smart cities

Article

Artificial Neural Networks and Ensemble Learning for Enhanced Liquefaction Prediction in Smart Cities

Yuxin Cong [1] and Shinya Inazumi [2,*]

[1] Graduate School of Engineering and Science, Shibaura Institute of Technology, Tokyo 135-8548, Japan; na23107@shibaura-it.ac.jp
[2] College of Engineering, Shibaura Institute of Technology, Tokyo 135-8548, Japan
[*] Correspondence: inazumi@shibaura-it.ac.jp

Highlights:

What are the main findings?

- The bagging prediction model demonstrated approximately 20% higher accuracy compared to the single ANN model.
- Accurate prediction of bearing layer depth is critical for improving urban resilience and infrastructure planning in smart cities.

What are the implications of the main finding?

- The improved accuracy of the bagging model supports more reliable geotechnical investigations, which can lead to safer urban development in earthquake-prone areas.
- Improved prediction models for bearing layer depth can reduce the need for extensive in situ testing, lowering costs and increasing the efficiency of construction projects.

Abstract: This paper examines how smart cities can address land subsidence and liquefaction in the context of rapid urbanization in Japan. Since the 1960s, liquefaction has been an important topic in geotechnical engineering, and extensive efforts have been made to evaluate soil resistance to liquefaction. Currently, there is a lack of machine learning applications in smart cities that specifically target geological hazards. This study aims to develop a high-performance prediction model for estimating the depth of the bearing layer, thereby improving the accuracy of geotechnical investigations. The model was developed using actual survey data from 433 points in Setagaya-ku, Tokyo, by applying two machine learning techniques: artificial neural networks (ANNs) and bagging. The results indicate that machine learning offers significant advantages in predicting the depth of the bearing layer. Furthermore, the prediction performance of ensemble learning improved by about 20% compared to ANNs. Both interdisciplinary approaches contribute to risk prediction and mitigation, thereby promoting sustainable urban development and underscoring the potential of future smart cities.

Keywords: artificial neural networks; ensemble learning; geotechnical information; prediction; smart cities

check for updates

Citation: Cong, Y.; Inazumi, S. Artificial Neural Networks and Ensemble Learning for Enhanced Liquefaction Prediction in Smart Cities. *Smart Cities* **2024**, *7*, 2910–2924. https://doi.org/10.3390/smartcities7050113

Academic Editor: Pierluigi Siano

Received: 2 September 2024
Revised: 4 October 2024
Accepted: 7 October 2024
Published: 8 October 2024

1. Introduction

Japan's urban landscape, characterized by rapid urbanization and cutting-edge technological advances, is at the forefront of addressing complex challenges in the construction and infrastructure sectors. However, in Japan, significant structural damage is often caused by settlement or overturning of structures due to liquefaction of saturated sandy soils during large earthquakes [1].

The primary cause of liquefaction is the loss of shear strength due to increased pore water pressure and a reduction in effective soil stress, which ultimately causes sandy soils to exhibit a fluid-like behavior [2–4]. As shown in Figure 1, liquefaction is a phenomenon that occurs when an earthquake causes a strong shock to the ground, causing soil particles that were previously in contact and supporting each other to separate, transforming the entire soil into a viscous, liquid-like state. When liquefaction occurs, water can gush out of the ground, causing previously stable soil to suddenly become soft. This can cause buildings to sink or tilt, manholes and buried pipes to rise to the surface, and the entire soil to flow downward. As shown in Table 1, liquefaction is more likely to occur when the following three conditions combine: loose soil, high water table, and an earthquake. Liquefaction has been an important topic in geotechnical engineering since the 1960s, and considerable effort has been devoted to evaluating the liquefaction resistance of in situ soils. The current methodology for this assessment typically relies on in situ measurements, such as the SPT-N value (the number of blows required to penetrate 30 cm in the standard penetration test [5,6]). Historical records indicate that most severe liquefaction disasters have occurred in geologically young, sandy deposits, such as those found on artificial islands and in former river channels, filled lakes, swamps, and pipeline backfill areas [7–9]. The sudden ground instability during such events can cause catastrophic damage to buildings and infrastructure, resulting in significant economic losses and tragic loss of life. This critical issue has been further highlighted in [1,10,11].

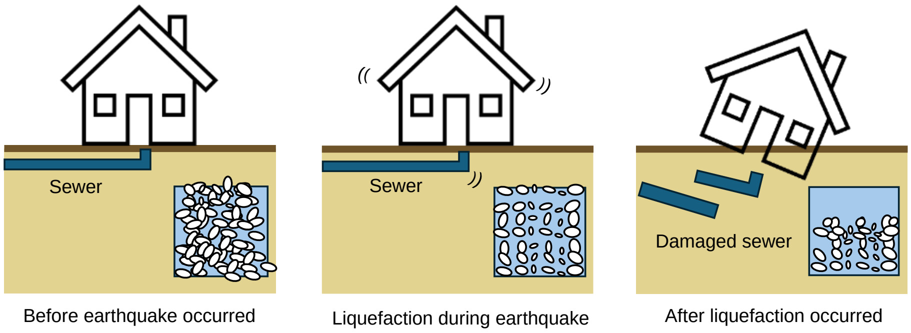
Sewer
Before earthquake occurred
Sewer
Liquefaction during earthquake
Damaged sewer
After liquefaction occurred

Figure 1. Specific manifestations of the liquefaction phenomenon.

Table 1. Three factors contributing to liquefaction and their detailed explanations.

Causes of Liquefaction	Details
Loose ground	Sandy soils with an N value of 20 or less, indicating soil hardness, and particle sizes between 0.03 mm and 0.5 mm.
High groundwater	Groundwater level within 10 m of the ground surface.
Earthquake	Commonly observed along coastlines, near river mouths, on reclaimed land, and in river alluvial fans. Earthquake intensity of 5 or higher. The longer the shaking lasts, the greater the damage.

The government should take effective measures to prevent and control liquefaction in residential areas in order to promote research, publish relevant findings, and deepen public understanding of liquefaction hazards.

The importance of smart cities is obvious. Smart cities use various information technologies or innovative concepts to integrate the systems and services that make up the city, with the aim of improving the efficiency of resource utilization, optimizing urban management and services, and improving the quality of life for citizens. Specifically, the concept of "intelligence" allows people to manage production and daily life in a more sophisticated and dynamic way through the application of next-generation information technologies. The emergence of the Internet of Things (IoT) enables access to remote sensor data and remote control of the physical world, allowing cities to effectively monitor and manage essential elements such as water supply, building operations, and transportation networks [12]. Data vitalization introduces a new paradigm for analyzing large datasets and provides ubiquitous data support for top-level smart city applications [13,14].

In this study, as shown in Figure 2, an AI-driven predictive model integrates data from various databases, including geotechnical and geographic information, to enhance urban resilience and promote the development of a safer and more sustainable society. This approach contributes to the sustainable growth of smart cities and ensures the safety of their inhabitants [15].

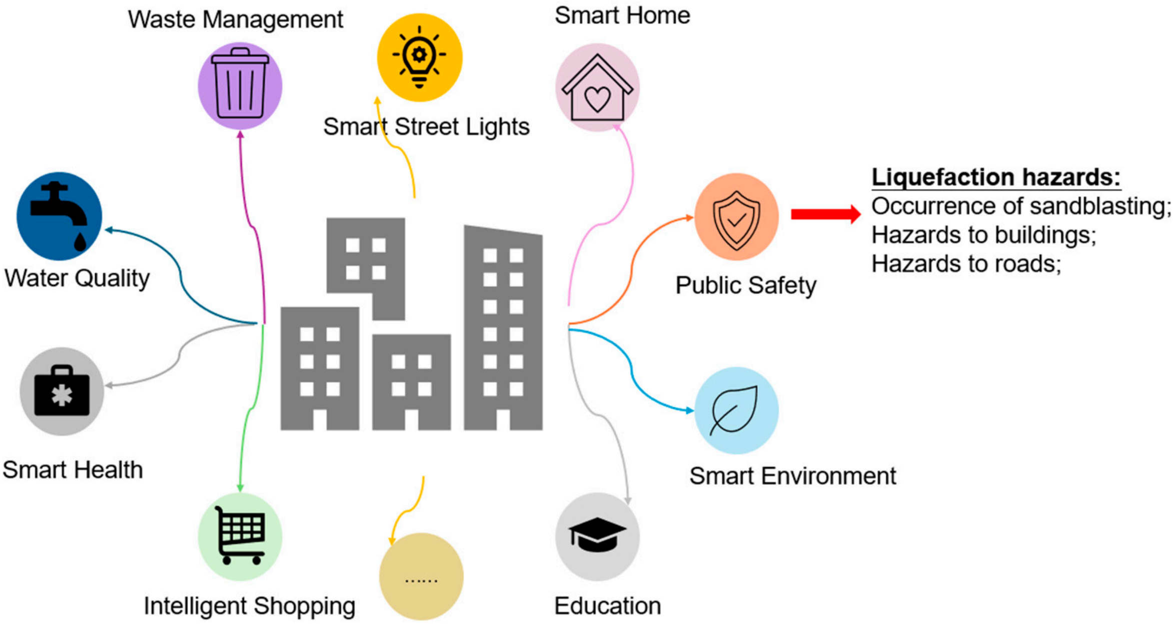

Figure 2. Visualization of smart cities and their relevance to this study.

Although monitoring technology for geologic hazards and liquefaction has continuously improved, significant limitations remain. The main problem is that traditional empirical methods lack reliability and universality. Therefore, the field of machine learning is expected to play a key role in improving prediction accuracy. Techniques such as artificial neural networks (ANNs) [16] and ensemble learning [17–20], which improve prediction through algorithmic diversity, are at the forefront of spatial and temporal data analysis. Currently, machine learning is widely used in many countries for applications such as address prediction and other purposes [21,22]. Similarly, ANNs can be used to predict potential geological risks [23]. Machine learning can build a fine-grained 3D geological model that provides an integrated representation of stratigraphic and lithologic information, demonstrating its effectiveness [24]. By using machine learning to analyze geological data and make predictions, it can significantly aid urban planning. This also reflects the widespread use and impact of machine learning worldwide.

The main objective of this study is to use AI technology that surpasses traditional methods to predict geological information, analyze reliable data sets, and develop a new prediction model. Since this is an emerging field, this study is divided into two parts: creating a single model using ANNs and developing an integrated model by combining multiple models using bagging for analysis and comparison. By comparing the results, the model with the superior predictive performance will be identified to address Japan's liquefaction problem, with the ultimate goal of achieving smart infrastructure and data-driven smart cities. This work also marks the beginning of a new era in urban development. By integrating ever-evolving technologies with traditional studies, we can better address urban development challenges and ensure a sustainable future for all.

2. Data and Methods

The purpose of this study is to predict the depth of the bearing layer. The bearing layer is defined as "a layer that is strong enough to support a given structure". In other words, it is "a layer that is strong and does not easily undergo detrimental deformation when loaded". A 63.5 ± 0.5 kg hammer is used to strike the drill rod at a free fall height of 76 ± 1 cm, driving the sampler 30 cm into the ground in the standard penetration test. The number of impacts required is referred to as the N value. The N value reflects the relationship between soil moisture, clay, and organic matter content and is used to estimate soil bearing capacity and the degree of settlement after drainage. An N value of 20 or less often indicates instability as a foundation layer for civil engineering structures. In general, soil with an N value of 20 or more or rock is desirable as a foundation layer. If the N value is between 30 and 50, the layer is considered suitable as a foundation for civil engineering structures. If the N value is 50 or more, it is considered to be very solid and suitable as a bearing layer for large structures such as high-rise buildings [25]. Therefore, in this study, an N value greater than 50 over a continuous depth of more than 3 m is defined as a bearing layer [26].

Machine learning tools aim to solve two classical statistical tasks: classification (pattern recognition) and regression (function approximation) [27]. The goal of regression is to predict real-valued labels for data, while the goal of classification is to predict discrete labels [28]. First, it is necessary to determine whether the problem is a regression or a classification problem. Clearly, predicting the depth of the bearing layer is a continuous value, so it is a regression problem.

This study used geological survey data from Setagaya-ku, Tokyo. The location information (latitude and longitude) of the survey sites and the depth of the bearing layer were obtained from the results of ground surveys in the Kanto region based on standard penetration tests and mini-ram sounding tests. The specific source of the data on the depth of the bearing layer was the actual experimental investigations conducted in recent years, which were provided by the cooperating company involved in the study. Elevation data were obtained from the Tokyo Geographical Research Institute, which provides a real-time elevation query service. By providing the latitude and longitude of the desired location, the corresponding elevation can be obtained in real time. A specific example of the data is shown in Table 2. A total of 433 data points were used in this study, and the analysis of these 433 data points is shown in Table 3. In addition, the distribution of the 433 data points on the map is shown in Figure 3, with the four locations listed in Table 2 also marked in Figure 3.

Table 2. Specific data examples from Setagaya-ku, Tokyo.

Latitude	Longitude	Bearing Layer Depth (m)	Elevation (m)
35.6290	139.674	13.38	38.8
35.6114	139.632	11.00	11.0
35.6582	139.649	12.80	37.3
35.6679	139.669	13.53	36.5

Table 3. General analysis of 433 data points used for training.

Area (km^2)	Data Density (pcs/km^2)	Standard Deviation of the Data
58.1	7.46	9.53

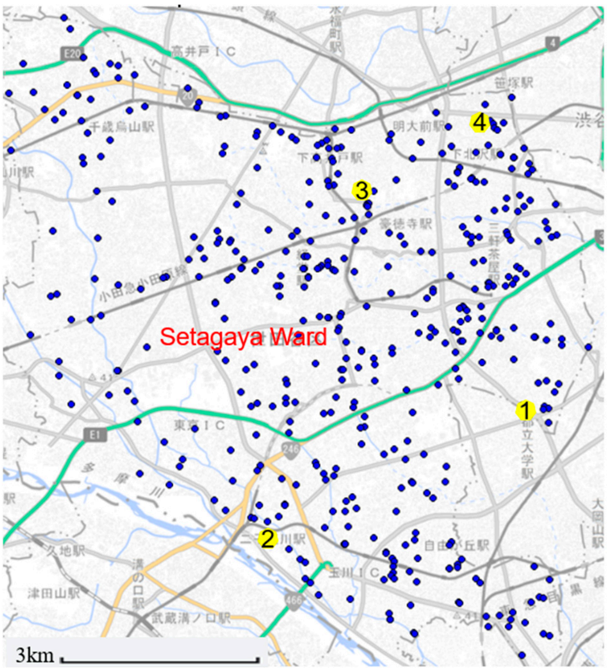

Figure 3. Specific distribution of 433 data and four examples in Table 2 on the map.

Since the term "machine learning" was coined by Arthur Samuel in 1959 [29], a wide variety of models have been developed in the field. In many cases, these models provide better results [30] due to their ability to handle complex data more effectively than classical statistical methods [31,32]. Recent advances in computing technologies have led to the development of several machine learning algorithms, in particular ANNs, which can operate in a nonlinear fashion [33]. Ensemble learning combines multiple machine learning algorithms to produce weakly predictive results based on features extracted from different data projections and fuses these results using different voting mechanisms to achieve better performance than any single algorithm alone [34].

The purpose of this study is to use ANNs and bagging techniques to learn from a dataset of 433 records, develop a prediction model for bearing layer depth, and compare the prediction performance.

3. Building Artificial Neural Networks (ANNs)

ANNs are mathematical or computational models that mimic the structure and function of biological neural networks and are used to estimate or approximate functions. ANNs have become popular and useful models for classification, clustering, pattern recognition, and prediction in various disciplines. ANNs are a type of model for machine learning (ML) and have become relatively competitive with traditional regression and statistical models in terms of effectiveness [35]. The great potential of ANNs lies in their high-speed processing capabilities, especially in massively parallel implementations, which has increased the interest in studying this field [36,37].

To build a PyTorch-based ANN model for output prediction, the process is divided into the following seven steps: defining the problem, preparing the dataset, defining the model, defining the loss function and optimizer, training the model, evaluating the model, and tuning the model.

3.1. Preparing the Dataset

After obtaining the data, duplicate entries were removed using the drop_duplicates function. Data types and missing information were then checked. Of the three variables, only the bearing layer depth variable had missing values. There were no variables with a high enough percentage of missing values to warrant deletion, and all variables were retained for modeling. Lines with missing values were simply removed. Because all variables in this study are numerical, they are easy to process and improve model performance. After confirming the relevance and importance of the features, the dataset was finalized. A total of 433 datasets were used in this study, and the data were divided into training, validation, and test sets at a ratio of 7:2:1.

3.2. Creating the Model

This study uses a model with three input neurons and one output neuron. The purpose of this model is to determine the weights on the connections between the input and output units in order to fit the results to the given data. By adding a hidden layer, the model is capable of handling nonlinear relationships. As shown in Figure 4, an ANN model is implemented using PyTorch 2.0.

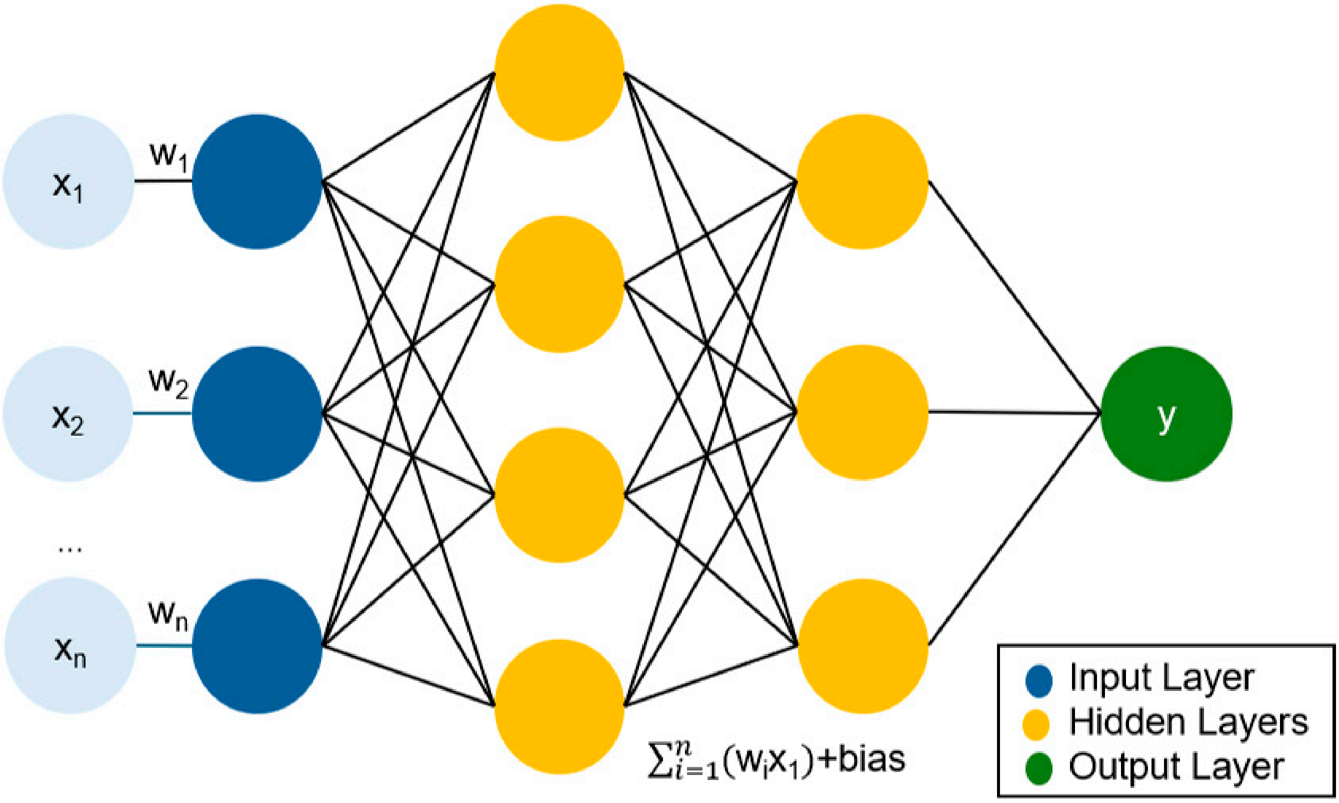

Figure 4. Prototype of ANN model for Case 1.

The activation function chosen is ReLU, which is commonly used in deep ANNs, and is defined as follows in Equation (1):

$$\text{ReLU (x)} = \max\,(0, x) \tag{1}$$

when x is less than 0, the output is 0, and when x is greater than 0, the output is x. Since ReLU is used as the activation function, the initial value specific to ReLU, recommended by He et al. [38] and known as the "He initialization", is applied. For the number of nodes (n) in the previous layer, the He initialization uses a Gaussian distribution with a standard deviation of $\sqrt{\frac{2}{n}}$ [38].

For a model with three inputs and one output, it is first defined as a simple fully connected network. In an ANN, fully connected means that every neuron in the current layer is directly connected to all neurons in the previous layer. This connection implies that the output from each neuron in the previous layer is passed to every neuron in the current layer, where a weighted summation of these inputs is performed, followed by applying an activation function to produce the output.

ANNs train the network through two processes: forward propagation and backpropagation. During forward propagation, input data is processed by applying weights and activation functions, passing the output from layer to layer until the final output is produced. During backpropagation, errors are propagated from the output layer to the input layer based on the difference between the predicted output and the true labels. During this process, gradients for each layer are calculated. Using these gradients, the gradient descent method is applied to update the weights and biases in order to minimize the error function and improve network performance. These steps are repeated until the maximum number of iterations is reached.

It is important to note the parameters that are set during the model creation. One key parameter is 'hidden_layer_sizes', which defines the number of hidden layers and the number of neurons in each layer. Initially, the array [2,5] was set, meaning the network had two hidden layers, with 5 neurons in the first layer and 2 neurons in the second layer. However, this configuration resulted in an accuracy of only 40%. To improve the predictive performance, the hidden layer configuration was modified to three hidden layers, each containing 10 neurons. This adjustment led to an improved accuracy of approximately 90%.

For regression problems, the mean square error (MSE) was used as the loss function, and stochastic gradient descent (SGD) was selected as the optimizer. Hyperparameter tuning was performed manually.

4. Building Bagging

Ensemble learning methods, which involve building and combining multiple learners, have been shown to produce better results and achieve improved generalization compared to any individual classifier alone [39–41]. Many methods for constructing ensembles have been developed, but bagging, boosting, and stacking are the most commonly used techniques [42]. In short, bagging (also known as bootstrap aggregation; [43]) improves the stability and accuracy of machine learning algorithms by training the same algorithm multiple times using different subsets sampled from the training data [44].

The aggregation of multiple learners results in lower variance for the model, although its bias may remain unchanged, based on the bias-variance decomposition of error for machine learning models. Given multiple models of the same machine learning algorithm trained on different training datasets, the bias represents the similarity between the models' average prediction and the ground truth, while the variance reflects the variability between the predictions [44]. Random Forest [45], as illustrated in Figure 5, is a well-known implementation of bagging that uses decision trees and introduces additional randomness in the feature selection process during training [46].

After creating the model, hyper-parameter tuning is still required. Manual parameter tuning was used for optimization. It was found that when the number of decision trees (n_estimators) was set to 91, the model achieved optimal performance. The specific hyperparameter values are listed in Table 4. 'n_estimators' represents the number of decision trees generated by sampling the original dataset with replacement. 'max_depth' specifies the maximum depth of each decision tree; a value of 'None' indicates that the depth of the sub-tree is not limited when building the optimal model. 'max_features' defines the maximum number of features considered when splitting a node; 'auto' means that the maximum number of features is set to the square root of the number of features (N).

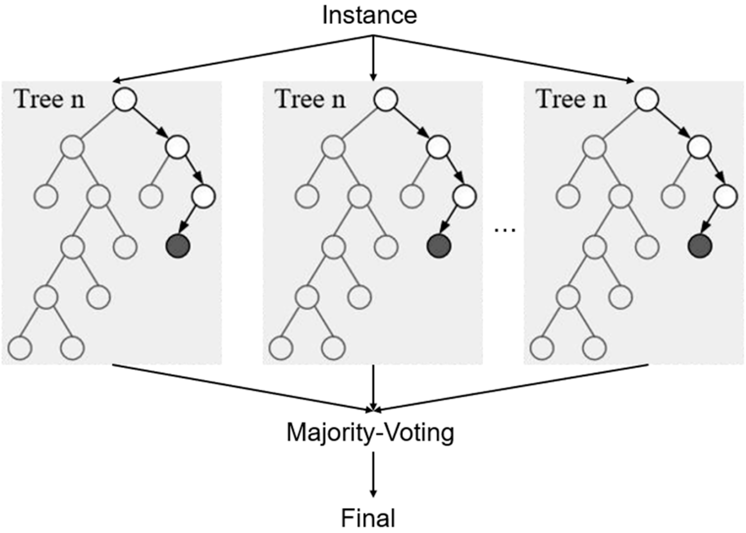

Figure 5. Specific process of the random forest model for Case 2.

Table 4. Hyper-parameter values of bagging in Case 2.

Hyperparameters	Value
N_esimators	91
Max features	Auto
Max_depth	None

The steps to creating the model are as follows: First, bootstrap sampling is used to extract samples from the original dataset, forming multiple sub-datasets. Second, a decision tree is constructed for each sub-dataset. At each node, a random subset of features is selected for splitting. These steps are repeated until 91 decision trees are generated. Finally, the predictions of these 91 decision trees are averaged to obtain the final prediction. This averaging process is represented by Equation (2):

$$Y = \frac{1}{N}\sum_{n=1}^{N} X_n \tag{2}$$

where, Y is the predicted value of the ensemble, X_n is the prediction from an individual decision tree, and N is the total number of decision trees.

A diagram of a decision tree is shown in Figure 6, where X [0], X [1], and X [2] represent latitude, longitude, and elevation, respectively.

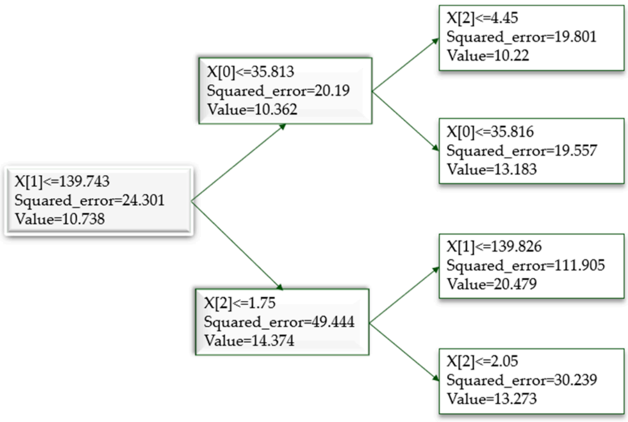

Figure 6. Diagram of a decision tree.

5. Results and Discussion

In the context of smart city development, the integration of AI technology can significantly improve urban management. This study focuses on two case studies in Setagaya, Tokyo. The accuracy and effectiveness of these methods are crucial for smart city applications, including urban planning and environmental monitoring. In Case 1, ANNs were used to create a predictive model, while in Case 2, bagging was used to develop a predictive model. The explanatory variables for both cases were latitude, longitude, elevation, and bearing layer depth, with the target variable being bearing layer depth.

5.1. Results on Predicting Bearing Layer Depth

In Case 1, the model created using ANNs was used to predict the bearing layer depth at 10 locations in Setagaya-ku, Tokyo. The actual measurements at these locations were used to evaluate the accuracy of the predictions, and the error values between the predicted and actual measurements were calculated. Table 5 and Figure 7 show the specific prediction results and errors for the 10 points in both cases. Figure 8 shows the prediction results for all points in Case 1, demonstrating the accuracy of the spatial prediction method. Similarly, in Case 2, the same data were used to make predictions using the model created with bagging. Figure 9 shows the prediction results for all points in Case 2.

Table 5. Average prediction error of bearing layer depth in both cases.

Predicted Location	Error of Case 1 (m)	Error of Case 2 (m)
1	1.40	0.75
2	0.80	0.53
3	5.30	3.41
4	0.70	1.95
5	0.89	0.09
6	1.40	0.22
7	0.56	0.02
8	0.70	0.10
9	0.26	0.26
10	0.78	1.26
Average error (m)	1.27	0.86
CI	10.16 ± 0.77	10.56 ± 1.05

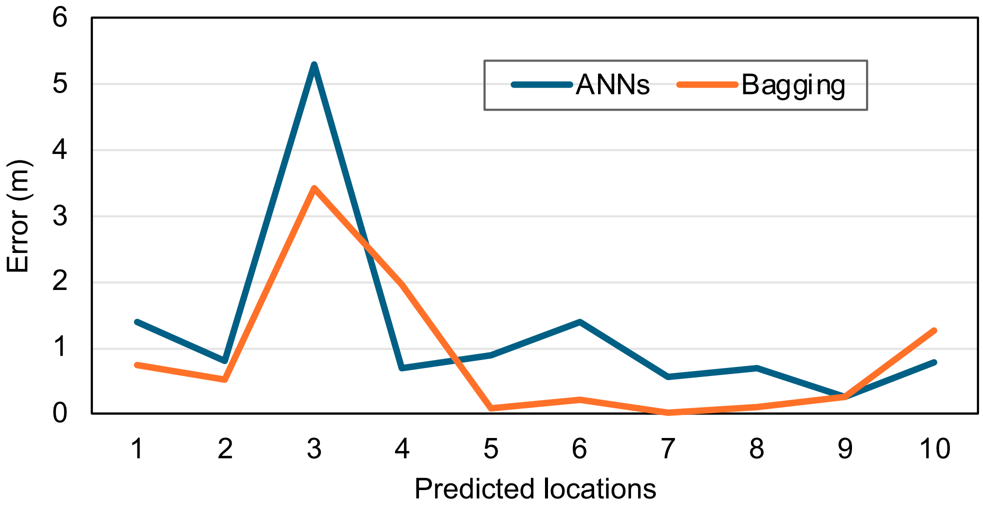

Figure 7. Prediction errors for ten points using ANNs and bagging.

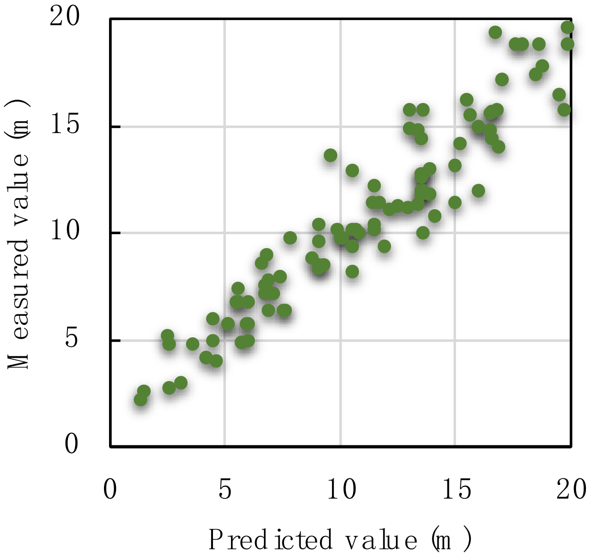

Figure 8. Prediction results for Case 1 using ANNs.

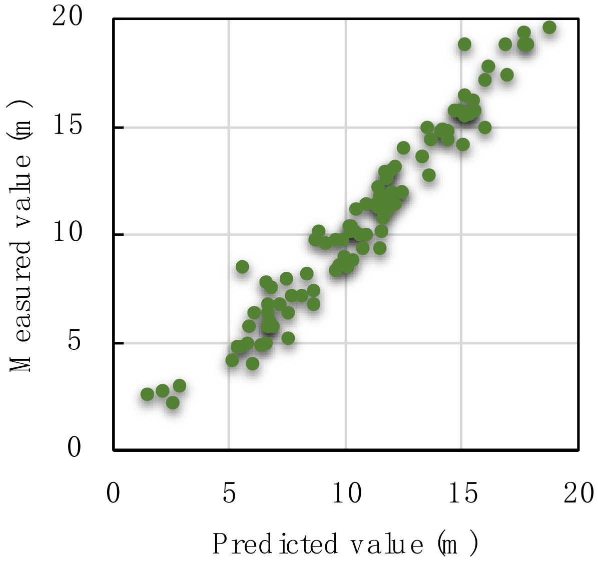

Figure 9. Prediction results for Case 2 using bagging.

5.2. Comparison of Prediction Results between ANNs and Bagging

Table 6 provides a detailed comparison of the prediction results using ANNs and bagging methods, showing that the prediction model based on the bagging method is more accurate than that of ANNs. Table 6 uses four metrics—MAE, MSE, RMSE, and confidence interval—to evaluate the prediction accuracy of the models in Case 1 and Case 2.

Table 6. Results of bearing layer depth prediction using ANNs and bagging.

	MAE	MSE	RMSE
ANNs	1.07	2.89	1.70
Bagging	0.86	1.79	1.34

Mean absolute error (MAE) represents the average of the absolute differences between the actual and predicted values in the data set, measuring the average residuals. Mean squared error (MSE) is the average of the squared differences between the actual and predicted values in the data set. Root mean squared error (RMSE) is the square root of the

MSE and measures the standard deviation of the residuals. The lower the values of MAE, MSE, and RMSE, the more accurate the regression model is, indicating a better fit to the data set.

Based on these metrics, it can be concluded that the predictive model developed using the bagging method is superior. The Confidence Interval (CI) is an estimated range that provides an interval that is likely to contain an unknown population parameter at a given confidence level. In other words, it provides an estimate of the possible value of the population parameter based on the sample data. In this study, a 95% confidence level with a normal distribution was chosen and the confidence interval was calculated accordingly. The equation for the CI is shown in Equation (3):

$$CI = Sample\ mean \pm (Critical\ value \times standard\ error) \tag{3}$$

where, the $Critical\ value$ is 1.96.

The difference in performance between the models motivates a deeper analysis of the strengths and weaknesses of each method, particularly in the context of their application in smart city planning and development.

It can be observed that, compared to a single model, the prediction performance of the ensemble model is improved by about 20%.

A single model refers to a method that uses only one base model for prediction or classification, such as the ANNs used in this study. The advantages of a single model are its simplicity, ease of understanding, and ease of implementation. However, there are also some disadvantages, primarily in the following aspects:

(1) Limited Generalization Ability: A single model is easily affected by data noise, outliers, and overfitting, resulting in poor performance on new data.

(2) Low Stability: A single model is highly sensitive to data distribution and feature selection, meaning slight changes in the data or features can significantly alter the prediction results.

(3) Limited Expressive Power: A single model can often only capture certain aspects of the data, making it difficult to represent the complexity and diversity inherent in the data.

To address the limitations of a single model, an ensemble model can be used to combine multiple base models, thereby improving the prediction performance and generalization ability. In this study, bagging was employed for this purpose. The diversity and complementarity of different models can be leveraged to obtain more robust and accurate predictions. Ensemble learning allows for the integration of the “wisdom” of multiple models by combining their results through voting, weighting, or other techniques, which enhances the model’s resistance to noise and its generalization ability, as concluded by numerous studies [47,48].

Furthermore, individual learners often have different decision boundaries, which may result in errors. By combining multiple learners, a more reasonable decision boundary can be established, reducing the overall error rate and yielding better results. When the dataset is small, partitioning and resampling techniques can be used to generate different data subsets, which are then used to train different learners, ultimately merging them into a stronger model. Additionally, when the data partition boundary is too complex for a single linear model to adequately describe, training multiple models and then fusing them can result in better overall model performance.

The performance of the bagging model has been improved. This improvement is due to factors such as the increased depth of support for observation points, the addition of the bagging process, and the overall improvement of the system. As shown in Figure 10, Figure 3 is divided into four central points, each with a radius of 1 km, to create a contour map of the depth of support within the specified area.

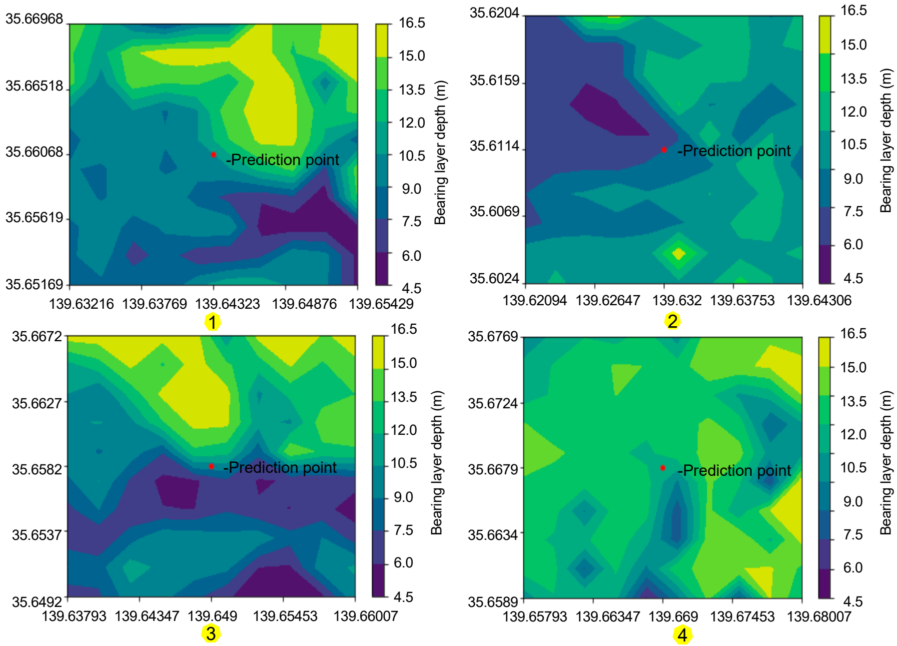

Figure 10. Contour of the bearing layer depth within a 1 km radius at four points (the four numbers are listed in Figure 3).

In the context of smart cities, bagging can provide support in the following areas:

(1) Geological data analysis and forecasting: Smart cities rely on rich data for decision making. Bagging can improve the analysis of geological and other related data by reducing model variance and improving prediction performance.

(2) Hazard detection: Timely detection of anomalies is critical for smart cities. The predictive model of bearing layer depth created by bagging can be used as the basis for creating disaster maps, which in turn helps to detect and respond to abnormal situations more effectively.

(3) Resource optimization: Based on the model developed in this study, Bagging can help optimize resource allocation, such as establishing a trusted bearing layer depth, predicting unknown points before construction, and omitting geological survey steps when a trusted value is exceeded, thereby reducing costs.

These aspects demonstrate the potential of bagging in smart cities, helping city managers make more informed decisions by improving data analysis and forecasting performance.

6. Conclusions

At the core of smart city development lies the critical role of predictive analytics, which uses data to anticipate future scenarios and support decision-making processes. This study establishes a high-precision prediction method for unknown points and areas, demonstrating the significant potential of machine learning in geotechnical engineering. The goal of smart cities is to promote the optimal use of scarce resources and improve the quality of life for residents. Data collection technology is central to advancing smart city planning and achieving these objectives. Data-driven insights enable local governments to improve urban planning and service deployment, thereby enhancing residents' quality of life. This study demonstrates the potential of smart cities to use data for urban improvement. The key findings are as follows:

(1) By using "latitude", "longitude", "altitude", and "bearing layer depth" as input features, high-precision prediction of bearing layer depth was achieved. This accuracy is critical for smart cities, as understanding the geotechnical properties of the ground can significantly impact infrastructure development, from building construction to transportation network design.

(2) Compared to single models such as ANNs, ensemble learning using bagging demonstrated superior prediction performance, with an increase in accuracy of approximately 20%. Bagging enables better data analysis, promoting more effective urban planning.
(3) When employing the ensemble learning method bagging to predict geotechnical engineering survey results, it was found that even small changes in the depth of the training data could significantly affect model performance. This finding underscores the importance of ensuring data accuracy.

However, the current study still has significant limitations. First, the reliability of the prediction results cannot be fully determined. In this study, the prediction performance was evaluated by comparing predictions at measured locations with actual values. However, ensuring the credibility of predictions at completely unknown points remains an important issue for future research. Second, the number of features used in the machine learning model is limited. The three features used in this study are not comprehensive enough to fully represent the depth of the bearing layer. Additional relevant ground conditions will be introduced in future studies. Including more features and ensuring their relevance will help further improve model performance. Since groundwater presence significantly affects liquefaction, future predictions will be divided into two specific scenarios. For coastal areas, a variable representing the distance to the coast will be added, while for non-coastal areas, two variables such as groundwater level depth and distance to the nearest water source will be included. Finally, there are many methods for ensemble learning, and determining the optimal approach remains an important question for future consideration.

This study not only confirms the effectiveness of ensemble learning in geological prediction but also demonstrates its potential in smart city applications.

Author Contributions: Conceptualization, S.I.; methodology, S.I.; software, Y.C.; validation, S.I.; formal analysis, Y.C.; investigation, Y.C.; resources, S.I.; data curation, Y.C.; writing—original draft preparation, Y.C.; writing—review and editing, S.I.; visualization, S.I.; supervision, S.I.; project administration, S.I.; funding acquisition, S.I. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data that support the findings of this study are available from the corresponding author, Shinya Inazumi, upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Cong, Y.; Inazumi, S. Integration of Smart City Technologies with Advanced Predictive Analytics for Geotechnical Investigations. *Smart Cities* **2024**, *7*, 1089–1108. [CrossRef]
2. Lashkari1, A.; Karimi, A.; Fakharian, K.; Kaviani-Hamedani, F. Prediction of undrained behavior of isotropically and anisotropically consolidated firoozkuh sand: In stability and flow liquefaction. *Int. J. Geomech.* **2017**, *17*, 1–17. [CrossRef]
3. Dobry, R.; Abdoun, T. Recent findings on liquefaction triggering in clean and silty sands during earthquakes. *J. Geotech. Geoenviron. Eng.* **2017**, *143*, 1778. [CrossRef]
4. Bao, X.H.; Jin, Z.Y.; Cui, H.Z.; Chen, X.S.; Xie, X.Y. Soil liquefaction mitigation in geotechnical engineering: An overview of recently developed methods. *Soil Dyn. Earthq. Eng.* **2019**, *120*, 273–291. [CrossRef]
5. Seed, H.B.; Idriss, I.M. *Ground Motions and Soil Liquefaction during Earthquakes*; Monograph Series; Earthquake Engineering Research Institute, University of California: Berkeley, CA, USA, 1982.
6. Tatsuoka, F.; Iwasaki, T.; Tokida, K.; Yasuda, S.; Hirose, M.; Imai, T.; Kon-no, M. Standard penetration tests and soil liquefaction potential evaluation. *Soils Found* **1980**, *20*, 95–111. [CrossRef] [PubMed]
7. Youd, T.L.; Perkins, D.M. Mapping liquefaction-induced ground failure potential. *J. Geotech. Eng.* **1978**, *104*, 433–446. [CrossRef]
8. Wakamatsu, K. Liquefaction history, 416–1997, in Japan. In Proceedings of the 12th WCEE, Auckland, New Zealand, 30 January–4 February 2000; p. 2270.
9. Towhata, I.; Taguchi, Y.; Hayashida, T.; Goto, S.; Shintakus, Y.; Hamada, Y.; Aoyama, S. Liquefaction perspective of soil ageing. *Geotechnique* **2016**, *67*, 467–478. [CrossRef]

10. Lo, R.C.; Wang, Y. Lessons learned from recent earthquakes-geoscience and geotechnical perspectives. In *Advances in Geotechnical Earthquake Engineering–Soil Liquefaction and Seismic Safety of Dams and Monuments*; IntechOpen: Rijeka, Croatia, 2012; pp. 1–42. [CrossRef]
11. Hazout, L.; Zitouni, Z.E.A.; Belkhatir, M.; Schanz, T. Evaluation of static liquefaction characteristics of saturated loose sand through the mean grain size and extreme grain sizes. *Geotech. Geol. Eng.* **2017**, *35*, 2079–2105. [CrossRef]
12. Kopetz, H. *Real-Time Systems*; Springer: New York, NY, USA, 2011; pp. 307–323.
13. Yuan, Y.M.; Qin, X.; Wu, C.L.; Tang, T.Z. Architecture and data vitalization of smart city. *Adv. Mater. Res.* **2012**, *403–408*, 2564–2568. [CrossRef]
14. Yin, C.T.; Xiong, Z.; Chen, H.; Wang, J.Y.; Cooper, D.; David, B. A literature survey on smart cities. *Sci. China* **2015**, *58*, 1–18. [CrossRef]
15. Katsuumi, A.; Cong, Y.; Inazumi, S. AI-Driven Prediction and Mapping of Soil Liquefaction Risks for Enhancing Earthquake Resilience in Smart Cities. *Smart Cities* **2024**, *7*, 1836–1856. [CrossRef]
16. Ren, X.; Hou, J.; Song, S.; Liu, Y.; Chen, D.; Wang, X.; Dou, L. Lithology identification using well logs: A method by integrating artificial neural networks and sedimentary patterns. *J. Pet. Sci. Eng.* **2019**, *182*, 106336. [CrossRef]
17. Mienye, I.D.; Sun, Y. A survey of ensemble learning: Concepts, algorithms, applications, and prospects. *IEEE Access* **2022**, *10*, 99129–99149. [CrossRef]
18. Yang, P.; Yang, Y.H.; Zhou, B.B.; Zomaya, A.Y. A review of ensemble methods in bioinformatics. *Curr. Bioinform.* **2010**, *5*, 296–308. [CrossRef]
19. Sun, J.; Li, Q.; Chen, M.; Ren, L.; Huang, G.; Li, C.; Zhang, Z. Optimization of models for a rapid identification of lithology while drilling-A win-win strategy based on machine learning. *J. Pet. Sci. Eng.* **2019**, *176*, 321–341. [CrossRef]
20. Xie, Y.; Zhu, C.; Zhou, W.; Li, Z.; Liu, X.; Tu, M. Evaluation of machine learning methods for formation lithology identification: A comparison of tuning processes and model performances. *J. Pet. Sci. Eng.* **2017**, *160*, 182–193. [CrossRef]
21. Binh, T.P.; DieuTien, B.; Indra, P.; Dholakia, M.B. Evaluation of predictive ability of support vector machines and naive Bayes trees methods for spatial prediction of landslides in Uttarakhand state (India) using GIS. *J. Geomat.* **2016**, *10*, 71–79.
22. Pakawan, C.; Saowanee, W. redicting Urban Expansion and Urban Land Use Changes in Nakhon Ratchasima City Using a CA-Markov Model under Two Different Scenarios. *Land* **2019**, *8*, 140. [CrossRef]
23. Li, H.; Wan, B.; Chu, D.P.; Wang, R.; Ma, G.M.; Fu, J.M.; Xiao, Z.C. Progressive Geological Modeling and Uncertainty Analysis Using Machine Learning. *Int. J. Geo-Inf.* **2023**, *12*, 97. [CrossRef]
24. Zhang, Z.Q.; Wang, G.W.; Carranza, E.; Liu, C.; Li, J.J.; Liu, X.X.; Chen, C.; Fan, J.J.; Dong, Y.L. An integrated machine learning framework with uncertainty quantification for three-dimensional lithological modeling from multi-source geophysical data and drilling data. *Eng. Geol.* **2023**, *324*, 107255. [CrossRef]
25. Shan, S.; Pei, X.; Zhan, W. Estimating Deformation Modulus and Bearing Capacity of Deep Soils from Dynamic Penetration Test. *Adv. Civ. Eng.* **2021**, *2021*, 1082050. [CrossRef]
26. Cong, Y.; Inazumi, S. Ensemble learning for predicting subsurface bearing layer depths in Tokyo. *Results Eng.* **2024**, *23*, 102654. [CrossRef]
27. Salman, R.; Kecman, V. Regression as classification. In Proceedings of the IEEE Southeastcon 2012, Orlando, FL, USA, 15–18 March 2012; pp. 1–6. [CrossRef]
28. Stewart, L.; Bach, F.; Berthet, Q.; Vert, J. Regression as classification: Influence of task formulation on neural network features. In Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS), Valencia, Spain, 25–27 April 2023; p. 206.
29. Kohavi, R.; Provost, F. Glossary of terms. *Mach. Learn.* **1998**, *30*, 271–274.
30. Rogan, J.; Franklin, J.; Stow, D.; Miller, J.; Woodcock, C.; Roberts, D. Mapping land-cover modifications over large areas: A comparison of machine learning algorithms. *Remote Sens. Environ.* **2008**, *112*, 2272–2283. [CrossRef]
31. Breiman, L.; Friedman, J.; Olshen, R.; Stone, C. *Classification and Regression Trees*; Belmont: Wadsworth, IL, USA, 1984.
32. Li, W.; Michael, H. Coastal wetland mapping using ensemble learning algorithms: A comparative study of bagging, boosting and stacking techniques. *Remote Sens.* **2020**, *12*, 1683. [CrossRef]
33. Jalloh, A.B.; Kyuro, S.; Jalloh, Y.; Barrie, A.K. Integrating artificial neural networks and geostatistics for optimum 3D geological block modeling in mineral reserve estimation: A case study. *Int. J. Min. Sci. Technol.* **2016**, *26*, 581–585. [CrossRef]
34. Krawczyk, B.; Minku, L.L.; Gama, J.; Stefanowski, J.; Woźniak, M. Ensemble learning for data stream analysis: A survey. *Inf. Fusion* **2017**, *37*, 132–156. [CrossRef]
35. Dave, V.S.; Dutta, K. Neural network-based models for software effort estimation: A review. *Artif. Intell. Rev.* **2014**, *42*, 295–307. [CrossRef]
36. Izeboudjen, N.; Larbes, C.; Farah, A. A new classification approach for neural networks hardware: From standards chips to embedded systems on chip. *Artif. Intell. Rev.* **2014**, *41*, 491–534. [CrossRef]
37. Oludare, I.A.; Aman, J.; Abiodun, E.O.; Kemi, V.D.; Nachaat, A.M.; Humaira, A. State-of-the-art in artificial neural network applications: A survey. *Heliyon* **2018**, *4*, 11. [CrossRef]
38. He, K.M.; Zhang, X.Y.; Ren, S.Q.; Sun, J. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In Proceedings of the International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1026–1034.

39. Opitz, D.; Maclin, R. Popular ensemble methods: An empirical study. *J. Artif. Intell. Res.* **1999**, *11*, 169–198. [CrossRef]
40. Polikar, R. Ensemble based systems in decision making. *IEEE Circuits Syst. Mag.* **2006**, *6*, 21–45. [CrossRef]
41. Ghimire, B.; Rogan, J.; Galiano, V.R.; Panday, P.; Neeti, N. An evaluation of bagging, boosting, and random forests for land-cover classification in Cape Cod, Massachusetts, USA. *GIScience Remote Sens.* **2012**, *49*, 623–643. [CrossRef]
42. Zhou, Z.H. *Ensemble Methods: Foundations and Algorithms*; CRC Press: Boca Raton, FL, USA; London, UK; New York, NY, USA, 2012.
43. Breiman, L. Bagging predictors. *Mach. Learn.* **1996**, *24*, 123–140. [CrossRef]
44. Kohavi, R.; Wolpert, D.H. Bias plus variance decomposition for zero-one loss functions. In Proceedings of the Thirteenth International Conference on Machine Learning (ICML'96), Bari, Italy, 3–6 July 1996.
45. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
46. Giang, N.; Rodney, B.; Rohitash, C. Evolutionary bagging for ensemble learning. *Neurocomputing* **2022**, *510*, 1–14. [CrossRef]
47. Lun, D.; Xiaozhou, S.; Yanlin, W.; Ensheng, S.; Shi, H.; Dongmei, Z. Is a single model enough? mucos: A multi-model ensemble learning for semantic code search. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management, Gold Coast, QLD, Australia, 1–5 November 2021; pp. 2994–2998. [CrossRef]
48. Xin, Y.; Quansheng, L.; Yucong, P.; Xing, H.; Jian, W.; Xinyu, W. Strength of stacking technique of ensemble learning in rockburst prediction with imbalanced data: Comparison of eight single and ensemble models. *Nat. Resour. Res.* **2021**, *30*, 1795–1815. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.