molecules

MDPI

Review

Ensemble Learning, Deep Learning-Based and Molecular Descriptor-Based Quantitative Structure–Activity Relationships

Yasunari Matsuzaka [1,2] and Yoshihiro Uesawa [1,*]

[1] Department of Medical Molecular Informatics, Meiji Pharmaceutical University, Kiyose 204-8588, Japan
[2] Division of Molecular and Medical Genetics, Center for Gene and Cell Therapy, The Institute of Medical Science, The University of Tokyo, Minato-ku, Tokyo 108-8639, Japan
* Correspondence: uesawa@my-pharm.ac.jp; Tel.: +81-42-495-8983

Abstract: A deep learning-based quantitative structure–activity relationship analysis, namely the molecular image-based DeepSNAP–deep learning method, can successfully and automatically capture the spatial and temporal features in an image generated from a three-dimensional (3D) structure of a chemical compound. It allows building high-performance prediction models without extracting and selecting features because of its powerful feature discrimination capability. Deep learning (DL) is based on a neural network with multiple intermediate layers that makes it possible to solve highly complex problems and improve the prediction accuracy by increasing the number of hidden layers. However, DL models are too complex when it comes to understanding the derivation of predictions. Instead, molecular descriptor-based machine learning has clear features owing to the selection and analysis of features. However, molecular descriptor-based machine learning has some limitations in terms of prediction performance, calculation cost, feature selection, etc., while the DeepSNAP–deep learning method outperforms molecular descriptor-based machine learning due to the utilization of 3D structure information and the advanced computer processing power of DL.

Keywords: DeepSNAP; ensemble learning; neural network; pharmacokinetics; regression model

check for updates

Citation: Matsuzaka, Y.; Uesawa, Y. Ensemble Learning, Deep Learning-Based and Molecular Descriptor-Based Quantitative Structure–Activity Relationships. *Molecules* **2023**, *28*, 2410. https://doi.org/10.3390/molecules28052410

Academic Editors: Huiyong Sun, Peichen Pan and Jingyu Zhu

Received: 6 February 2023
Revised: 28 February 2023
Accepted: 1 March 2023
Published: 6 March 2023

1. Introduction

Machine learning (ML) is a data analysis method that discovers associated regularities and rules through the repeated learning of past cases and data. ML methods can be classified into three types: supervised learning, unsupervised learning, and reinforcement learning [1]. Supervised learning methods involve making a computer learn the correct data and produce the correct output for the input data. In contrast, unsupervised learning methods do not set correct data, and hence the computer itself learns the data characteristics through a large amount of data. In reinforcement learning methods, a computer makes judgments so that the numerical value of the output result is the highest. Therefore, in ML, predictions and inference are made by applying the regularities obtained through learning to unknown and future cases. Furthermore, deep reinforcement learning combines deep learning (DL) and reinforcement learning and is used when a small amount of labeled data and a large amount of unlabeled data are prepared. Supervised learning is divided into two tasks: classification to sort learning data into predetermined classifications and regression to predict future values of continuous data [2,3]. The main purpose of classification is to predict the classification to which the data for analysis belong. However, the main purpose of regression is to make predictions based on trends in continuous values. Regression analysis is a statistical technique that examines the relationship between a result number and a factor number. At this time, the factor value is called the explanatory variable and the resulting value is called the objective variable. An ML model is a mechanism that outputs results for input data and analyzes the input data for evaluation and judgments based on some evaluation criteria. That is, ML is a system that realizes a mechanism equivalent to human learning with computers. Based on a certain calculation method or algorithm, in an ML method, a computer discovers patterns and rules from input data, and by applying those patterns and rules to data, it predicts data. DL is an ML method in which a neural network consisting of many layers is used. The concept of DL is focused on this multilayered neural network, but DL makes it possible for computers to extract feature values themselves when discovering patterns and rules, even if the feature amount is not set in advance. The pattern is recognized by checking how well the input pattern matches the prepared pattern. It is a big breakthrough that feature values are created by themselves without being instructed in this way. In DL in the field of image recognition, the main focus is the combination of a large number of images and a convolutional neural network (CNN), which is a mathematical model that mimics the neural circuits of the brain and returns an output for an input value. In a CNN, the neural network has convolutional layers and pooling layers that function by the characterization of two main factors: recognition in the "local receptive field" and "extraction with weight sharing" [4,5]. When humans recognize an object, they do not grasp the whole image at once but recognize the object little by little by extracting each partial area. The property that responds to the clipped area is called a local receptive field, and a characteristic of a CNN is that it responds to only a small part of the input data, such as the local receptive field. Weight sharing is a mechanism for recognizing the features deemed as important at a specific position in an image as having high importance at another position. CNNs have been applied for classification, object detection, and voice recognition in various fields, including quantitative structure–activity relationship (QSAR). In this review, we summarize the advances in and give an overview of novel QSAR systems, including CNNs, ensemble learning, supervised learning for regression, and molecular descriptor-based ML.

2. Classification of Images by a Neural Network

The term "neural" in neural networks stands for neurons, i.e., nerve cells. It is one of the ML methods that artificially reproduces the mechanism of the cranial nerves with a computer program. DL is a more complex version of these neural networks [6]. A neural network of artificial neurons can be regarded as a function that receives multiple data and outputs the calculated results. For example, if a photograph is captured with a 1-megapixel CCD camera, there are 1,000,000 pieces of data, and if colors are defined by red, green, and blue (RGB) values, the image comprises 3 million data points. By specifying the color information and positions of one-million-pixel data, the photograph can be reproduced. Inputting many data and creating a well-fitting function that can be judged well is supervised ML. By constructing a function that can be judged well, even if a new photo is input, it can be judged with a high probability. A structure called a multilayer perceptron is often used for a neural network, and a more complicated network with 10 or more intermediate layers is used in DL. The input layer has a large amount of data, and the data obtained by multiplying the numerical values by some coefficient and totaling them become the data of the middle layer in the first layer, and the second layer of the middle layer data is obtained by multiplying the data by some coefficient and totaling them. An output is derived from these data, but it involves a huge number of calculations. A function that can be judged by deriving the optimum value of the kind of coefficient combination that can be judged most correctly is completed. In addition, a recurrent network is a type of neural network that has a regressing, recurrent structure that can use the information previously incorporated into the model for predicting continuous data, such as time-series data and language data. DL is a neural network with a large number of intermediate layers. By increasing the number of hidden layers, highly complex problems can be solved and the prediction accuracy can be improved. However, DL models are sometimes called black-box models because they are too complex for humans to understand how the prediction is derived [7].

3. Evaluation of Predictive Models and Predictive Performance

In supervised learning, the collected data are divided into training data and evaluation data for learning. Thus, the separated data evaluation is called "cross validation" [8]. Supervised learning builds a model that makes predictions on other data after a machine learns from human labels for correct answers. The step of constructing a model with a "correct label" is performed using training data, and prediction is performed using evaluation data. When building a model using all training data, a model can be formed that can fit the data but never fit the unknown data that comes later—this phenomenon is called "overfitting" [9]. To prevent overfitting, the data at hand are divided into training data and evaluation data to build and predict the model. For example, in supervised learning for classification, when the evaluation data are classified by the prediction model built from the training data, the degree of correctness among the total data is evaluated. Thus, we consider a model, i.e., line of separation, that is too strict for some training data; however, the accuracy of prediction for different data decreases. Therefore, it is a good idea to balance the goodness of fit with the simplicity of the model. In statistical analysis, information content criteria are set to judge the balance. If the dimensionality or number of variables of the data used to build the model is very large, the combination of variables increases exponentially and the amount of calculations increases tremendously. Then, a sufficient learning performance may not be obtained with the data on hand. This problem is called the "curse of dimensionality"; that is, the ML efficiency decreases because of too many data dimensions [10]. To avoid this problem, multiple variables can be combined into one feature amount (feature amount creation) or the combination of effective feature amounts can be narrowed down (feature amount selection). Thus, overfitting occurs when the dimensionality of the polynomial is too large for the number of training data; that is, the coefficient w tends to take a large value. Therefore, if the coefficient can be restricted to a small value, overfitting can be suppressed. Regularization is a technique based on this idea [11]. A regularization term, namely a penalty term, is added to the sum-of-squares error given in Equation (1) that penalizes the coefficients for growing large (2). Here, λ is the regularization factor and it is an arbitrary value.

$$\mathrm{E(w)} = 1/2\ \|\mathrm{Xw} - \mathrm{t}\|^2 \tag{1}$$

$$\mathrm{E(w)} = 1/2\ \|\mathrm{Xw} - \mathrm{t}\|^2 + \lambda/2\|\mathrm{w}\|^2 \tag{2}$$

The larger the value of w, the larger is the error function. To find a w that minimizes Equation (2), we expand and differentiate with respect to w to obtain the following formula.

$$W = \left(X^T X + \lambda I\right)^{-1X^T t} \tag{3}$$

In Equation (2), $\lambda/2\|\mathrm{w}\|^2$ was added to the sum of squared errors, but the regularization factor can generally be expressed as shown in Equation (5):

$$\frac{1}{2}\lambda \sum_{j=1}^{M} (|wj|)^q \leq \ \eta \tag{4}$$

Equation (2) corresponds to q = 2. Adding a regularization term means limiting the coefficient wj to the range of Equation (5).

$$\sum_{j=1}^{M} (|wj|)^q \leq \ \eta \tag{5}$$

In supervised learning for regression, when performing regression of evaluation data with a model created with training data, the target of evaluation is the difference between the predicted value of the training data and the actual value of the evaluation data. Taking into account the evaluation value of the prediction result, such as the correct answer rate of the output classification and the regression value, we examine whether the model has actual versatility. Typical cross validation methods include the holdout method and the k-fold method [12,13]. The k-fold method is a cross validation method that divides the training data into several (k) pieces and repeats model construction and verification for the number of divided data (i.e., k times). One of the divided data groups is used for validation and the remaining data are used for model building. First, one data group is used for validation, and the first result is checked from the fit of the model built on the rest of the data. In the second round, another data group is used for validation, the rest of the data are used for model building, etc. Model construction and verification are performed for the number of divided data, and the average value of verification that is repeated k times is taken as the result. This prevents models from fitting only to specific data increases by testing different data multiple times for verification. The holdout method is a method adopted for confirming model accuracy and is performed by dividing one dataset into training data and evaluation data. Training data and evaluation data must be separated because they affect the accuracy of the model. In the holdout method, the data used as training data are never used as evaluation data. Similarly, the data used as evaluation data will not be used as training data. In this method, it is necessary to be careful not to confuse the data. Linear regression is a type of regression analysis that predicts the value of a target variable based on the values of another explanatory variable [14,15]. Predicting one target variable with one explanatory variable is called single regression analysis [16]. The relationship between the two datasets that make the prediction can be expressed in the form of a linear equation (Equation (6)), which is the most basic model used in regression [17,18]. If a (slope) and b (Y intercept) are known, y can be predicted from x. The accuracy of the prediction is expressed by the correlation coefficient, i.e., the coefficient of determination [19]. Analysis with two or more explanatory variables, i.e., two or more dimensions, is called multiple regression analysis [20,21]. Multiple selection of appropriate variables makes it possible to set up a prediction formula, i.e., Equation (7), that is easy to calculate and has few errors.

$$\mathrm{y} = \mathrm{ax} + \mathrm{b} \tag{6}$$

$$\mathrm{y} = \mathrm{a1} \times 1 + \mathrm{a2} \times 2 + \mathrm{a3} \times 3 + \mathrm{a4} \times 4 \cdots\cdots + \mathrm{b0} \tag{7}$$

Furthermore, to improve the prediction performance of a model, it is necessary to minimize its generalization error, which can be divided into three parts, namely bias and variance, which are minimizable errors, and noise, which is an irreducible error. This division is called bias–variance decomposition [22,23]. Minimizing bias requires learning more from the training data. However, if the bias is too small, the variance will become large. In contrast, if the variance is reduced too much, the bias will increase. It is necessary to find an optimal solution that balances both. When minimizing two prediction errors, bias and variance, we simultaneously need to consider the tradeoff. Hence, this situation is called the bias–variance dilemma [24]. The relationship between bias and variance follows the relationship that if one side wins, the other side will not win, i.e., the bias–variance tradeoff [25]. Bias in ML and statistics model prediction refers to the difference between the predicted and true values, that is the bias error, arising from incorrect model assumptions, where bias is the error due to the simplification of the actual problem [26]. For example, linear regression simplifies the problem.

Variance refers to the spread of predicted values, that is the variance error, which arises from fluctuations in the training data [27]. If the model prediction has a lot of bias, the model cannot accurately represent the relationship between inputs and outputs. In other words, even training data cannot be predicted accurately; this phenomenon is called underfitting [28]. Additionally, if the variance in the model prediction is too large, the model has learned noise in the training data; in this case, unknown data such as test data cannot be accurately predicted and overfitting occurs [29]. For high accuracy, the variance must be kept low.

4. Ensemble Learning

Ensemble learning (EL) is a method of taking a majority vote and learning to improve the prediction ability for unlearned data by combining the data trained as individual learners [30,31]. It refers to training multiple models and outputting a predicted value by a majority vote or average. Two concepts, namely the bias and variance, are important in EL, which is learning to collect information with low accuracy and increase accuracy. However, if the accuracy does not improve enough, the balance between bias and variance may be poor. The bias is simply the difference between the actual and predictive values. The smaller the difference, the higher the accuracy and the more accurate is the prediction, i.e., low bias results in accurate values, resulting from inadequate training. Variance, on the other hand, simply means the degree to which the predicted values are scattered. A state in which the degree of dispersion is high is called a high variance state, and the accuracy is low in this case. The high variance is due to overfitting, which is caused by overtraining [32].

The model used for EL should be a weak learner, that is a learner with low accuracy when used alone, as the name suggests. In terms of the bias–variance tradeoff (there exists a tradeoff relationship between model complexity and simplification), although the bias–variance tradeoff will fall, if the model is too simple, the generalization performance of the training data cannot be improved, and often the bias is high and the variance is low. That is, a simple model that is not overfitted is obtained. In addition, the characteristics of EL are not only used simply for ML algorithms, such as regression and classification, but also as an auxiliary method when obtaining learning coefficients for other ML algorithms. The effectiveness of EL is that it can take a majority vote using weak learners. In the case of simple binary classification, when classification is performed with a normal classifier, if the classifier misclassifies, an incorrect result will be returned. However, since EL employs a majority decision, if there are m learners, the answer is corrected as long as $(m + 1)/2$ or more learners do not misjudge. For classification problems that are prone to mistakes, EL is very useful because it allows the results of multiple classifiers, such as neural networks, SVM, and naïve Bayes, to be true [33]. Furthermore, assuming that each weak learner is statistically independent, and assuming that the error judgment probability of each weak learner is uniformly θ, out of m weak learners, the probability of k false positives is as follows:

$$P(k) = mCk\theta^{k}(1-\theta)^{m-k} \tag{8}$$

To simplify the explanation, consider that if k of m weak learners make mistakes, the smaller the value of m, the lower the mislearning rate [34]. One of the main EL methods, bagging is a method of training that uses some of the information in the training data rather than all of it and then combines all the training results [35]. Each training can be computed independently, thereby allowing parallel processing. Bagging involves selecting weak learners and merging them into the final learner using the bootstrap method. The basic bagging method is quite simple:

1. Repeat the following steps B times.
    a. Create a new dataset by m-time split sampling from the training data.
    b. Build a weak learner h based on the divided dataset.
2. Construct the final learning result using B times weak learners h.

$$\text{Classification: } H(x) = \text{arg max } |\{i \,|\, hi = y\}| \tag{9}$$

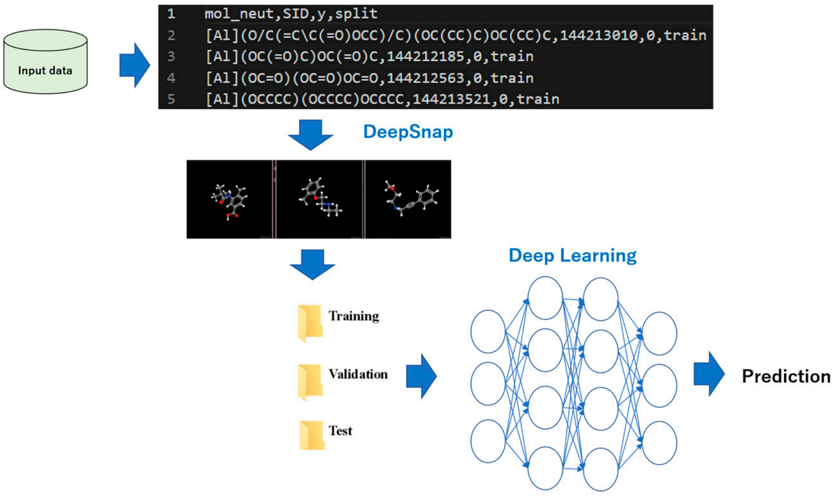
$$\text{Regression}: H(x) = \frac{1}{2B}\sum\nolimits_{i=1}^{B} hi \tag{10}$$

The formula for the part that finalizes the final learning result is given above in (10) and (11); in the case of classification problems, each weak learner is sorted so that the overall accuracy is the highest [36]. On the other hand, for regression, each weak learner is normalized by the overall value. An example of a well-known ML algorithm that uses bagging is the random forest algorithm. Using a part of the training data and merging it at the end is a common feature in bagging; boosting is the process of reusing previously used data to literally provide a boost. Hence, parallelism is not possible as with bagging. Boosting is an EL algorithm that sequentially builds weak learners.

In EL, high accuracy is also achieved by using many weak learners (e.g., decision trees) that do not have high accuracy alone. Bagging uses both the weak learners in parallel and the overall results of each model. In random forest, which combines bagging and decision trees, variance can possibly be suppressed by creating a number of decision tree models for the data and aggregating each result to output the final result.

5. DeepSNAP: DL and EL

QSAR is a method for in silico prediction of chemical substances with physiological activity. In particular, it is one of the key components of integrated toxicology assessment systems, which are highly likely to cause adverse effects to chemical structures and are useful in prioritizing and narrowing down chemicals requiring safety assessment. It can also contribute to alternatives to or minimization of animal testing. In QSAR, the correlation between the structure and activity of compounds such as pharmaceuticals is determined quantitively as numerical values. These values are handled via supervised learning that calculates feature values using chemical information for compounds and builds prediction models [37]. Molecular descriptors, which are the characteristic quantities that reflect the structure of a compound in QSAR, include fingerprints that determine the presence or absence of partial structures and the measured and estimated values of the physicochemical properties of compounds [38]. Using the descriptors, QSAR models are constructed by several algorithms, including random forest, support vector machine, extreme Gradient Boosting (XGBoost), Bayesian networks, multiple linear regression, polynomial regression (PLR), decision tree regression, and neural networks [39,40]. However, modeling in QSAR has some limitations related to the prediction performance, feature selections, calculation cost, etc. Therefore, Prof. Uesawa developed a new deep learning-based QSAR system, DeepSNAP, which generates an omnidirectional snapshot depicting the three-dimensional (3D) structure of chemical compounds (Figure 1) [41]. In DeepSNAP, each chemical structure is optimized for steric conformation and portrayed to depict a ball-and-stick model with different colors representing different atoms. Using datasets of approximately 9000 chemical structures in the simplified molecular input line entry system (SMILES) format and the corresponding activity scores, which represent the agonist or antagonist levels of nuclear receptors and stress response proteins, from a database composed of high-throughput quantitative screening results, two datasets were prepared by defining “active” or “inactive” agonist or antagonist activities. The aforementioned database was derived from the Toxicology in the 21st Century (Tox21) 10k library composed of chemicals from commercial sources, such as pesticides, industrial chemicals, food additives, and drugs [42–49]. Then, the SMILES format was applied to a 3D conformational import to generate the SDF files of the chemical database. In this database, the 3D chemical structures of compounds were depicted as 3D ball-and-stick models and captured continuously and automatically as snapshots with user-defined angle increments on the x-, y-, and z-axes. This was done to extract the spatial and temporal features in the images. Finally, 256 × 256 pixel resolution PNG files (RGB) were saved and automatically split into three datasets, namely the training, validation, and test datasets. Prediction models were created by using training and validation datasets, and the performance with the test dataset was examined. In addition, by optimizing DeepSNAP parameters such as zoom factor, atom size, and molecular bond radius, the performance of the prediction models can be improved because of the powerful feature discrimination capability of DeepSNAP without the need for extracting and selecting features. Furthermore, a combined system of DeepSNAP–DL with molecular descriptor-based methods was reported to construct regression models of rat clearance (CL), i.e., in vivo pharmacokinetic parameters among the parameters of absorption, distribution, metabolism, and excretion. These models outperformed models based on molecular descriptor-based methods (Figure 2) [50,51]. DeepSNAP–DL predicted compound performance from correlations based on large experimental datasets of compounds. ML can perform a correct action in response to input data, collect feedback on how well it performed, and improve it. Despite the significance of CL prediction in the field of drug discovery, few in silico prediction systems with sufficient performance have been established to date. Therefore, this novel system using a combination of DeepSNAP and DL and the molecular descriptor-based methods will be a useful tool for CL prediction. In the ensemble method, a molecular descriptor is selected by DataRobot, in which the feature importance of the descriptors calculated by the permutation importance of the prediction models is more than 0.5 times the highest average effect [50]. Some available EL methods in QSAR that help improve the ML performance have also been reported [52,53].



```
mol_neut,SID,y,split
[Al](O/C(=C\C(=O)OCC)/C)(OC(CC)C)OC(CC)C,144213010,0,train
[Al](OC(=O)C)OC(=O)C,144212185,0,train
[Al](OC=O)(OC=O)OC=O,144212563,0,train
[Al](OCCCC)(OCCCC)OCCCC,144213521,0,train
```

Figure 1. DeepSNAP–DL. The input data of a chemical compound are converted into the SMILES format, and snapshots are produced from different angles as image data. These data are split into three datasets, namely the training, validation, and test datasets, automatically. The prediction model is constructed by DL using these image data.

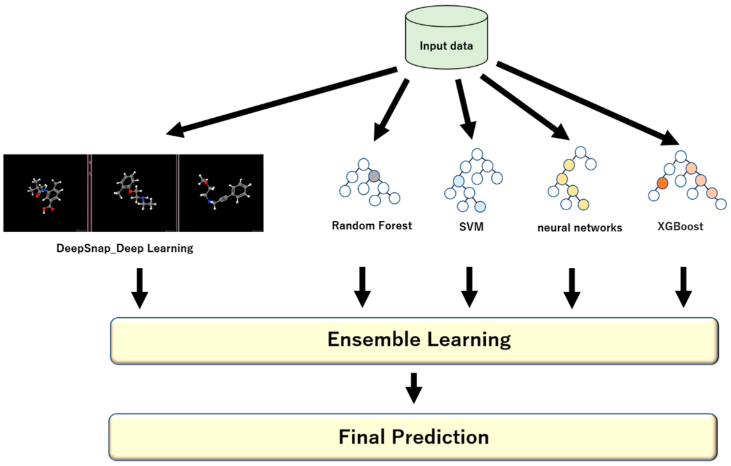


Figure 2. Ensemble learning with DeepSNAP–DL and descriptor-based ML, including random forest, SVM, neural networks, and XGBoost.

6. Application of DL in New Drug Development and Medicine

By using EL consisting of DeepSNAP–DL with molecular descriptor-based methods, a prediction model system for CL, which can be regarded as the blood cleaning speed by the processed organ, with regard to pharmacokinetics (PK) parameters in absorption, distribution, metabolism, and excretion, indicated a high prediction performance [50]. Furthermore, a highly accurate regression analysis was achieved by incorporating DeepSNAP–DL probability as an explanatory variable into the descriptors of the conventional ML methods [51]. In building a CL prediction model using the DeepSNAP–DL method, conditions including four angles (65°, 85°, 105°, and 145°), five learning rates (0.0000001 to 0.001), and five maximum epochs (15 to 300) were considered. As a result, the receiver operating characteristic area under the curve (ROC AUC) with the highest prediction performance was calculated (0.8974) under the conditions of a learning rate of 0.000001, a maximum epoch of 300, and a 145° angle. In addition, the ROC AUC was calculated to be 0.943 in predictive model construction by the ensemble model method, in which the average value of the predicted probabilities obtained by the DeepSNAP–DL method and the descriptor-based random forest method was used as the predicted probability of a new combination of prediction models. Balanced accuracy (0.868), F-measure (0.845), and Matthew's correlation coefficient (0.739) were used as the other evaluation indices. Prediction models are constructed by QSAR analysis for various prediction targets such as toxicity and pharmacokinetic parameters, but the problem is that the prediction accuracy is insufficient. This research focused on rat CL and developed a new prediction accuracy improvement method using the DeepSNAP–DL method with descriptors of conventional ML.

In addition, the application of artificial intelligence technology to the drug discovery field is expected to accelerate new drug development and realize innovative new drugs. However, because drugs and their target proteins have different types of structures, it is difficult to predict drug–protein combinations that are effective in treating diseases with high-throughput and high-performance. Therefore, based on the knowledge of chemistry and biology, molecular interaction prediction methods for drug development are being advanced by combining DL methods. In addition, because conventional methods that do not use the 3D structure of drugs or proteins cannot specify the interaction site from prediction results, there is a problem in interpreting the results. However, new technologies that can identify and visualize interaction sites are expected to accelerate the task of narrowing down new drugs from a large number of candidates. In addition, large-scale searches for drugs candidates by computers are expected to lead to the development of innovative new drugs that cannot be reached with human knowledge and experience alone. In general, drugs can be expressed as atoms and bonds as graph structure data, and proteins can be expressed as amino acid sequences as sequence structure data. Therefore, graph neural networks and CNNs, which are DL methods suitable for each drug data and protein data, respectively, are applied to each data point, by which feature vectors that appropriately capture the properties of the drug and protein are calculated. By learning this feature vector using large-scale data of drugs and proteins, it is possible to predict the presence or absence of interactions [54].

While DL can make fast and highly accurate predictions, it is difficult to interpret the prediction results. When dealing with chemical and biological data, it is necessary to judge the validity of results by comparing the results of automatic predictions made by computers with the chemical and biological knowledge already possessed by humans. Therefore, ML techniques with easy-to-interpret results are important.

7. Conclusions

A deep learning-based QSAR analysis, DeepSNAP, outperformed molecular descriptor-based conventional ML using 3D chemical structures and automatic extraction of features from image data. This system is mainly constructed of a neural network with multilayers, and it is characterized by a high performance of classification and object detection. Furthermore, by combining DeepSNAP–DL and conventional ML (such as EL), regression models can be constructed. The final prediction model obtained by combining multiple weak learning models is constructed. However, EL is not always a versatile learning method. It is important to consider that classifiers are diverse and accurate when bagging is used.

Author Contributions: Y.U. edited the manuscript. Y.M. drafted the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Karlin, E.A.; Lin, C.C.; Meftah, M.; Slover, J.D.; Schwarzkopf, R. The Impact of Machine Learning on Total Joint Arthroplasty Patient Outcomes: A Systemic Review. *J. Arthroplast.* **2022**. [CrossRef]
2. Sourlos, N.; Wang, J.; Nagaraj, Y.; van Ooijen, P.; Vliegenthart, R. Possible Bias in Supervised Deep Learning Algorithms for CT Lung Nodule Detection and Classification. *Cancers* **2022**, *14*, 3867. [CrossRef]
3. Jeng, F.C.; Jeng, Y.S. Implementation of Machine Learning on Human Frequency-Following Responses: A Tutorial. *Semin. Hear.* **2022**, *43*, 251–274. [CrossRef]
4. Ingrosso, A.; Goldt, S. Data-driven emergence of convolutional structure in neural networks. *Proc. Natl. Acad. Sci. USA* **2022**, *119*, e2201854119. [CrossRef] [PubMed]
5. Zhang, Y.; Xie, F.; Song, X.; Zhou, H.; Yang, Y.; Zhang, H.; Liu, J. A rotation meanout network with invariance for dermoscopy image classification and retrieval. *Comput. Biol. Med.* **2022**, *151*, 106272. [CrossRef] [PubMed]
6. Xu, Y. Deep Neural Networks for QSAR. *Methods Mol. Biol.* **2022**, *2390*, 233–260. [CrossRef] [PubMed]
7. Kaveh, M.; Mesgari, M.S. Application of Meta-Heuristic Algorithms for Training Neural Networks and Deep Learning Architectures: A Comprehensive Review. *Neural Process Lett.* **2022**, *in press*. [CrossRef]
8. Yates, L.; Aandahl, Z.; Richards, S.A.; Brook, B.W. Cross validation for model selection: A primer with examples from ecology. *arXiv* **2022**, arXiv:2203.04552v1. Available online: https://arxiv.org/abs/2203.04552 (accessed on 9 March 2022).
9. Cao, Y.; Chen, Z.; Belkin, M.; Gu, Q. Benign Overfitting in Two-layer Convolutional Neural Networks. *arXiv* **2022**, arXiv:2202.06526v3. Available online: https://arxiv.org/abs/2202.06526 (accessed on 14 February 2022).
10. Hou, C.K.J.; Behdinan, K. Dimensionality Reduction in Surrogate Modeling: A Review of Combined Methods. *Data Sci. Eng.* **2022**, *4*, 402–427. [CrossRef]
11. Kukačka, J.; Golkov, V.; Cremers, D. Regularization for Deep Learning: A Taxonomy. *arXiv* **2017**, arXiv:1710.10686v1. Available online: https://arxiv.org/abs/1710.10686 (accessed on 29 October 2017).
12. Raschka, S. Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning. *arXiv* **2018**, arXiv:1811.12808v33. Available online: https://arxiv.org/abs/1811.12808 (accessed on 13 November 2018).
13. Dehghani, A.; Glatard, T.; Shihab, E. Subject Cross Validation in Human Activity Recognition. *arXiv* **2019**, arXiv:1904.02666v2. Available online: https://arxiv.org/abs/1904.02666 (accessed on 4 April 2019).
14. Battey, H.S.; Reid, N. Inference in High-dimensional Linear Regression. *arXiv* **2021**, arXiv:2106.12001v2. Available online: https://arxiv.org/abs/2106.12001 (accessed on 22 June 2021).
15. Brannath, W.; Scharpenberg, M. Interpretation of Linear Regression Coefficients under Mean Model Miss-Specification. *arXiv* **2014**, arXiv:1409.8544v4. Available online: https://arxiv.org/abs/1409.8544 (accessed on 30 September 2014).
16. Gutknecht, A.J.; Barnett, L. Sampling distribution for single-regression Granger causality estimators. *arXiv* **2018**, arXiv:1911.09625v2. Available online: https://arxiv.org/abs/1911.09625 (accessed on 21 November 2019). [CrossRef]
17. Schultheiss, C.; Bühlmann, P. Ancestor regression in linear structural equation models. *arXiv* **2022**, arXiv:2205.08925v2. Available online: https://arxiv.org/abs/2205.08925 (accessed on 18 May 2022). [CrossRef]
18. Yevkin, G.; Yevkin, O. On regression analysis with Padé approximants. *arXiv* **2022**, arXiv:2208.09945v1. Available online: https://arxiv.org/abs/2208.09945 (accessed on 21 August 2022).
19. Choi, J.-E.; Shin, D.W. Quantile correlation coefficient: A new tail dependence measure. *arXiv* **2018**, arXiv:1803.06200v1. Available online: https://arxiv.org/abs/1803.06200 (accessed on 16 May 2018). [CrossRef]
20. O'Neill, B. Multiple Linear Regression and Correlation: A Geometric Analysis. *arXiv* **2021**, arXiv:2109.08519v1. Available online: https://arxiv.org/abs/2109.08519 (accessed on 13 September 2021).
21. Gupta, I.; Mittal, H.; Rikhari, D.; Singh, A.K. MLRM: A Multiple Linear Regression based Model for Average Temperature Prediction of A Day. *arXiv* **2022**, arXiv:2203.05835v1. Available online: https://arxiv.org/abs/2203.05835 (accessed on 11 March 2022).
22. Rocks, J.W.; Mehta, P. Bias-variance decomposition of overparameterized regression with random linear features. *Phys. Rev. E* **2022**, *106*, 025304. [CrossRef]
23. Gao, J. Bias-variance decomposition of absolute errors for diagnosing regression models of continuous data. *Patterns* **2021**, *2*, 100309. [CrossRef] [PubMed]
24. Voncken, L.; Albers, C.J.; Timmerman, M.E. Bias-Variance Trade-Off in Continuous Test Norming. *Assessment* **2021**, *28*, 1932–1948. [CrossRef] [PubMed]
25. Zhang, J.; Li, J. Mitigating Bias and Error in Machine Learning to Protect Sports Data. *Comput. Intell. Neurosci.* **2022**, *2022*, 4777010. [CrossRef] [PubMed]
26. Zhang, W.; Dimiccoli, M.; Lim, B.Y. Debiased-CAM to mitigate systematic error with faithful visual explanations of machine learning. *arXiv* **2022**, arXiv:2201.12835v2. Available online: https://arxiv.org/abs/2201.12835 (accessed on 30 January 2022).

27. Bashir, D.; Montanez, G.D.; Sehra, S.; Segura, P.P.; Lauw, J. An Information-Theoretic Perspective on Overfitting and Underfitting. *arXiv* **2020**, arXiv:2010.06076v2. Available online: https://arxiv.org/abs/2010.06076 (accessed on 12 October 2020).
28. Li, Z.; Liu, L.; Dong, C.; Shang, J. Overfitting or Underfitting? Understand Robustness Drop in Adversarial Training. *arXiv* **2020**, arXiv:2010.08034v1. Available online: https://arxiv.org/abs/2010.08034 (accessed on 15 October 2020).
29. Zhu, X.; Hu, J.; Xiao, T.; Huang, S.; Wen, Y.; Shang, D. An interpretable stacking ensemble learning framework based on multi-dimensional data for real-time prediction of drug concentration: The example of olanzapine. *Front. Pharmacol.* **2022**, *13*, 975855. [CrossRef]
30. Suri, J.S.; Bhagawati, M.; Paul, S.; Protogerou, A.D.; Sfikakis, P.P.; Kitas, G.D.; Khanna, N.N.; Ruzsa, Z.; Sharma, A.M.; Saxena, S.; et al. A Powerful Paradigm for Cardiovascular Risk Stratification Using Multiclass, Multi-Label, and Ensemble-Based Machine Learning Paradigms: A Narrative Review. *Diagnostics* **2022**, *12*, 722. [CrossRef]
31. Ghiasi, M.M.; Zendehboudi, S. Application of decision tree-based ensemble learning in the classification of breast cancer. *Comput. Biol. Med.* **2021**, *128*, 104089. [CrossRef] [PubMed]
32. Ghojogh, B.; Crowley, M. The Theory Behind Overfitting, Cross Validation, Regularization, Bagging, and Boosting: Tutorial. *arXiv* **2019**, arXiv:1905.12787v1. Available online: https://arxiv.org/abs/1905.12787 (accessed on 28 May 2019).
33. Chang, O.; Yao, Y.; Williams-King, D.; Lipson, H. Ensemble Model Patching: A Parameter-Efficient Variational Bayesian Neural Network. *arXiv* **2019**, arXiv:1905.09453v1. Available online: https://arxiv.org/abs/1905.09453 (accessed on 23 May 2019).
34. Kumar, R.; Subbiah, G. Zero-Day Malware Detection and Effective Malware Analysis Using Shapley Ensemble Boosting and Bagging Approach. *Sensors* **2022**, *22*, 2798. [CrossRef]
35. Lin, E.; Lin, C.H.; Lane, H.Y. A bagging ensemble machine learning framework to predict overall cognitive function of schizophrenia patients with cognitive domains and tests. *Asian J. Psychiatr.* **2022**, *69*, 103008. [CrossRef] [PubMed]
36. Ngo, G.; Beard, R.; Chandra, R. Evolutionary bagging for ensemble learning. *arXiv* **2022**, arXiv:2208.02400v3. Available online: https://arxiv.org/abs/2208.02400 (accessed on 4 August 2022). [CrossRef]
37. Song, H.; Dong, C.; Zhang, X.; Wu, W.; Chen, C.; Ma, B.; Chen, F.; Chen, C.; Lv, X. Rapid identification of papillary thyroid carcinoma and papillary microcarcinoma based on serum Raman spectroscopy combined with machine learning models. *Photodiagn. Photodyn. Ther.* **2022**, *37*, 102647. [CrossRef] [PubMed]
38. Yang, J.; Cai, Y.; Zhao, K.; Xie, H.; Chen, X. Concepts and applications of chemical fingerprint for hit and lead screening. *Drug Discov. Today* **2022**, *27*, 103356. [CrossRef]
39. Bamisile, O.; Cai, D.; Oluwasanmi, A.; Ejiyi, C.; Ukwuoma, C.C.; Ojo, O.; Mukhtar, M.; Huang, Q. Comprehensive assessment, review, and comparison of AI models for solar irradiance prediction based on different time/estimation intervals. *Sci. Rep.* **2022**, *12*, 9644. [CrossRef]
40. Zhao, X.; Lu, Y.; Li, S.; Guo, F.; Xue, H.; Jiang, L.; Wang, Z.; Zhang, C.; Xie, W.; Zhu, F. Predicting renal function recovery and short-term reversibility among acute kidney injury patients in the ICU: Comparison of machine learning methods and conventional regression. *Ren. Fail.* **2022**, *44*, 1326–1337. [CrossRef]
41. Uesawa, Y. Quantitative structure-activity relationship analysis using deep learning based on a novel molecular image input technique. *Bioorg. Med. Chem. Lett.* **2018**, *28*, 3400–3403. [CrossRef] [PubMed]
42. Matsuzaka, Y.; Uesawa, Y. A Deep Learning-Based Quantitative Structure-Activity Relationship System Construct Prediction Model of Agonist and Antagonist with High Performance. *Int. J. Mol. Sci.* **2022**, *23*, 2141. [CrossRef]
43. Matsuzaka, Y.; Totoki, S.; Handa, K.; Shiota, T.; Kurosaki, K.; Uesawa, Y. Prediction Models for Agonists and Antagonists of Molecular Initiation Events for Toxicity Pathways Using an Improved Deep-Learning-Based Quantitative Structure-Activity Relationship System. *Int. J. Mol. Sci.* **2021**, *22*, 10821. [CrossRef] [PubMed]
44. Matsuzaka, Y.; Uesawa, Y. A Molecular Image-Based Novel Quantitative Structure-Activity Relationship Approach, Deepsnap-Deep Learning and Machine Learning. *Curr. Issues Mol. Biol.* **2021**, *42*, 455–472. [CrossRef] [PubMed]
45. Matsuzaka, Y.; Uesawa, Y. Molecular Image-Based Prediction Models of Nuclear Receptor Agonists and Antagonists Using the DeepSnap-Deep Learning Approach with the Tox21 10K Library. *Molecules* **2020**, *25*, 2764. [CrossRef]
46. Matsuzaka, Y.; Hosaka, T.; Ogaito, A.; Yoshinari, K.; Uesawa, Y. Prediction Model of Aryl Hydrocarbon Receptor Activation by a Novel QSAR Approach, DeepSnap-Deep Learning. *Molecules* **2020**, *25*, 1317. [CrossRef]
47. Matsuzaka, Y.; Uesawa, Y. DeepSnap-Deep Learning Approach Predicts Progesterone Receptor Antagonist Activity with High Performance. *Front. Bioeng. Biotechnol.* **2020**, *7*, 485. [CrossRef]
48. Matsuzaka, Y.; Uesawa, Y. Prediction Model with High-Performance Constitutive Androstane Receptor (CAR) Using DeepSnap-Deep Learning Approach from the Tox21 10K Compound Library. *Int. J. Mol. Sci.* **2019**, *20*, 4855. [CrossRef]
49. Matsuzaka, Y.; Uesawa, Y. Optimization of a Deep-Learning Method Based on the Classification of Images Generated by Parameterized Deep Snap a Novel Molecular-Image-Input Technique for Quantitative Structure-Activity Relationship (QSAR) Analysis. *Front. Bioeng. Biotechnol.* **2019**, *7*, 65. [CrossRef]
50. Mamada, H.; Nomura, Y.; Uesawa, Y. Prediction Model of Clearance by a Novel Quantitative Structure-Activity Relationship Approach, Combination DeepSnap-Deep Learning and Conventional Machine Learning. *ACS Omega* **2021**, *6*, 23570–23577. [CrossRef]
51. Mamada, H.; Nomura, Y.; Uesawa, Y. Novel QSAR Approach for a Regression Model of Clearance That Combines DeepSnap-Deep Learning and Conventional Machine Learning. *ACS Omega* **2022**, *7*, 17055–17062. [CrossRef] [PubMed]

52. Daghighi, A.; Casanola-Martin, G.M.; Timmerman, T.; Milenković, D.; Lučić, B.; Rasulev, B. In Silico Prediction of the Toxicity of Nitroaromatic Compounds: Application of Ensemble Learning QSAR Approach. *Toxics* **2022**, *10*, 746. [CrossRef] [PubMed]
53. Chen, C.H.; Tanaka, K.; Kotera, M.; Funatsu, K. Comparison and improvement of the predictability and interpretability with ensemble learning models in QSPR applications. *J. Cheminform.* **2020**, *12*, 19. [CrossRef] [PubMed]
54. Tsubaki, M.; Tomii, K.; Sese, J. Compound-protein interaction prediction with end-to-end learning of neural networks for graphs and sequences. *Bioinformatics* **2019**, *35*, 309–318. [CrossRef] [PubMed]

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.