testing (33/30/20/15%) Validation (–)	RMSE, R
Castelli et al. [95]	1028	Predicting the compressive strength of high-performance concrete	Geometric Semantic Genetic Operators (GSGO)	Training (70%) Testing (15%) Validation (15%)	RMSE
Khademi et al. [123]	173	Predicting the compressive strength of concrete	ANN, ANFIS, MLR	Training (85%) Testing (15%) Validation (–)	R^2
Yaseen et al. [96]	91	Predicting the compressive strength of foamed concrete	ELM, MARS, SVM, M5 tree	Training (95%) Testing (5%) Validation (–%)	MSE, RMSE, R
Qi et al. [124]	2000	Determining the load resisting capacity of wood members	ANN	Training (80%) Testing (20%) Validation (–)	MAE, MSE, RMSE, R^2
Kellouche et al. [103]	300	Determining carbonation in concrete	ANN	Training (60%) Testing (20%) Validation (20%)	MSE
Abuodeh et al. [125]	110	Assess compressive strength of UHPC concrete	ANN	Training (70%) Testing (15%) Validation (15%)	NMSE
Abdalla and Hawileh [126]	50	Predict energy dissipated in steel reinforcing bars in reinforced concrete members	ANN	Training (70%) Testing (15%) Validation (15%)	MSE, NMSE, MAE, R, Absolute error



(a) Year of publication



(b) Algorithm frequency

Fig. 17 Details on AI-based methods often used in earthquake engineering [Note: GA, ANN, and GP rank the highest]

5.2 Earthquake Engineering

Earthquake is a complex and disastrous event that can significantly damage structures, and hence seismic engineering has been an evolving research area over the past decades [127]. The large number of factors that may influence the seismic behavior of structures complicate the study of structural performance. As a result, the use of AI-based methods has been extensively explored in the last two decades (see Fig. 17). For example, Arsalan [128] presented a novel approach for obtaining factors governing earthquake resistance of RC structures using the ANN algorithm. This researcher examined 256 RC buildings of 4 and 7 storey high via pushover analysis. Post a validation with an accuracy of about 92–99%, a sensitivity analysis was conducted and revealed that shear wall ratio and short column formation are the most significant structural components that influence seismic performance. On the other hand, concrete strength and transverse reinforcement were of negligible influence.

Mangalathu and Burton [129] evaluated seismic damage through a DL variant (long short-term memory (LSTM)) by examining 3423 buildings. They reported an accuracy of 59–94%, with those attaining the lowest accuracy being related to buildings comprising 7% of the total datasets. This report showcases the significance of the size of databases in the model's prediction capability. Zhang et al. [130] examined four story RC special moment frames (935 classified response patterns and 93,500 damage patterns) using CART and RF and reported high prediction accuracy of 91% and 88%. Hwang et al. [131] applied regression- and classification-based ML techniques to infer the damage state of RC frame buildings after an earthquake. They noted that adaptive boost (AdaBoost) and ExGBT algorithms have better

Table 2 Summary of works on ai-based methods tackling earthquake engineering

Study	Dataset	Objective	Algorithm(s)	Training procedure (%)	Performance metrics
Arslan [128]	256	Determining changes in load-bearing systems during earthquakes	Various ANN topologies	Training (42%) Testing (58%) Validation (–)	R^2, traditional error
Zhang and Burton [130]	935	Assessing post-earthquake structural safety	CART, RF	Training (75%) Testing (25%) Validation (–) tenfold cross- validations	Sensitivity, specificity, receiver operating characteristic (ROC)
Mangalathu and Burton [129]	3,423	Classification of building damages from textural document	Long short-term memory (LSTM)	Training (75%) Testing (25%) Validation (–)	Confusion matrix
Hwang et al. [131]	137	Predicting the seismic response and structural collapse	MLR, ridge regression, DT, RF, AdaBoost, ExGBT, Naıve Bayes (NB), KNN	Training (70%) Testing (30%) Validation (–)	R^2, RMSE, confusion matrix
Luo and Paal [133]	160	Quantification of seismic behavior of RC buildings	Locally weighted least squares support vector machines for regression (LWLS-SVMR), coupled simulated annealing (CSA), Grid search (GS)	Training (70%) Testing (30%) Validation (–) tenfold cross-validation leave-one-out cross-validation	R^2, RMSE, MAPE
Morfidis and Kostinakis [132]	30	Predicting seismic damage state	ANN	Training (70%) Testing (15%) Validation (15%)	R, MSE
Oh et al. [134]	13,230	Predicting the seismic response of structures	CNN	Training (85%) Testing (15%) Validation (–)	RMSE
Asteris [135]	4,026	Predicting the fundamental period of vibration of infilled frame reinforced concrete structures	Artificial bee colony (ABC)	Training (70%) Testing (15%) Validation (15%)	RMSE, MAPE, R^2
Su and He [136]	45,360	Detecting damage in reinforced concrete frames	DT	Training (50%) Testing (50%) Validation (–)	Confusion matrix, accuracy, standard deviation
Liu and Zhang [137]	500	Predicting damage of steel frame structures	ANN	Training (70%) Testing (30%) Validation (–)	MAE

performance for collapse status classification for future earthquake ground motions.

Morfidis and Kostinakis [132] applied ANN to rapidly assess the seismic performance of structures by examining 65 actual ground motions. The ANN was used to identify the "damage state' of seismically damaged buildings in real-time. Luo and Paal [133] adopted locally weighted least squares support vector machines for regression (LWLS-SVMR) to predict the degree in RC structures. These researchers report that the LWLS-SVMR was found superior to other examined approaches in predicting drift capacity in RC flexure-, shear-, and flexure–critical shear columns. The reader is invited to refer to Table 2 for a summary of additional works that adopted AI-based methods to evaluate the seismic performance of structures.

5.3 Wind Engineering

Wind effects are considered to be one of the critical natural forces, particularly in high-rise structures. The dynamic and distinctive nature of wind loads often impose complications (and limitations) to structural tests and simulations. This is where AI-methods become handy, especially when there is a lack of proper guidelines for designing structures with unique shapes against the wind or when laboratory conditions are costly or constraints [138]. Figure 18 presents the application of various AI-based methods in the area of wind engineering in the past decade.

In one study, Hu et al. [139] conducted an examination of multiple algorithms, namely, DT, RF, ExGBT, and generative adversarial networks (GANs), on 2664 cases of tall buildings. In this work, the selected algorithms were

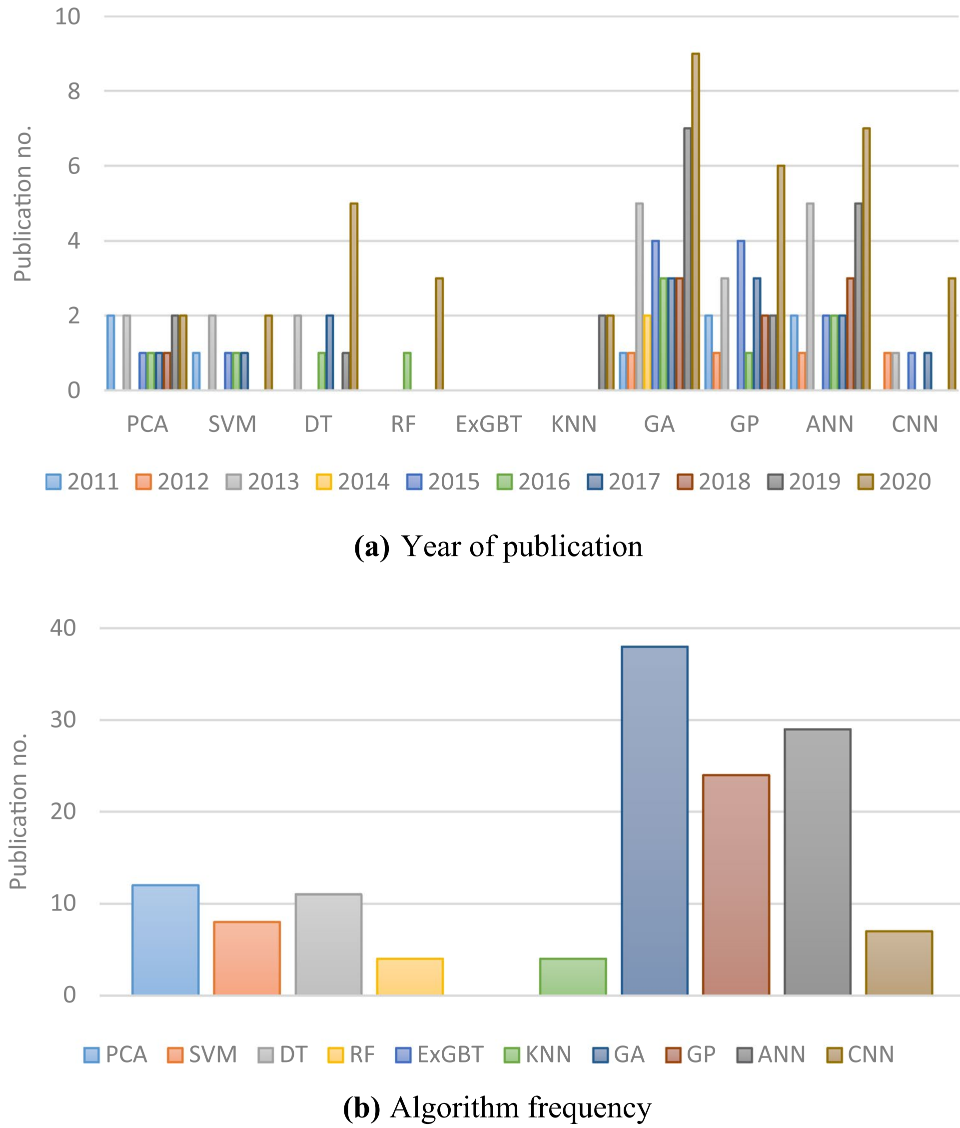


Fig. 18 Algorithm reoccurrences in wind engineering [Note: GA, ANN, and GP rank the highest]

examined on different portions of the dataset ranging from 10 to 90%. Hu et al. [139] report that GANs were able to accurately predict wind pressure coefficients on the principal building by using 30% of a dataset to predict—thereby reducing the cost associated with wind tunnel tests. Payán-Serrano et al. [140] applied a back-propagation ANN to investigate the effect of wind at different velocities on high-rise structures with different configurations using a large dataset (25,600 cases) with success. These researchers deployed seven different ANN models by varying the number of neurons ranging from 1 to 30 and showed that reducing neurons below 20 had no significant effect on reducing model error.

Dongmei et al. [79] carried out an investigation to predict pressure coefficients via proper orthogonal decomposition (POD-BPNN). The outcome of their research shows that this modified algorithm can attain a small error margin between 3–5%. The same algorithm can also predict all wind forces, moments imposed on structures due to wind force, and any spectrum or coherent functions. Nikose and Sonparote [141] presented an ANN capable of predicting the dynamic across-wind response of tall buildings as per the provisions given in the Indian Wind Code (IWC). Both researchers recommended using at least 2000 data points to realize adequate accuracy (around 99.5%). Other works in this branch of structural engineering are further summarized and examined in Table 3.

5.4 Fire Engineering

Structural fire engineering is a niche branch [148, 149]. Unlike earthquake and wind effects, fires are not bound to a geographical region nor season. This makes the problem of structures fires a unique one. Most existing works in this area

Table 3 Summary of works on ai-based methods applied in wind engineering

Study	Dataset	Objective	Algorithm(s)	Training procedure (%)	Performance metrics
Nikose and Sonparote [141]	2900	Predicting the dynamic wind response of tall buildings	ANN	Training (70%) Testing (15%) Validation (15%)	R, MSE
Hu et al. [139]	2664	Evaluating wind interference effects of buildings	DT, RF, ExGBoost, generative adversarial networks (GANs)	Training (80%) Testing (20%) Validation (–) tenfold cross-validations	R^2
Payán-serrano et al. [140]	25,600	Investigating maximum story drift	ANN	Training (85%) Testing (15%) Validation (–)	MSE, Sum square error (SSE)
Dongmei et al. [79]	14 story buildings	Predicting wind loads	ANN	Training (-%) Testing (–%) Validation (–)	RMS, RMSE
Paul and Dalui [142]	3 parametric buildings	Determining pressure coefficient (C_p)	ANN	Training (70%) Testing (30%) Validation (–)	SSE, R^2, RMSE
Oh et al. [143]	2100	Monitoring wind response of tall buildings	CNN	Training (95%) Testing (5%) Validation (–)	RMSE
Gavalda et al. [144]	118 + 90	Determining pressure coefficient (C_p) in low rise buildings and gable-roofed structures	ANN	NA	R, MSE
Bairagi and Dalui [145]	Three sets	Investigating wind incidence angles	ANN	NA	MSE, sum of square due to regression (SSR), sum of square error (SSE), the total sum of square (SSTO) is the sum of SSR and SSE
Abbas et al. [146]	48	Predicting force time histories	ANN	Training (80%) Testing (20%) Validation (–)	peak and root mean square (RMS), new metrics for time history comparison
Le and Caracoglia [147]	500	Fragility analysis	ANN	Training (70%) Testing (15%) Validation (15%)	NA

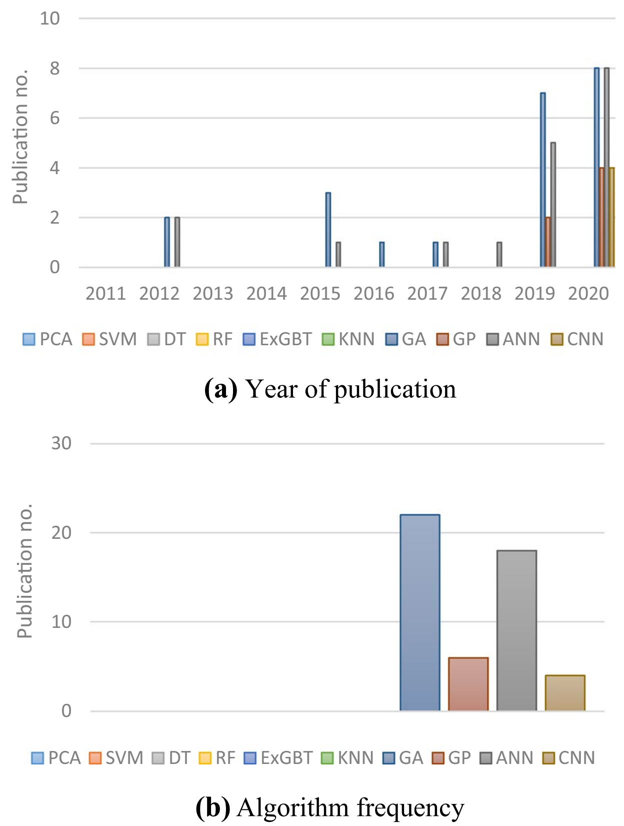


Fig. 19 Algorithm reoccurrences in fire engineering [Note: GA, ANN, and GP rank the highest]

applied traditional and perspective approaches; however, there has been some interest in exploring the use of AI, ML, and DL as of late (see Fig. 19) [6, 150]. For example, Bilgehan and Kurtoğlu [151] examined the effect of temperature rise and spalling on concrete structures through an adaptive neuro-fuzzy inference system (ANFIS) and achieved 98% accuracy. Fu [152] examined the fire response of structural metal frames against high-temperature loads using DT, KNN, and ANN. Fu [152] developed a dataset using Monte Carlo simulation and random sampling and noted that both KNN and ANN achieve better performance than Dz in classifying progressive collapse under fire.

Another system-level analysis was conducted by Naser and Kodur [46, 153], who applied GP and GA algorithms to classify bridges vulnerable to fire conditions. These researchers noted the dire need for data points to enable more friendly use of AI methods in structural fire engineering applications. Panev et al. [154] adopted SVM to predict the fire response of composite shallow floor systems. They reported an accuracy exceeding 96% using 150 data points. Lazarevska et al. [155] applied a fuzzy-neural network (FNN) to evaluate the fire response of eccentrically loaded concrete columns. They successfully developed a prognostic model to determine the fire resistance of such columns with ease by examining close to 400 data points. Ketabdari et al. [156] performed tests on steel bolts often used in joints in high-rise buildings, which are generally susceptible to heat during fires. In this study, the GA algorithm was applied to 420 data samples from different bolts types (8.8 mm, 10.9 mm) to predict the properties of the afferent bolts. These researchers reported all relative errors to be less than 10%, and concluded that GA has a high ability to predict properties of bolts at elevated temperatures.

The use of AI-based methods also explores the properties of concrete materials [38, 157–161]. Lee et al. [157] applied a BPNN to predict the thermal conductivity property of concrete and achieved an accuracy of 99%. McKinney and Ali [162] also applied ANNs to classify RC columns with high vulnerability to fire-induced spalling. Other works on spalling also include the following [38, 163, 164].

Despite the works that leveraged AI, ML, and DL in concrete and metallic structures, very few works were directed to timber structures under fire conditions [165, 166]. One such work is that of Tasdemir et al. [167], who investigated the behavior of wooden structures made from three distinct timbers (Pine, Fir, and Popular), making 150 data points. These researchers pointed out that the developed ANN could capture cross-sectional sizes of damaged wooden specimens with relative MSE error (~0.0055). Cachim [168] applied ANN to predict temperature rise in timber beams and residual resistance in wooden rectangular sections after heating. Tung [169] determined wooden roofs' thermal resistance using the ANN algorithm with a small error of 4%. A summary of other works is listed in Table 4.

5.5 Structural Health Monitoring

It is of utmost importance to structural engineers to be able to trace the response of structures at ambient (day-to-day) conditions, as well as during extreme events. This is often practiced through structural health monitoring networks, which utilize a series of sensors spread throughout a given structure. Given the massive amount of generated data, analyzing data from such sensor networks is a hectic procedure [171]. It is due to the above that AI-based methods can come in handy and are becoming of interest to structural engineers (see Fig. 20) [172].

Chun et al. [173] applied RF and ANN to monitor the cracking of concrete bridges and the associated corrosion rate of steel rebars. In their work, Chun et al. [173] compiled data from a campaign of 24 experimental tests to train the AI algorithms. Overall, RMSE and R^2 were reported to be 0.52 and 0.89, respectively, which implies the good performance of the AI models. The outcome of this work shows that adopting AI methods can accelerate the inspection of RC structures. Diez et al. [174] were able to successfully classify failure of joints in the Sydney Harbour bridge using

Table 4 Summary of works on ai-based methods adopted in structural fire engineering

Study	Dataset	Objective	Algorithm(s)	Training procedure (%)	Performance metrics
Naser [153]	150	Examining vulnerability of bridges to fire	DL, DT, GA, GEP	Training (60%) Testing (20%) Validation (20%)	MAE, R^2
Naser [159]	84	Predicting the fire-induced spalling and fire resistance of RC columns	GA	Training (70%) Testing (30%) Validation (–)	MAE, R^2
Seitlllari and Naser [72]	89	Evaluating of spalling phenomena in RC columns	MLR, ANN, ANFIS, GA	Training (80%) Testing (20%) Validation (–)	MSE, MARE, R^2
Bilgehan and Kurtoğlu [151]	520	Predicting the ultimate moment capacity of RC slabs under fire	ANFIS	Training (55%) Testing (45%) Validation (–)	R^2, RMSE, Mean bias error (MBE)
Fu [152]	1 steel framed building	Predicting fire resistance of structural frames	DT, KNN, ANN	Training (80%) Testing (20%) Validation (–)	NA
Lazarevska et al. [155]	398	Determining fire resistance capacity of eccentric loaded RC members	ANFIS	Training (80%) Testing (20%) Validation (–)	NA
Lee et al. [157]	152	Predicting thermal conductivity of concrete	ANN	Training (80%) Testing (20%) Validation (–)	MSE, R
Tasdemir et al. [167]	180	Predicting burned cross section of wooden structures	ANN	Training (85%) Testing (15%) Validation (–)	MSE
Ketabdari et al. [156]	420	Investigation material properties	GEP	Training (70%) Testing (–) Validation (30%)	RMSE, MAE, R
Liu and Zhang [164]	265	Predicting spalling	ANN	Tenfold cross-validation	Greedy trial-and-error method
Liu and Zhang [170]	306	Predicting spalling	ANN	Tenfold cross-validation	Greedy trial-and-error method
Tung and Hung [169]	36	Predicting fire resistance ratings of the wooden floor assemblies	ANN	Training (80%) Testing (10%) Validation (10%)	R^2, MSE

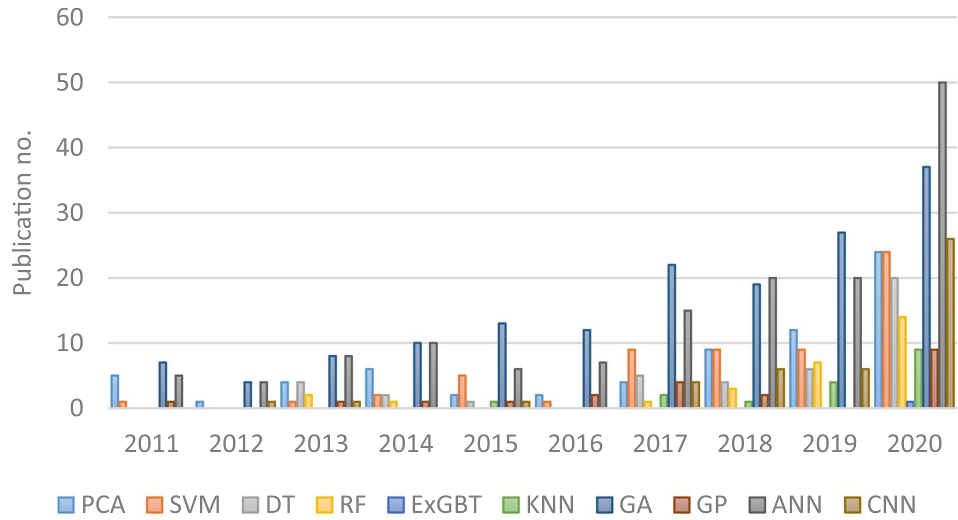
vibration signals generated by 23,849 events of the passage of vehicles and the k-mean clustering algorithm.

Kurian and Liyanapathirana [175] applied three algorithms KNN, SVM, and RF, to three-story structures. It was found that the RF algorithm for both damaged and intact structures achieved good accuracy in predicting data compared to the other two algorithms. The accuracy is reported at 92% for the RF algorithm and 85% and 80% for the SVM and KNN algorithms, respectively. Athanasiou et al. [176] applied multifractal analysis and DT to 119 collected images from laboratory tests as a mean to identify faults in concrete shell structures. These researchers obtained an accuracy of 89.3%, which was deemed acceptable.

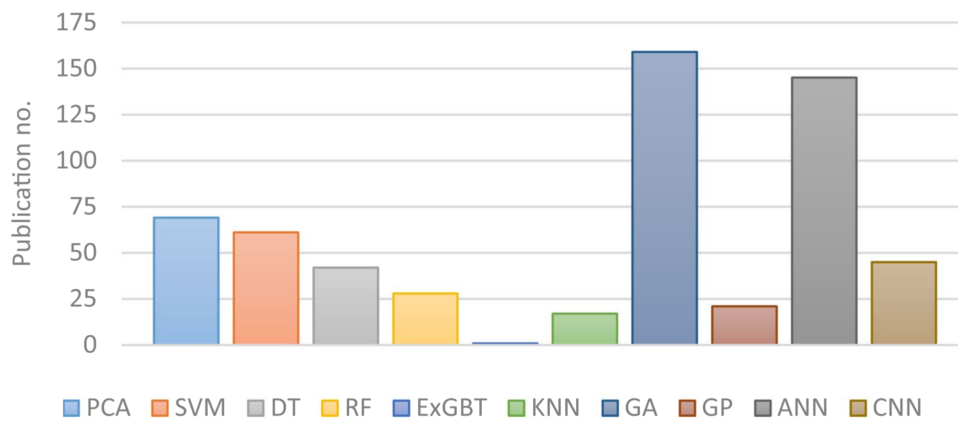
The use of AI, ML, and DL opened the door for automatic damage detection through analyzing imagery [177, 178]. In one study, Noh et al. [179] developed a fuzzy-c-mean algorithm capable of predicting cracks with 0.3–1 mm diameter of 1 mm and from a distance of one meter [179]. Other works adopted DL algorithms, such as Xu et al. [180], who achieved 80% accuracy using a modified faster R-CNN algorithm through Matlab software. Xu et al. started with 400 photo samples and then augmented these photos by rotation to realize 2400 samples. Of these samples, 90% were used for training and 10% for testing, which achieved an overall average precision of 80%.

Dung and Anh [181] showed that using a fully convolutional neural network (FCNN) can detect the mode of failure with high reliability. In their research, 40,000 data samples of cracking were used in training the FCNN to achieve an accuracy of 90%. Li and Zhao [182] also examined the CNN algorithm on 60,000 datasets consisting of various images (blurry crack, shadow, rusty surface, and rough) and reported high accuracy of 99% in training and testing regimes. Rashidi et al. [183] compared SVM,

Fig. 20 Algorithm reoccurrences in structural health monitoring [Note: GA, ANN, and PCA rank the highest]



(a) Year of publication



(b) Algorithm frequency

MLP, and Radial Basis Function (RBF) in categorizing concrete, red brick, and oriented strand board (OSB). Their analysis shows that the noted algorithms perform well in detecting materials with distinct color and appearance while struggle in classifying materials of similar color and appearance properties. Overall, the SVM was reported to be of the highest accuracy. Other studies that tackled this research area can be found herein [184, 185] and in Table 5.

5.6 Structural Systems and Structural Members

This section showcases works applied to exploring the use of AI, ML and DL algorithms in problems related to structural systems and structural members. Due to the breadth of this branch, this section can be further split into four subsections, namely, RC members, FRP-strengthened members, and other types of members. Figure 21 demonstrates the trend in this area, along with the most frequently used algorithms.

5.6.1 RC Members

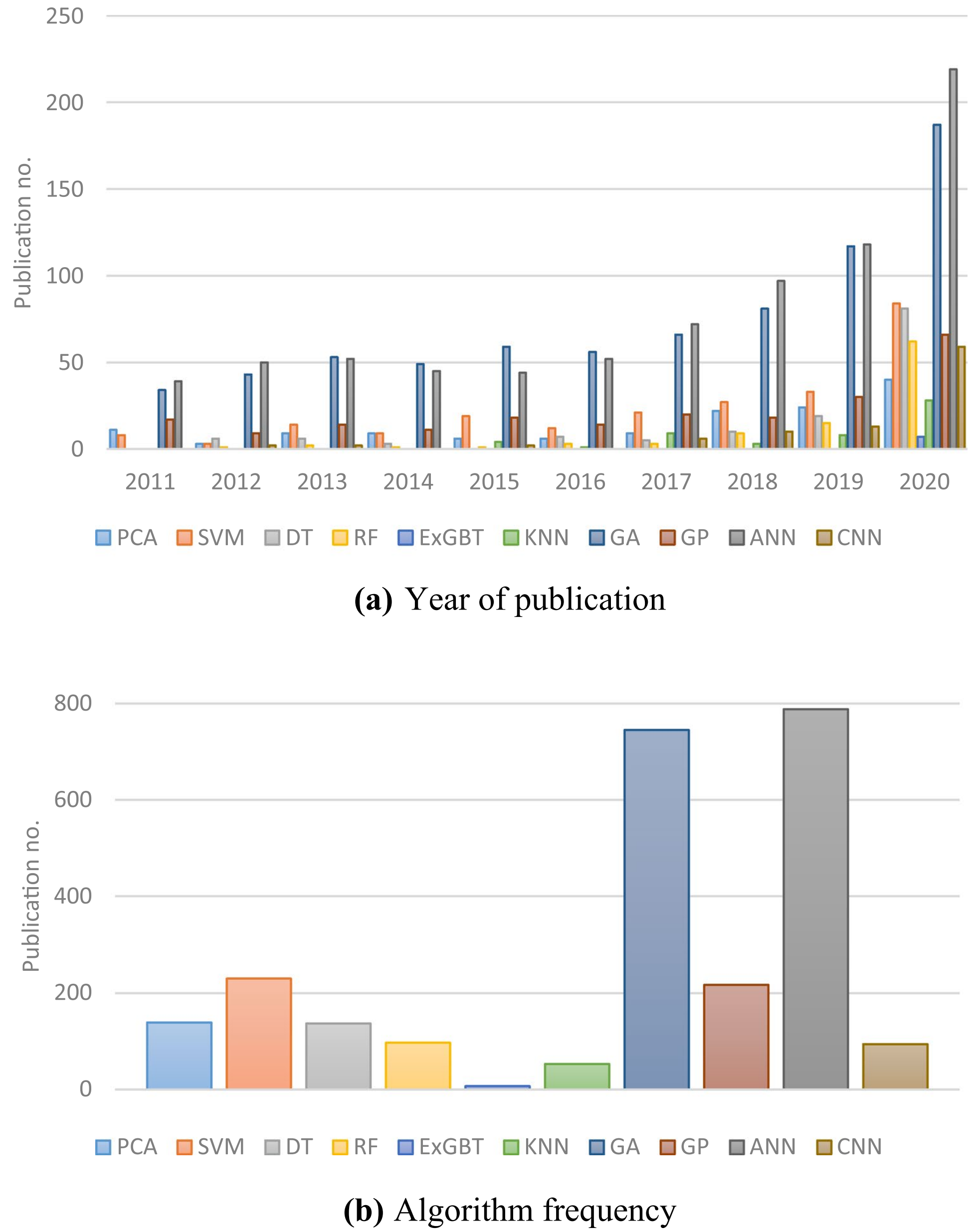
Mangalathu et al. [192] examined data points from 393 tests on RC shear walls using eight ML algorithms, including KNN, DT, RF, Adaboost, ExGBT, Naïve Bayes, LightGBM, and CatBoost. These researchers reported 86% accuracy in identifying the failure mode of shear walls and developed an open-source tool as well that can be used with ease to evaluate failure modes. Another study was conducted on RC shear walls by Chen et al. [193], who examined the shear capacity of 139 squat walls using ANN-PSO. The evaluation of this model compared to other models shows the model's high accuracy reaching R^2 of 97.6%, as compared

Table 5 Summary of works on ai-based methods tackling structural health monitoring problems

Study	Dataset	Objective	Algorithm(s)	Training procedure (%)	Performance metrics
Chang and Chang [186]	7-story building	Determining damage location and severity	ANN	Training (75%) Testing (25%) Validation (–)	MSE, R
Diez et al. [174]	28,511 + 27,407	Classifying joints and damage in bridge structures	K-means clustering	Training (77/60%) Testing (23/40%) Validation (–)	Confusion Matrix
Kurian and Liyanapathirana [187]	8,192	Damage detection	KNN, SVM, RF	Training (80%) Testing (20%) Validation (–)	Confusion Matrix
Chun et al. [173]	392	Evaluating internal damage	RF	Training (75%) Testing (25%) Validation (–) Leave one out cross-validation (LOOCV)	RMSE, R^2
Hoang and Nguyen [188]	1000	Classifying surface damage in concrete	linear population size reduction SVM	Training (90%) Testing (10%) Validation (–)	Accuracy, precision, recall, negative predictive value (NPV), and F1 score
Liu and Zheng [189]	8259	Classifying of damage in steel elements	CNN	Training (80%) Testing (20%) Validation (–)	Accuracy
Satpal et al. [190]	325	Identifying damage location in Aluminum cantilever beams	SVM	Training (95%) Testing (5%) Validation (–)	Average percentage error
Mariniello et al. [191]	Two RC. frames	Detecting localized damage in structure	DT	Training (50%) Testing (50%) Validation (–)	Accuracy, confidence of probabilistic predictions, and localization errors
Athanasiou et al. [176]	119	Quantifying crack patterns	DT	tenfold cross validation	Confusion matrix
Noh et al. [179]	50	Identifying cracks	Fuzzy c-mean clustering	–	Recalled precision
Xu et al. [180]	400	Seismic damage identification	Faster-R-CNN	Training (90%) Testing (10%) Validation (–)	Average mini-batch loss and accuracy, precision and recall
Cha et al. [81]	297	Developing a structural damage detection method	Faster R-CNN	Training (70%) Testing (20%) Validation (–)	Average precision (AP)
Dung and Anh [181]	40,000 + 600	Proposing a crack detection	fully convolutional net (FCN)	Training (66%) Testing (16%) Validation (16%)	F1 score, average precision (AP), bounding box
Cha et al. [82]	332	A vision-based approach for detecting cracks on concrete	CNN	Training (85%) Testing (15%) Validation (–)	Accuracy, confusion matrix
Li and Zhao [182]	1455	Detecting cracks	CNN	Training (85%) Testing (15%) Validation (–)	Accuracy
Rashidi et al. [183]	750	Detecting of building materials	ANN, Radial Basis Function (RBF), SVM	Training (80%) Testing (20%) Validation (–)	Confusion matrix

to 35.8% and 13.7%, using the American Concrete Institute (ACI) design guide and the Canadian Standards Association (CSA A23.3–04) codal provisions, respectively.

Feng et al. [115] applied ensemble learning mode of failure and shear capacity in 254 cyclically-loaded RC columns. These researchers noted that Adaboost managed to achieve



(a) Year of publication

(b) Algorithm frequency

Fig. 21 Algorithm reoccurrences in structural systems and structural members [Note: ANN, GA, and SVM rank the highest]

high predictive capacity than the Chinese design code provisions. Ketabdari et al. [194] examined the shear strength of circular RC columns using PSO and GEP algorithms. Then, compared the predicted performance of the algorithms against empirical equations in provisions such as ASCE-ACI 426, ACI -318. Ketabdari et al. [194] noted that relative error between measured and code-predicted data was in the range of 25–30%. This error was reduced to 9–13% by incorporating the above two algorithms.

Ly et al. [195] examined 463 experimental data on RC beams points through a real-coded genetic algorithm (RCGA) and the firefly algorithm (FFA) and noted the better performance of the former over the latter. Ababneh et al. [196] conducted a study to obtain the shear strength of unreinforced beams made of recycled aggregate concrete (RAC) via ANN. In this study, it was observed that algorithm predictions are about eight percentage away from actual measurements. Also, in this study, it was shown that the input parameters are influential factors in accurately predicting the shear strength of the sample.

Solhmirzaei et al. [197] applied classifiers and regression models to predict failure mode and shear capacity of more than 200 ultra-high performance concrete (UHPC) beams. Besides, GEP, SVM, ANN, and KNN were used. It was shown that among the algorithms designated for failure mode, the ANN algorithm with 98% accuracy. On the other hand, the SVM and KNN algorithms in shear-flexure failure mode have better data prediction capability than flexural or shear failure mode. Bai et al. [198] investigated the deflection history of 120 RC beams using the "bagging technique" and its combination with other algorithms such as ANN, SVM, and ANFIS. They showed that SVM-ANFIS has the best-predicting capability. Other works on beams can be found elsewhere [49, 199, 200].

5.6.2 FRP-Strengthened Members

The use of fiber-reinforced polymer (FRP) composites has been well established in the structural engineering fields and dates back to a few decades ago. FRPs are often used to retrofit or strengthened weakened RC and metallic structures by means of adhesively and anchored systems, given their use of installation, high strength to weight ratio, and cost-effectiveness [201–203].

Lee and Lee [199] developed an ANN algorithm to investigate the shear strength of FRP-reinforced flexural members. ANN predictions from 288 cases were compared against those obtained from codal provisions such as the British Institution of Structural Engineers guidelines (BISE) and the Japan Society of Civil Engineers (JSCE) and noted high accuracy. Naderpour et al. [204], in a similar paper, it was shown that ANN has high accuracy ($R^2 = 92\%$) in predicting shear capacity in FRP joints to concrete beams and pointed out that the depth of cross-section has the most impact on the ANN model predictions. Mansouri et al. [205] investigated the behavior of concrete elements bounded by FRP sheets using four algorithms ANN, ANFIS, multivariate adaptive regression splines (MARS), and M5 TREE. Their results show that ANN and ANFIS did not perform as well as MARS and M5 Tree in predicting strain ratios of FRPs.

Nguyen et al. [206] used 331 laboratory samples and ANN and ANFIS algorithms to establish a relationship to predict the compressive strength of self-compacting concrete reinforced with FRP. Naderpour et al. [207] applied a three-layer ANN to investigate shear strength of FRP-reinforced concrete beams without longitudinal reinforcements and achieved 9.72% error rate as compared to codal models such as (ACI-440, ISIS Canadian design manual (ISIS-M03-07), and BISE). The bond strength between FRP and concrete is a key factor that also was explored in detail. For example, Su et al. [208], and Köroglu [209] examined the bond performance of FRPs via several algorithms. Other works examined the use of FRP in other structural elements such as beams, columns, joints, and slabs can be found elsewhere [200, 210–216].

5.6.3 Other Types of Structural Members

In lieu of RC and FRP-strengthened structural members, the open literature also contains studies that explored the use of AI-based in other types of structural members and components. To name a few, Degtyarev [217] developed an ANN to predict the shear strength of cold-formed steel channels and attained high accuracy by finetuning this ANN. Degtyarev [217] also explored the influence of various tuning options and observed that two-hidden layer ANNs could show better performance metrics than ANNs with one-hidden layer. Le [218] examined the axial capacity of concrete-filled steel tubes using the Gaussian process regression (GPR) algorithm and noted that column slenderness to be of the highest influence. Nguyen et al. [219] applied the invasive weed optimization (IWO) algorithm to predict the axial strength of 99 rectangular CFST columns and reported remarkable success ($R^2 \sim 98\%$). Thai et al. [220, 221] carried out a series of works using various algorithms on CFST columns. Other works were applied toward predicting the behavior of wood members [124] and structural connections [222–224], as well as those listed in Table 6.

6 Analysis of Observed Practices

This section presents further insights into observations noted from this scientometrics review. The focus of this analysis is to identify common trends and practices posed by works that leveraged AI-based methods in structural engineering over the past decade. Special attention is paid towards frequently adopted algorithms, applied model development procedures, size of datasets used, and employed performance metrics.[2]

6.1 Frequently Adopted Algorithms

Figure 22 lists the most frequently used algorithms as collected herein. As one can see, ANN, GA, GP, and SVM top all other algorithms. Of these four algorithms, ANN and GA ranked the heights with about 2–3 more times re-occurrences than GP or SVM (i.e., 55.9% of the time). This may stem from the notion that ANN and GA are well-established algorithms that have found home in this domain a while ago, which could explain the familiarity of structural engineers with these algorithms. In addition, ANN and GA comprise visual architecture that can be easier to visualize and apply. Finally, these two algorithms can be used in a wide range of problems, are often associated with little data processing, and can be incorporated into other algorithms (to create hybrid tools) with ease. Figure 22 and d show knowledge maps for ANN and GA as obtained through our analysis.

[2] It is worth noting that the analysis displayed herein is based primarily based on our observations and constraints of this work. We do believe that a more systematic examination by means of social trends, surveys, and peer practices etc. is warranted.

Table 6 Summary of works on ai-based methods applied in various structural systems

Study	Dataset	Objective	Algorithm(s)	Training procedure (%)	Performance metrics
Cachim [168]	5	Predicting temperature rise within timber structures	ANN	Training (45%) Testing (60%) Validation (–)	RMSE, MAE, MAPE, R^2
Mangalathua et al. [192]	393	Failure mode identification of concrete shear walls	KNN, DT, RF, Adaboost, ExGBoost, Light GBM	Training (70%) Testing (30%) Validation (–)	Confusion matrix
Ketabdari et al. [194]	200	Predicting shear strength of short circular RC columns	PSO, GA	Training (70%) Testing (–) Validation (30%)	RMSE, MAE, R
Ababneh et al. [196]	231	Predicting concrete contribution in the shear capacity of recycled aggregate concrete beams	ANN	Training (80%) Testing (20%) Validation (–)	mean relative error (MRE), MAE, R^2
Solhmirzaei e al. [197]	360	Predicting failure mode and shear capacity of UHPC beams	SVM, ANN, KNN, GP	Training (70%) Testing (30%) Validation (–) 20-fold cross-validation	Confusion matrix, ROC curve, R^2
Zarringol et al. [35]	2,686	Predicting the ultimate strength of CFST columns	ANN	Training (85%) Testing (15%) Validation (–)	MSEREG, R^2, MSE
Le [218]	314	Predicting the axial load of square concrete-filled steel tubular (CFST) columns	Gaussian Process Regression (GPR)	Training (-) Testing (–) Validation (-)	R, MAPE, MAE, RMSE
Nguyen et al. [219]	99	Predicting axial strength of concrete filled in steel tubes	invasive weed optimization (IWO), ANN	Training (60%) Testing (40%) Validation (–)	RMSE, MAE, R^2
Ly et al. [195]	463	Predicting the ultimate shear capacities of concrete beams reinforced with steel fiber	ANN, GA, Firefly algorithm	Training (70%) Testing (30%) Validation (–)	RMSE, MAE, R
Razavi et al. [225]	6	Predicting first crack of CFRP strengthened RC one-way slabs	General regression neural network (GRNN)	Training (85%) Testing (15%) Validation (–)	MSE, RMSE
Degtyarev [217]	3,512	Predicting the elastic shear buckling loads and the ultimate shear strengths of the channels with slotted webs	ANN	tenfold cross-validation	MSE, MAE, MAPE), R^2
Naser [89]	12,000	Predicting sectional capacity of FRP-strengthened members	ANN, GA	Training (70%) Testing (30%) Validation (–)	MAE, R, R^2
Fathi et al. [110]	70	Predicting the modulus of elasticity (MOE) and modulus of rupture (MOR) of wood with varying moisture contents (MC)	Group method of data handling (GMDH)	Training (75%) Testing (25%) Validation (–)	R^2

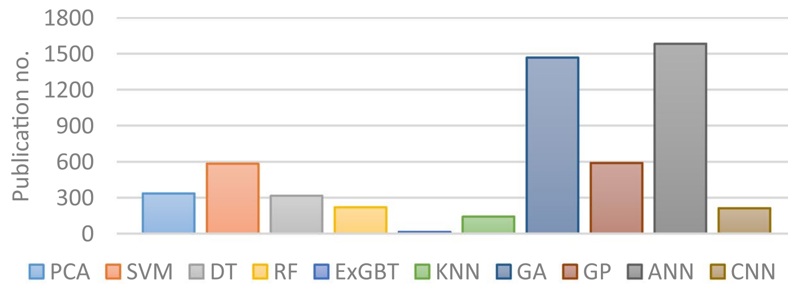
Table 6 (continued)

Study	Dataset	Objective	Algorithm(s)	Training procedure (%)	Performance metrics
Lee and Lee[199]	106	Predicting shear strength of slender fiber reinforced polymer (FRP) reinforced concrete flexural members without stirrups	ANN	Training (73%) Testing (27%) Validation (–)	Coefficient of variation (COV), RMSE, R^2
Naderpour [207]	177	Predicting shear resistance of concrete beams	ANN	Training (60%) Testing (20%) Validation (20%)	MSE, R^2
Abuodeh et al. [200]	120	Study the behavior of shear-deficient reinforced concrete (RC) beams strengthened in shear with side-bonded and U-wrapped fiber-reinforced polymers (FRP) laminates	RBPNN, Recursive feature elimination (RFE)	Training (70%) Testing (15%) Validation (15%)	RMSE, R^2
Su et al. [208]	122 + 136	Establishing correlation between influencing variables and the interfacial bond strength and then to predict the IBS	MLR, SVR, ANN	Training (80%) Testing (20%) Validation (–) tenfold validation results	RMSE, MAE, mean relative error (MRE), R^2
Köroğlu [209]	408	Predicting the bond strength of FRP bars in concrete	ANN	Training (85%) Testing (15%) Validation (–)	RMSE, R^2
Naderpour et al. [226]	150	Predicting the bond strength	ANFIS	Training (80%) Testing (20%) Validation (–)	RMSE, MAE, R^2
Naderpour et al. [204]	110	Extracting a new equation to predict the shear strength of concrete beams reinforced with FRP bars	ANN	Training (70%) Testing (20%) Validation (20%)	MAE, RMSE, MSE
Ma et al. [211]	102 + 68	Simulating the FRP-repaired concrete subjected to pre-damaged loading	ANN	Training (70%) Testing (15%) Validation (15%)	R^2
Mansouri et al. [205]	1,153	Predicting ultimate conditions of fiber-reinforced polymer (FRP)-confined concrete	ANN, ANFIS, multivariate adaptive regression splines (MARS), M5 Model Tree (M5Tree)	Training (60%) Testing (20%) Validation (20%)	RMSE, and average absolute error (AAE), MARE
Vu and Hoang [212]	82	Predicting the ultimate punching shear capacity of FRP- reinforced slabs	least squares support vector machine (LS-SVM), FA (firefly algorithm)	Training (90%) Testing (10%) Validation (–) tenfold cross-validation	RMSE, MAPE, R^2
Nguyen et al. [206]	131	Predicting the 28-day compressive strength of fiber-reinforced high-strength self-compacting concretes	ANN, ANFIS	Training (70%) Testing (15%) Validation (15%)	R^2, MSE, RMSE, and a20 – index
Feng and Fu [213]	86	Predicting the shear strength of RC beam to column connections	Gradient Boosting Regression Tree (GBRT)	Training (80%) Testing (20%) Validation (–) 10- fold cross-validation	R2, RMSE, MAE

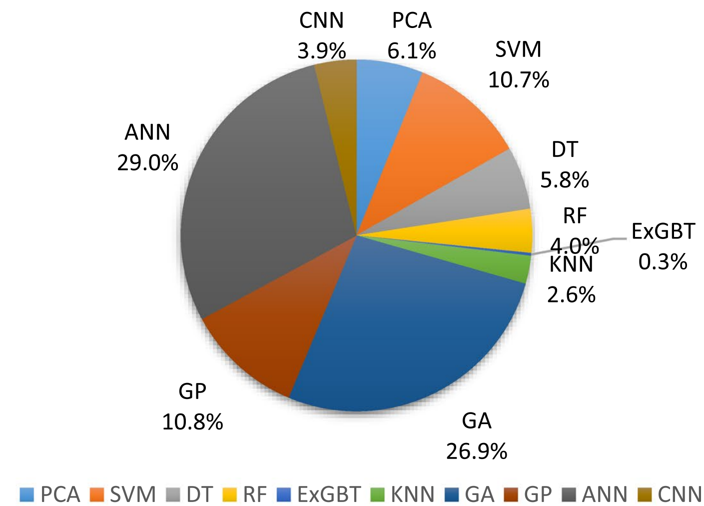
Table 6 (continued)

Study	Dataset	Objective	Algorithm(s)	Training procedure (%)	Performance metrics
Allahyari et al. [214]	90	Predicting the shear strength of Perfobond rib shear connector in steel–concrete composite structures	ANN	Training (85%) Testing (15%) Validation (–)	Normalized mean square of errors (NMSE), R
Mirrashid [215]	149	Predicting the shear strength of none-ductile RC joints	ANFIS, GMDH, ANN	Training (85%) Testing (15%) Validation (–)	R^2, MAE, RMSE
Yaseen et al. [227]	98	Predicting the joint shear behavior of beam to column structures	GA, DNN	Training (–) Testing (–) Validation (–)	MAE, RMSE, mean relative error (MRE), R^2
Alwanas et al. [216]	153	Predicting behavior of beam to column connections	ELM, MARS	Training (80%) Testing (20%) Validation (–)	Scatter index (SI), MAPE, RMSE, MAE, root mean square relative error (RMSRE), MRE, BIAS
Shariati et al. [222]	1,010 + 2,896	Predicting the behavior of channel shear connectors	ANN, ANFIS, ELM	Training (70%) Testing (30%) Validation (–)	RMSE, R, R^2
Chen et al. [193]	139	Predicting the shear strength of Squat RC walls	ANN, PSO	Training (80%) Testing (20%) Validation (–)	R^2, relative root mean square error (RRMSE), MAPE
Kotsovou et al. [224]	150	Predicting the behavior of RC exterior beam to column connections	ANN	Training (60%) Testing (20%) Validation (20%)	MSE
Luo and Paal [228]	262	Developing design curves for flexure- and shear-critical columns	Grid search algorithm	Training (90%) Testing (10%) Validation (–) tenfold cross-validation	RMSE, R^2
Bai et al. [198]	120	Assessing deflection in RC beams	ANN, ANFIS, SVM	Training (80%) Testing (20%) Validation (–) tenfold cross-validation	RMSE, R^2, RMSE, VAF, and MAPE,
Sujith Mangalathua [68]	536	Predicting the shear strength of beam to column joints	LR, Lasso, Discriminant analysis, Naïve Bayes (NB), DT, Extreme learning machines (ELM), Stepwise regression (SR)	Training (70%) Testing (30%) Validation (–)	MSE, absolute error, R^2

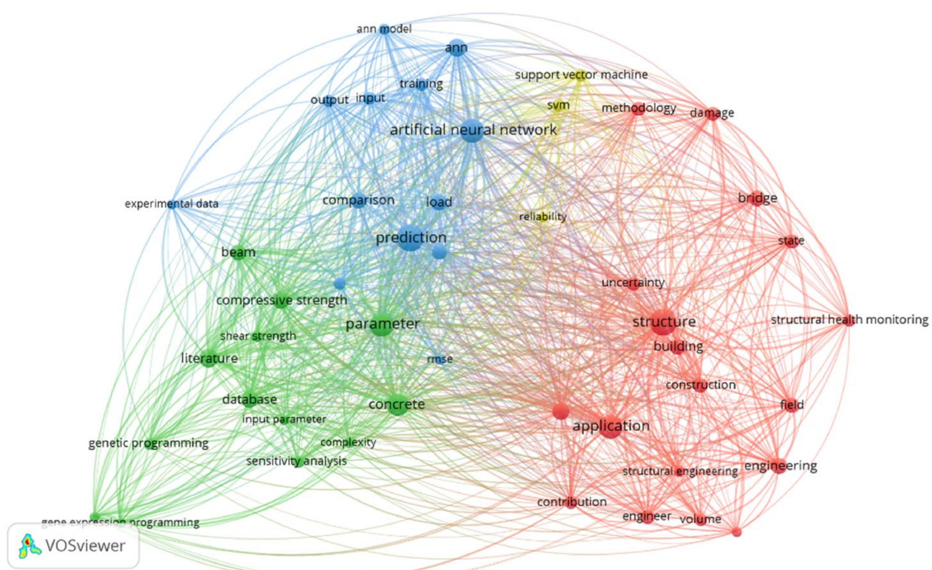
Fig. 22 Insights into most frequently used algorithms in structural engineering between (2011–2020)



(a) No. of publications per surveyed algorithm

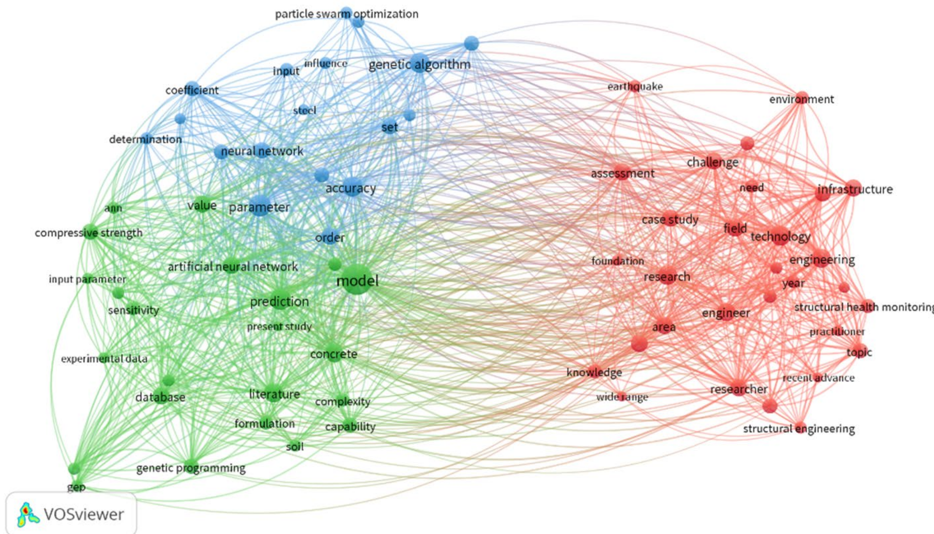


(b) Breakout of used algorithms



(c) ANN

Fig. 22 (continued)



(d) GA

6.2 Frequently Applied Model Development Procedures

The result of our analysis shows that the majority of scholarly works seem to primarily adopt one of two model development procedures: train/test splits or *k*-fold cross-validation (see Fig. 23). Historically, earlier works applied train/test splits where a database is split into two subsets. A larger subset of about 85% of reviewed works applied splits ranging between 70 and 80% of the original database size in training the ML model, and the smaller set was used to validate and test the predictivity of the model. Recent trends are moving towards applying a variant of *k*-fold cross validation on a more regular basis due to its inherent benefits over the train/test splitage.

Recent works properly articulate the need for data scaling, or normalization. However, we have not seen a consist approach to such and it appears it follows the researchers' preferences. An interesting observation was that the bulk of the reviewed works did not report applying feature cleansing, feature engineering or feature selection techniques and only presented the final inputs used for analysis. It is not clear if feature selection techniques were applied beforehand but never reported nor if researchers naturally relied on domain expertise to identify the features of importance—which seems to be a common denominator. In parallel, feature selection techniques were noted in the works that applied PCA or data reduction pre-analysis, such as those related to structural health monitoring problems. Future works are advised to carry out data analytics (database health examination) to ensure the generalizability of data points and subsequent ML models developed through such data points. Finally, future works are also invited to investigate the influence of algorithms tweaking options such as hyperparameter tuning [229] and algorithm architecture [230].

6.3 Frequently Used Size of Datasets

Our analysis indicates that there was considerable variability in the size of databases used in the reviewed works herein (see Fig. 24). For studies with datasets with less than 1000 point, the average dataset size was around 247 points. In general, datasets used in DNN and computer vision problems tend to have significantly larger data points. Still, the size of a utilized database in a particular study was merely discussed from a data-quality or -quantity points of view, but rather was primarily disclosed.[3] Most works examined the utilized dataset via basic statistical treatments such as a correlation analysis/matrix or via frequency plots. In most instances, information with regard to data range (i.e., max, min, average) and distribution were provided, however, only in a small number of instances, additional information with regard to kurtosis or skewness of data were provided. It is not clear how questions such as, does a used dataset include most examples of the combinations of in the search space a study is targeting? Does a dataset contain biased or imbalanced data? We hope to see answers to such questions, as well as others, in future works.

[3] Some works reported a practice of eliminating data points with up to a certain degree of deviation from the global trend of data [91].

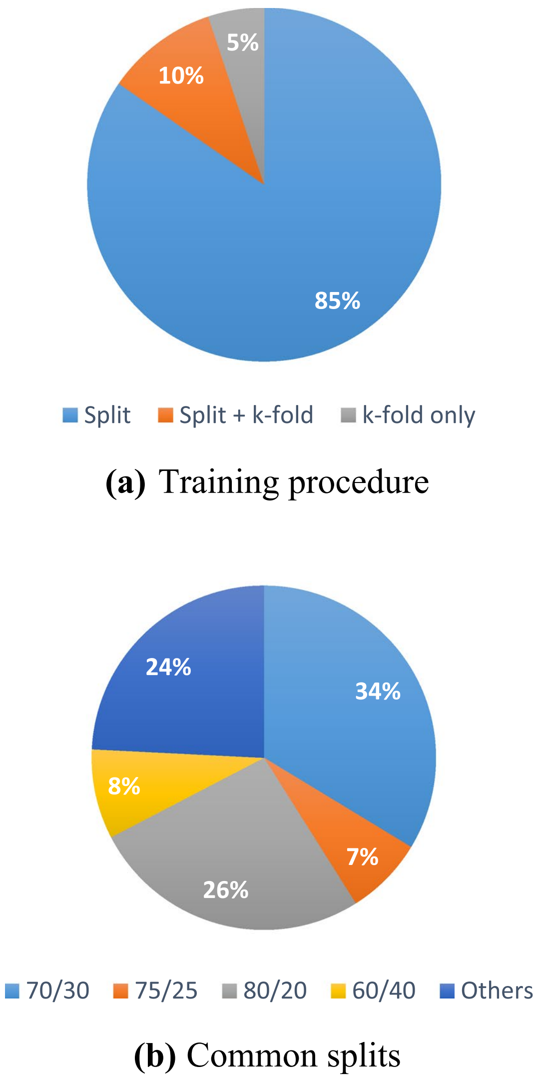


Fig. 23 Insights into most frequently ML model development procedures in structural engineering between (2011–2020)

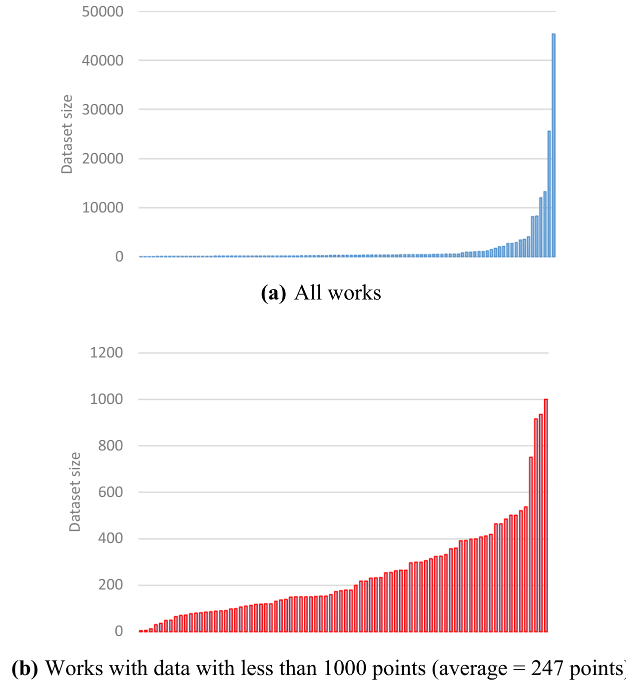


Fig. 24 Insights into most frequently used size of datasets in structural engineering between (2011–2020)

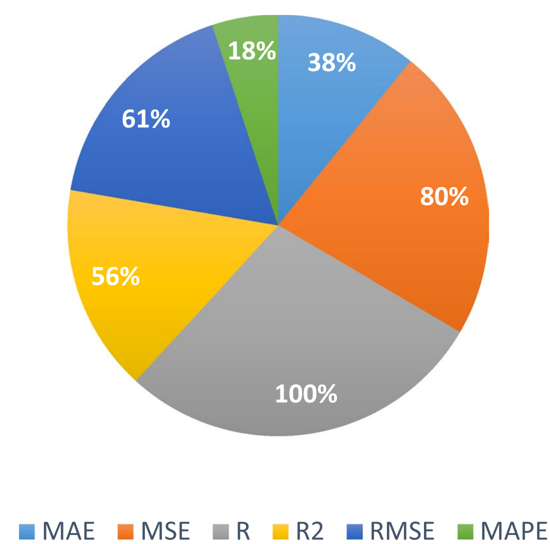


Fig. 25 Insights into most frequently used performance metrics in ML models in structural engineering between (2011–2020)

6.4 Frequently Employed Performance Metrics

The five key metrics were found to be MAE, MSE, R, R^2, RMSE, and MAPE with R being the most used metric followed by MSE and RMSE—see Fig. 25. We note that reviewed works favored the use of traditional performance metrics (i.e., R, R^2, RMSE, etc.). This is understandable, especially since, as shown in Sec. 5, the use of ML is tightly linked to regression problems (e.g., prediction of properties of sectional capacities). We also speculate the inherent familiarity of structural engineers with such metrics, which are commonly used in experimental tests. In some instances, reviewed researchers created a problem-specific metric or objective functions. We find such efforts to be of value as they go beyond the application of traditional metrics into more so of "performance-based" metrics. In classification problems, prominently used metrics were those related to the confusion matrix.

We would also like to point out that while a good number of works applied a sole algorithm to tackle a problem. In more recent efforts, researchers seem to favor adopting two or more algorithms (whether individually or by means of ensembles). In the case of the latter, then traditional performance metrics can be applied; on the other hand, additional performance metrics that are primarily designed to compare ML models can be used (however, rarely reported). Such metrics include; Wilcoxon signed-rank test [231, 232], 5 × 2CV paired t-test [233] and McNemar's Test [234]. In the meantime, the use of those metrics does not seem to find a reoccurring home in the reviewed works as the performance of competing models continues to be examined through traditional metrics. Finally, performance metrics can be applied globally (whole dataset), as well as regionally (on training, validation, testing subsets) and locally (especially near data regions with extreme values). A comprehensive application of metrics throughout the whole data range can be helpful in understanding the predictivity of a model.

6.5 Where to Go from Here?

The domains of AI, ML, and DL continue to expand with new technologies being added on a regular and more frequent basis. This implies that we can expect such advancements to reach the domain of structural engineering in the years, if not months, to come. Perhaps we can work to facilitate a smooth and more *accessible transfer* of such technologies to enable a transition towards modern structural engineering that could fully harness the potential of AI-based methods [235].

To fully harness this potential, research efforts are invited to join hands with industry partners. As the demand for faster, safer, and more intelligent solutions rises to match future structures, we expect to see a shift towards adopting AI-based methods. This also brings in the dimension of *education*, as future structural engineers will need to learn how to utilize AI, ML, and DL, as well as future technologies. Given their dense curriculum, innovative short works or seminar-like workshops can come in handy to help bridge this gap. In reality, such efforts can even start at the early stages of education [236].

Noting the results of our survey as discussed in this section, we did not identify a standard procedure to select, develop, deploy, or examine AI-tools [237]. In practical scenarios, AI-based methods, when used in the field of structural engineering for design/practical scenarios, are expected to be *rigorously vetted* in a similar manner to that of commonly adopted codal provisions to limit bias, overfitting, and ensure reliability as well as consistency; to name a few. Perhaps this would be a good time to start to formulate task groups/committees that can lead this effort.

As we move towards a more AI-adopting structural engineering, a need for *transparent* and *reproducible* AI solutions that break the notion of the blackbox will be on the rise. Herein, resolutions such as open access databases, code sharing, whitebox AI models, and citizen scientists can be of merit and can facilitate *trust* between structural engineers themselves, as well as with AI tools [238]. One way to establish reproducibility is by adopting *benchmarked* and well validated databases and case studies.

One of the rarely discussed topics is the use of properly designed *visualizations* to illustrate the outcome of AI analyses [239]. For example, the outcome of most regression models can be integrated with error bars, bounds, and confidence intervals that can visually illustrate the suitability of model predictions. Equivalent tools can also be supplemented in classification-based models. Noting the various visualization options available as packages in different programming languages, AI users may benefit from supplementing their works with such useful tools to further disseminate and enhance the delivery of their works.

A distinction should be drawn wherein some problems may necessitate embracing a "*chased accuracy*" mindset while tackling a phenomenon (where a model is heavily pushed to attain high performance metrics), as opposed to when a model is to be used to try to pinpoint the underlying mechanics or hidden patterns governing a phenomenon. In the former, such as in optimization-like problems, small improvement in model performance can be displayed via comparing up to 3–7 significant figures against earlier works, or measurements. In the latter, attaining close to unity performance metrics may not be necessary, especially if model interpretation can be beneficial or helpful to guiding researchers in exploring new dimensions of a phenomenon, as opposed to attempting to solve a phenomenon [240]. The reader is to be cognizant that performance metrics reflect a model's performance upon the available dataset used in developing such model—which may or may not be a reflection on the actual underlying mechanics in the real world. In a way, there is a good room to explore AI in different setting and problems.

The above notes some of the key observations that arose during our survey, and we invite future works to extend this survey further and to explore other dimensions and scholarly databases that were not present herein—such as that with regard to implementing unsupervised [241] and reinforcement learning [242]. A viable look will be towards causal AI [243, 244] and those pertaining to the role of AI in the education of civil engineers [245].

7 Conclusions

This paper presents a scientometrics review of artificial intelligence, machine learning, and deep learning with particular attention to structural engineering. This review starts by introducing big ideas within AI, ML, and DL in terms of its commonly used algorithms and techniques. Then, this review maps the latest knowledge within this domain by examining works published within the last ten years. Special attention is given to the application of AI, ML, and DL in earthquake, wind, and fire engineering, as well as structural health monitoring, damage detection, and prediction of properties of structural materials as collected from over 4000 sources. The following list of inferences can be drawn from this review:

- The past decade sets the stage for more eminent adoption of AI, ML, and DL in structural engineering, as noted by the significant rise in publications.
- Collectively, ANN, GA, GP, and SVM were used more frequently than other algorithms. ANN and GA have the lion share with about 55.9% of the time.
- 85% of reviewed works seem to favor adopting a split-based training procedure wherein the dataset is unequally split into a training set and a testing set. This model development procedure was then followed by a k-fold training procedure.
- The open literature shows a large variation in the size of used datasets. While the majority or works adopted datasets in the vicinity of 100–300 datapoints, others have reported the use of data points ranging between exceeding 10,000 points.
- Commonly used performance metrics were found to be R, MSE and RMSE. It is worth noting that some studies incorporated composite and exotic metrics that combines traditional metrics into ne metrics.
- Arising challenges such as; the need for AI education, transparency, reproducibility, and benchmarking databases and methods can be overcome in the coming years via collective/domain efforts.

Declarations

Conflict of interest The authors declare no conflict of interest.

References

1. Ziegel ER (2003) The elements of statistical learning. Technometrics. https://doi.org/10.1198/tech.2003.s770
2. Russell S, Norvig P (2010) Artificial intelligence a modern approach, 3rd edn. Pearson Education Inc, London
3. Anderson J, Smith A (2014) AI, robotics, and the future of jobs. Technol Rev 6:51
4. Zhou G, Zhang C, Li Z, Ding K, Wang C (2020) Knowledge-driven digital twin manufacturing cell towards intelligent manufacturing. Int J Prod Res. https://doi.org/10.1080/00207543.2019.1607978
5. Randhawa GS, Soltysiak MPM, El Roz H, de Souza CPE, Hill KA, Kari L (2020) Machine learning using intrinsic genomic signatures for rapid classification of novel pathogens: COVID-19 case study. PLoS ONE. https://doi.org/10.1371/journal.pone.0232391
6. Naser MZ (2021) Mechanistically informed machine learning and artificial intelligence in fire engineering and sciences. Fire Technol 57:1–44. https://doi.org/10.1007/s10694-020-01069-8
7. Mills H, Treagust D (2003) Engineering education. Is problem-based or project-based learning the answer? Australas J Eng Educ 3:2–16
8. Stevens R, O'connor K, Garrison L, Jocuns A, Amos DM (2008) Becoming an engineer: toward a three dimensional view of engineering learning. J Eng Educ. https://doi.org/10.1002/j.2168-9830.2008.tb00984.x
9. CSI, SAP2000 (2016) Analysis reference manual. Comput. Struct. INC. CSI Berkeley
10. Behnood A, Golafshani EM (2020) Machine learning study of the mechanical properties of concretes containing waste foundry sand. Constr Build Mater. https://doi.org/10.1016/j.conbuildmat.2020.118152
11. Zhu XQ, Law SS (2015) Structural health monitoring based on vehicle-bridge interaction: accomplishments and challenges. Adv Struct Eng. https://doi.org/10.1260/1369-4332.18.12.1999
12. Naser MZ (2020) Enabling cognitive and autonomous infrastructure in extreme events through computer vision. Innov Infrastruct Solut 5:99. https://doi.org/10.1007/s41062-020-00351-6
13. Sun H, Burton HV, Huang H (2021) Machine learning applications for building structural design and performance assessment: state-of-the-art review. J Build Eng 33:101816. https://doi.org/10.1016/j.jobe.2020.101816
14. Xie Y, Ebad-Sichani M, Padgett JE, DesRoches R (2020) The promise of implementing machine learning in earthquake engineering: a state-of-the-art review. Earthq Spectra. https://doi.org/10.1177/8755293020919419
15. Naser MZ (2020) Autonomous fire resistance evaluation. ASCE J Struct Eng 146:04020103. https://doi.org/10.1061/(ASCE)ST.1943-541X.0002641
16. Somvanshi M, Chavan P, Tambade S, Shinde SV (2017) A review of machine learning techniques using decision tree and support vector machine. In: Proc - 2nd Int Conf Comput Commun Control Autom. ICCUBEA 2016. https://doi.org/10.1109/ICCUBEA.2016.7860040
17. Salehi H, Burgueño R (2018) Emerging artificial intelligence methods in structural engineering. Elsevier, Amsterdam
18. Lu P, Chen S, Zheng Y (2012) Artificial intelligence in civil engineering. Math Probl Eng 2012:1–23. https://doi.org/10.1155/2012/145974
19. Ullah Z, Al-Turjman F, Mostarda L, Gagliardi R (2020) Applications of artificial intelligence and machine learning in smart cities. Comput Commun. https://doi.org/10.1016/j.comcom.2020.02.069
20. Shukla H, Piratla K (2020) Leakage detection in water pipelines using supervised classification of acceleration signals. Autom Constr. https://doi.org/10.1016/j.autcon.2020.103256
21. Tran-Ngoc H, Khatir S, Le-Xuan T, De Roeck G, Bui-Tien T, Abdel Wahab M (2020) A novel machine-learning based on the global search techniques using vectorized data for damage detection in structures. Int J Eng Sci 157:103376. https://doi.org/10.1016/j.ijengsci.2020.103376
22. Naser MZ, Zhou H (2021) Machine learning to derive unified material models for steel under fire conditions. In: Deo RC, Samui P, Kisi O, Yaseen ZM (eds) Intelligent data analytics for decision-support systems in hazard mitigation: theory and practice of hazard mitigation. Singapore, Springer, pp 213–225
23. Naser MZZ, Thai S, Thai H-THT (2021) Evaluating structural response of concrete-filled steel tubular columns through machine learning. J. Build. Eng. 34:101888. https://doi.org/10.1016/j.jobe.2020.101888
24. Flood I (1989) A neural network approach to the sequencing of construction tasks. In Proceedings of the 6th international

symposium on automation and robotics in construction. https://doi.org/10.22260/isarc1989/0026
25. Vanluchene RD, Sun R (1990) Neural networks in structural engineering. Comput Civ Infrastruct Eng. https://doi.org/10.1111/j.1467-8667.1990.tb00377.x
26. Adeli H (2001) Neural networks in civil engineering: 1989–2000. Comput Civ Infrastruct Eng 16:126–142. https://doi.org/10.1111/0885-9507.00219
27. Zhang Q, Barri K, Jiao P, Salehi H, Alavi AH (2020) Genetic programming in civil engineering: advent, applications and future trends. Artif Intell Rev. https://doi.org/10.1007/s10462-020-09894-7
28. Mirrashid M, Naderpour H (2020) Recent trends in prediction of concrete elements behavior using soft computing (2010–2020). Arch Comput Methods Eng. https://doi.org/10.1007/s11831-020-09500-7
29. Penadés-Plà V, García-Segura T, Martí JV, Yepes V (2016) A review of multi-criteria decision-making methods applied to the sustainable bridge design. Sustainability 8:1295. https://doi.org/10.3390/su8121295
30. Aldwaik M, Adeli H (2014) Advances in optimization of high-rise building structures. Struct Multidiscip Optim 50:899–919. https://doi.org/10.1007/s00158-014-1148-1
31. Burnwal AP, Das SK, Kumar A, Das B, Burnwal B (2013) On soft computing techniques in various areas. Comput Sci Inf Technol 3:59–68. https://doi.org/10.5121/csit.2013.3206
32. Esmin AAA, Lambert-Torres G, Alvarenga GB (2006) Hybrid evolutionary algorithm based on PSO and GA mutation. In: Proceedings - Sixth Int Conf Hybrid Intell Syst Fourth Conf Neuro-Computing Evol Intell HIS-NCEI 2006 57. https://doi.org/10.1109/HIS.2006.264940
33. Magdalena L (2010) What is soft computing? Revisiting possible answers. Int J Comput Intell Syst 3:148–159. https://doi.org/10.1080/18756891.2010.9727686
34. Bohlin TP (2013) Practical grey-box process identification: theory and applications. Springer, New York
35. Zarringol M, Thai HT, Thai S, Patel V (2020) Application of ANN to the design of CFST columns. Structures. https://doi.org/10.1016/j.istruc.2020.10.048
36. Bishop C (2007) Pattern recognition and machine learning. Technometrics. https://doi.org/10.1198/tech.2007.s518
37. Murphy KP (2012) Machine learning: a probabilistic perspective (adaptive computation and machine learning series)
38. Naser MZZ (2021) Observational analysis of fire-induced spalling of concrete through ensemble machine learning and surrogate modeling. J Mater Civ Eng 33:04020428. https://doi.org/10.1061/(ASCE)MT.1943-5533.0003525
39. Caruana R, Niculescu-Mizil A (2006) An empirical comparison of supervised learning algorithms. In: ACM Int Conf Proceeding Ser. ACM Press, New York, pp 161–168. https://doi.org/10.1145/1143844.1143865
40. Iguyon I, Elisseeff A (2003) An introduction to variable and feature selection. J Mach Learn Res. https://doi.org/10.1162/153244303322753616
41. Zheng A, Casari A (2018) Feature engineering for machine learning: principles and techniques for data scientists. O'Reilly Media Inc, Newton
42. Unglert K, Radić V, Jellinek AM (2016) Principal component analysis vs. self-organizing maps combined with hierarchical clustering for pattern recognition in volcano seismic spectra. J Volcanol Geotherm Res 320:58–74. https://doi.org/10.1016/J.JVOLGEORES.2016.04.014
43. Mousavi SM, Aminian P, Gandomi AH, Alavi AH, Bolandi H (2012) A new predictive model for compressive strength of HPC using gene expression programming. Adv Eng Softw. https://doi.org/10.1016/j.advengsoft.2011.09.014
44. Farrar CR, Worden K (2012) Structural health monitoring: a machine learning perspective. Wiley, Hoboken
45. Pan L, Novák L, Lehký D, Novák D, Cao M (2021) Neural network ensemble-based sensitivity analysis in structural engineering: comparison of selected methods and the influence of statistical correlation. Comput Struct. https://doi.org/10.1016/j.compstruc.2020.106376
46. Kodur VK, Naser MZ (2021) Classifying bridges for the risk of fire hazard via competitive machine learning. Adv Bridg Eng. https://doi.org/10.1186/s43251-020-00027-2
47. Anguita D, Ghelardoni L, Ghio A, Oneto L, Ridella S (2012) The 'K' in K-fold cross validation. In: ESANN 2012 Proceedings, 20th Eur Symp Artif Neural Networks Comput. Intell Mach Learn
48. Das SK (2013) Artificial neural networks in geotechnical engineering: modeling and application issues. In: Yang XS, Gandomi AH, Talatahari S, Alavi AH (eds) Metaheuristics in water, geotechnical and transport engineering. Elsevier, Amsterdam
49. Naser M, Abu-Lebdeh G, Hawileh R (2012) Analysis of RC T-beams strengthened with CFRP plates under fire loading using ANN. Constr Build Mater 37:301–309. https://doi.org/10.1016/j.conbuildmat.2012.07.001
50. Babanajad SK, Gandomi AH, Alavi AH (2017) New prediction models for concrete ultimate strength under true-triaxial stress states: an evolutionary approach. Adv Eng Softw. https://doi.org/10.1016/j.advengsoft.2017.03.011
51. Naser M, Alavi A (2020) Insights into performance fitness and error metrics for machine learning. Under Rev
52. Botchkarev A (2019) A new typology design of performance metrics to measure errors in machine learning regression algorithms. Interdiscip J Inf Knowl Manag 14:045–076. https://doi.org/10.28945/4184
53. Seyedzadeh S, Pour Rahimian F, Rastogi P, Glesk I (2019) Tuning machine learning models for prediction of building energy loads. Sustain Cities Soc 47:101484. https://doi.org/10.1016/j.scs.2019.101484
54. Jollife IT, Cadima J (2016) Principal component analysis: A review and recent developments. Philos Trans R Soc A Math Phys Eng Sci 374:2065. https://doi.org/10.1098/rsta.2015.0202
55. Li J, Dackermann U, Xu Y-L, Samali B (2011) Damage identification in civil engineering structures utilizing PCA-compressed residual frequency response functions and neural network ensembles. Struct Control Heal Monit 18:207–226. https://doi.org/10.1002/stc.369
56. Scikit, sklearn.decomposition.PCA—scikit-learn 0.24.1 documentation (n.d.) https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html. Accessed 5 Apr 2021
57. Boser BE, Guyon IM, Vapnik VN (1992) Training algorithm for optimal margin classifiers. In: Proc Fifth Annu ACM Work Comput Learn Theory. https://doi.org/10.1145/130385.130401
58. Çevik A, Kurtoğlu AE, Bilgehan M, Gülşan ME, Albegmprli HM (2015) Support vector machines in structural engineering: a review. J Civ Eng Manag 21:261. https://doi.org/10.3846/13923730.2015.1005021
59. Cherkassky V, Ma Y (2004) Practical selection of SVM parameters and noise estimation for SVM regression. Neural Netw. https://doi.org/10.1016/S0893-6080(03)00169-2
60. Scikit 1.4. Support Vector Machines—scikit-learn 0.24.1 documentation (n.d.) https://scikit-learn.org/stable/modules/svm.html. Accessed 5 Apr 2021
61. Rokach L, Maimon O (2005) Top-down induction of decision trees classifiers: a survey. IEEE Trans Syst Man Cybern Part C Appl Rev 35:476. https://doi.org/10.1109/TSMCC.2004.843247
62. Huang H, Burton HV (2019) Classification of in-plane failure modes for reinforced concrete frames with infills using machine

learning. J Build Eng. https://doi.org/10.1016/j.jobe.2019.100767
63. Scikit sklearn.ensemble.RandomForestClassifier—scikit-learn 0.24.1 documentation (2020). https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html . Accessed 9 Feb 2021
64. Freund Y, Schapire RE (1997) A decision-theoretic generalization of on-line learning and an application to boosting. J Comput Syst Sci. https://doi.org/10.1006/jcss.1997.1504
65. Scikit sklearn.ensemble.GradientBoostingRegressor — scikit-learn 0.24.1 documentation (2020) https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingRegressor.html. accessed 9 Feb 2021
66. XGBoost Python Package, Python Package Introduction—xgboost 1.4.0-SNAPSHOT documentation (2020) https://xgboost.readthedocs.io/en/latest/python/python_intro.html#early-stopping . Accessed 10 Feb 2021
67. Gradient boosted tree (GBT) (2019) https://software.intel.com/en-us/daal-programming-guide-details-24 . Accessed 9 Apr 2019
68. Mangalathu S, Jeon JS (2018) Classification of failure mode and prediction of shear strength for reinforced concrete beam-column joints using machine learning techniques. Eng Struct. https://doi.org/10.1016/j.engstruct.2018.01.008
69. Scikit sklearn.neighbors.NearestNeighbors—scikit-learn 0.24.1 documentation (2021) https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.NearestNeighbors.html?highlight=knearest#sklearn.neighbors.NearestNeighbors.kneighbors. Accessed 5 Apr 2021
70. Koza JR (1992) A genetic approach to finding a controller to back up a tractor-trailer truck. In: Proc 1992 Am Control Conf
71. Alavi AH, Gandomi AH, Sahab MG, Gandomi M (2010) Multi expression programming: a new approach to formulation of soil classification. Eng Comput 26:111–118. https://doi.org/10.1007/s00366-009-0140-7
72. Seitlllari A, Naser MZZ (2019) Leveraging artificial intelligence to assess explosive spalling in fire-exposed RC columns. Comput Concr 24:271–282. https://doi.org/10.12989/cac.2019.24.3.271
73. Willis M-J (2005) Genetic programming: an introduction and survey of applications. https://doi.org/10.1049/cp:19971199
74. Aslam F, Farooq F, Amin MN, Khan K, Waheed A, Akbar A, Javed MF, Alyousef R, Alabdulijabbar H (2020) Applications of gene expression programming for estimating compressive strength of high-strength concrete. Adv Civ Eng 2020:1–23. https://doi.org/10.1155/2020/8850535
75. Langdon WB (2020) Big data driven genetic improvement for maintenance of legacy software systems. ACM SIGEVOlution. https://doi.org/10.1145/3381343.3381345
76. Ferreira C (2002) Gene expression programming in problem solving. Soft Comput Ind. https://doi.org/10.1007/978-1-4471-0123-9_54
77. McCulloch WS, Pitts W (1943) A logical calculus of the ideas immanent in nervous activity. Bull Math Biophys. https://doi.org/10.1007/BF02478259
78. Cobaner M, Unal B, Kisi O (2009) Suspended sediment concentration estimation by an adaptive neuro-fuzzy and neural network approaches using hydro-meteorological data. J Hydrol 367:52–61. https://doi.org/10.1016/J.JHYDROL.2008.12.024
79. Dongmei H, Shiqing H, Xuhui H, Xue Z (2017) Prediction of wind loads on high-rise building using a BP neural network combined with POD. J Wind Eng Ind Aerodyn 170:1–17. https://doi.org/10.1016/j.jweia.2017.07.021
80. Scikit 1.17. Neural network models (supervised)—scikit-learn 0.24.1 documentation (2021) https://scikit-learn.org/stable/modules/neural_networks_supervised.html . Accessed 5 Apr 2021
81. Cha YJ, Choi W, Suh G, Mahmoudkhani S, Büyüköztürk O (2018) Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types. Comput Civ Infrastruct Eng 33:731–747. https://doi.org/10.1111/mice.12334
82. Cha YJ, Choi W, Büyüköztürk O (2017) Deep learning-based crack damage detection using convolutional neural networks. Comput Civ Infrastruct Eng. https://doi.org/10.1111/mice.12263
83. Cireşan DC, Meier U, Masci J, Gambardella LM, Schmidhuber J (2011) Flexible, high performance convolutional neural networks for image classification. IJCAI Int Jt Conf Artif Intell. https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-210
84. van Eck NJ, Waltman L (2010) Software survey: VOSviewer, a computer program for bibliometric mapping. Scientometrics 84:523–538. https://doi.org/10.1007/s11192-009-0146-3
85. Cioffi R, Travaglioni M, Piscitelli G, Petrillo A, De Felice F (2020) Artificial intelligence and machine learning applications in smart production: progress, trends, and directions. Sustainability 12:492. https://doi.org/10.3390/su12020492
86. Dimensions, Dimensions.ai (2021) https://www.dimensions.ai/
87. Thelwall M (2018) Dimensions: a competitor to scopus and the web of science? J Informetr. https://doi.org/10.1016/j.joi.2018.03.006
88. Gao J, Koopialipoor M, Armaghani DJ, Ghabussi A, Baharom S, Morasaei A, Shariati A, Khorami M, Zhou J (2020) Evaluating the bond strength of FRP in concrete samples using machine learning methods. Smart Struct Syst. https://doi.org/10.12989/sss.2020.26.4.403
89. Naser MZ (2020) Machine learning assessment of fiber-reinforced polymer-strengthened and reinforced concrete members. ACI Struct J. https://doi.org/10.14359/51728073
90. Marani A, Nehdi ML (2020) Machine learning prediction of compressive strength for phase change materials integrated cementitious composites. Constr Build Mater 265:120286. https://doi.org/10.1016/j.conbuildmat.2020.120286
91. Javed MF, Amin MN, Shah MI, Khan K, Iftikhar B, Farooq F, Aslam F, Alyousef R, Alabduljabbar H (2020) Applications of gene expression programming and regression techniques for estimating compressive strength of bagasse ash based concrete. Crystals 10:1–17. https://doi.org/10.3390/cryst10090737
92. Nguyen T, Kashani A, Ngo T, Bordas S (2019) Deep neural network with high-order neuron for the prediction of foamed concrete strength. Comput Civ Infrastruct Eng 34:316–332. https://doi.org/10.1111/mice.12422
93. Jalal M, Grasley Z, Gurganus C, Bullard JW (2020) Experimental investigation and comparative machine-learning prediction of strength behavior of optimized recycled rubber concrete. Constr Build Mater 256:119478. https://doi.org/10.1016/j.conbuildmat.2020.119478
94. Sultana N, Zakir Hossain SM, Alam MS, Islam MS, Al Abtah MA (2020) Soft computing approaches for comparative prediction of the mechanical properties of jute fiber reinforced concrete. Adv Eng Softw 149:102887. https://doi.org/10.1016/j.advengsoft.2020.102887
95. Castelli M, Vanneschi L, Silva S (2013) Prediction of high performance concrete strength using Genetic Programming with geometric semantic genetic operators. Expert Syst Appl. https://doi.org/10.1016/j.eswa.2013.06.037
96. Yaseen ZM, Deo RC, Hilal A, Abd AM, Bueno LC, Salcedo-Sanz S, Nehdi ML (2018) Predicting compressive strength of lightweight foamed concrete using extreme learning machine model. Adv Eng Softw. https://doi.org/10.1016/j.advengsoft.2017.09.004
97. Ben Seghier MEA, Ouaer H, Ghriga MA, Menad NA, Thai DK (2020) Hybrid soft computational approaches for modeling the maximum ultimate bond strength between the corroded steel

reinforcement and surrounding concrete. Neural Comput Appl 6:6905. https://doi.org/10.1007/s00521-020-05466-6
98. Gorphade VG, Rao HS, Beulah M (2014) Development of genetic algorithm based neural network model for predicting workability and strength of high performance concrete. Int J Invent Eng Sci 2:2319–9598
99. Naseri H, Jahanbakhsh H, Hosseini P, Moghadas Nejad F (2020) Designing sustainable concrete mixture by developing a new machine learning technique. J Clean Prod 258:120578. https://doi.org/10.1016/j.jclepro.2020.120578
100. Huang Y, Zhang J, Tze Ann F, Ma G (2020) Intelligent mixture design of steel fibre reinforced concrete using a support vector regression and firefly algorithm based multi-objective optimization model. Constr Build Mater 260:120457. https://doi.org/10.1016/j.conbuildmat.2020.120457
101. Golafshani EM, Ashour A (2016) Prediction of self-compacting concrete elastic modulus using two symbolic regression techniques. Autom Constr 64:7–19. https://doi.org/10.1016/j.autcon.2015.12.026
102. Okazaki Y, Okazaki S, Asamoto S, Jo-Chun P (2020) Applicability of machine learning to a crack model in concrete bridges. Comput Civ Infrastruct Eng 35:775. https://doi.org/10.1111/mice.12532
103. Kellouche Y, Boukhatem B, Ghrici M, Tagnit-Hamou A (2019) Exploring the major factors affecting fly-ash concrete carbonation using artificial neural network. Neural Comput Appl 31:969–988. https://doi.org/10.1007/s00521-017-3052-2
104. Salami BA, Rahman SM, Oyehan TA, Maslehuddin M, Al-Dulaijan SU (2020) Ensemble machine learning model for corrosion initiation time estimation of embedded steel reinforced self-compacting concrete. Meas J Int Meas Conf 165:108141. https://doi.org/10.1016/j.measurement.2020.108141
105. Ben Chaabene W, Flah M, Nehdi ML (2020) Machine learning prediction of mechanical properties of concrete: critical review. Constr Build Mater 260:119889. https://doi.org/10.1016/j.conbuildmat.2020.119889
106. Guo S, Yu J, Liu X, Wang C, Jiang Q (2019) A predicting model for properties of steel using the industrial big data based on machine learning. Comput Mater Sci. https://doi.org/10.1016/j.commatsci.2018.12.056
107. Chen CT, Gu GX (2019) Machine learning for composite materials. MRS Commun. https://doi.org/10.1557/mrc.2019.32
108. Wei J, Chu X, Sun X, Xu K, Deng H, Chen J, Wei Z, Lei M (2019) Machine learning in materials science. InfoMat. https://doi.org/10.1002/inf2.12028
109. Abdalla JA, Hawileh RA (2013) Artificial neural network predictions of fatigue life of steel bars based on hysteretic energy. J Comput Civ Eng. https://doi.org/10.1061/(asce)cp.1943-5487.0000185
110. Fathi H, Nasir V, Kazemirad S (2020) Prediction of the mechanical properties of wood using guided wave propagation and machine learning. Constr Build Mater 262:120848. https://doi.org/10.1016/j.conbuildmat.2020.120848
111. Bal L, Buyle-Bodin F (2014) Artificial neural network for predicting creep of concrete. Neural Comput Appl 25:1359–1367. https://doi.org/10.1007/s00521-014-1623-z
112. Ince R (2010) Artificial neural network-based analysis of effective crack model in concrete fracture. Fatigue Fract Eng Mater Struct 33:595–606. https://doi.org/10.1111/j.1460-2695.2010.01469.x
113. Ahmad A, Kotsovou G, Cotsovos DM, Lagaros ND (2018) Assessing the accuracy of RC design code predictions through the use of artificial neural networks. Int J Adv Struct Eng 10:349–365. https://doi.org/10.1007/s40091-018-0202-4
114. Kaloop MR, Kumar D, Samui P, Hu JW, Kim D (2020) Compressive strength prediction of high-performance concrete using gradient tree boosting machine. Constr Build Mater 264:120198. https://doi.org/10.1016/j.conbuildmat.2020.120198
115. Feng DC, Liu ZT, Wang XD, Jiang ZM, Liang SX (2020) Failure mode classification and bearing capacity prediction for reinforced concrete columns based on ensemble machine learning algorithm. Adv Eng Inform 45:101126. https://doi.org/10.1016/j.aei.2020.101126
116. Chou JS, Tsai CF, Pham AD, Lu YH (2014) Machine learning in concrete strength simulations: multi-nation data analytics. Constr Build Mater. https://doi.org/10.1016/j.conbuildmat.2014.09.054
117. Thanh Duong H, Chi Phan H, Le TT, Duc Bui N (2020) Optimization design of rectangular concrete-filled steel tube short columns with Balancing Composite Motion Optimization and data-driven model. Structures 28:757–765. https://doi.org/10.1016/j.istruc.2020.09.013
118. Yan Y, Ren Q, Xia N, Shen L, Gu J (2015) Artificial neural network approach to predict the fracture parameters of the size effect model for concrete. Fatigue Fract Eng Mater Struct 38:1347–1358. https://doi.org/10.1111/ffe.12309
119. Golafshani EM, Rahai A, Sebt MH, Akbarpour H (2012) Prediction of bond strength of spliced steel bars in concrete using artificial neural network and fuzzy logic. Constr Build Mater 36:411–418. https://doi.org/10.1016/j.conbuildmat.2012.04.046
120. Naik U, Kute S (2013) Span-to-depth ratio effect on shear strength of steel fiber-reinforced high-strength concrete deep beams using ANN model. Int J Adv Struct Eng 5:1–12. https://doi.org/10.1186/2008-6695-5-29
121. Hoang ND, Tran XL, Nguyen H (2020) Predicting ultimate bond strength of corroded reinforcement and surrounding concrete using a metaheuristic optimized least squares support vector regression model. Neural Comput Appl 32:7289–7309. https://doi.org/10.1007/s00521-019-04258-x
122. Akin OO, Abejide OS (2019) Modelling of concrete compressive strength admixed with GGBFS using gene expression programming. J Soft Comput Civ Eng 3:43–53
123. Khademi F, Akbari M, Jamal SM, Nikoo M (2017) Multiple linear regression, artificial neural network, and fuzzy logic prediction of 28 days compressive strength of concrete. Front Struct Civ Eng 11:90–99. https://doi.org/10.1007/s11709-016-0363-9
124. Qi P, He M, Li M, Zheng X, Li Z, Liu C, Zeng X, Tao D, Qi X, Ma Z (2020) Machine learning-based modeling for the duration of load effect in wood structural members under long-term sustained load. IEEE Access 8:17903–17915. https://doi.org/10.1109/ACCESS.2020.2966883
125. Abuodeh OR, Abdalla JA, Hawileh RA (2020) Assessment of compressive strength of Ultra-high Performance Concrete using deep machine learning techniques. Appl Soft Comput J. https://doi.org/10.1016/j.asoc.2020.106552
126. Abdalla JA, Hawileh RA (2021) Assessment of effect of strain amplitude and strain ratio on energy dissipation using machine learning. Lect Notes Civ Eng. https://doi.org/10.1007/978-3-030-51295-8_9
127. Asim KM, Martínez-Álvarez F, Basit A, Iqbal T (2017) Earthquake magnitude prediction in Hindukush region using machine learning techniques. Nat Hazards. https://doi.org/10.1007/s11069-016-2579-3
128. Arslan MH (2010) An evaluation of effective design parameters on earthquake performance of RC buildings using neural networks. Eng Struct 32:1888–1898. https://doi.org/10.1016/j.engstruct.2010.03.010
129. Mangalathu S, Burton HV (2019) Deep learning-based classification of earthquake-impacted buildings using textual damage descriptions. Int J Disaster Risk Reduct 36:101111. https://doi.org/10.1016/j.ijdrr.2019.101111

130. Zhang Y, Burton HV, Sun H, Shokrabadi M (2018) A machine learning framework for assessing post-earthquake structural safety. Struct Saf. https://doi.org/10.1016/j.strusafe.2017.12.001
131. Hwang SH, Mangalathu S, Shin J, Jeon JS (2020) Machine learning-based approaches for seismic demand and collapse of ductile reinforced concrete building frames. J Build Eng 34:101905. https://doi.org/10.1016/j.jobe.2020.101905
132. Morfidis K, Kostinakis K (2018) Approaches to the rapid seismic damage prediction of r/c buildings using artificial neural networks. Eng Struct 165:120–141. https://doi.org/10.1016/j.engstruct.2018.03.028
133. Luo H, Paal SG (2019) A locally weighted machine learning model for generalized prediction of drift capacity in seismic vulnerability assessments. Comput Civ Infrastruct Eng 34:935–950. https://doi.org/10.1111/mice.12456
134. Oh BK, Park Y, Park HS (2020) Seismic response prediction method for building structures using convolutional neural network. Struct Control Heal Monit 27:1–17. https://doi.org/10.1002/stc.2519
135. Asteris PG, Nikoo M (2019) Artificial bee colony-based neural network for the prediction of the fundamental period of infilled frame structures. Neural Comput Appl 31:4837–4847. https://doi.org/10.1007/s00521-018-03965-1
136. Su L, He HJ (2019) Decision tree–based seismic damage prediction for reinforcement concrete frame buildings considering structural micro-characteristics. Adv Struct Eng 22:2097–2109. https://doi.org/10.1177/1369433219832508
137. Liu Z, Zhang Z (2018) Artificial neural network based method for seismic fragility analysis of steel frames. KSCE J Civ Eng 22:708–717. https://doi.org/10.1007/s12205-017-1329-8
138. Kareem A (2020) Emerging frontiers in wind engineering: computing, stochastics, machine learning and beyond. J Wind Eng Ind Aerodyn. https://doi.org/10.1016/j.jweia.2020.104320
139. Hu G, Liu L, Tao D, Song J, Tse KT, Kwok KCS (2020) Deep learning-based investigation of wind pressures on tall building under interference effects. J Wind Eng Ind Aerodyn 201:104138. https://doi.org/10.1016/j.jweia.2020.104138
140. Payán-Serrano O, Bojórquez E, Bojórquez J, Chávez R, Reyes-Salazar A, Barraza M, López-Barraza A, Rodríguez-Lozoya H, Corona E (2017) Prediction of maximum story drift of MDOF structures under simulated wind loads using artificial neural networks. Appl Sci 7:563. https://doi.org/10.3390/app7060563
141. Nikose TJ, Sonparote RS (2020) Computing dynamic across-wind response of tall buildings using artificial neural network. J Supercomput 76:3788–3813. https://doi.org/10.1007/s11227-018-2708-8
142. Paul R, Dalui SK (2020) Prognosis of wind-tempted mean pressure coefficients of cross-shaped tall buildings using artificial neural network, period. Polytech Civ Eng 64:1124–1143. https://doi.org/10.3311/PPci.16311
143. Oh BK, Glisic B, Kim Y, Park HS (2019) Convolutional neural network-based wind-induced response estimation model for tall buildings. Comput Civ Infrastruct Eng 34:843–858. https://doi.org/10.1111/mice.12476
144. Gavalda X, Ferrer-Gener J, Kopp GA, Giralt F (2011) Interpolation of pressure coefficients for low-rise buildings of different plan dimensions and roof slopes using artificial neural networks. J Wind Eng Ind Aerodyn 99:658–664. https://doi.org/10.1016/j.jweia.2011.02.008
145. Bairagi AK, Dalui SK (2020) Forecasting of wind induced pressure on setback building using artificial neural network, period. Polytech Civ Eng 64:751–763. https://doi.org/10.3311/PPci.15769
146. Abbas T, Kavrakov I, Morgenthal G, Lahmer T (2020) Prediction of aeroelastic response of bridge decks using artificial neural networks. Comput Struct 231:106198. https://doi.org/10.1016/j.compstruc.2020.106198
147. Le V, Caracoglia L (2020) A neural network surrogate model for the performance assessment of a vertical structure subjected to non-stationary, tornadic wind loads. Comput Struct 231:106208. https://doi.org/10.1016/j.compstruc.2020.106208
148. Buchanan AH, Abu AK (2016) Fire safety in buildings. Taylor & Francis, Milton Park
149. Kodur V, Naser MZM (2020) Structural fire engineering, 1st edn. McGraw Hill Professional, New York
150. Dexters A, Leisted RR, Van Coile R, Welch S, Jomaas G (2019) Testing for knowledge: maximising information obtained from fire tests by using machine learning techniques. In: Interflam 2019. http://hdl.handle.net/1854/LU-8622485
151. Bilgehan M, Kurtoğlu AE (2016) ANFIS-based prediction of moment capacity of reinforced concrete slabs exposed to fire. Neural Comput Appl 27:869–881. https://doi.org/10.1007/s00521-015-1902-3
152. Fu F (2020) Fire induced progressive collapse potential assessment of steel framed buildings using machine learning. J Constr Steel Res. https://doi.org/10.1016/j.jcsr.2019.105918
153. Naser MZ (2019) Can past failures help identify vulnerable bridges to extreme events? A biomimetical machine learning approach. Eng Comput. https://doi.org/10.1007/s00366-019-00874-2
154. Panev Y, Kotsovinos P, Deeny S, Flint G (2021) The use of machine learning for the prediction of fire resistance of composite shallow floor systems. Fire Technol. https://doi.org/10.1007/s10694-021-01108-y
155. Lazarevska M, Gavriloska AT, Laban M, Knezevic M, Cvetkovska M (2018) Determination of fire resistance of eccentrically loaded reinforced concrete columns using fuzzy neural networks. Complexity 2018:1–12. https://doi.org/10.1155/2018/8204568
156. Ketabdari H, Saedi-Daryan A, Hassani N (2019) Predicting post-fire mechanical properties of grade 8.8 and 10.9 steel bolts. J Constr Steel Res 162:105735. https://doi.org/10.1016/j.jcsr.2019.105735
157. Lee JHJ, Lee JHJ, Cho BS (2012) Effective prediction of thermal conductivity of concrete using neural network method. Int J Concr Struct Mater 6:177–186. https://doi.org/10.1007/s40069-012-0016-x
158. Naser MZ, Uppala VA (2020) Properties and material models for construction materials post exposure to elevated temperatures. Mech Mater 142:103293. https://doi.org/10.1016/j.mechmat.2019.103293
159. Naser MZ (2019) Heuristic machine cognition to predict fire-induced spalling and fire resistance of concrete structures. Autom Constr 106:102916. https://doi.org/10.1016/J.AUTCON.2019.102916
160. Naser MZ (2019) Properties and material models for modern construction materials at elevated temperatures. Comput Mater Sci 160:16–29. https://doi.org/10.1016/J.COMMATSCI.2018.12.055
161. Naser MZ (2019) Properties and material models for common construction materials at elevated temperatures. Constr Build Mater 10:192–206. https://doi.org/10.1016/j.conbuildmat.2019.04.182
162. McKinney J, Ali F (2014) Artificial neural networks for the spalling classification & failure prediction times of high strength concrete columns. J Struct Fire Eng. https://doi.org/10.1260/2040-2317.5.3.203
163. Naser MZZ, Seitllari A (2019) Concrete under fire: an assessment through intelligent pattern recognition. Eng Comput 36:1–14. https://doi.org/10.1007/s00366-019-00805-1

164. Liu JC, Zhang Z (2020) A machine learning approach to predict explosive spalling of heated concrete. Arch Civ Mech Eng. https://doi.org/10.1007/s43452-020-00135-w
165. Cachim PB (2011) Using artificial neural networks for calculation of temperatures in timber under fire loading. Constr Build Mater. https://doi.org/10.1016/j.conbuildmat.2011.04.054
166. Naser MZ (2019) Fire resistance evaluation through artificial intelligence: a case for timber structures. Fire Saf J 105:1–18. https://doi.org/10.1016/j.firesaf.2019.02.002
167. Tasdemir SS, Altin M, Pehlivan GF, Saritas I, Didem S, Erkis B, Tasdemir SS (2015) Determining fire resistance of wooden construction elements through experimental studies and artificial neural network. J Int Mater Metal Eng 9:209–213
168. Cachim P (2019) ANN prediction of fire temperature in timber. J Struct Fire Eng 10:233–244. https://doi.org/10.1108/JSFE-06-2018-0012
169. Tung PT, Hung PT (2020) Predicting fire resistance ratings of timber structures using artificial neural networks. J Sci Technol Civ Eng - NUCE 14:28–39. https://doi.org/10.31814/stce.nuce2020-14(2)-03
170. Liu JC, Zhang Z (2020) Neural network models to predict explosive spalling of PP fiber reinforced concrete under heating. J Build Eng. https://doi.org/10.1016/j.jobe.2020.101472
171. Farrar CR, Worden K (2007) An introduction to structural health monitoring. Philos Trans R Soc A Math Phys Eng Sci 365:303. https://doi.org/10.1098/rsta.2006.1928
172. Wu RT, Jahanshahi MR (2020) Data fusion approaches for structural health monitoring and system identification: past, present, and future. Struct Heal Monit. https://doi.org/10.1177/1475921718798769
173. Jo-Chun P, Ujike I, Mishima K, Kusumoto M, Okazaki S (2020) Random forest-based evaluation technique for internal damage in reinforced concrete featuring multiple nondestructive testing results. Constr Build Mater 253:119238. https://doi.org/10.1016/j.conbuildmat.2020.119238
174. Diez A, Khoa NLD, Makki Alamdari M, Wang Y, Chen F, Runcie P (2016) A clustering approach for structural health monitoring on bridges. J Civ Struct Health Monit 6:429–445. https://doi.org/10.1007/s13349-016-0160-0
175. Kurian B, Liyanapathirana R (2020) Machine Learning Techniques for Structural Health Monitoring. Lect Notes Mech Eng. https://doi.org/10.1007/978-981-13-8331-1_1
176. Athanasiou A, Ebrahimkhanlou A, Zaborac J, Hrynyk T, Salamone S (2020) A machine learning approach based on multifractal features for crack assessment of reinforced concrete shells. Comput Civ Infrastruct Eng 35:565–578. https://doi.org/10.1111/mice.12509
177. Chen C, Fu J, Lu N, Chu Y, Hu J, Guo B, Zhao X (2019) Knowledge-based identification and damage detection of bridges spanning water via high-spatial-resolution optical remotely sensed imagery. J Indian Soc Remote Sens. https://doi.org/10.1007/s12524-019-01036-z
178. Nguyen DT, Ofli F, Imran M, Mitra P (2017) Damage assessment from social media imagery data during disasters. In: Proc 2017 IEEE/ACM Int Conf Adv Soc Networks Anal Mining, ASONAM 2017. https://doi.org/10.1145/3110025.3110109
179. Noh Y, Koo D, Kang YM, Park DG, Lee DH (2017) Automatic crack detection on concrete images using segmentation via fuzzy C-means clustering. In: Proc 2017 IEEE Int Conf Appl Syst Innov Appl Syst Innov Mod Technol ICASI 2017, pp 877–880. https://doi.org/10.1109/ICASI.2017.7988574
180. Xu Y, Wei S, Bao Y, Li H (2019) Automatic seismic damage identification of reinforced concrete columns from images by a region-based deep convolutional neural network. Struct Control Heal Monit 26:1–22. https://doi.org/10.1002/stc.2313
181. Dung CV, Anh LD (2019) Autonomous concrete crack detection using deep fully convolutional neural network. Autom Constr. https://doi.org/10.1016/j.autcon.2018.11.028
182. Li S, Zhao X (2019) Image-based concrete crack detection using convolutional neural network and exhaustive search technique. Adv Civ Eng 2019:1–12. https://doi.org/10.1155/2019/6520620
183. Rashidi A, Sigari MH, Maghiar M, Citrin D (2016) An analogy between various machine-learning techniques for detecting construction materials in digital images. KSCE J Civ Eng 20:1178–1188. https://doi.org/10.1007/s12205-015-0726-0
184. Anay R, Soltangharaei V, Assi L, DeVol T, Ziehl P (2018) Identification of damage mechanisms in cement paste based on acoustic emission. Constr Build Mater. https://doi.org/10.1016/j.conbuildmat.2017.12.207
185. Hasni H, Alavi AH, Lajnef N, Abdelbarr M, Masri SF, Chakrabartty S (2017) Self-powered piezo-floating-gate sensors for health monitoring of steel plates. Eng Struct. https://doi.org/10.1016/j.engstruct.2017.06.063
186. Chang CMCW, Lin TK, Chang CMCW (2018) Applications of neural network models for structural health monitoring based on derived modal properties. Meas J Int Meas Confed 129:457–470. https://doi.org/10.1016/j.measurement.2018.07.051
187. Kurian B, Liyanapathirana R (2018) Proceedings of the international conference on e-learning, ICEL. Springer Singapore. https://doi.org/10.1007/978-981-13-8331-1
188. Hoang ND, Nguyen QL (2020) A novel approach for automatic detection of concrete surface voids using image texture analysis and history-based adaptive differential evolution optimized support vector machine. Adv Civ Eng 2020:1–15. https://doi.org/10.1155/2020/4190682
189. Liu H, Zhang Y (2019) Image-driven structural steel damage condition assessment method using deep learning algorithm. Meas J Int Meas Confed 133:168–181. https://doi.org/10.1016/j.measurement.2018.09.081
190. Satpal SB, Guha A, Banerjee S (2015) Damage identification in aluminum beams using support vector machine: numerical and experimental studies. Struct Control Heal Monit. https://doi.org/10.1002/stc.1773
191. Mariniello G, Pastore T, Menna C, Festa P, Asprone D (2020) Structural damage detection and localization using decision tree ensemble and vibration data. Comput Civ Infrastruct Eng 2020:1–21. https://doi.org/10.1111/mice.12633
192. Mangalathu S, Jang H, Hwang SH, Jeon JS (2020) Data-driven machine-learning-based seismic failure mode identification of reinforced concrete shear walls. Eng Struct. https://doi.org/10.1016/j.engstruct.2020.110331
193. Chen XL, Fu JP, Yao JL, Gan JF (2018) Prediction of shear strength for squat RC walls using a hybrid ANN–PSO model. Eng Comput 34:367–383. https://doi.org/10.1007/s00366-017-0547-5
194. Ketabdari H, Karimi F, Rasouli M (2020) Shear strength prediction of short circular reinforced-concrete columns using soft computing methods. Adv Struct Eng 23:3048–3061. https://doi.org/10.1177/1369433220927270
195. Ly HB, Le TT, Thi-Vu HL, Tran VQ, Le LM, Pham BT (2020) Erratum: Computational hybrid machine learning based prediction of shear capacity for steel fiber reinforced concrete beams. Sustainability 12:2709. https://doi.org/10.3390/su12177029
196. Ababneh A, Alhassan M, Abu-Haifa M (2020) Predicting the contribution of recycled aggregate concrete to the shear capacity of beams without transverse reinforcement using artificial neural networks. Case Stud Constr Mater 13:e00414. https://doi.org/10.1016/j.cscm.2020.e00414
197. Solhmirzaei R, Salehi H, Kodur V, Naser MZ (2020) Machine learning framework for predicting failure mode and shear capacity of ultra high performance concrete beams. Eng Struct. https://doi.org/10.1016/j.engstruct.2020.111221

198. Bai C, Nguyen H, Asteris PG, Nguyen-Thoi T, Zhou J (2020) A refreshing view of soft computing models for predicting the deflection of reinforced concrete beams. Appl Soft Comput J 97:106831. https://doi.org/10.1016/j.asoc.2020.106831
199. Lee S, Lee C (2014) Prediction of shear strength of FRP-reinforced concrete flexural members without stirrups using artificial neural networks. Eng Struct 61:99–112. https://doi.org/10.1016/j.engstruct.2014.01.001
200. Abuodeh OR, Abdalla JA, Hawileh RA (2020) Prediction of shear strength and behavior of RC beams strengthened with externally bonded FRP sheets using machine learning techniques. Compos Struct 234:111698. https://doi.org/10.1016/j.compstruct.2019.111698
201. Fardis MN, Khalili HH (1982) FRP-encased concrete as a structural material. Magn Concr Res. https://doi.org/10.1680/macr.1982.34.121.191
202. Ritchie PA, Thomas DA, Lu LW, Connelly GM (1991) External reinforcement of concrete beams using fiber reinforced plastics. ACI Struct J. https://doi.org/10.14359/2723
203. Naser MZ, Hawileh RA, Abdalla JA (2019) Fiber-reinforced polymer composites in strengthening reinforced concrete structures: a critical review. Eng Struct 198:109542
204. Naderpour H, Haji M, Mirrashid M (2020) Shear capacity estimation of FRP-reinforced concrete beams using computational intelligence. Structures 28:321–328. https://doi.org/10.1016/j.istruc.2020.08.076
205. Mansouri I, Ozbakkaloglu T, Kisi O, Xie T (2016) Predicting behavior of FRP-confined concrete using neuro fuzzy, neural network, multivariate adaptive regression splines and M5 model tree techniques. Mater Struct Constr 49:4319–4334. https://doi.org/10.1617/s11527-015-0790-4
206. Nguyen TT, Pham Duy H, Pham Thanh T, Vu HH (2020) Compressive strength evaluation of fiber-reinforced high-strength self-compacting concrete with artificial intelligence. Adv Civ Eng 2020:1–12. https://doi.org/10.1155/2020/3012139
207. Naderpour H, Poursaeidi O, Ahmadi M (2018) Shear resistance prediction of concrete beams reinforced by FRP bars using artificial neural networks. Meas J Int Meas Confed 126:299–308. https://doi.org/10.1016/j.measurement.2018.05.051
208. Su M, Zhong Q, Peng H, Li S (2020) Selected machine learning approaches for predicting the interfacial bond strength between FRPs and concrete. Constr Build Mater 270:121456. https://doi.org/10.1016/j.conbuildmat.2020.121456
209. Köroglu MA (2019) Artificial neural network for predicting the flexural bond strength of FRP bars in concrete. Sci Eng Compos Mater 26:12–29. https://doi.org/10.1515/secm-2017-0155
210. Abdalla JA, Elsanosi A, Abdelwahab A (2007) Modeling and simulation of shear resistance of R/C beams using artificial neural network. J Franklin Inst. https://doi.org/10.1016/j.jfranklin.2005.12.005
211. Ma CK, Lee YH, Awang AZ, Omar W, Mohammad S, Liang M (2019) Artificial neural network models for FRP-repaired concrete subjected to pre-damaged effects. Neural Comput Appl 31:711–717. https://doi.org/10.1007/s00521-017-3104-7
212. Vu DT, Hoang ND (2016) Punching shear capacity estimation of FRP-reinforced concrete slabs using a hybrid machine learning approach. Struct Infrastruct Eng. https://doi.org/10.1080/15732479.2015.1086386
213. Feng DC, Fu B (2020) Shear strength of internal reinforced concrete beam-column joints: intelligent modeling approach and sensitivity analysis. Adv Civ Eng 2020:1–19. https://doi.org/10.1155/2020/8850417
214. Allahyari H, Nikbin IM, Rahimi S, Heidarpour A (2018) A new approach to determine strength of Perfobond rib shear connector in steel-concrete composite structures by employing neural network. Eng Struct 157:235–249. https://doi.org/10.1016/j.engstruct.2017.12.007
215. Mirrashid M (2017) Comparison study of soft computing approaches for estimation of the non-ductile RC joint shear strength. J Soft Comput Civ Eng 1:9–25. https://doi.org/10.22115/scce.2017.46318
216. Alwanas AAH, Al-Musawi AA, Salih SQ, Tao H, Ali M, Yaseen ZM (2019) Load-carrying capacity and mode failure simulation of beam-column joint connection: application of self-tuning machine learning model. Eng Struct. https://doi.org/10.1016/j.engstruct.2019.05.048
217. Degtyarev VV (2021) Neural networks for predicting shear strength of CFS channels with slotted webs. J Constr Steel Res. https://doi.org/10.1016/j.jcsr.2020.106443
218. Le TT (2020) Practical machine learning-based prediction model for axial capacity of square CFST columns. Mech Adv Mater Struct. https://doi.org/10.1080/15376494.2020.1839608
219. Nguyen HQ, Ly HB, Tran VQ, Nguyen TA, Le TT, Pham BT (2020) Optimization of artificial intelligence system by evolutionary algorithm for prediction of axial capacity of rectangular concrete filled steel tubes under compression. Materials (Basel) 13:1205. https://doi.org/10.3390/MA13051205
220. Thai S, Thai HT, Uy B, Ngo T (2019) Concrete-filled steel tubular columns: test database, design and calibration. J Constr Steel Res. https://doi.org/10.1016/j.jcsr.2019.02.024
221. Thai S, Thai H, Uy B, Ngo T, Naser M (2019) Test database on concrete-filled steel tubular columns. https://doi.org/10.17632/3XKNB3SDB5.1
222. Shariati M, Mafipour MS, Mehrabi P, Bahadori A, Zandi Y, Salih MNA, Nguyen H, Dou J, Song X, Poi-Ngian S (2019) Application of a hybrid artificial neural network-particle swarm optimization (ANN-PSO) model in behavior prediction of channel shear connectors embedded in normal and high-strength concrete. Appl Sci 9:5534. https://doi.org/10.3390/app9245534
223. Shariati M, Mafipour MS, Mehrabi P, Shariati A, Toghroli A, Trung NT, Salih MNA (2020) A novel approach to predict shear strength of tilted angle connectors using artificial intelligence techniques. Eng Comput. https://doi.org/10.1007/s00366-019-00930-x
224. Kotsovou GM, Cotsovos DM, Lagaros ND (2017) Assessment of RC exterior beam-column Joints based on artificial neural networks and other methods. Eng Struct 144:1–18. https://doi.org/10.1016/j.engstruct.2017.04.048
225. Razavi SV, Jumaat MZ, Ei-Shafie AH, Mohammadi P (2011) General regression neural network (GRNN) for the first crack analysis prediction of strengthened RC one-way slab by CFRP. Int J Phys Sci 6:2439–2446. https://doi.org/10.5897/IJPS10.578
226. Naderpour H, Mirrashid M, Nagai K (2020) An innovative approach for bond strength modeling in FRP strip-to-concrete joints using adaptive neuro–fuzzy inference system. Eng Comput 36:1083–1100. https://doi.org/10.1007/s00366-019-00751-y
227. Yaseen ZM, Afan HA, Tran MT (2018) Beam-column joint shear prediction using hybridized deep learning neural network with genetic algorithm. IOP Conf Ser Earth Environ Sci 143:012025. https://doi.org/10.1088/1755-1315/143/1/012025
228. Luo H, Paal SG (2018) Machine learning-based backbone curve model of reinforced concrete columns subjected to cyclic loading reversals. J Comput Civ Eng 32:04018042. https://doi.org/10.1061/(asce)cp.1943-5487.0000787
229. Yogatama D, Mann G (2014) Efficient transfer learning method for automatic hyperparameter tuning. J Mach Learn Res
230. Arel I, Rose D, Coop R (2009) DeSTIN: a scalable deep learning architecture with application to high-dimensional robust pattern recognition. In: AAAI Fall Symp. - Tech. Rep
231. Krishnamoorthy K (2020) Wilcoxon signed-rank test. Handb Stat Distrib Appl. https://doi.org/10.1201/9781420011371-34

232. Anaene Oyeka IC, Ebuh GU (2012) Modified wilcoxon signed-rank test. Open J Stat 2:172. https://doi.org/10.4236/ojs.2012.22019
233. Bouckaert RR, Frank E (2004) Evaluating the replicability of significance tests for comparing learning algorithms. In: Lect Notes Comput Sci (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). https://doi.org/10.1007/978-3-540-24775-3_3
234. Kim S, Lee W (2017) Does McNemar's test compare the sensitivities and specificities of two diagnostic tests? Stat Methods Med Res. https://doi.org/10.1177/0962280214541852
235. Bundy A (2017) Preparing for the future of artificial intelligence. AI Soc. https://doi.org/10.1007/s00146-016-0685-0
236. Kim K, Park Y (2017) A development and application of the teaching and learning model of artificial intelligence education for elementary students. J Korean Assoc Inf Educ. https://doi.org/10.14352/jkaie.2017.21.1.139
237. Jones DT (2019) Setting the standards for machine learning in biology. Nat Rev Mol Cell Biol. https://doi.org/10.1038/s41580-019-0176-5
238. Loyola-Gonzalez O (2019) Black-box vs White-Box: understanding their advantages and weaknesses from a practical point of view. IEEE Access 7:154096. https://doi.org/10.1109/ACCESS.2019.2949286
239. Witten IH, Frank E, Hall MA (2011) Data mining: practical machine learning tools and techniques. Google eBook
240. Yin M, Vaughan JW, Wallach H (2019) Understanding the effect of accuracy on trust in machine learning models. In: Conf Hum Factors Comput Syst—Proc. https://doi.org/10.1145/3290605.3300509
241. Alloghani M, Al-Jumeily D, Mustafina J, Hussain A, Aljaaf AJ (2020) A systematic review on supervised and unsupervised machine learning algorithms for data science. Springer, Cham
242. Sutton RS, Barto AG (2017) Reinforcement learning : an introduction 2nd (19 June, 2017), Neural Networks IEEE Trans
243. Naser MZ (2022) Causality, causal discovery, and causal inference in structural engineering. https://doi.org/10.48550/arxiv.2204.01543
244. Naser MZ (2021) Demystifying ten big ideas and rules every fire scientist & engineer should know about blackbox, whitebox & causal artificial intelligence. https://arxiv.org/abs/2111.13756v1 . Accessed 26 Jan 2022
245. Naser M (2022) A faculty's perspective into infusing artificial intelligence to civil engineering education. J Civ Eng Educ. https://doi.org/10.1061/(ASCE)EI.2643-9115.0000065

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.