ARTIFICIAL INTELLIGENCE IN CHEMISTRY

EDITED BY: José S. Torrecilla, John C. Cancilla, Jose Omar Valderrama
and Charalampos Vasilios Proestos

PUBLISHED IN: Frontiers in Chemistry

frontiers Research Topics

ISSN 1664-8714
ISBN 978-2-88963-870-3
DOI 10.3389/978-2-88963-870-3

About Frontiers

Frontiers is more than just an open-access publisher of scholarly articles: it is a pioneering approach to the world of academia, radically improving the way scholarly research is managed. The grand vision of Frontiers is a world where all people have an equal opportunity to seek, share and generate knowledge. Frontiers provides immediate and permanent online open access to all its publications, but this alone is not enough to realize our grand goals.

Frontiers Journal Series

The Frontiers Journal Series is a multi-tier and interdisciplinary set of open-access, online journals, promising a paradigm shift from the current review, selection and dissemination processes in academic publishing. All Frontiers journals are driven by researchers for researchers; therefore, they constitute a service to the scholarly community. At the same time, the Frontiers Journal Series operates on a revolutionary invention, the tiered publishing system, initially addressing specific communities of scholars, and gradually climbing up to broader public understanding, thus serving the interests of the lay society, too.

Dedication to Quality

Each Frontiers article is a landmark of the highest quality, thanks to genuinely collaborative interactions between authors and review editors, who include some of the world's best academicians. Research must be certified by peers before entering a stream of knowledge that may eventually reach the public - and shape society; therefore, Frontiers only applies the most rigorous and unbiased reviews.
Frontiers revolutionizes research publishing by freely delivering the most outstanding research, evaluated with no bias from both the academic and social point of view. By applying the most advanced information technologies, Frontiers is catapulting scholarly publishing into a new generation.

What are Frontiers Research Topics?

Frontiers Research Topics are very popular trademarks of the Frontiers Journals Series: they are collections of at least ten articles, all centered on a particular subject. With their unique mix of varied contributions from Original Research to Review Articles, Frontiers Research Topics unify the most influential researchers, the latest key findings and historical advances in a hot research area! Find out more on how to host your own Frontiers Research Topic or contribute to one as an author by contacting the Frontiers Editorial Office: researchtopics@frontiersin.org

ARTIFICIAL INTELLIGENCE IN CHEMISTRY

Topic Editors:
José S. Torrecilla, Complutense University of Madrid, Spain
John C. Cancilla, Scintillon Institute, United States
Jose Omar Valderrama, Centro de Información Tecnológica (CIT), Chile
Charalampos Vasilios Proestos, National and Kapodistrian University of Athens, Greece

Citation: Torrecilla, J. S., Cancilla, J. C., Valderrama, J. O., Proestos, C. V., eds. (2020). Artificial Intelligence in Chemistry. Lausanne: Frontiers Media SA. doi: 10.3389/978-2-88963-870-3

Table of Contents

EDITORIAL
published: 09 April 2020
doi: 10.3389/fchem.2020.00275

Editorial: Artificial Intelligence in Chemistry

John C. Cancilla[1], José S. Torrecilla[2]*, Charalampos Vasilios Proestos[3] and José Omar Valderrama[4]*

[1] Scintillon Institute, San Diego, CA, United States, [2] Departamento de Ingeniería Química y de Materiales, Universidad Complutense de Madrid, Madrid, Spain, [3] Laboratory of Food Chemistry, Department of Chemistry, National and Kapodistrian University of Athens, Athens, Greece, [4] Centro de Información Tecnológica (CIT), La Serena, Chile

Keywords: artificial intelligence, machine learning, artificial neural networks, deep learning, modern chemistry, chemical applications

Editorial on the Research Topic

Artificial Intelligence in Chemistry

Within our Research Topic, six unique manuscripts which contain different trained machine and deep learning algorithms to model chemical processes have been published. The optimized intelligent tools cover applications within several scopes including (i) chemical and physicochemical molecule property predictions, (ii) compound ranking, identification, and classification, (iii) monitoring and aiding drug discovery, as well as (iv) quality evaluation and classification of chemicals and foods.

In the successive paragraphs, each of the accepted publications are presented in chronological order and briefly described:

1) **e-Sweet: A Machine-Learning Based Platform for the Prediction of Sweetener and Its Relative Sweetness** (Zheng et al.). The authors of this work have designed and made available a free machine learning software platform called "e-Sweet" that predicts the relative sweetness of different molecules. They used a database containing the structures of many different compounds, both sweeteners and non-sweeteners, to train an array of machine learning models (e.g., support vector machine, random forest, or deep neural networks) that label each molecule tested with a relative sweetness value. Their aspiration is to empower food scientists to discover and develop new molecules with enhanced sweetness by harnessing the power of their intelligent platform.
2) **Deep Neural Network Classifier for Virtual Screening Inhibitors of (S)-Adenosyl-L-Methionine (SAM)-Dependent Methyltransferase Family** (Li et al.). In this study, the research team developed a deep learning-based neural network model to classify active vs. inactive compounds in relation to their ability to inhibit SAM-dependent methyltransferases. These targets are enzymes that possess a relevant epigenetic role and are pharmacologically significant as they are involved in the pathogenesis of several genetic disorders as well as cancer. To train their model, 12 unique targets (methyltransferases) were analyzed, using up to 1,740 different ligands (potential inhibitors) as samples to be classified, reaching improved statistical performances when compared to previous studies.
3) **Neural Networks Are Promising Tools for the Prediction of the Viscosity of Unsaturated Polyester Resins** (Molina et al.). Here, a neural network model was designed and optimized to determine a physicochemical property such as viscosity of unsaturated polyester resins, which are employed to synthesize composite materials. Viscosities are directly related to the performance of these materials, which leads to the intrinsic value of the accurate intelligent mathematical algorithm developed for the industry.

Edited and reviewed by:
Thomas S. Hofer,
University of Innsbruck, Austria

****Correspondence:***
John C. Cancilla
jcancilla@scintillon.org
José S. Torrecilla
jstorre@ucm.es

Specialty section:
This article was submitted to Theoretical and Computational Chemistry, a section of the journal Frontiers in Chemistry

Received: *14 January 2020*
Accepted: *20 March 2020*
Published: *09 April 2020*

Citation:
Cancilla JC, Torrecilla JS, Proestos CV and Valderrama JO (2020) Editorial: Artificial Intelligence in Chemistry. Front. Chem. 8:275. doi: 10.3389/fchem.2020.00275

4) **Prediction of the Antioxidant Response Elements' Response of Compound by Deep Learning** (Bai et al.). During this research, the authors trained several deep learning algorithms to identify compounds that can hypothetically activate antioxidant response elements, which may lead to elevated toxicities linked to the appearance of oxidative stress. The strong performance offered by the team's optimized deep neural network (their most accurate model) implies the usefulness of machine learning to assess the safety of novel drugs and their future development, as molecules that potentially activate antioxidant response elements could be screened out.
5) **Development of Predictive Models for Identifying Potential S100A9 Inhibitors Based on Machine Learning Methods** (Lee et al.). In this work, the researchers analyzed a large dataset containing over six million compounds with the goal set to identify potential S100A9 inhibitors via machine learning algorithms including random forest classifiers. Their intelligent tool is relevant as S100A9 has been identified as a therapeutic target for various diseases including cancer and Alzheimer's, reason why facilitating the discovery of inhibiting drugs while vastly reducing costs is highly valuable for the field.
6) **Deep Learning Techniques to Improve the Performance of Olive Oil Classification** (Vega-Márquez et al.). In this final article, the authors cover the use of deep learning neural networks to classify olive oil samples in terms of quality by using data gathered via gas chromatography coupled to ion mobility spectrometry of over 700 samples to train the algorithms. The basic goal of their work is to reach tools that can help ensure the safety of olive oils in terms of health (being suitable for human consumption) and to avoid fraud (selling lower grade products as high quality ones).

These six articles are great examples which showcase the application of artificial intelligence in the shape of mathematical algorithms and machine learning to solve different technological and/or scientific problems of the chemical field. These manuscripts help readers understand the usefulness of these intelligent models and empowers them to design such tools to extract the most out of their experimental results to tackle problems of their own specific lines of research or technological development.

Artificial intelligence is offering alternatives and solutions that catalyze the creation and implementation of applications that would have been unconceivable or even impossible only 10 years ago. For this reason, many novel chemical industries, research projects, and ideas can greatly benefit from the inclusion of artificial intelligence, setting new frontiers while reaching a modernized and intelligent field of chemistry.

AUTHOR CONTRIBUTIONS

JC and JT wrote the editorial while CP and JV reviewed it.

Conflict 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.

ORIGINAL RESEARCH
published: 30 January 2019
doi: 10.3389/fchem.2019.00035

e-Sweet: A Machine-Learning Based Platform for the Prediction of Sweetener and Its Relative Sweetness

Suqing Zheng[1,2]*, *Wenping Chang*[1], *Wenxin Xu*[1], *Yong Xu*[3] *and Fu Lin*[1]*

[1] School of Pharmaceutical Sciences, Wenzhou Medical University, Wenzhou, China, [2] Chemical Biology Research Center, Wenzhou Medical University, Wenzhou, China, [3] Center of Chemical Biology, Guangzhou Institutes of Biomedicine and Health, Chinese Academy of Sciences, Guangzhou, China

Artificial sweeteners (AS) can elicit the strong sweet sensation with the low or zero calorie, and are widely used to replace the nutritive sugar in the food and beverage industry. However, the safety issue of current AS is still controversial. Thus, it is imperative to develop more safe and potent AS. Due to the costly and laborious experimental-screening of AS, *in-silico* sweetener/sweetness prediction could provide a good avenue to identify the potential sweetener candidates before experiment. In this work, we curate the largest dataset of 530 sweeteners and 850 non-sweeteners, and collect the second largest dataset of 352 sweeteners with the relative sweetness (RS) from the literature. In light of these experimental datasets, we adopt five machine-learning methods and conformational-independent molecular fingerprints to derive the classification and regression models for the prediction of sweetener and its RS, respectively via the consensus strategy. Our best classification model achieves the 95% confidence intervals for the accuracy (0.91 ± 0.01), precision (0.90 ± 0.01), specificity (0.94 ± 0.01), sensitivity (0.86 ± 0.01), F1-score (0.88 ± 0.01), and NER (Non-error Rate: 0.90 ± 0.01) on the test set, which outperforms the model (NER = 0.85) of Rojas et al. in terms of NER, and our best regression model gives the 95% confidence intervals for the R^2(test set) and ΔR^2 [referring to $|R^2$(test set)- R^2(cross-validation)$|$] of 0.77 ± 0.01 and 0.03 ± 0.01, respectively, which is also better than the other works based on the conformation-independent 2D descriptors (e.g., 2D Dragon) according to R^2(test set) and ΔR^2. Our models are obtained by averaging over nineteen data-splitting schemes, and fully comply with the guidelines of Organization for Economic Cooperation and Development (OECD), which are not completely followed by the previous relevant works that are all on the basis of only one random data-splitting scheme for the cross-validation set and test set. Finally, we develop a user-friendly platform "e-Sweet" for the automatic prediction of sweetener and its corresponding RS. To our best knowledge, it is a first and free platform that can enable the experimental food scientists to exploit the current machine-learning methods to boost the discovery of more AS with the low or zero calorie content.

Keywords: sweet taste, sweetener prediction, relative sweetness prediction, machine learning method, QSAR

OPEN ACCESS

Edited by:
John C. Cancilla,
Scintillon Institute, United States

Reviewed by:
Ramesh L. Gardas,
Indian Institute of Technology Madras, India
Hamid Reza Marateb,
Universitat Politecnica de Catalunya, Spain

***Correspondence:**
Suqing Zheng
zsq_2016@126.com
Fu Lin
lin1449@126.com

Specialty section:
This article was submitted to Theoretical and Computational Chemistry,
a section of the journal Frontiers in Chemistry

Received: *22 August 2018*
Accepted: *14 January 2019*
Published: *30 January 2019*

Citation:
Zheng S, Chang W, Xu W, Xu Y and Lin F (2019) e-Sweet: A Machine-Learning Based Platform for the Prediction of Sweetener and Its Relative Sweetness. Front. Chem. 7:35. doi: 10.3389/fchem.2019.00035

INTRODUCTION

Sweet taste, eliciting a pleasant sensation, provides an instinctive means to find the energy source such as the carbohydrates, which usually taste sweet. The taste perception of the sweetness is a complex mechanism involving the multiple disciplines (e.g., chemistry, biology, and physiology), however, it is generally assumed to be predominantly mediated by the taste receptors type 1 (Tas1Rs) on the taste buds in the oral cavity (Roper and Chaudhari, 2017). Interestingly, Tas1Rs are also expressed in numerous different organs (e.g., gut and pancreas), implicating that they are intricately participated in various physiological processes such as intestinal absorption, glucose homeostasis, and metabolic regulation (Laffitte et al., 2014).

Human sweet taste receptor (*h*STR) functions as a heterodimer of two subunits (*h*Tas1R2 and *h*Tas1R3) belonging to the class C family of G-protein coupled receptors (GPCRs), whereas each subunit contains three distinct domains: a large extracellular venus flytrap domain (VFD), a short cysteine-rich domain (CRD), and seven-transmembrane domain (TMD) (Meyers and Brewer, 2008). *h*STR harbors at least four different binding sites revealed by the biochemical characterization such as the chimeras or site-directed mutagenesis experiment, and thereby can recognize a variety of sweeteners (Masuda et al., 2012): sugars (e.g., sucrose and glucose), amino acids (e.g., D-trypotophan and D-glycine), artificial sweeteners (e.g., saccharin and aspartame), and sweet proteins (e.g., monellin and thaumatin). According to the content of calorie, these chemically diverse sweeteners can be generally categorized into two types (Dubois and Prakash, 2012): the nutritive sweeteners with the high calorie (e.g., sucrose), and the non-nutritive sweeteners (e.g., saccharin and aspartame) with the low or zero calorie that mainly refer to the artificial sweeteners in this work.

Nowadays, the non-nutritive sweeteners are broadly used as the food additives to substitute for the nutritive sweeteners such as sucrose, since the over-consumption of high-calorie nutritive sweeteners in the functional food and beverage will lead to the elevated risks of the metabolic disorders (e.g., type II diabetes) and cardiovascular diseases (Fernstrom, 2015). Therefore, a multitude of non-nutritive sweeteners with the low calorie yet preserving the sweetness have been manually synthesized or directly extracted from the natural plants to prevent these risks.

Hitherto, none of the currently available non-nutritive sweeteners (especially the artificial sweeteners) can accurately replicate the same sweetness profile (e.g., concentration/response function, temporal profile, and adaption behaviors) of the natural sucrose (Dubois, 2016), since they usually exhibit the slow sweetness onset, lingering sweetness aftertaste, apparent off-taste, or moderate/strong adaption upon the iterative tasting, which are generally not preferred by most of consumers. Moreover, the heavy use of the artificial sweeteners, one major class of non-nutritive sweeteners, are reported to cause some side-effects such as an increased risk of cancer in human (Mishra et al., 2015). Therefore, it is still desirable to discover more novel and safe non-nutritive sweeteners.

As we know, the sweetener discovery using the human taste-panel or cell-based high-throughput screening is an expensive, laborious and slow process. Hence *in-silico* sweetener prediction could be a good alternative to rapidly identify the most likely sweetener candidates with the high potency prior to the time-consuming and arduous experiment. Currently, there are two main computational methods for the sweetener prediction: structure-based and ligand-based methods. Structure-based method is to rationally design the compound based on *h*STR. Nevertheless, the crystal structure of full *h*STR is still unraveled, albeit there are several homology models based on the templates with the limited sequence identities (Shrivastav and Srivastava, 2013; Jean-Baptiste et al., 2017; Kim et al., 2017; Acevedo et al., 2018). In addition, a compound that can bind with *h*STR could be also a sweetness inhibitor (e.g., lactisole) (Jiang et al., 2005), rather than the sweetener of our interest. However, the data-driven machine-learning method, emerging as a vibrant area of ligand-based method, can directly predict the sweetener and its relative sweetness (RS), provided that there is sufficient experimental dataset to build the predictive model. More specifically, the sweetener/non-sweetener classification models based on the machine-learning methods can be employed to predict the sweetener, and the regression models derived from the machine-learning methods can be utilized to forecast the RS of the sweetener.

Rojas et al. comprehensively review the sweet/bitter (Rojas et al., 2016a; Banerjee and Preissner, 2018), sweet/tasteless (Rojas et al., 2016a), and sweet/sweetless (Rojas et al., 2017) classification models, and also provide a systematic overview on the regression models for the RS prediction of sweetener (Rojas et al., 2016a,b,c). In our study, only the typical works about the sweet/sweetless classification model on the relatively large dataset are briefly summarized here, because the sweet/sweetless pair is more reasonable and practical for the sweetener prediction due to the inclusion of bitter and tasteless compounds into the sweetless dataset. Meanwhile, only the representative works regarding to the regression model also on the basis of the comparatively large dataset will be shortly recapitulated in our study, while the pioneering works of the sweeteners prediction model based on the congeneric systems or small dataset, contributing significantly to the subsequent works in this research area, have been thoroughly summarized in Rojas et al. (2016b) and thereby will be not reviewed here due to the limited space. It should be noted that only the works about the classification and regression models on the basis of the comparatively large dataset will be shortly reviewed in this study and the relatively large dataset here refers to the dataset with at least two hundreds samples, since the relatively large dataset affords the more extended applicability-domain of model.

As for the classification model, Rojas et al. develop the sweet/sweetless classification model based on the relatively large dataset (649 compounds) consisting of 435 sweeteners and 214 non-sweeteners (133 tasteless and 81 bitterants). In their work, the partial least squares discriminant analysis (PLSDA) and K-nearest neighbors (KNN) coupled with the 2D Dragon descriptors (https://chm.kode-solutions.net/) are used to train the models, respectively, which are combined to form a consensus model. Their consensus model gives the sensitivity, specificity and NER (Non-Error Rate, the average of sensitivity and specificity in the binary classification) of 0.88, 0.82, and 0.85, respectively on the test set including 108 sweeteners and 53 non-sweeteners that are randomly selected from the whole dataset (Rojas et al., 2017). However, only 81 bitterants are adopted as the sweetless compounds in their work. Hence the numerous bitterants curated by BitterDB (Wiener et al., 2012) could be treated as the sweetless compounds to further leverage the applicability-domain of the sweeteners/non-sweeteners classification model.

Regarding to the regression model, Zhong et al. (2013) build the regression models based on the comparatively large dataset including the 320 sweeteners (214 for the training set and 106 for the hold-out test set) with RS. The regression models are trained with the multi-linear regression (MLR) and support vector machine (SVM), respectively in combination with the mixed 2D and 3D descriptors from ADRIANA.Code program (Molecular Networks GmbH, Erlangen, Germany). The MLR and SVM models give the R^2 of 0.77 and 0.78, respectively on the test set consisting of 106 randomly selected sweeteners. Moreover, Goel et al. harness the genetic function approximation (GFA) and artificial neural network (ANN) coupled with the mixed 2D and 3D molecular descriptors (e.g., LUMO eigenvalue) from Material Studio v6.0 (MS6) (BIOVIA, San Diego, USA) to establish the regression model on the dataset of 455 sweeteners (319 for the training set and 136 for the hold-out test set), which is the largest so far. Both GFA and ANN models offer the impressive performance with the same R^2 of 0.83 on the test set consisting of 136 randomly selected sweeteners (Goel et al., 2018). However, the conformation-dependent 3D descriptors are included in both works from Zhong et al. and Goel et al. and this will hamper the reproducibility of prediction result due to the versatile conformations for the same flexible compound, because most of the sweeteners are quite flexible. Moreover, some other potential issues introduced by the 3D descriptors have been discussed in the work of Rojas et al. (2016a).

Therefore, the conformation-independent 2D descriptors are advocated to be used in the prediction of RS, especially for the rapid and large-scale screening of potent sweeteners. Rojas et al. employ MLR and 2D Dragon descriptors to establish the regression model on the dataset of 233 sweeteners (163 for the training set and 70 for the hold-out test set). This model provides R^2 of 0.70 on the test set including 70 sweeteners, which are selected by the K-mean cluster analysis (Rojas et al., 2016c). Ojha et al. utilize the partial least squares regression analysis (PLSRA) and 2D Dragon/PaDEL descriptors (Wei, 2011) to build the regression model on the dataset of 299 sweeteners (239 for the training set and 60 for the hold-out test set). This model achieves R^2 of 0.75 on the test set composed of 60 randomly selected sweeteners (Ojha and Roy, 2017). Cheron et al. make full use of random forest (RF) and SVM methods combined with the 2D and 3D Dragon descriptors, respectively to construct the regression model on the dataset of 225 sweeteners (134 for the training set and 91 for the hold-out test set). The RF-2D, SVM-2D, RF-3D, and SVM-3D models offer R^2 of 0.74, 0.83, 0.76, and 0.85, respectively on the test set comprising of 91 randomly chosen sweeteners. Nevertheless, their models may be prone to the over-fitting or under-fitting, since the respective model performances on the training set and test set differ significantly, which can be observed from R^2 of 0.96, 0.69, 0.98, and 0.69 for RF-2D, SVM-2D, RF-3D, and SVM-3D models, respectively on the training set (Chéron et al., 2017). Thus, the performance evaluation by only R^2(test set) is probably not enough.

In spite of the individual merits and pitfalls in each work, there are several common concerns in the aforementioned works about the classification and regression models. Firstly, only one data-splitting scheme for the training set and hold-out test set is used in those works, which may lead to the biased performance of the models. Thus, model would be more robust if it can be trained on the multiple data-splitting schemes to alleviate the bias from the single random data-splitting. Secondly, all these works fail to fully comply with the guidelines of Organization for Economic Cooperation and Development (OECD), since most of works are short of either Y-randomization test to evaluate the robustness of their models, or the clear and pragmatic definition for the domain-applicability of their models. Thirdly, all the works do not provide any convenient and practical programs for the users to predict the sweeteners and their RS, which will greatly restrict the application of their models. At last, all these works adopt PLSDA, PLSRA, MLR, KNN, SVM, RF, GFA, or ANN method, while the current state-of-the-art machine-learning methods such as Deep Neuron Network (DNN) and Gradient Boosting Machine (GBM), which often demonstrate the encouraging performance in the Kaggle competitions, were never exploited in the prediction of sweetener or RS before. Therefore, it is highly desirable to overcome these issues and develop a convenient and comprehensive software for the experimental food scientists to predict the sweetener and its corresponding RS.

In order to tentatively address the problems as mentioned above, we plan to build the informative models for the prediction of sweetener and its RS, which will be systematically derived with diverse machine-learning methods (KNN, SVM, GBM, RF, and DNN) and conformation-independent 2D molecular fingerprints based on the multiple data-splitting schemes and will be completely in accordance with the guidelines of OECD. For the convenience of the experimental food scientists, a machine-learning based platform called "e-Sweet" will be developed to automate the prediction of sweetener and its RS via the simple mouse-click on the graphic user interface. The detail of these functions and their implementation will be elaborated below.

MATERIALS AND METHODS

Sweetener Prediction Based on the Multiple Machine Learning Methods

In our previous work about the bitterant prediction (Zheng et al., 2018), we develop a systematic and general protocol to build the classification model, which makes full use of the multiple machine-learning methods (KNN, SVM, GBM, RF, and DNN) by the consensus voting and adopts the Extended-connectivity Fingerprint (ECFP) (Rogers and Hahn, 2010) as the molecular descriptor. In practice, this protocol can be further adapted to generate the regression model. In this work, we will exploit this protocol (**Figure 1**) to derive the machine-learning based models for the prediction of sweeteners and its RS.

In our work, 530 sweeteners are curated from SuperSweet (Ahmed et al., 2011) and SweetenersDB (Chéron et al., 2017) and additionally gathered from the literature (Rojas et al., 2016a; Banerjee and Preissner, 2018), while 850 non-sweeteners consist of 718 bitter compounds downloaded from BitterDB (Wiener et al., 2012) and 132 tasteless compounds retrieved from the literature (Rojas et al., 2016a). Four criteria are defined for the data curation above. (1) Only the larger fragment is kept for the disconnected structures such as salt. (2) Only the compounds containing the elements C, H, O, N, S, P, Si, F, Cl, Br, or I are considered. (3) The same compound with the different taste modalities is excluded. (4) The duplicated compounds from the different sources are eliminated. Based on these standards, all the compounds are finally saved as the Tripos mol2 files, which are integrated into e-Sweet platform for the public access.

In order to train and test the classification model, the whole dataset is randomly divided into two parts: the dataset for the cross-validation (Dataset-CV) and the hold-out test set for the independent validation (Dataset-test). The detailed data-splitting scheme is given as follows: 80% of sweeteners (339 compounds) and 80% of non-sweeteners (544 compounds) randomly selected from the whole dataset are adopted to train the model with the five-fold cross-validation, while the rest of them (221 compounds) are used as the hold-out test set. Finally, this whole data-splitting will be repeated for nineteen or three times to reduce the bias from the random data-splitting. Concretely, nineteen data-splitting schemes are performed for KNN, SVM, GBM, and RF, while three data-splitting schemes are carried out for deep neuron network (DNN) on account of its much higher computational burden.

Besides the indispensable dataset and its partition above, the molecular descriptors are also required for the machine-learning method. Extended-connectivity Fingerprint (ECFP), which is extensively used in the quantitative structure-activity relationship (QSAR) studies (Ekins et al., 2010; Chen et al., 2011; Hu et al., 2012; Koutsoukas et al., 2016; Braga et al., 2017; Rodríguez-Pérez et al., 2017), is adopted as the molecular descriptor in this work. Four ECFPs (1024bit-ECFP4, 2048bit-ECFP4, 1024bit-ECFP6, and 2048bit-ECFP6) are generated for all the curated compounds in the aforementioned dataset with our own implementation of ECFP in e-Bitter program (Zheng et al., 2018), which uniquely offers the intuitive visualization of each "1" bit of ECFP in the context of 3D structure and is also integrated into e-Sweet platform.

Furthermore, feature selection is generally applied in the machine-learning method. In this work, both full-feature without the feature selection and feature-subset with the feature selection are considered. Here the feature selection is performed according to the feature importance (**Figure 1**), which is derived from the model-training with the random forest (RF) method that will be described in the following paragraph about the model-training. In total, 76 runs of model-training with RF are conducted by considering the combination of four ECFP fingerprints and nineteen random data-splitting of the dataset, which will lead to 76 models and the attendant 76 sets of feature importance. Then the feature importance for all the bits in the ECFP fingerprint is sorted in the descending order and plotted in **Figures S1–S4**. Thus the top 512, 256, and 128 important features (**Figures S1–S4**) are selected, respectively as the typical feature subsets for the following model-training, since the exhaustive and systematic scan of feature-number ranging from 1 to fingerprint-length is really time-consuming especially for the training of deep neuron networks (DNN).

Five machine-learning methods (KNN, SVM, GBM, RF, and DNN) are utilized to train the model, which are minutely introduced in our previous work about the bitterant prediction (Zheng et al., 2018) and briefly summarized as follows. K-nearest neighbors (KNN) method conducts the classification and regression based on the closest instances in the training set. Support vector machine (SVM) performs the classification and regression via constructing the hyper-planes in the high-dimensional space. Random forest (RF) and gradient boosting machine (GBM) belong to the decision-tree based ensemble method. RF builds a multitude of decision trees by the bootstrap-sampling of training set and random-selection of feature-subset. GBM generates a series of decision trees in a step-wise manner, rather than in a random way as RF. Deep neuron network (DNN) is a neural network with more than one hidden layer between the input and output layers. Nowadays, thousands of neurons in each layer can be routinely adopted in DNN, which can combine the advanced regularization technique such as the dropout to avoid the overfitting. In this work, the deep neuron networks with two hidden layers (DNN2 in **Figure S5**) and three hidden layers (DNN3 in **Figure S6**) are employed. All the key parameters for each method are listed in **Table S1**, which will be fine-tuned in the five-fold cross-validation (**CV**) to achieve the optimal performance.

The performance of models on the training set and test set are evaluated by the following metrics: the accuracy, precision, specificity, sensitivity, Matthews correlation coefficient (MCC), non-error rate (NER) and F1-score (Equations 1–6). It should be noted that F1-score (Equation 1) is adopted as the criterion to select the best model, albeit F1-score, MCC, and NER are commonly used to measure the quality of the classification.

$$\text{F1-score} = 2\times\text{TP}/(2\times\text{TP} + \text{FP} + \text{FN}) \tag{1}$$

$$\text{Accuracy} = (\text{TP} + \text{TN})/(\text{TP} + \text{TN} + \text{FP} + \text{FN}) \tag{2}$$

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \tag{3}$$

$$\text{Specificity} = \text{TN}/(\text{TN} + \text{FP}) \tag{4}$$

$$\text{Sensitivity} = \text{TP}/(\text{TP} + \text{FN}) \tag{5}$$

$$MCC = \frac{(\text{TP} \times \text{TN} - \text{FP} \times \text{FN})}{\sqrt{(\text{TP} + \text{FP})\,(\text{TP} + \text{FN})\,(\text{TN} + \text{FP})\,(\text{TN} + \text{FN})}} \tag{6}$$

$$\begin{aligned}\Delta\text{F1-score} = {} & |\text{F1-score(cross-validation)} \\ & - \text{F1-score(test set)}|\end{aligned} \tag{7}$$

$$\text{NER} = (\text{Sensitivity} + \text{Specificity})/2 \tag{8}$$

TP, TN, FP, and FN are the numbers of true sweeteners, true non-sweeteners, false sweeteners, and false non-sweeteners, respectively. NER is short for non-error rate and is the arithmetic

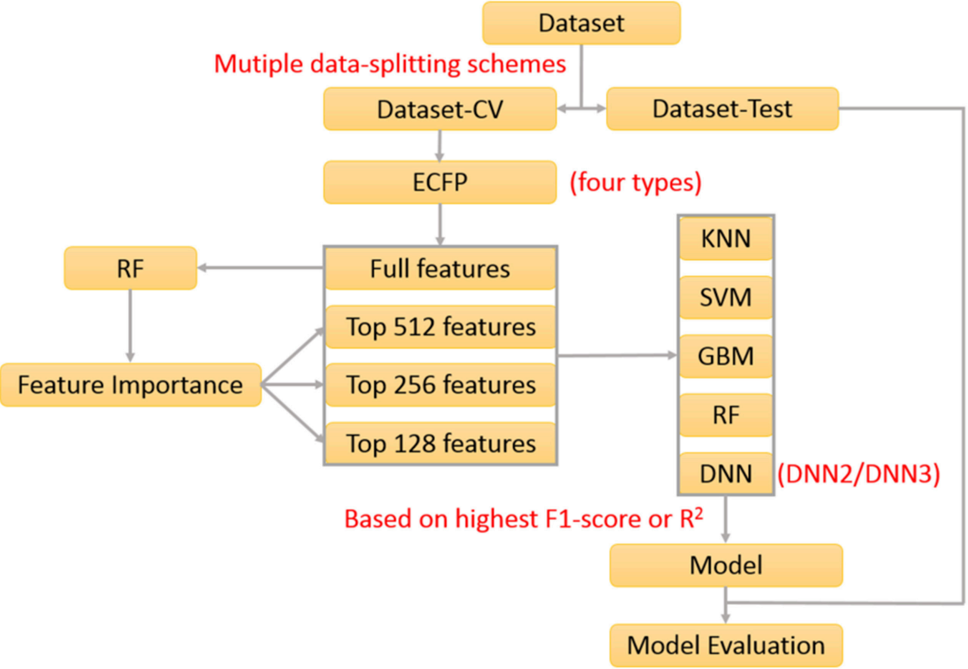


FIGURE 1 | The protocol to derive the classification and regression model used in this work.

mean of sensitivity and specificity in the binary classification. ΔF1-score is calculated to monitor the potential over-fitting or under-fitting.

Upon completion of the model-training with the five-fold cross-validation, totally 1312 models including 328 models without feature selection and 984 models with feature selection are harvested according to the highest F1-score, and are further gauged on the respective hold-out test sets with the evaluation metrics: accuracy, precision, specificity, sensitivity, F1-score, MCC, and NER, which are listed in **Table S2**. To reduce the bias from the random splitting of the whole dataset, 96 average models (AM) are derived from 1,312 individual models by averaging over the different data-splitting schemes and are tabulated in **Table S3**.

Following the guidelines of OECD, Y-randomization test for our models should be performed and the applicability-domain of our models should be also defined practically. To inspect the robustness of all the models, Y-randomization test is done with the following procedure. Firstly, the experimentally observed labels for Dataset-CV are randomly shuffled (**Table S4**). Subsequently, the five-fold cross-validation on this noisy dataset is performed with exactly the same molecular descriptors and the same protocol mentioned in the previous section about the model-training. The best models are also determined based on the highest F1-score assessed on the internal validation dataset during the cross-validation, and further evaluated on the hold-out test set (**Dataset-Test**) without any random shuffling. All the results are collected in **Tables S5–S6**. Meanwhile, with regard to the definition of the applicability domain, it is generally hypothesized that the compound, which is highly dissimilar to all the compounds used in the model-training, may not be predicted confidently (Tropsha, 2010). With this assumption in our mind, the applicability domain of our models is defined on the basis of the ECFP based Tanimoto-similarity between the compound of interest and its five closest neighboring compounds in our training set (Dataset-CV).

Finally, 1,312 individual models (M0001–M1312 in **Table S2**) and 96 average models (AM01–AM96 in **Table S3**) are obtained after the model training and validation. Based on these models, four consensus models are proposed according to the criteria such as the performance, speed and diversity of machine-learning based models, and are integrated into our e-Sweet platform. All the constitute models for each consensus model are provided in **Tables S7–S10** and the performances of these four consensus models are given in **Table 1**. More specifically, Consensus model 1 (CM01) selects 19 best individual models (**Table S7**) with all the methods except DNN purely based on the highest F1-scores in each data-splitting scheme from the perspective of performance and speed. Consensus model 2 (CM02) selects 19 best individual models (**Table S8**) with all the methods including DNN solely based on the highest F1-scores in each data-splitting scheme according to the model performance. Consensus model 3 (CM03) considers the top five average models (**Table S9**) with the highest F1-scores. Consensus model 4 (CM04) chooses the five average models (**Table S10**) considering each machine-learning method with the highest F1-score to balance the performance and diversity of machine-learning based models. All the evaluation metrics for each consensus model (**Table 1**) are obtained by averaging over all the constituent models.

TABLE 1 | The performance of four consensus models (CM01–CM04) for the sweetener/non-sweetener classification.

Model	Accuracy (test set)	Precision (test set)	Specificity (test set)	Sensitivity (test set)	F1-score (test set)	MCC (test set)	NER (test set)	F1-score (CV)	NER (CV)	ΔF1-score	ΔNER
MEAN(STANDARD DEVIATION)											
CM01	0.91 (0.01)	0.90 (0.03)	0.94 (0.02)	0.85 (0.02)	0.88 (0.02)	0.80 (0.02)	0.90 (0.01)	0.85 (0.01)	0.87 (0.01)	0.03 (0.02)	0.03 (0.01)
CM02	0.91 (0.01)	0.90 (0.03)	0.94 (0.02)	0.86 (0.03)	0.88 (0.02)	0.81 (0.03)	0.90 (0.01)	0.85 (0.01)	0.87 (0.01)	0.04 (0.02)	0.03 (0.02)
CM03	0.89 (0.00)	0.89 (0.01)	0.93 (0.01)	0.83 (0.00)	0.86 (0.00)	0.77 (0.01)	0.88 (0.00)	0.85 (0.01)	0.87 (0.00)	0.02 (0.01)	0.02 (0.01)
CM04	0.89 (0.01)	0.89 (0.02)	0.94 (0.01)	0.82 (0.01)	0.85 (0.01)	0.76 (0.01)	0.88 (0.01)	0.84 (0.00)	0.87 (0.00)	0.02 (0.00)	0.02 (0.00)
95% CONFIDENCE INTERVAL: MEAN ± MARGIN OF ERROR											
CM01	0.91 ± 0.01	0.90 ± 0.01	0.94 ± 0.01	0.85 ± 0.01	0.88 ± 0.01	0.80 ± 0.01	0.90 ± 0.01	0.85 ± 0.01	0.87 ± 0.01	0.03 ± 0.01	0.03 ± 0.01
CM02	0.91 ± 0.01	0.90 ± 0.01	0.94 ± 0.01	0.86 ± 0.01	0.88 ± 0.01	0.81 ± 0.01	0.90 ± 0.01	0.85 ± 0.01	0.87 ± 0.01	0.04 ± 0.01	0.03 ± 0.01
CM03	0.89 ± 0.00	0.89 ± 0.01	0.93 ± 0.01	0.83 ± 0.00	0.86 ± 0.00	0.77 ± 0.00	0.88 ± 0.00	0.85 ± 0.00	0.87 ± 0.00	0.02 ± 0.00	0.02 ± 0.00
CM04	0.89 ± 0.01	0.89 ± 0.02	0.94 ± 0.01	0.82 ± 0.01	0.85 ± 0.01	0.77 ± 0.01	0.88 ± 0.01	0.84 ± 0.00	0.87 ± 0.00	0.02 ± 0.00	0.02 ± 0.00

(1) The number in each parenthesis is the standard deviation, which is derived based on the multiple random data-splitting schemes; (2) ΔF1-score and ΔNER refer to | F1-score (test set)–F1-score (cross-validation) | and | NER (test set)–NER (cross-validation) | respectively; (3) "MCC," "CV," and "NER" are short for "Matthews correlation coefficient," "cross-validation," and "Non-error Rate," respectively.

Sweetness Prediction Based on Multiple Machine Learning Methods

In our work, all the sweeteners with RS are gathered from the literature (Iwamura, 1981; Drew et al., 1998; Kinghorn and Soejarto, 2002; Vepuri et al., 2007; Yang et al., 2011), and subjected to the filtering with the following criteria. (1) Only the larger fragment is saved for the disconnected structures such as salts. (2) Only the compounds containing the elements C, H, O, N, S, P, Si, F, Cl, Br, or I are considered. (3) Only one compound is kept for the duplicated compounds from the different sources. (4) Only the compound with the experimental RS, which is only measured relative to the 5% (w/v) sucrose, is taken account. After the filtering with these conditions, 352 sweeteners are curated for our subsequent training with the machine learning methods. All the structures with the Tripos mol2 files, and their corresponding $\log_{10}$RS (common logarithm of the relative sweetness) used as the dependent variable (Y) are publicly available in our e-Sweet platform. To train and validate the model, the whole dataset is sorted ascendingly according to $\log_{10}$RS. Twenty percent of the whole dataset (71 compounds) is randomly selected from every five compounds in the ascending order to form the hold-out test set (Dataset-Test) with the even distribution of $\log_{10}$RS. The rest of them (281 compounds) are adopted to train the model with the five-fold cross-validation (Dataset-CV). Similarly, the whole data-splitting is repeated for the multiple times as well.

To derive the regression model for RS, nearly the same protocol (**Figure 1**) as the sweetener/non-sweetener classification is adopted. According to this protocol, all the combination of the molecular fingerprints, feature selection, feature number, data-splitting schemes, and machine-learning methods is taken into account in the model-training, and thereby 1,312 best individual models are also achieved based on the highest R^2 (square of the coefficient of determination) after the five-fold cross-validation, and are further assessed on the respective hold-out test sets with the evaluation metrics: R^2, mean absolute error (MAE), mean squared error (MSE), and ΔR^2 (referring to $|R^2(\text{test set})–R^2(\text{cross-validation})|$), which are summarized in **Table S11**. Subsequently, 96 average models are also obtained based on 1,312 individual models by averaging over the different data-splitting schemes, whose performances are given in **Table S12**. Furthermore, Y-randomization test (**Tables S13–S15**) and defining the applicability-domain for our models are also carried out with the similar protocol in the previous section about the classification model. Finally, three consensus models (CM01–CM03 in **Tables S16–S18**) are suggested on the basis of 1,312 individual models and 96 averages models and are embedded into our e-Sweet platform.

RESULTS AND DISCUSSION

Overview of e-Sweet Platform

e-Sweet is a machine-learning based platform for the automatic prediction of the sweetener and its RS, which is developed based on our previous e-Bitter program (Zheng et al., 2018). This e-Sweet platform can be easily installed via the simple click of mouse and can smoothly run both in the modes of graphic user-interface and command-line, which are well tested on the Win7, Win8, and Win10. The whole program including the manual and tutorials can be freely from the link (https://www.dropbox.com/sh/1fmlv7nf6wofgcp/AADBJzFbbbiNRJUP0806wSyna?dl=0).

In the current version of e-Sweet, there are several major helpful functions for the food scientists. (1) Visualize and inquiry our curated datasets for the classification of sweetener/non-sweetener or the regression of RS. (2) Predict the sweetener and its RS with the multiple machine-learning methods by evoking the external scikit-learn (v0.19.1), Keras (v1.1.0), and Theano (v1.0.1) python libraries fully integrated in the free Anaconda (v2-5.2.0) that can also be handily installed on the windows in the simple way. (3) Virtual screening of database to enrich the possible sweetener candidates. (4) Generate and visualize the ECFP fingerprint, which is adopted as the molecular descriptor and is also natively implemented in this platform. (5) View the fingerprint bit in the context of 3D structure, and synchronously display the feature importance of fingerprint bit contributing to the classification of sweetener/non-sweetener or regression of RS. The detailed usage of all those functions is articulated in the manual and tutorials, while only the key functions (**Figure 2**) will be detailed as follows.

In a nutshell, e-Sweet is the first, free, and convenient standalone software for the experimental food scientists to automate the prediction of the sweetener and its corresponding RS with the machine-learning based classification and regression models, and also offers several key auxiliary functions relevant to the prediction.

The Chemical Space of Our Curated Datasets Embedded in e-Sweet

Our curated datasets for the classification of sweetener/non-sweetener and the regression of RS are publicly available and fully integrated into our e-Sweet platform, with which users can simultaneously visualize the chemical structures and the corresponding labels (or $\log_{10}$RS) and can conveniently enquiry our datasets with the compounds of users' interests by Tanimoto-similarity based structure search (**Figure S7**).

Our dataset for the sweetener/non-sweetener classification consists of 530 sweeteners and 850 non-sweeteners, which is the largest dataset so far. In the latest sweetener/non-sweetener classification model from Rojas et al. 435 sweeteners and 214 non-sweeteners are utilized, which is much less than ours. To examine the difference of chemical spaces between the sweeteners and non-sweeteners in our dataset, the molecular weight (MW), logP, and the numbers of hydrogen-bond donor (N_{HBD}) and hydrogen-bond acceptor (N_{HBA}) are calculated by OpenBabel v2.4 (Oboyle et al., 2011). The scatter plots of logP vs. MW (**Figure S8**) and N_{HBD} vs. N_{HBA} (**Figure S9**) showcase that the distributions for the sweeteners are very similar to the counterpart for the non-sweeteners. Hence the intuitive discrimination between the sweeteners and non-sweeteners by the simple descriptors such as logP, MW, N_{HBD}, and N_{HBA} is not effective. Furthermore, ECFP based similarity-matrix (**Figure S10**) illustrates that the overall pairwise Tanimoto-similarities between the sweeteners and non-sweeteners are quite low with the average value of 0.08 over the entire matrix, indicating that ECFP fingerprint may be a promising molecular descriptor for the classification of sweeteners and non-sweeteners.

In addition, our dataset for the regression of RS is composed of 352 sweeteners, and is larger than the datasets utilized in most of relevant works (Zhong et al., 2013; Rojas et al., 2016c; Chéron et al., 2017; Ojha and Roy, 2017), but is smaller than the dataset used in the work of Goel et al. which is made up of 455 sweeteners that is not directly accessible to the other researchers (Goel et al., 2018). It is worth mentioning that both works glean the sweeteners with RS from the same source (Iwamura, 1981; Drew et al., 1998; Kinghorn and Soejarto, 2002; Vepuri et al., 2007; Yang et al., 2011), thus the different number of sweeteners used in both works is presumably resulted from the distinct curation criteria. To check the conformational flexibility of the sweeteners in this dataset, the numbers of the freely rotatable bonds (N_{FRB}) for all the sweeteners are computed by OpenBabel v2.4 and the histogram of N_{FRB} (**Figure S11**) demonstrates that most of the sweeteners are quite flexible and have many conformers, which may bring about the irreproducible result for the model prediction if the conformation-dependent 3D molecular descriptors are used to establish the model. Therefore, ECFP based 2D molecular descriptors are used in this work.

In a word, our dataset for the sweeteners/non-sweeteners classification is the largest and the dataset for the sweeteners with RS is the second largest, and both datasets are publicly available to other researchers. ECFP based similarity-matrix indicates that ECFP based 2D descriptor could be beneficial to the classification of sweeteners and non-sweeteners, and the analysis of conformational flexibility of the sweeteners in this dataset casts light on the potential weakness of the conformation-dependent 3D molecular descriptors.

Prediction of Sweetener by the Classification Model in e-Sweet

For the sweetener/non-sweetener classification, 1,312 individual classification models (M0001–M1312 in **Table S2**) and 96 average classification models (AM01–AM96 in **Table S3**) are harvested. The scatter-plot of ΔF1-score vs. F1-score for all the models is plotted in **Figure 3A**, since F1-score is the performance indicator of the classification model and ΔF1-score is used to examine the possible over-fitting or under-fitting of the classification model. **Figure 3A** demonstrates that ΔF1-score for most of individual and average classification models is lower than 0.04, suggesting that the model performance on the test set and in the cross-validation is quite similar. Thus, most of our models do not suffer from the obvious over-fitting or under-fitting from this perspective. Moreover, the orange dots (**Figure 3A**) standing for 96 average classification models based on the multiple data-splitting schemes have much narrower distribution than the blue dots (**Figure 3A**) denoting 1,312 individual classification models on the basis of the single data-splitting scheme, which provides an important clue that the different random data-splitting schemes

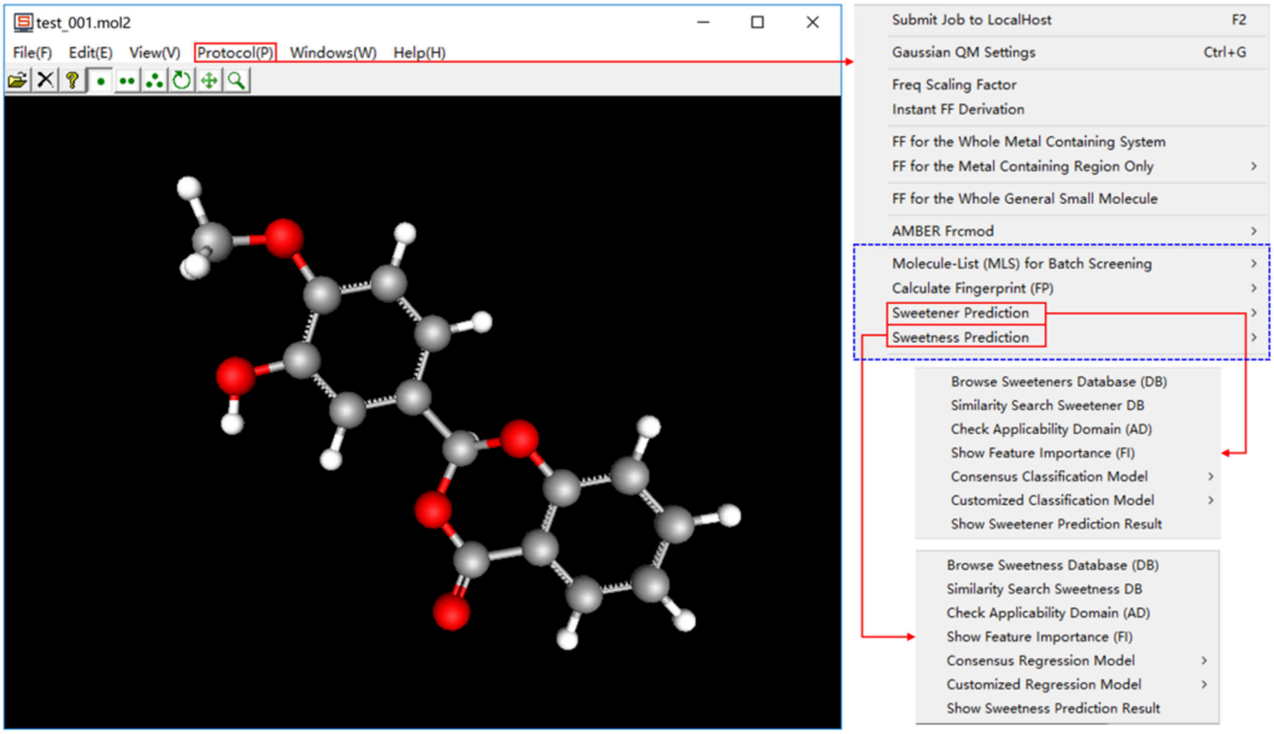

test_001.mol2
File(F)
Edit(E)
View(V)
Protocol(P)
Windows(W)
Help(H)
Submit Job to LocalHost F2
Gaussian QM Settings Ctrl+G
Freq Scaling Factor
Instant FF Derivation
FF for the Whole Metal Containing System
FF for the Metal Containing Region Only
FF for the Whole General Small Molecule
AMBER Frcmod
Molecule-List (MLS) for Batch Screening
Calculate Fingerprint (FP)
Sweetener Prediction
Sweetness Prediction
Browse Sweeteners Database (DB)
Similarity Search Sweetener DB
Check Applicability Domain (AD)
Show Feature Importance (FI)
Consensus Classification Model
Customized Classification Model
Show Sweetener Prediction Result
Browse Sweetness Database (DB)
Similarity Search Sweetness DB
Check Applicability Domain (AD)
Show Feature Importance (FI)
Consensus Regression Model
Customized Regression Model
Show Sweetness Prediction Result

FIGURE 2 | The main features of e-Sweet platform for the sweetener and sweetness prediction.

have dramatic effects on the model performance. Therefore, it is a good practice for the machine-learning practitioners to repeat the data-splitting for the multiple times to gain more objective models.

To further inspect the robustness of all 1,312 individual and 96 average classification models, Y-randomization test is performed for all the classification models by the random shuffling of experimental labels in Dataset-CV (**Table S4**), and all the results are tabulated in **Tables S5–S6**. For the better illustration, the scatter plot of F1-score(test set) vs. MCC(test set) for all the models is plotted in **Figure S12**, which clearly demonstrates that the model performances after Y-randomization is drastically decreased relative to the models without Y-randomization. Accordingly, all our previous models without Y-randomization are quite robust and not obtained by chance.

However, it is not very efficient to harness all 1,312 individual and 96 average classification models simultaneously for the pragmatic prediction of sweeteners, consequently four typical consensus models (CM01–CM04 in **Tables S7–S10**) are suggested based on the performance, speed, and diversity of the models, and are incorporated into our e-Sweet platform. Observed from **Table 1**, the overall performances of all these consensus models on the test set (**Table 1**) are very promising, while the best model CM02 with the highest F1-score can achieve the 95% confidence intervals for the accuracy (0.91 ± 0.01), precision (0.90 ± 0.01), specificity (0.94 ± 0.01), sensitivity (0.86 ± 0.01), F1-score (0.88 ± 0.01), MCC (0.81 ± 0.01), and NER (0.90 ± 0.01) on the test set by averaging over the 19 data-splitting schemes.

To demonstrate the advantage of our models, CM01–CM04 in **Tables S7–S10** are compared with the model in the work of Rojas et al. which is the only published work about the sweetener/non-sweeteners classification based on the relatively large dataset and affords the NER values of 0.85 and 0.83 on the test set and in the cross-validation, respectively, whereas the evaluation metrics such as F1-score and MCC are not reported in their work. The procedure for the statistical comparison is given as follows. (1) Bland-Altman analysis (Martin Bland and Altman, 1986) is firstly conducted to examine whether NER(test set) and NER (cross-validation) of the models from Rojas et al. match well within the limits of agreement (LoA) in the Bland-Altman plots based on our consensus classification models. (2) If NER(test set) and NER (cross-validation) of the models from Rojas et al. agree well, it indicates that their model probably does not suffer from the evident over-fitting or under-fitting. Subsequently, further comparison will be performed to check whether their model is within the 95% confidence intervals of ΔNER (referring to |NER(test set)–NER(cross-validation)|) and NER(test set), respectively.

According to this comparison protocol, **Figure S14** clearly illustrates that NER(test set) and NER(cross-validation) of the model from Rojas et al. agree very well in all the Bland-Altman plots (**Figure S14**) based on CM01, CM02, CM03, and CM04. Subsequently, NER(test set) and ΔNER will be used as

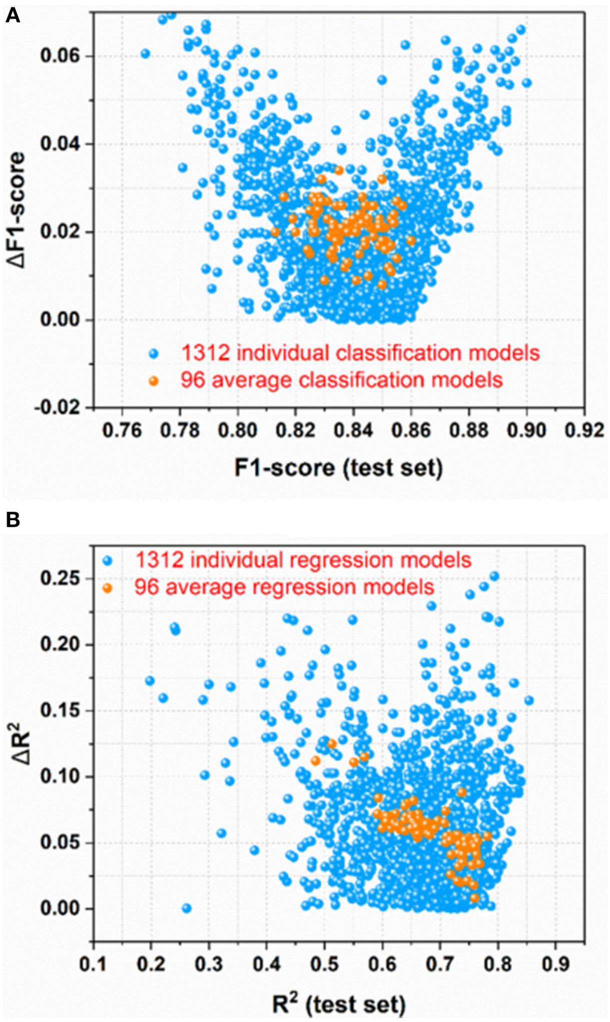


FIGURE 3 | (A) the scatter-plot of ΔF1-score vs. F1-score for all the classification models; **(B)** The scatter plot of ΔR^2 vs. R^2(test set) for all the regression models. ΔF1-score [referring to |F1-score (test set)–F1-score(cross-validation)|] and ΔR^2 [referring to $|R^2$(test set)–R^2(cross-validation)|] are used to monitor the potential overfitting or underfitting.

the performance indicators for the further comparisons. More specifically, ΔNER of the model from Rojas et al. is 0.02, and is within the 95% confidence intervals of ΔNER from our CM01, CM02, CM03, and CM04, which are 0.03 ± 0.01, 0.03 ± 0.01, 0.02 ± 0.00, and 0.02 ± 0.00, respectively. Meanwhile, NER(test set) of the model from Rojas et al. is 0.85, and is consistently lower than the 95% confidence intervals of NER (test set) from our CM01, CM02, CM03, and CM04, which are 0.90 ± 0.01, 0.90 ± 0.01, 0.88 ± 0.00, and 0.88 ± 0.01, respectively. Therefore, all four consensus sweetener/non-sweeteners classification models are better than the model from Rojas et al.

In short, the robust sweetener/non-sweetener classification models based on the largest dataset, multiple data-splitting schemes and manifold machine-learning methods are derived, and our proposed four consensus models are demonstrated to outperform the model from Rojas et al. that is based on the single data-splitting scheme.

Prediction of Relative Sweetness by the Regression Model in e-Sweet

For the prediction of RS, 1,312 individual regression models (M0001–M1312 in **Table S11**) and 96 average regression models (AM01–AM96 in **Table S12**) are achieved. The scatter plot of ΔR^2 [referring to $|R^2$(test set)–R^2(cross-validation)|] vs. R^2(test set) for all the models is made for the assessment of overall performance, because R^2(test set) is the performance indicator of regression model and ΔR^2 is used to monitor the potential over-fitting or under-fitting of regression model. From **Figure 3B**, it illustrates that ΔR^2 for most of the individual and average regression models is <0.10, implying that the models achieve the consistently similar performance on the hold-out test set and in the cross-validation, respectively. Thus, most of the models do not exhibit the noticeable over-fitting or under-fitting from this point of view. In addition, observed from **Figure 3B**, the more compact distribution of the average models relative to the individual models also emphasizes that the average models based on the multiple data-splitting schemes are more convergent than the individual models based on the single data-splitting scheme.

To further ensure the robustness of all the individual and average regression models, Y-randomization test is also conducted for all the regression models by the random shuffling of experimental $\log_{10}^{RS}$ in Dataset-CV (**Table S13**), and all the outcomes are given in **Tables S14**, **S15**. For the sake of intuitive description, the scatter plot of R^2(test set) vs. MAE(test set) for all the models before and after Y-randomization in **Figure S13** unambiguously illustrates that our regression models without Y-randomization are reliable.

Nevertheless, it is not realistic to utilize all the 1,312 individual and 96 average regression models at the same time for the practical prediction of RS, hence three representative consensus models (CM01-CM03 in **Tables S16–S18**) are proposed and integrated into our e-Sweet platform. **Table 2** illustrates that our consensus models (CM01–CM03) on the basis of the individual and average models afford R^2(test set) ranging from 0.77 to 0.78. CM02 has the highest R^2(test set) with the 95% confidence interval of 0.78 ± 0.02, while CM03 provides the lowest ΔR^2 with the 95% confidence interval of 0.03 ± 0.01.

For the sake of the easier comparison with the other works about the prediction of RS, R^2(test set) and R^2(cross-validation) are generally reported in the respective works and compiled in **Table S19**, which are all based on only one data-splitting scheme to prepare the hold-out test set and training set in the other works. The statistical comparison between ours and other models is very similar to the aforementioned comparisons between the classification models and will be carried out as follows: (1) Bland-Altman method is firstly adopted to check whether R^2(test set) and R^2(cross-validation) of the models from other works agree well within the limits of agreement in the Bland-Altman plots based on our consensus regression models. (2) If R^2(test set) and R^2(cross-validation) of the models from other works agree well, the 95% confidence intervals of $|R^2$(test set)-R^2(cross-validation)| and R^2(test set) for our models are used for the further comparison with the models from other works. Otherwise, the model may suffer from the over-fitting or under-fitting due to the distinct difference between R^2(test set) and R^2(cross-validation) and thereby will be excluded in the subsequent comparison.

From **Figure S15**, all the models from other works exceed the upper or lower limits of agreement (LoA) and their 95% confident intervals, which reveals that the model performances of other models on the test set and in the cross-validation do not agree well compared to the counterpart of our consensus model CM03. Thus, CM03 is the best model in term of the agreement between R^2(test set) and R^2(cross-validation). However, all the constituent models in CM03 are derived from DNN method, which are much slower relative to the models derived from the other machine-learning methods such as KNN, SVM, GBM, and RF. Therefore, we proposed two other consensus models (CM01 and CM02). CM02 is constructed on 19 best constituent models in 19 data-splitting schemes. However, in CM02 there is still one constituent model that comes from the time-consuming DNN method. Thus, CM01 is suggested also based on 19 best constituent models by excluding the model from DNN method. As a result, CM01 has very similar constituent models relative to CM02, but is much faster than CM02 and thereby is suitable for the database screening. Thus, it is understandable that Bland-Altman plots (**Figure S15**) of CM01 and CM02 are very similar.

Therefore, the Bland-Altman plot (**Figure S15A**) based on CM01 is taken as an instance. Five 3D descriptors based models are very close to the limits of agreement (LoA), however, those models can be still assumed that R^2(test set) and R^2(cross-validation) of these five models are agreeable according to the Bland-Altman plot based on CM01 (**Figure S15A**), while only one 3D descriptors based model completely locates outside the upper and lower LoA and their 95% confident intervals. These five acceptable 3D descriptors based models in Bland-Altman plot (**Figure S15A**) are MLR model from Zhong et al. SVM model from Zhong et al. GFA model from Goel et al. ANN model from Goel. et al. and SVM model from Cheron et al. which afford ΔR^2 with the values of 0.04, 0.05, 0.03, 0.06, and 0.16, respectively (**Table S19**). According to **Table 2**, the 95% confidence interval of ΔR^2 for our CM01 is 0.07 ± 0.02. Consequently, the ΔR^2 of SVM model from Cheron et al. is much larger than the 95% confidence interval (0.07 ± 0.02) from CM01. Finally, four remaining models will be further compared with our model CM01 based on R^2(test set). MLR model with R^2(test set) value of 0.77 and SVM model with R^2(test set) value of 0.78 from Zhong et al. (**Table S19**) are still within the 95% confidence interval (0.77 ± 0.02) of R^2(test set) for our CM01, while GFA model with R^2(test set) value of 0.83 and ANN model with R^2(test set) value of 0.83 from Goel et al. (**Table S19**) is higher than the 95% confidence interval (0.77 ± 0.02) of R^2(test set) for our CM01 (**Table 2**). Therefore, our CM01 has a similar performance with the MLR and SVM models from Zhong et al. and shows the lower performance than the GFA and ANN models from Goel et al. It is worth mentioning that this conclusion also holds for CM02. Nevertheless, Goel et al. employed the conformation-dependent 3D molecular descriptors such as the LUMO eigenvalue, which requires the time-consuming quantum mechanical (QM) calculation particularly for the large molecules. Moreover, the flexible sweeteners usually possess very diverse conformations due to a number of freely rotatable bonds, which may provide the totally different molecular descriptors for the same compound and thereby may lead to the irreproducible result in the practical prediction. Actually the work of Rojas et al. also well addresses this issue and suggests to adopt the 2D molecular descriptors for the simplicity and the fast speed. Thus, ECFP based 2D molecular descriptors are adopted in our work.

As such, 2D descriptors based models including ours will be the main focus for the comparison of model performance. Two 2D descriptors based models from other works are very close to the limits of agreement (LoA), albeit they are still in the acceptable region. One model from Rojas et al. is trained with MLR and 2D Dragon descriptors, and gives R^2(test set), R^2(cross-validation), and ΔR^2 of 0.70, 0.78, and 0.08, respectively, while the other from Cheron et al. is built with SVM and 2D Dragon descriptors, and offers R^2(test set), R^2(cross-validation), and ΔR^2 of 0.83, 0.69, and 0.14, respectively. However, the 95% confidence interval of ΔR^2 for our CM01 model is 0.07 ± 0.02. Hence only the model from Rojas et al. is within the 95% confidence interval (0.07 ± 0.02) of ΔR^2. Finally, the model comparison based on R^2(test set) illustrates that CM01 is better than the model from Rojas et al. since R^2(test set) with the value of 0.70 from Rojas et al. is much lower than the 95% confidence interval (0.77 ± 0.02) of R^2(test set) from CM01. It is noteworthy that this conclusion can also apply to CM02.

In sum, our consensus regression model CM03 is prominently promising than all the models from other works in term of agreement between R^2(test set) and R^2(cross-validation) based on the Bland-Altman plot of CM03, while CM01/CM02 remarkably outperforms the 2D descriptors based models from other works according to the full analysis of Bland-Altman plot and the 95% confident intervals of ΔR^2 and R^2(test set), but is inferior to the 3D descriptors based models from Goel et al. that are derived from the single data-splitting scheme. However, the 3D descriptors based models are not pragmatic for the prediction by other users. Furthermore, it still should be taken with caution that R^2(test set) from the single data-splitting scheme is adopted to compare the model performance, since different data-splitting schemes have apparent effects on the model performance.

Automatic Inspection of Applicability Domain in e-Sweet

To comply the guideline of OECD, the applicability domain of the models should be defined appropriately. In this work, the applicability domain of our models is defined on the basis of the concept "average-similarity." More Concretely , the automatic procedure implemented in our e-Sweet is given as follows: (1) each compound in the test set (Dataset-Test) is compared with all the compounds in the cross-validation dataset (Dataset-CV) according to the Tanimoto-similarity based on 2048bit-ECFP6; (2) five most similar compounds from Dataset-CV are retrieved and treated as five nearest neighbors for the given compound in Dataset-Test, and the average of five similarities is defined as the "average-similarity" between this given compound and these five nearest neighbors; (3) each compound in Dataset-Test retrieves five nearest neighbors in Dataset-CV to calculate

TABLE 2 | The performance of three consensus models (CM01–CM03) for the regression of relative sweetness (RS).

Model	R^2 (test set)	MSE (test set)	MAE (test set)	R^2 (CV)	$\triangle R^2$
MEAN(STANDARD DEVIATION)					
CM01	0.77 (0.05)	0.27 (0.06)	0.39 (0.03)	0.72 (0.05)	0.07 (0.05)
CM02	0.78 (0.05)	0.28 (0.06)	0.40 (0.03)	0.71 (0.05)	0.07 (0.05)
CM03	0.77 (0.01)	0.58 (0.31)	0.58 (0.17)	0.74 (0.01)	0.03 (0.01)
95% CONFIDENCE INTERVAL: MEAN ± MARGIN OF ERROR					
CM01	0.77 ± 0.02	0.27 ± 0.03	0.39 ± 0.01	0.72 ± 0.02	0.07 ± 0.02
CM02	0.78 ± 0.02	0.28 ± 0.03	0.40 ± 0.01	0.71 ± 0.02	0.07 ± 0.02
CM03	0.77 ± 0.01	0.58 ± 0.27	0.58 ± 0.15	0.74 ± 0.01	0.03 ± 0.01

(1) The number in each parenthesis is the standard deviation, which is obtained on the basis of the multiple random data-splitting schemes; (2) ΔR^2 referring to | R^2(test set)–R^2(cross-validation) | is employed to monitor the potential over-fitting/under-fitting; (3) "CV" is short for the cross-validation.

the average-similarity. Similarly, each compound in Dataset-CV also finds five nearest neighbors in Dataset-CV to compute its corresponding average-similarity; (4) the histograms of the average-similarity for Dataset-Test and Dataset-CV are given in **Figure 4** to address the applicability domain of our models.

For the classification model, **Figure 4A** shows that the average-similarity of 0.1 could be used as the threshold for the definition of the applicability domain of our classification models. If the average-similarity of the compound of interest is larger than this threshold (0.1), it means that this compound is located inside the applicability domain of our models, the prediction for this compound is a confident inference. Otherwise, this prediction may be a bold extrapolation. Similarly, for the regression model, **Figure 4B** reveals that the average-similarity of 0.1 can also serve as the threshold to define the applicability domain of our regression models. In order to automatically check whether the compound to be predicted is within the applicability domain of our classification and regression models, we have implemented a convenient function via simple clicking of the menu in our e-Sweet platform.

In brief, our classification and regression models for the prediction of sweetener and its RS have the pragmatically defined applicability domain, which is not commonly or explicitly mentioned in other relevant works and can be automatically inspected by our e-Sweet.

Model Interpretation for our Classification and Regression Models in e-Sweet

Model interpretation suggested by OECD, will be considered based on the feature importance, which underscores the importance of each ECFP fingerprint bit contributing to the sweeteners/non-sweeteners classification or the regression of RS. Our e-Sweet platform can advantageously offer the appealing function to synchronously visualize the structural feature in the context of 3D structure and the associated feature importance for the ECFP fingerprint bit "1."

In order to visualize the structural features and the corresponding feature importance for all the bits in ECFP, it would be better to adopt the model trained with the full features, since the feature selection will obviously lose some ECFP bits and hamper us to view the complete bits. Hence the average feature importance, which is from the average classification (AM22 in **Table S3**) and regression model (AM22 in **Table S12**) trained with RF and full features (2048bit-ECFP6), is embedded in our e-Sweet for the fully interactive visualization of ECFP fingerprint-bit, structural feature, and feature importance of ECFP bit.

For the purpose of concise demonstration, visualization of the feature importance (FI) contributing to the sweeteners/non-sweeteners classification is taken as an example and only the structural feature with the largest feature importance is considered here. In this case, the bit with the largest feature importance (FI = 0.019821) is 1138-bit (**Figure S16**). In our sweeteners/non-sweeteners dataset, the ECFPs of 228 sweeteners and 20 non-sweeteners contain the "1" in the 1138-bit. Here only one sweet molecule containing "1" in 1138-bit is taken as an instance for the better illustration (**Figure S16**). The structure feature for 1138-bit is highlighted with the yellow color in the 3D viewer window, the corresponding feature importance for 1138-bit is shown in the window titled "FI." Based on the feature importance, it means that 1138-bit is very important for the sweeteners/non-sweeteners classification.

Concisely, our e-Sweet platform provide a convenient and intuitive visualization function for the model interpretation, which makes our classification and regression models fully conform to the OECD guidelines.

The Limitation and Prospect of This Work

Admittedly, our work has some shortcomings. (1) Our curated dataset only considers the organic compounds, ignores the inorganic compounds and mixtures, and also neglects the effects of purity, moisture content, and temperature. In addition, the sweet taste assessment results given by the trained taste panelists have some inevitable noise, because the taste panelists possess some subjective factors (e.g., some mixed tastes that are very difficult to be clearly discriminated in qualitative or quantitative manner) and objective reasons (e.g., the individual gene-polymorphism of sweet taste receptor). (2) The consensus strategy is used to balance the pros and cons of each machine-learning method. However, it will bring some extra computational burden, because the final prediction is obtained

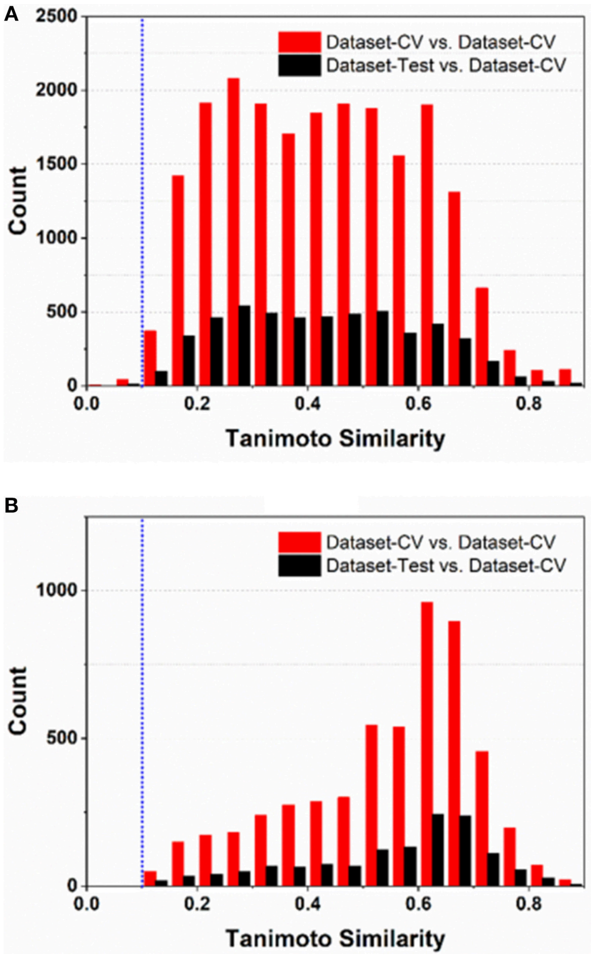


FIGURE 4 | The histograms of average-similarity are utilized to define the applicability-domain of our classification **(A)** and regression models **(B)**. Both average-similarity thresholds of 0.1 are defined and implemented in our e-Sweet platform to automatically check whether the compound to be predicted is within the applicability domain of our models.

by averaging over all the prediction results from each constituent model. (3) Applicability domain of the regression model for the relative sweetness is still limited, because the size of dataset for the regression model is relatively small compared to the size of sweetener/non-sweetener dataset for the classification and thereby needs further expansion.

In spite of these limitations, our work also possesses several advantages, which may provide some beneficial advice for the other researchers to develop more informative sweetener/sweetness prediction model. (1) Different data-splitting schemes have dramatic effects on the model training and model performance, which will be more obvious for the dataset with the limited size. Hence the multiple data-splitting schemes are highly recommended. (2) 2D descriptors based models are preferred over 3D descriptors based models for the practical prediction, because the sweeteners are usually very flexible molecules with diverse conformations that will cause the irreproducible outcome for the 3D descriptors based models. (3) The model evaluation solely based on R^2(test set) or F1-score(test set) may be not convincing enough. Thus, it is suggested to consider both R^2(test set) and $|R^2$(test set)-R^2(CV)$|$ for the regression models and both F1-score(test set) and $|$F1-score(test set)-F1-score(CV)$|$ for the classification model, since the model probably suffers from the over-fitting or under-fitting if $|R^2$(test set)-R^2(CV)$|$ or $|$F1-score(test set)-F1-score(CV)$|$ is large. (4) Deep neural network (DNN) method affords the consensus regression model **CM03** with the best agreement between R^2(test set) and R^2(CV) compared to all the models from other works. Thus, more exhaustive parameter optimization for DNN may offer a very good venue to further enhance the model performance, although there are so many hyper-parameters in DNN. (5) Consensus strategy is suggested to balance the pros and cons of each machine learning based model. (6) The full compliance with OECD guideline including the intuitive model interpretation and defined applicability domain of the model is strongly recommended. (7) Software development with the in-depth encapsulation of prediction model, fingerprint generation, and feature selection in the automatic manner is also very important for other users to apply the prediction model to their projects.

In the near future, we envision that the machine-learning based sweetener/sweetness prediction will become more and more effective and pragmatic, if it can be seamlessly fused with the other computational methods and experimental techniques. In our opinions, the performance of machine learning based model is heavily reliant on the initial high-quality dataset, which can be sustainably extended by the experimental high-throughput screening on the sweet taste receptor. Moreover, the machine-learning based sweetener/sweetness prediction belongs to the ligand-based approach and is expected to further combine with the structure-based sweetener prediction such as the molecular dynamics simulation, free energy calculation with the enhanced sampling or molecular docking methods on the basis of the modeled 3D structure of sweet taste receptor, although solving the crystal structure of the sweet taste receptor remains challenging so far. Thus, in the near future, the in-depth integration of machine-learning based sweetener/sweetness prediction, structure-based sweetener/sweetness prediction, and the experimental high-throughput screening based on the sweet taste receptor will provide a good paradigm for the discovery and development of novel sweeteners.

CONCLUSION

In this work, we present a machine-learning based platform "e-Sweet," which is developed for the experimental food scientists to automatically predict the sweetener and its corresponding RS. This platform provides several advantageous functions. (1) Users can visualize and inquiry our curated datasets that are all publicly available; (2) Four consensus sweetener/non-sweetener classification models in e-Sweet, derived from the largest dataset (530 sweeteners and 850 non-sweeteners), offer the best performance with the 95% confidence intervals for the accuracy (0.91 ± 0.01), precision (0.90 ± 0.01), specificity (0.94 ± 0.01), sensitivity (0.86 ± 0.01), F1-score (0.88 ± 0.01), MCC (0.81 ± 0.01), NER (0.90 ± 0.01), and ΔNER (0.03 ± 0.01), respectively on the test set, and prominently outperforms the results from the work of Rojas et al. (NER = 0.85); (3) The RS prediction model is harvested on the basis of the second largest dataset (352 sweeteners with the RS) and gives the robust outcome with the 95% confidence intervals for the R^2(test set) and ΔR^2 of 0.77 ± 0.01 and 0.03 ± 0.01, respectively, which is also better than other works based on the conformation-independent 2D molecular descriptors in terms of both R^2(test set) and ΔR^2. (4) Both the classification and regression models are trained with the multiple machine-learning methods and fully comply with the guidelines of OECD. (5) Interactive visualization of fingerprint bit, 3D structural feature, and feature importance. Therefore, we hope that this comprehensive platform can enable the experimental food scientists to exploit the machine-learning methods to boost the discovery and development of more novel sweeteners with the high potency.

AUTHOR CONTRIBUTIONS

SZ wrote the manuscript, SZ and WC performed the calculations and analysis. WC and WX curated the dataset and test the program, YX and FL revised the manuscript. SZ and FL conceived the workflow and developed the program.

FUNDING

This work was supported by the Natural Science Foundation of Zhejiang Province (Grant Number: Q7LY17B030007), the National Natural Science Foundation of China (Grant Number: 21502144), and the startup funding of Wenzhou Medical University.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fchem.2019.00035/full#supplementary-material

REFERENCES

Acevedo, W., Ramírez-Sarmiento, C. A., and Agosin, E. (2018). Identifying the interactions between natural, non-caloric sweeteners and the human sweet receptor by molecular docking. *Food Chem.* 264, 164–171. doi: 10.1016/j.foodchem.2018.04.113

Ahmed, J., Preissner, S., Dunkel, M., Worth, C. L., Eckert, A., and Preissner, R. (2011). SuperSweet-a resource on natural and artificial sweetening agents. *Nucleic Acids Res.* 39, 377–382. doi: 10.1093/nar/gkq917

Banerjee, P., and Preissner, R. (2018). BitterSweetForest: a random forest based binary classifier to predict bitterness and sweetness of chemical compounds. *Front. Chem.* 6:93. doi: 10.3389/fchem.2018.00093

Braga, R. C., Alves, V. M., Muratov, E. N., Strickland, J., Kleinstreuer, N., Trospsha, A., et al. (2017). Pred-Skin: a fast and reliable web application to assess skin sensitization effect of chemicals. *J. Chem. Inf. Model.* 57, 1013–1017. doi: 10.1021/acs.jcim.7b00194

Chen, L., Li, Y., Zhao, Q., Peng, H., and Hou, T. (2011). ADME evaluation in drug discovery. 10. predictions of P-glycoprotein inhibitors using recursive partitioning and naive bayesian classification techniques. *Mol. Pharm.* 8, 889–900. doi: 10.1021/mp100465q

Chéron, J. B., Casciuc, I., Golebiowski, J., Antonczak, S., and Fiorucci, S. (2017). Sweetness prediction of natural compounds. *Food Chem.* 221, 1421–1425. doi: 10.1016/j.foodchem.2016.10.145

Drew, M. G. B., Wilden, G. R. H., Spillane, W. J., Walsh, R. M., Ryder, C. A., and Simmie, J. M. (1998). Quantitative structure–activity relationship studies of sulfamates RNHSO3Na: distinction between sweet, sweet-bitter, and bitter molecules. *J. Agric. Food Chem.* 46, 3016–3026. doi: 10.1021/jf980095c

Dubois, G. E. (2016). Molecular mechanism of sweetness sensation. *Physiol. Behav.* 164, 453–463. doi: 10.1016/j.physbeh.2016.03.015

Dubois, G. E., and Prakash, I. (2012). Non-caloric sweeteners, sweetness modulators, and sweetener enhancers. *Annu. Rev. Food Sci. Technol.* 3, 353–380. doi: 10.1146/annurev-food-022811-101236

Ekins, S., Williams, A. J., and Xu, J. J. (2010). A predictive ligand-based bayesian model for human drug-induced liver injury. *Drug Metab. Dispos.* 38, 2302–2308. doi: 10.1124/dmd.110.035113

Fernstrom, J. D. (2015). Non-nutritive sweeteners and obesity. *Annu. Rev. Food Sci. Technol.* 6, 119–136. doi: 10.1146/annurev-food-022814-015635

Goel, A., Gajula, K., Gupta, R., and Rai, B. (2018). *In-silico* prediction of sweetness using structure-activity relationship models. *Food Chem.* 253, 127–131. doi: 10.1016/j.foodchem.2018.01.111

Hu, G., Kuang, G., Xiao, W., Li, W., Liu, G., and Tang, Y. (2012). Performance evaluation of 2D fingerprint and 3D shape similarity methods in virtual screening. *J. Chem. Inf. Model.* 52, 1103–1113. doi: 10.1021/ci300030u

Iwamura, H. (1981). Structure-sweetness relationship of L-aspartyl dipeptide analogs. A receptor site topology. *J. Med. Chem.* 24, 572–583. doi: 10.1021/jm00137a018

Jean-Baptiste, C., Jerome, G., Serge, A., and Sebastien, F. (2017). The anatomy of mammalian sweet taste receptors. *Proteins Struct. Funct. Bioinf.* 85, 332–341. doi: 10.1002/prot.25228

Jiang, P., Cui, M., Zhao, B., Liu, Z., Snyder, L. A., Benard, L. M. J., et al. (2005). Lactisole interacts with the transmembrane domains of human T1R3 to inhibit sweet taste. *J. Biol. Chem.* 280, 15238–15246. doi: 10.1074/jbc.M414287200

Kim, S. K., Chen, Y., Abrol, R., Goddard, W. A., and Guthrie, B. (2017). Activation mechanism of the G protein-coupled sweet receptor heterodimer with sweeteners and allosteric agonists. *Proc. Natl. Acad. Sci. U.S.A.* 114, 2568–2573. doi: 10.1073/pnas.1700001114

Kinghorn, A. D., and Soejarto, D. D. (2002). Discovery of terpenoid and phenolic sweeteners from plants. *Pure Appl. Chem.* 74, 1169–1179. doi: 10.1351/pac200274071169

Koutsoukas, A., St. Amand, J., Mishra, M., and Huan, J. (2016). Predictive toxicology: modeling chemical induced toxicological response combining circular fingerprints with random forest and support vector machine. *Front. Enrivon. Sci.* 4:11. doi: 10.3389/fenvs.2016.00011

Laffitte, A., Neiers, F., and Briand, L. (2014). Functional roles of the sweet taste receptor in oral and extraoral tissues. *Curr. Opin. Clin. Nutr. Metab. Care* 17, 379–385. doi: 10.1097/mco.0000000000000058

Martin Bland, J., and Altman, D. (1986). Statistical methods for assessing agreement between two methods of clinical measurement. *Lancet* 327, 307–310. doi: 10.1016/S0140-6736(86)90837-8

Masuda, K., Koizumi, A., Nakajima, K. I., Tanaka, T., Abe, K., Misaka, T., et al. (2012). Characterization of the modes of binding between human sweet taste receptor and low-molecular-weight sweet compounds. *PLoS ONE* 7:e35380. doi: 10.1371/journal.pone.0035380

Meyers, B., and Brewer, M. S. (2008). Sweet taste in man: a review. *J. Food Sci.* 73, 81–90. doi: 10.1111/j.1750-3841.2008.00832.x

Mishra, A., Ahmed, K., Froghi, S., and Dasgupta, P. (2015). Systematic review of the relationship between artificial sweetener consumption and cancer in humans: analysis of 599,741 participants. *Int. J. Clin. Pract.* 69, 1418–1426. doi: 10.1111/ijcp.12703

Oboyle, N. M., Banck, M., James, C. A., Morley, C., Vandermeersch, T., and Hutchison, G. R. (2011). Open babel: an open chemical toolbox. *J. Cheminform.* 3, 33–46. doi: 10.1186/1758-2946-3-33

Ojha, P. K., and Roy, K. (2017). Development of a robust and validated 2D-QSPR model for sweetness potency of diverse functional organic molecules. *Food Chem. Toxicol.* 112, 551–562. doi: 10.1016/j.fct.2017.03.043

Rodríguez-Pérez, R., Vogt, M., and Bajorath, J. (2017). Support vector machine classification and regression prioritize different structural features for binary compound activity and potency value prediction. *ACS Omega* 2, 6371–6379. doi: 10.1021/acsomega.7b01079

Rogers, D., and Hahn, M. (2010). Extended-connectivity fingerprints. *J. Chem. Inf. Model.* 50, 742–754. doi: 10.1021/ci100050t

Rojas, C., Ballabio, D., Consonni, V., Tripaldi, P., Mauri, A., and Todeschini, R. (2016a). Quantitative structure-activity relationships to predict sweet and non-sweet tastes. *Theor. Chem. Acc.* 135, 66–78. doi: 10.1007/s00214-016-1812-1

Rojas, C., Duchowicz, P., Diez, R., and Tripaldi, P. (2016b). "Applications of quantitative structure-relative sweetness relationships in food chemistry," in *Chemometrics Applications and Research*, eds A. G. Mercader, P. R. Duchowicz, and P. M. Sivakumar (Toronto, ON: Apple Academic Press), 317–339.

Rojas, C., Todeschini, R., Ballabio, D., Mauri, A., Consonni, V., Tripaldi, P., et al. (2017). A QSTR-based expert system to predict sweetness of molecules. *Front. Chem.* 5:53. doi: 10.3389/fchem.2017.00053

Rojas, C., Tripaldi, P., and Duchowicz, R. P. (2016c). A new QSPR study on relative sweetness. *Int. J. Quant. Struct. Prop. Relat.* 1, 78–93. doi: 10.4018/IJQSPR.2016010104

Roper, S. D., and Chaudhari, N. (2017). Taste buds: cells, signals and synapses. *Nat. Rev. Neurosci.* 18, 485–497. doi: 10.1038/nrn.2017.68

Shrivastav, A., and Srivastava, S. (2013). Human sweet taste receptor: complete structure prediction and evaluation. *Int. J. Chem. Anal. Sci.* 4, 24–32. doi: 10.1016/j.ijcas.2013.03.002

Tropsha, A. (2010). Best practices for QSAR model development, validation, and exploitation. *Mol. Inform.* 29, 476–488. doi: 10.1002/minf.201000061

Vepuri, S. B., Tawari, N. R., and Degani, M. S. (2007). Quantitative structure–activity relationship study of some aspartic acid analogues to correlate and predict their sweetness potency. *QSAR Comb. Sci.* 26, 204–214. doi: 10.1002/qsar.200530191

Wei, Y. C. (2011). PaDEL-descriptor: an open source software to calculate molecular descriptors and fingerprints. *J Comput. Chem.* 32, 1466–1474. doi: 10.1002/jcc.21707

Wiener, A., Shudler, M., Levit, A., and Niv, M. Y. (2012). BitterDB: a database of bitter compounds. *Nucleic Acids Res.* 40, 413–419. doi: 10.1093/nar/gkr755

Yang, X., Chong, Y., Yan, A., and Chen, J. (2011). *In-silico* prediction of sweetness of sugars and sweeteners. *Food Chem.* 128, 653–658. doi: 10.1016/j.foodchem.2011.03.081

Zheng, S., Jiang, M., Zhao, C., Zhu, R., Hu, Z., Xu, Y., et al. (2018). e-Bitter: bitterant prediction by the consensus voting from the machine-learning methods. *Front. Chem.* 6:82. doi: 10.3389/fchem.2018.00082

Zhong, M., Chong, Y., Nie, X., Yan, A., and Yuan, Q. (2013). Prediction of sweetness by multilinear regression analysis and support vector machine. *J. Food Sci.* 78, 1445–1450. doi: 10.1111/1750-3841.12199

Conflict of Interest Statement: 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.

frontiers
in Chemistry

ORIGINAL RESEARCH
published: 10 May 2019
doi: 10.3389/fchem.2019.00324

Deep Neural Network Classifier for Virtual Screening Inhibitors of (S)-Adenosyl-L-Methionine (SAM)-Dependent Methyltransferase Family

Fei Li[1,2†], Xiaozhe Wan[2,3†], Jing Xing[2,4], Xiaoqin Tan[2,3], Xutong Li[2,3], Yulan Wang[2], Jihui Zhao[2,3], Xiaolong Wu[2,5], Xiaohong Liu[2,6], Zhaojun Li[7], Xiaomin Luo[2], Wencong Lu[1*] and Mingyue Zheng[2*]*

[1] Department of Chemistry, College of Sciences, Shanghai University, Shanghai, China, [2] State Key Laboratory of Drug Research, Drug Discovery and Design Center, Shanghai Institute of Materia Medica, Chinese Academy of Sciences, Shanghai, China, [3] School of Pharmacy, University of Chinese Academy of Sciences, Beijing, China, [4] Department of Pediatrics and Human Development, Michigan State University, East Lansing, MI, United States, [5] School of Pharmacy, East China University of Science and Technology, Shanghai, China, [6] School of Life Science and Technology, Shanghai Tech University, Shanghai, China, [7] School of Information Management, Dezhou University, Dezhou, China

OPEN ACCESS

Edited by:
John C. Cancilla,
Scintillon Institute, United States

Reviewed by:
Yun Tang,
East China University of Science and Technology, China
Sixue Zhang,
Southern Research Institute, United States

****Correspondence:***
Xiaomin Luo
xmluo@simm.ac.cn
Wencong Lu
wclu@shu.edu.cn
Mingyue Zheng
myzheng@simm.ac.cn

†These authors have contributed equally to this work

Specialty section:
This article was submitted to Theoretical and Computational Chemistry,
a section of the journal Frontiers in Chemistry

Received: *06 March 2019*
Accepted: *23 April 2019*
Published: *10 May 2019*

Citation:
Li F, Wan X, Xing J, Tan X, Li X, Wang Y, Zhao J, Wu X, Liu X, Li Z, Luo X, Lu W and Zheng M (2019) Deep Neural Network Classifier for Virtual Screening Inhibitors of (S)-Adenosyl-L-Methionine (SAM)-Dependent Methyltransferase Family. Front. Chem. 7:324. doi: 10.3389/fchem.2019.00324

The (S)-adenosyl-L-methionine (SAM)-dependent methyltransferases play essential roles in post-translational modifications (PTMs) and other miscellaneous biological processes, and are implicated in the pathogenesis of various genetic disorders and cancers. Increasing efforts have been committed toward discovering novel PTM inhibitors targeting the (S)-Adenosyl-L-methionine (SAM)-binding site and the substrate-binding site of methyltransferases, among which virtual screening (VS) and structure-based drug design (SBDD) are the most frequently used strategies. Here, we report the development of a target-specific scoring model for compound VS, which predict the likelihood of the compound being a potential inhibitor for the SAM-binding pocket of a given methyltransferase. Protein-ligand interaction characterized by Fingerprinting Triplets of Interaction Pseudoatoms was used as the input feature, and a binary classifier based on deep neural networks is trained to build the scoring model. This model enhances the efficiency of the existing strategies used for discovering novel chemical modulators of methyltransferase, which is crucial for understanding and exploring the complexity of epigenetic target space.

Keywords: deep neural network, virtual screening, methyltransferase, epigenetic, drug design

INTRODUCTION

Methyltransferases (MTases) are a class of enzymes that transfer methyl groups to the substrates including DNA, proteins and small molecules (Zhang and Zheng, 2016). Based on different substrates, MTases can be divided into three classes: DNA methyltransferases (DNMTs) (Da Costa et al., 2017), protein methyltransferases (PMTs) (Boriack-Sjodin and Swinger, 2016) and MTases for small molecules like catecholamines (Bonifácio et al., 2007). Most methyltransferases use S-adenosyl-L-methionine (SAM) as a donor for methyl groups, where all have a SAM-binding pocket and a substrate-binding pocket (Martin and McMillan, 2002). These SAM-dependent MTases participate in numerous essential biological processes, including the epigenetic control of cell fate, cell signaling and degration of metabolites (Hu et al., 2015; Schapira, 2016). Consequently, the dysregulation of MTases have been implicated in diverse diseases including of many types of cancers (Kaniskan et al., 2015), metabolic disorders (Deng et al., 2013), cardiovascular disease (Bouras et al., 2013), inflammatory response (Sun et al., 2015), neurological disorders (Meaney and Ferguson-Smith, 2010), and so on. Therefore, SAM-dependent MTases have been considered as a type of intriguing targets for pharmacological intervention, and interest in developing potent MTase inhibitors continues to grow in both academic laboratories and pharmaceutical companies (Hu et al., 2016). Targeting the SAM-binding pocket is an effective strategy for designing methyltransferase inhibitors, akin to targeting the ATP-binding pocket of kinases (Wu et al., 2015). A number of inhibitors binding to SAM pocket have been reported, including SGI-1027 (Rilova et al., 2014), CPI-1205 (Vaswani et al., 2016),

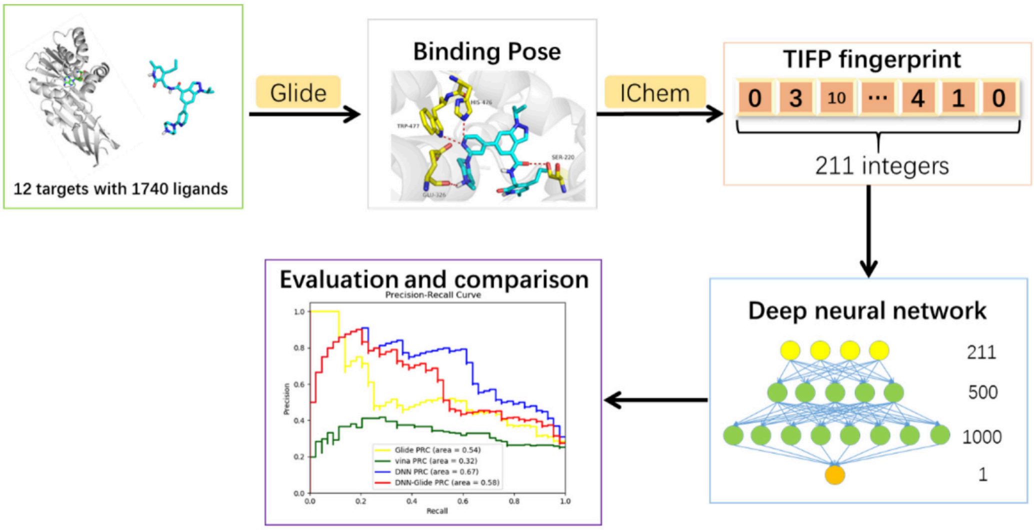


FIGURE 1 | Overall workflow of model construction.

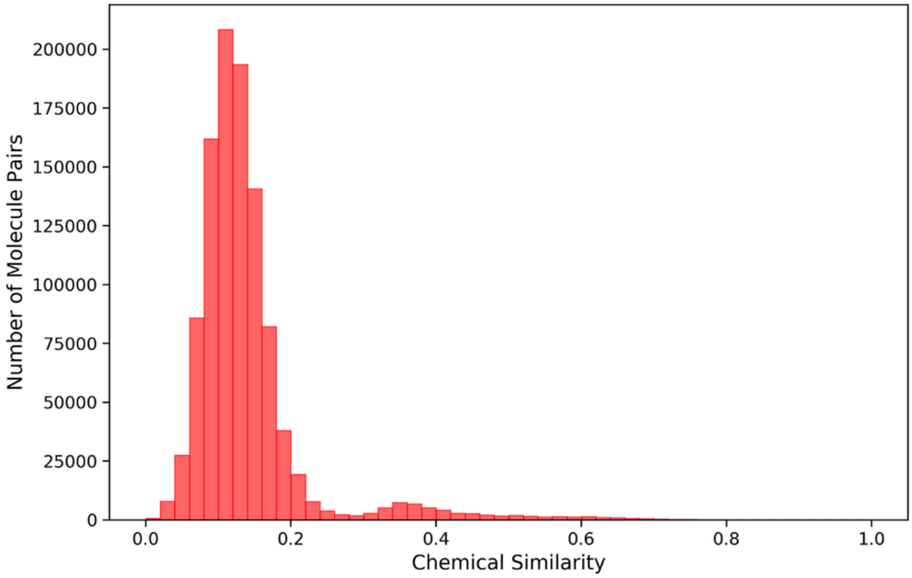


FIGURE 2 | Histogram showing the distribution of chemical similarity of any two molecules in the dataset.

EPZ-6438 (Kuntz et al., 2016), GSK-126 (McCabe et al., 2012), EPZ-5676 (Stein et al., 2018), and so on (Biswas and Rao, 2018). Among them, pyridone-based EZH2 inhibitors CPI-1205, EPZ-6438 and GSK-126 have been in phase I clinical trials. In addition, compound EPZ-5676 has finished phase I clinical trials for relapsed/refractory leukemias bearing a rearrangement of the MLL gene, and has modest clinical activity in adult acute leukemia. So far, there is still no small molecule MTases inhibitors being approved, and many projects were temporarily halted partially due to poor *in vivo* activity or unsatisfactory bioavailability of current chemo types. Therefore, finding of MTases inhibitors with novel scaffolds is still a challenging research area.

To discover and design new MTases inhibitors more efficiently, a variety of computational methods have been developed and used in combination with experiment methods (Kireev, 2016). For example, virtual screening based on molecular docking has been widely used to discover potential small molecule leads (Kireev, 2016). Existing molecular docking methods typically consists of conformation searching and a scoring function for complex binding affinity evaluation (Morris and Lim-Wilby, 2008). These molecular docking methods can produce the binding poses with acceptable accuracy, but they are less successful in scoring and active compound ranking, leading to high false positive rates in virtual screening campaigns (Berishvili et al., 2018). Furthermore, the performance of molecular docking for different targets may vary widely, especially with regard to the complexity of methyltransferase family targets. Previously our group constructed a knowledge-based general-purposed scoring function iPMF (Shen et al.,

TABLE 1 | The searched hyperparameters and their performance.

Hyperparameters				Performance							
				Train				Valid			
Dropout	Learning rate	Layer size	Stop epoch	Recall	Precision	ROC-AUC	PRC-AUC	Recall	Precision	ROC-AUC	PRC-AUC
0.1	0.0001	[500, 100]	42	0.75	0.86	0.96	0.91	0.55	0.76	0.82	0.76
0.2	0.0001	[500, 100]	46	0.68	0.67	0.90	0.79	0.57	0.68	0.80	0.72
0.1	0.0001	[100, 500]	50	0.74	0.92	0.97	0.93	0.58	0.74	0.84	0.75
0.2	0.0001	[100, 500]	40	0.59	0.94	0.94	0.88	0.42	0.88	0.81	0.75
0.1	0.0001	[320, 640]	31	0.68	0.93	0.97	0.92	0.47	0.86	0.81	0.76
0.2	0.0001	[320, 640]	40	0.69	0.91	0.96	0.91	0.47	0.86	0.81	0.74
0.1	0.0001	[500, 1,000]	29	0.79	0.87	0.97	0.91	0.58	0.70	0.84	0.76
0.2	0.0001	[500, 1,000]	22	0.42	0.94	0.86	0.77	0.28	0.94	0.75	0.71
0.1	0.001	[500, 100]	29	0.80	0.88	0.98	0.94	0.60	0.70	0.82	0.76
0.2	0.001	[500, 100]	27	0.54	0.84	0.92	0.82	0.49	0.70	0.82	0.74
0.1	0.001	[100, 500]	21	0.66	0.93	0.95	0.91	0.51	0.79	0.82	0.72
0.2	0.001	[100, 500]	78	0.79	0.93	0.98	0.95	0.55	0.78	0.80	0.75
0.1	0.001	[320, 640]	32	0.91	0.90	0.99	0.97	0.60	0.68	0.80	0.74
0.2	0.001	[320, 640]	102	0.83	0.99	0.99	0.98	0.58	0.84	0.81	0.72
0.1	0.001	[500, 1,000]	9	0.74	0.77	0.94	0.85	0.66	0.73	0.87	0.81
0.2	0.001	[500, 1,000]	34	0.85	0.99	0.99	0.98	0.60	0.78	0.83	0.78
0.1	0.0005	[500, 100]	18	0.67	0.80	0.93	0.85	0.60	0.78	0.79	0.74
0.2	0.0005	[500, 100]	21	0.67	0.73	0.92	0.82	0.60	0.73	0.81	0.75
0.1	0.0005	[100, 500]	43	0.88	0.96	0.99	0.99	0.62	0.83	0.83	0.77
0.2	0.0005	[100, 500]	51	0.84	0.93	0.98	0.96	0.57	0.79	0.84	0.77
0.1	0.0005	[320, 640]	28	0.82	0.96	0.99	0.98	0.51	0.71	0.81	0.74
0.2	0.0005	[320, 640]	24	0.77	0.93	0.97	0.94	0.49	0.68	0.80	0.74
0.1	0.0005	[500, 1,000]	17	0.79	0.82	0.95	0.88	0.60	0.65	0.78	0.70
0.2	0.0005	[500, 1,000]	14	0.74	0.84	0.95	0.87	0.60	0.73	0.80	0.74
0.1	0.00005	[500, 100]	82	0.72	0.92	0.97	0.93	0.53	0.78	0.84	0.77
0.2	0.00005	[500, 100]	131	0.69	0.97	0.97	0.94	0.49	0.81	0.83	0.76
0.1	0.00005	[100, 500]	55	0.51	0.94	0.92	0.84	0.36	0.83	0.78	0.72
0.2	0.00005	[100, 500]	87	0.57	0.97	0.95	0.89	0.42	0.92	0.79	0.75
0.1	0.00005	[320, 640]	40	0.57	0.95	0.93	0.87	0.42	0.88	0.77	0.73
0.2	0.00005	[320, 640]	50	0.64	0.83	0.92	0.84	0.43	0.68	0.82	0.71
0.1	0.00005	[500, 1,000]	46	0.68	0.96	0.96	0.93	0.45	0.96	0.81	0.79
0.2	0.00005	[500, 1,000]	71	0.77	0.91	0.97	0.93	0.51	0.75	0.81	0.73

2011), which utilizes the interative-extracted statistical potentials from protein-ligand complexes. However, the SAM-binding sites exhibit great polarity and structural flexibility; therefore, it is difficult for the general-purpose scoring functions like iPMF to perform satisfactorily for this system. It is therefore a practical compromise constructing a scoring function specific for SAM-dependent MTases. Many target-specific scoring functions have been constructed through different methods to improve the performance of existing scoring functions on certain targets to varying degree (Xing et al., 2017; Berishvili et al., 2018). Recently, our group developed a SAM-dependent methyl transferase-specific scoring function SAM-score using ε-SVR, and used this scoring function in discovery of a new class of DOT1L inhibitors (Wang et al., 2017). Regrettably, despite a lower rate of false positive in our in-house use, the SAM-score still leaves large room for improvement. For example, the Enrichment Factor (EF) (5%) of SAM-score was only 1.46 in one of our recent tests, which means that the screening power of the scoring model is not satisfactory.

Recently, deep learning-based approaches have emerged in the field of scoring function. For instance, Jiménez et al. constructed a general-purpose scoring function K_{DEEP} via 3D-convolutional neural networks (Jiménez et al., 2018). There are clear differences between deep learning and traditional machine learning methods, for example: traditional machine learning methods uses sparse representations to describe the input data, and learning-task related features are further extracted from the representations, which needs extensive domain knowledge and time investment, and may lose some important information in the process; while the representation learning framework of deep learning methods uses distributed representations for the dataset and then automatically extract features, which can extract abstract higher-level features and finally generate more accurate prediction results (LeCun et al., 2015).

In this study, we developed a SAM-dependent MTases-specific classifier based on a fully connected neural network to accurately distinguish between negative (inactive) and positive (active) MTases inhibitors. First, crystal structures of the SAM-dependent MTases and the compounds with experimental affinity data against these targets were collected. Decoys for each targets were also generated to expand the data set in this step. Then, molecular docking was used to produce protein-ligand interaction conformations. Here, the Fingerprinting Triplets of Interaction Pseudo atoms (TIFP) (Desaphy et al., 2013) were used to describe the predicted complex conformations. In the next step, these TIFPs were used as inputs to establish a fully connected neural network model by mining the structure and activity relationship of previously reported small molecules for different MTases. The performance of the DNN model were also compared with Glide, Autodock·vina, and the mixed model of DNN and Glide. The results showed that DNN model can significantly improve the screening power of docking and has the ability to prioritize active molecules with diverse scaffolds. Moreover, this model can also help to determine the selectivity of the compounds targeting different MTases, which may provide insight into developing novel inhibitors of SAM-dependent MTases.

TABLE 2 | The performances of 10 trained models on the validation set.

Model	Recall	Precision	Accuracy	ROC-AUC	PRC-AUC
1	0.622	0.742	0.874	0.853	0.689
2	0.800	0.653	0.856	0.876	0.793
3	0.725	0.518	0.782	0.849	0.688
4	0.638	0.750	0.845	0.822	0.746
5	0.738	0.660	0.845	0.856	0.791
6	0.682	0.612	0.810	0.863	0.785
7	0.718	0.718	0.874	0.859	0.760
8	0.547	0.690	0.787	0.817	0.723
9	0.436	0.750	0.776	0.800	0.681
10	0.535	0.767	0.845	0.813	0.714
Average	0.64 ± 0.09	0.69 ± 0.06	0.83 ± 0.03	0.84 ± 0.02	0.74 ± 0.04

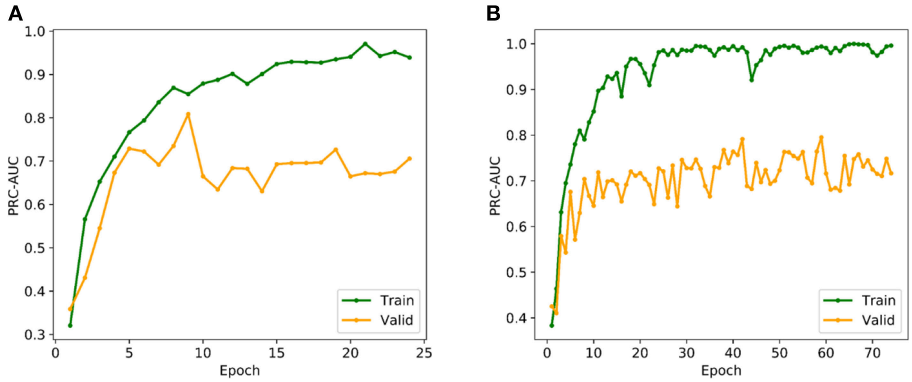


FIGURE 3 | (A) Variation tendency of PRC-AUC with epochs in DNN model. **(B)** Variation tendency of PRC-AUC with epochs in DNN-Glide model. The PRC-AUCs of the DNN model have reached the peak on the 9th epoch, while the DNN-Glide reached the peak on the 59th epoch.

RESULTS AND DISCUSSION

This research was aimed to build a target-specific classification model to distinguish whether a compound is a potential inhibitor of a given methyltransferase. The workflow contains deep neural network model construction and model evaluation steps, which will be explained in details below. The overall workflow of this study was shown in **Figure 1**.

Deep Neural Network Model Construction

Data Sources

Based on the previous work of our workgroup, the data used to build model include the same set of 12 SAM-dependent methyltransferases, which are DNA (cytosine-5)-methyltransferase 1 (DNMT1), coactivator-associated arginine methyltransferase 1 (CARM1), protein arginine N-methyltransferase 1 (PRMT1), protein arginine N-methyltransferase 3 (PRMT3), protein arginine N-methyltransferase 5 (PRMT5), protein arginine N-methyl-transferase 6 (PRMT6), euchromatic histone-lysine N-methyl-transferase 1 (EHMT1), euchromatic histone-lysine N-methyltransferase 2 (EHMT2), SET domain containing lysine methyltransferase 7 (SETD7), SET domain containing lysine methyltransferase 8 (SETD8), suppressor of variegation 3-9 homolog 2 (SUV39H2) and disruptor of telomeric silencing 1-like histone H3K79 methyltransferase (DOT1L). The crystal structures in the data set are derived from the Protein Data Bank (PDB) (https://www.rcsb.org), which are all complex crystal structures with a ligand occupying the SAM pocket. The structures and activities data of small molecule ligands for the 12 targets were collected from the ChEMBL database, and the IC50, EC50, and Ki values less than or equal to 10 micromole were used as positive data, and that more than 50 micromole as negative data. Totally, there were 919 positive samples and 366 negative samples. The IC_{50}, EC_{50}, and K_i values in the activity data were normalized to PIC_{50}, PEC_{50} or PK_i (PActivation = 9 – lg(Activation)). Furthermore, a total of 1212 decoys were generated in the DUD-E website (http://dude.docking.org/generate) (Mysinger et al., 2012) to better correspond to the fact of actual virtual screening where the negative data are much more than the positive data. Each molecule, either positive or negative, has at least one of 12 Mtase targets reported. The 211-bit TIFP interaction fingerprints (Desaphy et al., 2013) were used as inputs to construct the deep neural network classification model, due to its capability in characterizing directional molecular interactions such as hydrogen bonding and pi-pi stacking. Totally, 1740 molecules were compiled for deriving interaction features, which including 446 positive data and 1294 negative data. Tanimoto coefficients of Morgan fingerprints of any two molecules in the data set were calculated by RDKit python package (**Figure 2**), and most of them were below 0.2, suggesting that the data set has diverse chemical structures and would make the DNN model less biased.

Datasets Partition

(1) The total 1,740 samples were randomly divided into two parts with the proportion 1:10, in which the smaller one was used as a test set.
(2) The bigger one was shuffled and randomly divided into a validation set and a train set with the proportion of 1:8, which were used in the hyperparameter optimization processing.
(3) This process of step 2 was repeated for ten times to obtain ten different training/validation datasets, and the best model

TABLE 3 | The performances of 4 methods on the test set, and the best performed method and its metrics are shown in bold.

Method	ROC-AUC	PRC-AUC	EF (5%)
Glide	0.75	0.54	2.97
Autodock vina	0.61	0.32	0.99
DNN	**0.86**	**0.67**	**3.46**
DNN-Glide	0.80	0.58	3.46

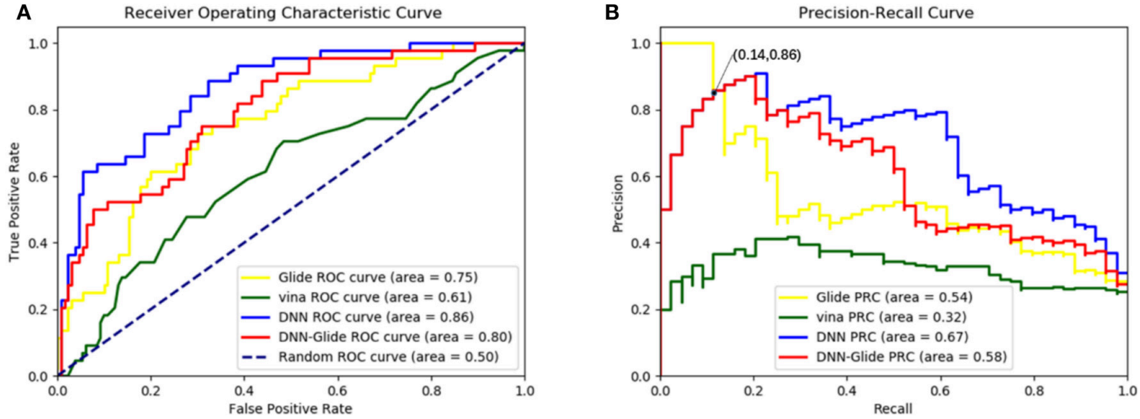


FIGURE 4 | (A) The ROC curves of Glide, Autodock vina, DNN model and DNN-Glide model on the test set. **(B)** The PRC curves of Glide, Autodock vina, DNN model and DNN-Glide model on the test set.

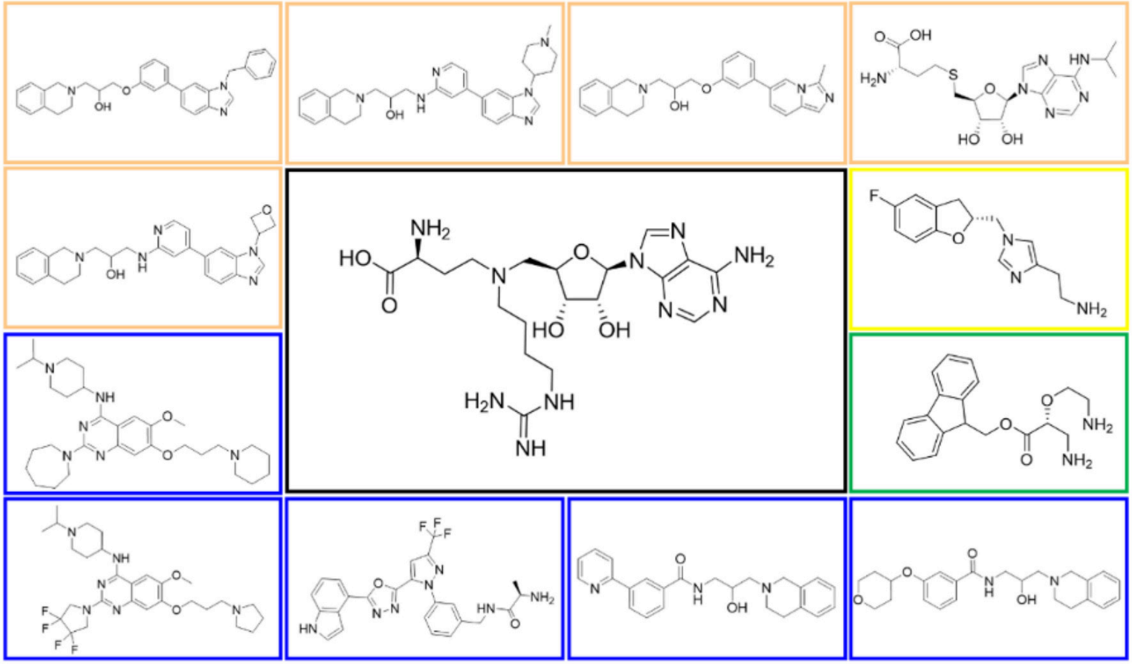


FIGURE 5 | Structures of the positive compounds predicted by Glide and DNN model before the intersection. Structures in darkorange and yellow box are predicted to be positive by Glide; structures in dark, blue and green box are predicted to be positive by DNN model.

among the models trained on the ten datasets was evaluated with the test set.

Hyperparameter Optimization

The multi-grid searching method was applied to the optimization of the hyperparameters. Because the area under Precision-Recall curve (PRC-AUC) is more informative than the area under receiver operating characteristic curve (ROC-AUC) when evaluating classifiers on imbalanced datasets (Saito and Rehmsmeier, 2015), PRC-AUC on the validation set was used for the evaluation of the hyperparameters. During training process, Adam optimizer was used for model optimization and cross-entropy was utilized as the loss function, which is a common loss function for classification model. Early stopping with a stop window size of 15 was used to save training time and to prevent over-fitting, i.e., training would be stopped if the PRC-AUC on the validation set did not increase for 15 consecutive epochs. The performance of evaluated hyperparameters in the hyperparametric search are shown in **Table 1**. According to the best set of hyperparameters, a fully connected three-layer neural network model with two hidden layers (500 × 1,000) was established. The input layer had 211 neurons, and the output layer was softmax-standardized dichotomous probability. Learning rate, weight decay penalty and dropout were set to 0.001, 0.0001, and 0.1, respectively. The activation function was set as ReLU. **Figure 3A** shows the variation tendency of PRC-AUC with epochs on training set and validation set when the DNN model was trained with the best set of hyperparameters. The PRC-AUCs of DNN model have reached the peak on the ninth epoch, and the model at that epoch was used for further evaluation.

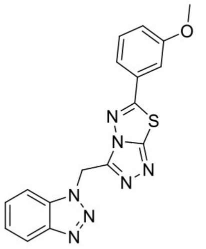
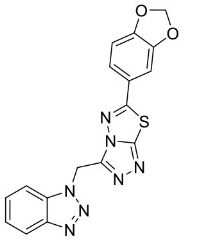
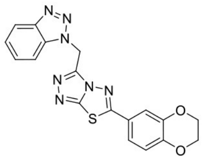
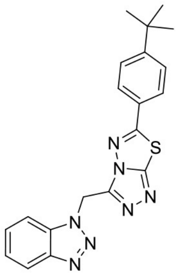
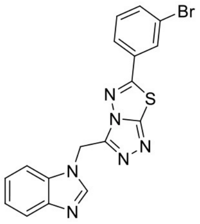
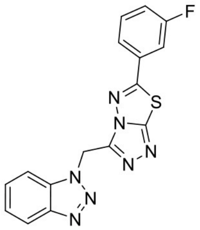
Model Evaluation and Comparison

DNN Model Evaluation

To validate the feasibility and effectiveness of the models, the searched best set of hyperparameters were then trained on 10 datasets and evaluated on the validation set. The performances of these 10 models were similar, as shown in **Table 2**, among which the performance of 2nd model has the best PRC-AUC and ROC-AUC (Bradley, 1997) here, which was selected for further evaluation on the test set. It showed PRC-AUC, ROC-AUC and EF (5%) of 0.67, 0.86 and 3.46, respectively, on the test set.

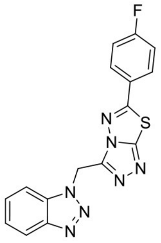
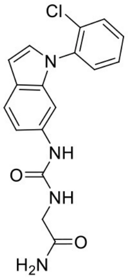
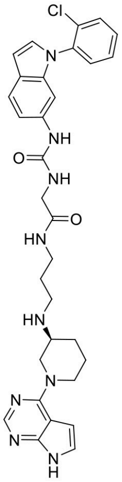
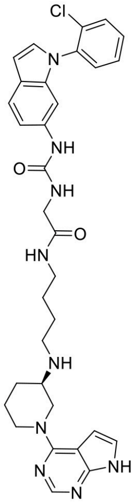
In order to evaluate the DNN model comprehensively, Glide and Autodock vina were compared with the DNN model. The docking score of the Glide SP was added as a descriptor to the end of interaction fingerprint, which was used to build a hybrid model named DNN-Glide. The DNN-Glide model was trained in the same way as DNN model and on the same datasets, and it obtained the same set of best hyperparameters as DNN model, although there is a delay of reaching the best PRC-AUC on the validation set (**Figure 3B**). By comparison, both the ROC curves and PRC curves of the DNN model were above that of the other models, indicating the high-quality performance of the DNN model (**Figure 4** and **Table 3**). Especially, the true positive rate of DNN is consistently higher than that of Glide and Autodock vina when the false positive rate was extremely

TABLE 4 | The ligands of DOT1L and their scores valued by Glide, Autodock vina and DNN model.

Label	Structure	Smiles	pIC$_{50}$	Glide score	Vina score	DNN score
C170206_10		COC1=CC=CC(=C1)C1=NN2C(CN3N=NC4=CC=CC=C34)=NN=C2S1	5.19	−7.627	−8.5	0.8542
C170206_15		C(N1N=NC2=CC=CC=C12)C1=NN=C2SC(=NN12)C1=CC=C2OCOC2=C1	5.40	−7.915	−8.4	0.9927
C170206_16		C(N1N=NC2=CC=CC=C12)C1=NN=C2SC(=NN12)C1=CC=C2OCCOC2=C1	5.08	−7.898	−9.9	0.9821
C170206_17		CC(C)(C)C1=CC=C(C=C1)C1=NN2C(CN3N=NC4=CC=CC=C34)=NN=C2S1	5.35	−5.374	−8.8	0.555
C170206_39		BrC1=CC(=CC=C1)C1=NN2C(CN3C=NC4=CC=CC=C34)=NN=C2S1	5.39	−8.233	−8.8	0.9007
C170206_6		FC1=CC=CC(=C1)C1=NN2C(CN3N=NC4=CC=CC=C34)=NN=C2S1	5.08	−6.629	−8.7	0.8713

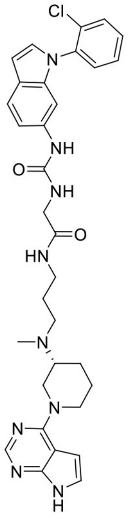
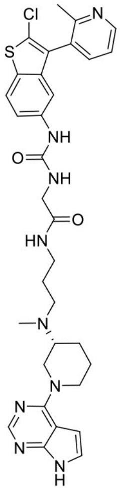
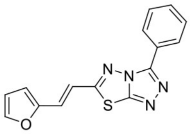
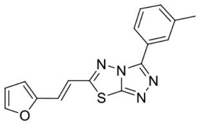
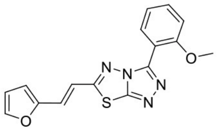
(Continued)

TABLE 4 | Continued

Label	Structure	Smiles	pIC_{50}	Glide score	Vina score	DNN score
C170206_9		FC1=CC=C(C=C1)C1=NN2C(CN3N=NC4=CC=CC=C34)=NN=C2S1	5.33	−7.839	−8.4	0.9439
C170214_3		NC(=O)CNC(=O)NC1=CC2=C(C=CN2C2=C(Cl)C=CC=C2)C=C1	5.05	−8.322	−8.3	0.6092
C170214_4		ClC1=CC=CC=C1N1C=CC2=C1C=C(NC(=O)NCC(=O)NCCCN[C@@H]1CCCN(C1)C1=C3C=CNC3=NC=N1)C=C2	8.4	−9.432	−8.3	0.0141
C170214_5		ClC1=CC=CC=C1N1C=CC2=C1C=C(NC(=O)NCC(=O)NCCCCN[C@@H]1CCCN(C1)C1=C3C=CNC3=NC=N1)C=C2	8.4	−7.773	−9.6	0.8417

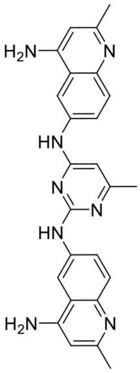
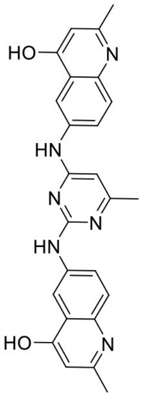
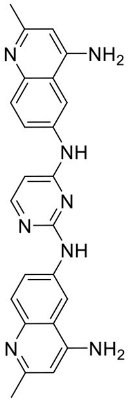
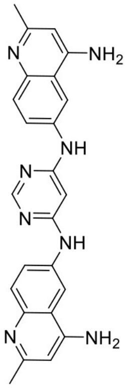
(Continued)

TABLE 4 | Continued

Label	Structure	Smiles	pIC_{50}	Glide score	Vina score	DNN score
C170214_6		CN(CCCNC(=O)CNC(=O)NC1=CC2=C(C=CN2C2=CC=CC=C2Cl)C=C1)[C@@H]1CCCN(C1)C1=C2C=CNC2=NC=N1	9.82	−10.445	−9.2	0.9239
C170214_7		CN(CCCNC(=O)CNC(=O)NC1=CC=C2SC(Cl)=C(C2=C1)C1=CC=CN=C1C)[C@@H]1CCCN(C1)C1=C2C=CNC2=NC=N1	8.52	−9.125	−8.8	0.9996
C180224_6		O1C=CC=C1C=CC1=NN2C(S1)=NN=C2C1=CC=CC=C1	5.15	−5.67	−7.9	0.8878
C180224_7		CC1=CC=CC(=C1)C1=NN=C2SC(C=CC3=CC=CO3)=NN12	5.03	−7.471	−8.3	0.6535
C180224_9		COC1=C(C=CC=C1)C1=NN=C2SC(C=CC3=CC=CO3)=NN12	5.01	−6.375	−7.8	0.9412

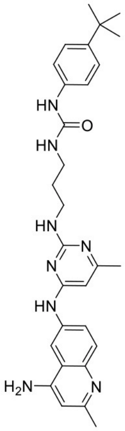
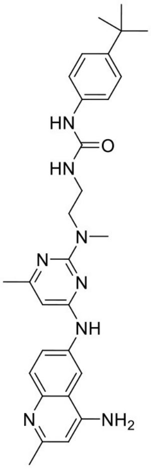
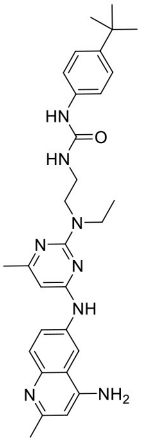
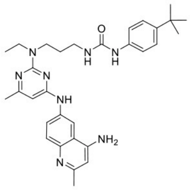
(Continued)

TABLE 4 | Continued

Label	Structure	Smiles	pIC_{50}	Glide score	Vina score	DNN score
C180722_3a		CC1=CC(N)=C2C=C(NC3=NC(NC4=CC=C5N=C(C)C=C(N)C5=C4)=CC(C)=N3)C=CC2=N1	5.82	−6.461	−9.8	0.7206
C180722_3b		CC1=NC(NC2=CC=C3N=C(C)C=C(O)C3=C2)=NC(NC2=CC=C3N=C(C)C=C(O)C3=C2)=C1	5.36	−7.745	−9.1	0.8655
C180722_3d		CC1=NC2=CC=C(NC3=CC=NC(NC4=CC=C5N=C(C)C=C(N)C5=C4)=N3)C=C2C(N)=C1	5.97	−8.945	−10.1	0.7807
C180722_3e		CC1=NC2=CC=C(NC3=CC(NC4=CC=C5N=C(C)C=C(N)C5=C4)=NC=N3)C=C2C(N)=C1	5.97	−6.465	−9.3	0.7952

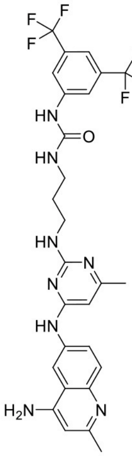
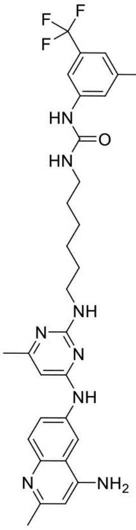
(Continued)

TABLE 4 | Continued

Label	Structure	Smiles	pIC_{50}	Glide score	Vina score	DNN score
C180722_8b		CC1=NC(NCCCNC(=O)NC2=CC=C(C=C2)C(C)(C)C)=NC(NC2=CC=C3N=C(C)C=C(N)C3=C2)=C1	5.11	−5.558	−8.3	0.9847
C180722_8f		CN(CCNC(=O)NC1=CC=C(C=C1)C(C)(C)C)C1=NC(NC2=CC=C3N=C(C)C=C(N)C3=C2)=CC(C)=N1	5.22	−7.641	−9.9	0.5408
C180722_8h		CCN(CCNC(=O)NC1=CC=C(C=C1)C(C)(C)C)C1=NC(NC2=CC=C3N=C(C)C=C(N)C3=C2)=CC(C)=N1	5.24	−5.082	−8.6	0.2014
C180722_8i		CCN(CCCNC(=O)NC1=CC=C(C=C1)C(C)(C)C)C1=NC(NC2=CC=C3N=C(C)C=C(N)C3=C2)=CC(C)=N1	5.1	−5.542	−9	0.9797

(Continued)

TABLE 4 | Continued

Label	Structure	Smiles	pIC_{50}	Glide score	Vina score	DNN score
C180722_9b	 F F F F F F HN O HN HN N N HN H_2N N	CC1=NC(NCCCNC(=O)NC2=CC(=CC(=C2)C(F)(F)F)C(F)(F)F)=NC(NC2=CC=C3N=C(C)C=C(N)C3=C2)=C1	5.06	−7.433	−8.7	0.9993
C180722_9e	 F F F F F F HN O HN NH N N NH N NH_2	CC1=NC(NCCCCCNC(=O)NC2=CC(=CC(=C2)C(F)(F)F)C(F)(F)F)=NC(NC2=CC=C3N=C(C)C=C(N)C3=C2)=C1	5.45	−7.664	−9.3	0.9985

low, which is an obvious merit for applications in virtual screening. Unfortunately, the added Glide SP didn't improve the performance of the DNN model. It is noteworthy that the PRC curve of the Glide and DNN model intersected each other at (0.14, 0.86), before the point (Recall <0.14), the precision of Glide is higher than the DNN model. **Figure 5** shows the structures of the positive compounds predicted by Glide and DNN model before the intersection point. We may find that Glide tends to retrieve compounds with one or two common scaffolds, while the DNN model is able to provide more diverse scaffolds, suggesting its generalization ability on recognizing active compounds.

To investigate the performance of the DNN model on a specific target, an external test set containing 25 molecules was collected, which were reported binding to SAM pocket of DOT1L recently (Möbitz et al., 2017; Wang et al., 2017; Song et al., 2018). The structures and the DNN model scores of the molecules were shown in the **Table 4**. There are two molecules "C180722_8h" and "C170214_4" predicted far lower than the threshold of 0.5, which means that they were wrongly classified. The reason of the wrong judge was considered to be improper inputs originated from inaccurate simulated binding conformations. Since the structure of DOT1L is flexible, especially in SAM-pocket region, crystal structures obtained from experiment are quite different, which leads to various simulated binding conformations in docking (**Figure 6**), and different conformations may cause different results. To prove the guess, a different PDB entry 5MVS (the previous used one was 1NW3) was used as receptor structure to generate input data with the two compounds. As expected, the C170214_4 and C180722_8h was evaluated with high scores of 0.90 and 0.89, respectively, which suggests that it is vital to select a suitable receptor structure for more accurate results.

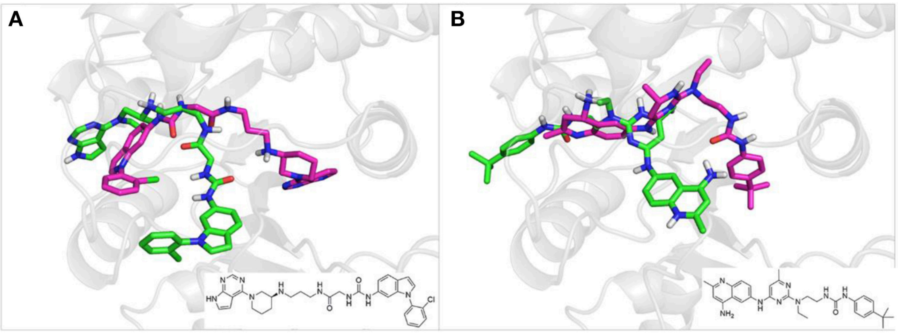


FIGURE 6 | (A) Docking poses of the molecule "C170214_4" in PDB entry of 1NW3 (green) and 5MVS (magentas). **(B)** Docking poses of the molecule "C180722_8h" in PDB entry of 1NW3 (green) and 5MVS (magentas).

Methods

Ligand-Protein Binding Conformations Generation

Accurate binding poses of protein-ligand complexes are required for extracting interaction information. In view of the fact that most collected molecules don't have available complex crystal structures with their related target, we used molecular docking to produce the binding conformations.

Cross docking was carried out to choose an appropriate receptor structure of each target for generating binding poses. At first, all the complex crystal structures of each target were aligned via pymol software (version 1.8.2.2) (Schrodinger, 2015), and then the Xglide module of Maestro version 10.2 (Schrödinger, LLC, New York, NY, 2015-2) was used for cross docking. In this process, every ligand extracted from a crystal structure was docked to collected crystal structures of the target, and the root-mean-square deviation (RMSD) values of the docked poses with reference to the corresponding native poses in the crystal structures were calculated. For every target, the crystal structure with the smallest average RMSD of all extracted ligands of this target was selected as the receptor structure for the next molecular docking. Selected protein crystal structure structures and the average RMSD values between the predict ligand binding conformations and the native conformations in crystal structures were shown in **Table 5**. According to the results of cross docking, the average RMSD between the ultimately chosen docking poses and the ligand original poses in crystal structures are all less than 1.5 Å, suggesting molecular docking is accurate in generating the protein-ligand binding conformations for MTases.

Then, all the small molecules in our dataset are docked into the chosen protein crystal structures in Glide of Maestro version 10.2. Each protein crystal structure was prepared by the Protein Preparation Wizard module of Maestro version 10.2, including adding hydrogens, assigning the bond level, creating disulfide bonds, converting selenomethionines to methionines, and filling in missing side chains using Prime, hydrogen bond network optimization and restrained minimization; removing all the water molecules and metal ions. The protein receptor grids were generated by the Maestro Receptor Grid Generation module of Maestro version 10.2, and the grid centers were set as the centroid of ligands binding in the SAM pocket. All small molecules were prepared by the LigPrep module of Maestro version 10.2, including creating 3D coordinates, calculating ionization states, generating tautomers and stereoisomers, and producing a low energy ring conformation. Grid docking was completed by the Glide module of Maestro Version 10.2, precision of which was set as SP (standard precision) and the number of poses to write out of which was limited to at most 1 per ligand. All other parameters were set as default. Only the binding pose with the best docking score was retained. According to the result of the molecular docking, some molecules preferentially bound other sites than the SAM-binding pocket, and some molecules showed lower docking scores. With the docking score of −8.2 as the threshold, the lower-scored conformations may not be the actual binding conformations, which are not studied in the virtual screening generally. These molecules were disregarded in the followed study. Similarly, binding conformers of decoys were generated through molecular docking by the same process.

Interaction Fingerprint Generation

The Fingerprinting Triplets of Interaction Pseudo atoms (TIFP) were used to encode the protein-ligand interaction patterns. Firstly, the interactions between protein and ligand are recognized, including hydrophobic contacts, aromatic interactions, hydrogen bonds, ionic interactions and metal complexation. Then, each interaction was abstract into a pseudo-atom, which is located in the position of the geometric center of the interaction, the acceptor interacted atom or the interacted ligand atom. Then, the number of triples are counted in 6 distance ranges: 0–4, 4–6, 6–9, 9–13, 13–17, 17+Å. Each type of triples is taken as one characteristic and the 211 most common characteristics are retained to form a 211-bit vector.

For each complex crystal structure used for docking, residues within 6 Å of the ligand were retained as binding site information, which was needed for the generation of TIFP fingerprints. The binding sites and selected ligand conformers were converted to the standard mol2 format using chimera (version 1.13). Standard formatted 211-bit TIFPs was generated using IChem software.

TABLE 5 | Protein crystal structure structures including the selected structure in cross-docking and the average RMSD between the predict ligand binding conformations and the native conformations.

Target	PDB ID	Ligand	Average RMSD	Selected PDB ID
CARM1	2y1w	SFG	0.32	2y1w
	5dx1	SFG	0.44	
	5is6	SFG	0.46	
	6arv	SAH	0.46	
	5dwq	SFG	0.48	
	5dxa	SFG	0.49	
	5dxj	SFG	0.49	
	5lv3	SAH	0.50	
	5dx8	SFG	0.56	
	5dx0	SFG	0.62	
	6arj	SAH	1.83	
	2v74	SAH	1.84	
	5ih3	SAH	2.09	
	5u4x	SAH	2.24	
	6d2l	FTG	3.34	
	2y1x	SAH	3.40	
	4ikp	4IK	3.45	
	5k8v	6RE	3.48	
	5is8	SAH	3.59	
	3b3f	SAH	3.65	
DNMT1	3swr	SFG	0.81	3swr
	5gut	SAH	0.89	
	3av5	SAH	0.90	
	3pta	SAH	0.98	
	3pt6	SAH	1.02	
	3pt9	SAH	1.13	
	4wxx	SAH	1.27	
	3av6	SAM	1.32	
	4da4	SAH	2.09	
DOT1L	1nw3	SAM	1.17	1nw3
	3sx0	SX0	1.19	
	4er0	AW1	1.24	
	4ek9	EP4	1.45	
	4ekg	0QJ	1.52	
	5juw	6NR	1.68	
	4eqz	AW0	1.74	
	4hra	EP6	1.74	
	3uwp	5ID	1.76	
	4er7	AW3	1.79	
	4eki	0QK	1.98	
	3qox	SAH	3.68	
	4er3	0QK	3.75	
	3sr4	TT8	3.91	
	3qow	SAM	3.97	
	4er6	AW2	4.03	
	5mw3	5JT	4.19	
	4wvl	3US	4.39	
	4er5	0QK	4.98	
	5mw4	5JU	5.02	
EHMT1	2igq	SAH	0.44	2igq
	3mo2	SAH	0.74	
	3mo5	SAH	0.89	
	3sw9	SFG	0.90	
	4i51	SAH	0.93	
	3fpd	SAH	0.95	
	5tuz	SAM	0.98	
	3hna	SAH	1.01	
	5vsd	SAM	1.11	
	3mo0	SAH	1.13	
	5vsf	SAM	1.15	
	2rfi	SAH	1.18	
	5ttg	SAM	2.49	
	3swc	SAH	2.57	
	5v9j	SAM	2.59	
EHMT2	3k5k	SAH	0.69	3k5k
	5t0m	SAM	0.71	
	5vse	SAM	0.72	
	5tuy	SAM	0.75	
	5v9i	SAM	0.75	
	5jhn	SAM	0.86	
	3rjw	SAH	0.89	
	4nvq	SAH	0.90	
	5t0k	SAM	0.92	
	5jj0	SAM	0.97	
	5jin	SAM	0.98	
	5ttf	SAM	1.02	
	5vsc	SAM	1.04	
	2o8j	SAH	1.14	
	5jiy	SAM	1.73	
SETD8	2bqz	SAH	1.27	2bqz
	1zkk	SAH	1.33	
	3f9z	SAH	1.34	
	5teg	SAM	1.50	
	3f9w	SAH	1.73	
	3f9x	SAH	2.59	
	4ij8	SAM	2.63	
	3f9y	SAH	2.66	
PRMT1	1or8	SAH	0.65	1or8
	1orh	SAH	0.94	
	1ori	SAH	0.97	
	3q7e	SAH	3.99	
PRMT3	1f3l	SAH	0.47	1f3l
	2fyt	SAH	3.95	
PRMT5	5emk	SFG	0.76	5emk
	5emm	SFG	0.91	
	4gqb	0XU	1.15	
	6ckc	F5J	1.29	
	4x63	SAH	1.54	
	5emj	SFG	1.56	
	3ua3	SAH	1.61	

(Continued)

TABLE 5 | Continued

Target	PDB ID	Ligand	Average RMSD	Selected PDB ID
	5c9z	SFG	2.03	
	5eml	SAM	2.10	
	4x60	SFG	2.30	
	5fa5	MTA	2.57	
	4g56	SAH	2.71	
	4x61	SAM	3.13	
PRMT6	4c04	SFG	0.64	4c04
	4y30	SAH	0.70	
	4qqk	37H	0.89	
	5wcf	SAH	0.89	
	4c03	SFG	0.92	
	4hc4	SAH	0.94	
	4c05	SAH	0.99	
	5fqo	SAH	1.24	
	4qpp	SAH	1.29	
	5hzm	SAH	1.29	
	5egs	SAH	1.37	
	4y2h	SAH	1.61	
	5fqn	SAH	3.03	
	5e8r	SAH	3.78	
	4lwp	SAH	3.91	
SETD7	3vv0	KH3	0.72	3vv0
	3vuz	K15	0.91	
	4e47	SAM	1.02	
	4j83	SAM	1.02	
	3m57	SAH	1.05	
	3m5a	SAH	1.07	
	1n6a	SAM	1.14	
	3m55	SAH	1.58	
	3m58	SAH	1.89	
	3cbm	SAH	2.00	
	3cbo	SAH	2.07	
	3m59	SAH	2.36	
	4j7i	SAH	2.38	
	4j7f	SAH	2.41	
	5eg2	SAH	2.46	
	3m53	SAH	2.54	
	4jlg	SAM	2.56	
	3m56	SAH	2.74	
	4j8o	SAH	2.77	
	3cbp	SFG	2.90	
	1o9s	SAH	3.07	
	3os5	SAH	3.07	
	2f69	SAH	3.08	
	3m54	SAH	3.27	
	1xqh	SAH	3.48	
	4jds	SAM	3.73	
	5ayf	SAM	4.31	
	1n6c	SAM	5.91	
	1mt6	SAH	7.82	
SUV39H2	2r3a	SAM	0.69	2r3a

DNN Model Construction and Evaluation

The DNN model was built by the MultitaskClassifier module of Deepchem (version 2.1.0), and the data set was randomly divided by the RandomSplitter of Deepchem. The Evaluator module of Deepchem was used to evaluate the performance of DNN models.

The evaluation indexed used to evaluate the performance of these modules were area under the precision-recall curve (PRC-AUC) and area under the Compute Receiver operating characteristic curve (ROC-AUC), which are widely used in evaluation the enrichment of scoring model. The closer that the AUC is to 1, the more likely it is that the model is an ideal classification model. Especially, when the ROC-AUC is close to 0.5, the model is close to a random classifier. When the PRC curve reports the evolutions of Recall and Precision, the ROC curve shows the changes of true positive rate (TPR) and false positive rate (FPR):

$$Recall = \frac{N_{TP}}{N_{TP} + N_{FN}} \tag{1}$$

$$Precision = \frac{N_{TP}}{N_{TP} + N_{FP}} \tag{2}$$

$$\mathrm{TPR} = \frac{N_{TP}}{N_{TP} + N_{FN}} \tag{3}$$

$$\mathrm{FPR} = \frac{N_{FP}}{N_{FP} + N_{TN}} \tag{4}$$

where $\mathrm{N_{TP,}}$ $\mathrm{N_{TN}}$, $\mathrm{N_{FP}}$, and $\mathrm{N_{FN}}$ refer to the numbers of true positives, true negatives, false positives, and false negatives, respectively.

The performance of Autodock vina in the test set was also compared with that of DNN model. Before applying Autodock vina (Version 1.1.2), the protein receptor structures and ligand structures were prepared using python scripts named "prepare_receptor4.py" and "prepare_ligand4.py" in AutoDockTools, respectively, which included standard steps such as adding hydrogens and electrons. The grid was also centered on the centroid of the ligand. The grid size was set to 25 Å × 25 Å × 25 Å, and the energy range was set to 4, and all other parameters were used the default settings. The conformation with the best affinity score of each ligand was selected for further study. All figures in this article were produced by Matplotlib and Seaborn python package.

CONCLUSIONS

In this study, we have developed a target-specific classifier for methyltransferases based on protein ligand interaction fingerprint and deep neural network. Binding poses of active and inactive compounds for 12 methyltransferase were generated via molecular docking. TIFP interaction fingerprints were employed as input features of full-connected deep neural network models. The performance of the DNN model on the test set showed that our classifier can classify active and inactive compounds more accurately. In comparison with

Glide Autodock vina and DNN-Glide hybrid model, the DNN model improved both classification performance and compound ranking capability.

Currently, the scoring model can be used in virtual screening and experimentally verified. As a target-specific classifier, this neural network model may be applied to other targets through transfer learning, or if the data used for training is appropriate, the classifier of other targets or even the general classifier can be constructed through the same workflow.

DATA AVAILABILITY

The raw data supporting the conclusions of this manuscript will be made available by the authors, without undue reservation, to any qualified researcher.

AUTHOR CONTRIBUTIONS

XLu, WL, and MZ designed research. FL, XWa, JX, XT, XLi, YW, and JZ performed research. XWu, XLiu, and ZL analyzed data. FL, XWa, and MZ wrote the paper.

FUNDING

We gratefully acknowledge financial support from the National Natural Science Foundation of China (81773634 and 81430084), National Science & Technology Major Project Key New Drug Creation and Manufacturing Program, China (Number: 2018ZX09711002), and Personalized Medicines—Molecular Signature-based Drug Discovery and Development, Strategic Priority Research Pro-gram of the Chinese Academy of Sciences (XDA12050201).

REFERENCES

Berishvili, V. P., Voronkov, A. E., Radchenko, E. V., and Palyulin, V. A. (2018). Machine learning classification models to improve the docking-based screening: a case of PI3K-tankyrase inhibitors. *Mol. Inform.* 37:e1800030. doi: 10.1002/minf.201800030

Biswas, S., and Rao, C. M. (2018). Epigenetic tools (The Writers, The Readers and The Erasers) and their implications in cancer therapy. *Eur. J. Pharmacol.* 837, 8–24. doi: 10.1016/j.ejphar.2018.08.021

Bonifácio, M. J., Palma, P. N., Almeida, L., and Soares-da-Silva, P. (2007). Catechol-O-methyltransferase and its inhibitors in Parkinson's disease. *CNS Drug Rev.* 13, 352–379. doi: 10.1111/j.1527-3458.2007.00020.x

Boriack-Sjodin, P. A., and Swinger, K. K. (2016). Protein methyltransferases: a distinct, diverse, and dynamic family of enzymes. *Biochemistry* 55, 1557–1569. doi: 10.1021/acs.biochem.5b01129

Bouras, G., Deftereos, S., Tousoulis, D., Giannopoulos, G., Chatzis, G., Tsounis, D., et al. (2013). Asymmetric dimethylarginine (ADMA): a promising biomarker for cardiovascular disease? *Curr. Top. Med. Chem.* 13, 180–200. doi: 10.2174/1568026611313020007

Bradley, A. P. (1997). The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recogn.* 30, 1145–1159. doi: 10.1016/S0031-3203(96)00142-2

Da Costa, E. M., McInnes, G., Beaudry, A., and Raynal, N. J. (2017). DNA methylation-targeted drugs. *Cancer J.* 23, 270–276. doi: 10.1097/PPO.0000000000000278

Deng, L., Zhang, L., Yao, Y., Wang, C., Redell, M. S., Dong, S., et al. (2013). Synthesis, activity and metabolic stability of non-ribose containing inhibitors of histone methyltransferase DOT1L. *Medchemcomm* 4, 822–826. doi: 10.1039/c3md00021d

Desaphy, J., Raimbaud, E., Ducrot, P., and Rognan, D. (2013). Encoding protein-ligand interaction patterns in fingerprints and graphs. *J. Chem. Inf. Model.* 53, 623–637. doi: 10.1021/ci300566n

Hu, H., Qian, K., Ho, M. C., and Zheng, Y. G. (2016). Small molecule inhibitors of protein arginine methyltransferases. *Expert Opin. Investig. Drugs* 25, 335–358. doi: 10.1517/13543784.2016.1144747

Hu, J., Chen, S., Kong, X., Zhu, K., Cheng, S., Zheng, M., et al. (2015). Interaction between DNA/histone methyltransferases and their inhibitors. *Curr. Med. Chem.* 22, 360–372. doi: 10.2174/0929867321666141106114538

Jiménez, J., Škalič, M., Martínez-Rosell, G., and De Fabritiis, G. (2018). K_{DEEP}: protein–ligand absolute binding affinity prediction via 3D-convolutional neural networks. *J. Chem. Inf. Model.* 58, 287–296. doi: 10.1021/acs.jcim.7b00650

Kaniskan, H. U., Konze, K. D., and Jin, J. (2015). Selective inhibitors of protein methyltransferases. *J. Med. Chem.* 58, 1596–1629. doi: 10.1021/jm501234a

Kireev, D. (2016). Structure-based virtual screening of commercially available compound libraries. *Methods Mol. Biol.* 1439, 65–76. doi: 10.1007/978-1-4939-3673-1_4

Kuntz, K. W., Campbell, J. E., Keilhack, H., Pollock, R. M., Knutson, S. K., Porter-Scott, M., et al. (2016). The importance of being me: magic methyls, methyltransferase inhibitors, and the discovery of tazemetostat. *J. Med. Chem.* 59, 1556–1564. doi: 10.1021/acs.jmedchem.5b01501

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature* 521, 436–444. doi: 10.1038/nature14539

Martin, J. L., and McMillan, F. M. (2002). SAM (dependent) I AM: the S-adenosylmethionine-dependent methyltransferase fold. *Curr. Opin. Struct. Biol.* 12, 783–793. doi: 10.1016/S0959-440X(02)00391-3

McCabe, M. T., Ott, H. M., Ganji, G., Korenchuk, S., Thompson, C., Van Aller, G. S., et al. (2012). EZH2 inhibition as a therapeutic strategy for lymphoma with EZH2-activating mutations. *Nature* 492, 108–112. doi: 10.1038/nature11606

Meaney, M. J., and Ferguson-Smith, A. C. (2010). Epigenetic regulation of the neural transcriptome: the meaning of the marks. *Nat. Neurosci.* 13:1313. doi: 10.1038/nn1110-1313

Möbitz, H., Machauer, R., Holzer, P., Vaupel, A., Stauffer, F., Ragot, C., et al. (2017). Discovery of potent, selective, and structurally novel Dot1L inhibitors by a fragment linking approach. *ACS Med. Chem. Lett.* 8, 338–343. doi: 10.1021/acsmedchemlett.6b00519

Morris, G. M., and Lim-Wilby, M. (2008). Molecular docking. *Methods Mol. Biol.* 443, 365–382. doi: 10.1007/978-1-59745-177-2_19

Mysinger, M. M., Carchia, M., Irwin, J. J., and Shoichet, B. K. (2012). Directory of useful decoys, enhanced (DUD-E): better ligands and decoys for better benchmarking. *J. Med. Chem.* 55, 6582–6594. doi: 10.1021/jm300687e

Rilova, E., Erdmann, A., Gros, C., Masson, V., Aussagues, Y., Poughon-Cassabois, V., et al. (2014). Design, synthesis and biological evaluation of 4-amino-N- (4-aminophenyl)benzamide analogues of quinoline-based SGI-1027 as inhibitors of DNA methylation. *ChemMedChem* 9, 590–601. doi: 10.1002/cmdc.201300420

Saito, T., and Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLoS ONE* 10:e0118432. doi: 10.1371/journal.pone.0118432

Schapira, M. (2016). Chemical inhibition of protein methyltransferases. *Cell Chem. Biol.* 23, 1067–1076. doi: 10.1016/j.chembiol.2016.07.014

Schrodinger, LLC. (2015). *The PyMOL Molecular Graphics System, Version 1.8.*

Shen, Q., Xiong, B., Zheng, M., Luo, X., Luo, C., Liu, X., et al. (2011). Knowledge-based scoring functions in drug design: 2. Can the knowledge base be enriched? *J. Chem. Inf. Model.* 51, 386–397. doi: 10.1021/ci100343j

Song, Y., Li, L., Chen, Y., Liu, J., Xiao, S., Lian, F., et al. (2018). Discovery of potent DOT1L inhibitors by AlphaLISA based high throughput screening assay. *Bioorg. Med. Chem.* 26, 1751–1758. doi: 10.1016/j.bmc.2018.02.020

Stein, E. M., Garcia-Manero, G., Rizzieri, D. A., Tibes, R., Berdeja, J. G., Savona, M. R., et al. (2018). The DOT1L inhibitor pinometostat reduces H3K79 methylation and has modest clinical activity in adult acute leukemia. *Blood* 131, 2661–2669. doi: 10.1182/blood-2017-12-818948

Sun, Q., Liu, L., Roth, M., Tian, J., He, Q., Zhong, B., et al. (2015). PRMT1 upregulated by epithelial proinflammatory cytokines participates in COX2 expression in fibroblasts and chronic antigen-induced pulmonary inflammation. *J. Immunol.* 195, 298–306. doi: 10.4049/jimmunol.1402465

Vaswani, R. G., Gehling, V. S., Dakin, L. A., Cook, A. S., Nasveschuk, C. G., Duplessis, M., et al. (2016). Identification of (R)-N-((4-Methoxy-6-methyl-2-oxo-1,2-dihydropyridin-3-yl)methyl)-2-methyl-1-(1-(1 -(2,2,2-trifluoroethyl) piperidin-4-yl)ethyl)-1H-indole-3-carboxamide (CPI-1205), a potent and selective inhibitor of histone methyltransferase EZH2, suitable for phase I clinical trials for B-cell lymphomas. *J. Med. Chem.* 59, 9928–9941. doi: 10.1021/acs.jmedchem.6b01315

Wang, Y., Li, L., Zhang, B., Xing, J., Chen, S., Wan, W., et al. (2017). Discovery of novel disruptor of silencing telomeric 1-like (DOT1L) inhibitors using a target-specific scoring function for the (S)-adenosyl-l-methionine (SAM)-dependent methyltransferase family. *J. Med. Chem.* 60, 2026–2036. doi: 10.1021/acs.jmedchem.6b01785

Wu, P., Nielsen, T. E., and Clausen, M. H. (2015). FDA-approved small-molecule kinase inhibitors. *Trends Pharmacol. Sci.* 36, 422–439. doi: 10.1016/j.tips.2015.04.005

Xing, J., Lu, W., Liu, R., Wang, Y., Xie, Y., Zhang, H., et al. (2017). Machine-learning-assisted approach for discovering novel inhibitors targeting bromodomain-containing protein 4. *J. Chem. Inf. Model.* 57, 1677–1690. doi: 10.1021/acs.jcim.7b00098

Zhang, J., and Zheng, Y. G. (2016). SAM/SAH analogs as versatile tools for SAM-dependent methyltransferases. *ACS Chem. Biol.* 11, 583–597. doi: 10.1021/acschembio.5b00812

Conflict of Interest Statement: 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.

frontiers in Chemistry

ORIGINAL RESEARCH
published: 27 May 2019
doi: 10.3389/fchem.2019.00375

Neural Networks Are Promising Tools for the Prediction of the Viscosity of Unsaturated Polyester Resins

Julien Molina[1,2]*, *Aurélie Laroche*[1,2], *Jean-Victor Richard*[1], *Anne-Sophie Schuller*[1] *and Christian Rolando*[2]

[1] Mäder Research, Mulhouse, France, [2] Faculté des Sciences et Technologies, Université de Lille, USR 3290 MSAP, Miniaturisation pour l'Analyse, la Synthèse et la Protéomique, Villeneuve d'Ascq, France

Unsaturated polyester resins are widely used for the preparation of composite materials and fulfill the majority of practical requirements for industrial and domestic applications at low cost. These resins consist of a highly viscous polyester oligomer and a reactive diluent, which allows its process ability and its crosslinking. The viscosity of the initial polyester and the reactive diluent mixture is critical for practical applications. So far, these viscosities were determined by trial and error which implies a time-consuming succession of manipulations, to achieve the targeted viscosities. In this work, we developed a strategy for predicting the viscosities of unsaturated polyesters formulation based on neural networks. In a first step 15 unsaturated polyesters have been synthesized through high-temperature polycondensation using usual monomers. Experimental Hansen solubility parameters (HSP) were determined from solubility experiment with HSPiP software and glass transition temperatures (T_g) were measured by Differential Scanning Calorimetry (DSC). Quantitative Structure—Property Relationship (QSPR) coupled to multiple linear regressions have been used to get a prediction of Hansen solubility parameters δ_d, δ_p, and δ_h from structural composition. A second QSPR regression has been done on glass transition temperature (prediction vs. experimental coefficient of determination $R^2 = 0.93$) of these unsaturated polyesters. These unsaturated polyesters were next diluted in several solvents with different natures (ethers, esters, alcohol, aromatics for example) at different concentrations. Viscosities at room temperature of these polyesters in solution were finally measured in order to create a database of 220 entries with 7 descriptors (polyester molecular weight, T_g, dispersity index *Đ*, polyester-solvent HSP RED, molar volume of the solvent, δ_h of the solvent, concentration of polyester in solvent). The QSPR method for predicting the viscosity from these 6 descriptors proved to be ineffective ($R^2 = 0.56$) as viscosities exhibit non-linear phenomena. A Neural Network with an optimized number of 12 hidden neurons has been trained with 179 entries to predict the viscosity. A correlation between experimental and predicted viscosities based on 41 testing instances gave a correlation coefficient R^2 of 0.88 and a predicted vs. measured slope of 0.98. Thanks to Neural Networks, new developments with eco-friendly reactive diluents can be accelerated.

Keywords: unsaturated polyester, viscosity, neural network, QSPR, hansen solubility parameters, prediction

OPEN ACCESS

Edited by:
John C. Cancilla,
Scintillon Institute, United States

Reviewed by:
Albert Poater,
University of Girona, Spain
Jose A. Gamez,
Covestro Deutschland AG, Germany

****Correspondence:***
Julien Molina
j.molina@mader-group.com

Specialty section:
This article was submitted to Theoretical and Computational Chemistry,
a section of the journal Frontiers in Chemistry

Received: *15 February 2019*
Accepted: *07 May 2019*
Published: *27 May 2019*

Citation:
Molina J, Laroche A, Richard J-V, Schuller A-S and Rolando C (2019) Neural Networks Are Promising Tools for the Prediction of the Viscosity of Unsaturated Polyester Resins. Front. Chem. 7:375. doi: 10.3389/fchem.2019.00375

INTRODUCTION

Today composite materials find many applications in the fields of transport, construction as well as in sports and leisure (Biron, 2013). The unsaturated polyester resins used for the preparation of these composite materials have several advantages, mainly a favorable price ratio with respect to the mechanical and thermal properties (Mishra et al., 2003), good durability and a relatively good resistance to corrosion (Dagher et al., 2004), a low maintenance cost as well as good electrical, phonic and thermal insulation properties. It also lightens the structures compared to conventional metallic materials allowing to obtain better energy performances (Song et al., 2009). The investment cost related to machining composite materials by hand lay-up is also low (Biron, 2013).

The unsaturated polyesters are synthesized by high temperature polycondensation of diols with saturated and unsaturated diacids. The most used unsaturated monomers are maleic anhydride or fumaric acid. The water produced by the esterification reaction is eliminated by condensation in a Dean-Stark during the reaction. The number average molecular weight of the obtained polyesters are $\sim$1,000 g.mol^{-1} (Fink, 2013). Depending on the monomers used in the polycondensation, the properties of polyester resins differ. For applications where the resin must be resistant to hydrolysis, monomers such as neopentyl glycol and isophthalic acid are particularly suitable. The use of diethylene glycol or dipropylene glycol makes possible to obtain flexible resins (Zaske and Goodman, 1998; Fink, 2013). Thus, there is a multitude of possible chemical structures depending on the intended application.

In order to be manipulated at room temperature and to be crosslinked, the polyesters are diluted in polymerizable solvents. The most commonly reactive diluent is styrene because it effectively reduces the viscosity of the unsaturated polyester in solution and efficiently copolymerizes with the fumarate units (Lewis and Mayo, 1948; Cousinet et al., 2015). However, styrene has been classified by the US Department of Health and Human Services as "reasonably anticipated to be a human carcinogen." It is a very volatile monomer that has also been classified as a hazardous air pollutant by the US Environmental Protection Agency (Cousinet et al., 2015). In Europe, styrene has been classified as "reproductive toxicity category 2" by the European Chemicals Agency (ECHA). Methacrylate monomers are commonly used to replace styrene (Fink, 2013). However, monomers such as methyl, ethyl or butyl methacrylates have strong odors. This is a disadvantage for open mold applications. In addition, their reactivity ratio with fumarate units does not allow good crosslinking (Bengough et al., 1967). Many publications deal with the search for alternative reactive diluents, sometimes bio-sourced, in order to be able to eliminate styrene and to provide resins with less volatile and less toxic organic compounds (Sadler et al., 2012; Cousinet et al., 2014, 2015; Li et al., 2014; Dai et al., 2017; Panic et al., 2017; Yadav et al., 2018).

To develop a new resin, it is now necessary to multiply time-consuming manipulations. Firstly a polyester with a defined structure is synthesized, then diluted in a reactive solvent and finally crosslinked. The properties of the resin such as its viscosity at room temperature and its mechanical properties need to be measured for assessing its performance. Performing all of these steps take several days for a single try. The multitude of possible chemical structures as well as the diversification of available reactive diluents considerably extends the time required for the development of a new resin. The viscosity of polyester resins at room temperature is an important parameter to be respected in a specification. Indeed, the resin must be in a certain range of viscosity depending on its mode of application (Fink, 2013). Developing property prediction tools that use only theoretical values without manipulation is therefore a strategic issue, particularly in the industrial sector.

Neural networks are machine learning tools for connecting non-linear data with one or more target properties (Gasteiger and Zupan, 1993; Svozil et al., 1997). This type of algorithm has been used effectively in many scientific fields, especially in environmental or chemical applications (Behler, 2011; Torrecilla et al., 2013; Wei et al., 2016). Several studies have already been published on the prediction of polymer properties using neural networks, such as the glass transition temperature (Joyce et al., 1995; Mattioni and Jurs, 2002; Chen et al., 2008; Liu and Cao, 2009), intrinsic viscosity (Gharagheizi, 2007a) or lower critical solution temperature (Gharagheizi F., 2007b).

In this work, a neural network was set up in order to predict the viscosity of unsaturated polyester resins from simple descriptors. Once a polyester is synthesized, its number average molecular weight and its glass transition temperature are measured. The experimental Hansen solubility parameters (HSP) (Hansen, 2002) of the polyester are then obtained by solubilization of the polymer in 40 solvents followed by processing results on the HSPiP software (Abbott, 2013). Then, the polyester is solubilized by varying its concentration in solvents of different natures among those previously used. A database of 220 entries of polymer-solvent combination was set up including for the polyesters, their number average molecular weight, their glass transition temperatures and their Hansen parameters, for the solvents their molar volumes, their δ_h and the concentration of the polyester in solution. The resulting viscosity of the polyester in solution was measured with a rheometer for each entry. The neural network was subsequently optimized and trained with this database.

To be able to predict unsaturated polyester viscosity exclusively based on theoretical values without manipulation, the glass transition temperature as well as Hansen parameters of unsaturated polyesters have been correlated according to the theoretical chemical structure of the polyesters. Prediction methods have already been described in the literature for the glass transition temperature (Katritzky et al., 1996; Bicerano, 2002; Camacho-Zuñiga and Ruiz-Treviño, 2003; Krevelen and Nijenhuis, 2009) as well as the Hansen solubility parameters of polymers (Stefanis and Panayiotou, 2008; Krevelen and Nijenhuis, 2009). However, these methods generally relate to high average molecular weight polymers and are not necessarily adapted to unsaturated polyesters. In this work, a Quantitative Structure—Property Relationship (QSPR) method was applied to propose a simple method for determining the glass transition temperature and Hansen solubility parameters for unsaturated polyesters. The experimental values used in the neural network can be replaced in the future by the predicted values obtained by QSPR.

Data capitalization and processing has become a strategic topic for predicting phenomena (Dong et al., 1996; Zhang et al., 1998; Marengo et al., 2004). Being able to predict the viscosity of polyester resins to see if they fulfill specifications and minimize the number of tests is undoubtedly of high added value for thermoset resins industrial companies. Today, the establishment of a machine learning system has become more accessible, so its use in chemical companies will certainly grow in the coming years.

MATERIALS AND METHODS

Reagents

Propylene glycol (PG), dipropylene glycol (DPG), neopentyl glycol (NPG), cyclohexanedimethanol also known as 1,4-bis(hydroxymethyl)cyclohexane (CHDM), 2-ethylhexanol (EH), benzyl alcohol (AB), maleic anhydride (AM), itaconic acid (IT), fumaric acid (AF), phthalic anhydride (PA), adipic acid (AA) were provided by the Mäder group. They were used as received without further purification.

All solvents used for the determination of Hansen parameters are laboratory grade and were used as received without further purification.

Synthesis of the Prepolymer

The prepolymer was synthesized by the melt polycondensation between diols and diacids. The components were mixed in a 1 L four-necked round-bottom flask connected with a stirrer, a temperature probe connected to the heater, a Dean–Stark, and a N_2 gas inlet. No catalyst was used in this work. The reaction was carried out at a temperature of 200°C under a nitrogen atmosphere. The reaction was carried out until the acid value reached 30. The acid value (AV) is defined as the number of milligrams of KOH needed to neutralize 1 g of resin and was measured according to ASTM D465-01. Around 1 g of resins was titrated with a KOH solution in isopropanol (0.1 M).

Prepolymer Characterization

The size exclusion chromatography (SEC) used was a Shimadzu Prominence fitted with a Refractive Index (RI) detector (RID-20A) and an UV detector (SPD-20A). The columns (KF-802 and KF-803L from Shodex) were eluted with tetrahydrofuran (THF) at a flow rate of 1 mL/min at 30°C. The samples were previously prepared by dissolving 10 mg of sample in 1 mL THF. The solution was then filtered through a PTFE filter with a pore diameter of 0.45 µm. A volume of 20 µL was injected into the size exclusion chromatography to carry out the analysis. The SEC has been calibrated with poly(styrene) standards. The number average molecular weights were determined from the UV detector absorbance.

The glass transition temperature (T_g) of the prepolymers was measured by differential scanning calorimetry, DSC, using a Q20 TA Instruments in hermetic aluminum capsules with a scan rate of 10°C/min from −80°C to 150°C under N_2 (50 mL/min). The second heating run was used to determine the T_g with the TA Instruments software.

Hansen Solubility Parameter Experimental Determination

The solubility of the polymers was assessed by dissolving 100 mg in 1 mL of solvent at room temperature. Solubility was assessed after 24 h of agitation using a Vortex-Genie 2 from Scientific Industries. The 40 solvents tested were acetic acid, acetone, acetonitrile, aniline, benzonitrile, benzyl alcohol, γ-butyrolactone, m-cresol, cyclohexane, cyclohexanone, o-dichlorobenzene, diethylene glycol, dimethyl formamide, 1,4-dioxane, ethanol, ethyl acetate, ethylene glycol, ethylene glycol monomethyl ether, formamide, formic acid, furan, hexane, isobutyl alcohol, methanol, methyl ethyl ketone, *N*-methyl formamide, methyl methacrylate, *N*-methyl-2-pyrrolidone, methylene dichloride, morpholine, nitrobenzene, 1-pentanol, 1-propanol, propionitrile, propylene carbonate, propylene glycol monomethyl ether, styrene, tetrahydrofuran, toluene, water (Delgove et al., 2017). The Hansen solubility parameters δ_d, δ_p, δ_h and the solubility sphere radius R_0 of the unsaturated polyesters were obtained using the HSPiP software. A sphere centered on the HSP of the polyester and radius R_0 constitutes the sphere of solubility of the polyester. Solvents whose HSP are inside the sphere allow the solubilization of the polyester. The polyester is insoluble in solvents having HSP outside the sphere.

Unsaturated Polyester—Solvent Compatibility Determination

Once the HSP of the polyesters were obtained, the compatibility of each polyester in solvents of different natures was quantified. Firstly, the distance R_a in a three-dimensional space between the Hansen parameters of the polyester (P) and the Hansen parameters of the solvent (S) was calculated using the Equation (1) (Krevelen and Nijenhuis, 2009).

$$R_a^2 = 4.0 \times (\delta_{dP} - \delta_{dS})^2 + \left(\delta_{pP} - \delta_{pS}\right)^2 + (\delta_{hP} - \delta_{hS})^2 \quad (1)$$

The Relative Energy Difference (*RED*) was then calculated by performing the ratio of R_a to R_0 (Equation 2) corresponding to the solubility radius of the unsaturated polyester (Krevelen and Nijenhuis, 2009).

$$RED = \frac{R_a}{R_0} \quad (2)$$

Thus, the *RED* gives a simple numerical value for characterizing the compatibility of a polymer in a solvent. According to Hansen's theory, two compounds are very compatible if their *RED* approaches 0 because their Hansen solubility parameters are very close. If their RED is equal to 1, it means that the polyester is at the limit of solubility in the solvent and therefore almost incompatible. A *RED* >1 means that the polyester is not soluble in the solvent tested (Krevelen and Nijenhuis, 2009).

Creation of the Polyester Resin Database

In order to develop the database, the unsaturated polyesters synthesized were diluted in various solvents among those used in Part 2.3 and at different concentrations. Apparent viscosities were measured at 23°C as a function of shear rate over the range 1–100 s^{-1} using the viscometry function of a controlled stress and strain rheometer (Anton Paar MCR 301). A parallel plate geometry has been used with a diameter plate of 25 mm (PP25) and a gap of 1 mm.

The database contains 220 entries including for each of them the number average molecular weight of the polyester M_n (obtained by SEC), its index polydispersity *Đ,* and its glass transition temperature T_g (obtained by DSC), the *RED* polymer-solvent compatibility (obtained via HSPiP), the molar volume of the solvent M_{vol}(obtained via HSPiP), the concentration of the polyester in the solution and the measured viscosity at 23°C of the polyester in solution. This database is provided in **Table S1**.

QSPR Modeling With Multiple Linear Regression (MLR)

Quantitative Structure—Property Relationship (QSPR) modelizations were carried out by multiple linear regression. Different descriptors x_i are correlated with one or more responses. The linear relation linking the descriptors to this response is given in Equation 3.

$$y = a_0 + a_1x_1 + a_2x_2 + \ldots + a_ix_i + e \tag{3}$$

The values a_i are the regression coefficients. The purpose of multiple linear regression is to determine the value of these coefficients by the least squares method. These modelizations were realized with the software Cosmoquick version 1.7 (COSMOlogic, Leverkusen, Germany) (Loschen and Klamt, 2012).

Artificial Neural Network

Neural networks are a type of machine learning tool which link several input data with output data by non-linear relations (Gasteiger and Zupan, 1993; Svozil et al., 1997). They present a real advantage over conventional linear mathematical approaches (Díaz-Rodríguez et al., 2014; Cancilla et al., 2016). The use of neural networks allows to find physico-chemical models already described in the literature or even to discover original models (Behler, 2011; Díaz-Rodríguez et al., 2015).

A neural network is divided into several layers, each composed of neurons and interconnected by synapses (Díaz-Rodríguez et al., 2014). The first layer, called the input layer, introduces into the neural network the values of the different descriptors influencing the target property at the output of the neural network. In this study, several physicochemical data describing both the polyesters as well as the solvents properties were used in this input layer.

The second part of the neural network is the hidden learning layer. It contains neurons that allow non-linear calculations to obtain the relationship between input and output data (Gasteiger and Zupan, 1993; Cancilla et al., 2014a,b). Each learning neuron performs a linear combination of input data multiplied by the weight of the synapses associated with that data. An additional constant, called bias, is added to this linear combination in order to add an extra degree of freedom to the neural network to better match input and output data. A function that can be linear or not transforms the value obtained in order to obtain the output signal of the neuron. The most common non-linear functions are the hyberbolic tangent or the sigmoid. A multitude of other activations functions exist and research are still on-going on the development of new functions (Xu et al., 2015). This output value is then introduced as an input value for the next layer of neurons.

The number of neurons in the hidden layer must be optimized in order to have the best learning and to get the best prediction accuracy. A low number of learning neurons will tend to limit the learning ability of complex problems by the neural network whereas an excessive number of neurons can lead to an over-fit of prediction and an increase in the gap compared to the experimental target values. Although different rules emerge to fix the number of hidden neurons based on the number of input and output data, it is also possible to test the evolution of the prediction error with respect to the experimental one by changing the number of learning neurons (Sheela and Deepa, 2013). In the initial state, values of the synapses weights are fixed randomly. The training protocol is based on an algorithm seeking to reduce the difference between the experimental target values compared to the values predicted by successive iterations that modify the weight of the synapses. There are different types of training algorithms, each of which is more suitable for a kind of applications (Torrecilla et al., 2008). A neural network can continue the iterations until the predicted values fit perfectly with the training data. However, this can cause over-fit due to the consideration of non-general trends such as experimental errors or noise. Verification of the reliability of the neural network can be performed with a set of data that have not been used for the modification of synaptic weights during training (Cancilla et al., 2014a). When the error between experimental values and predicted values begins to increase, it means that the training phase has undergone too many iterations.

Neural designer desktop version 2.9.5 (Artelnics, Salamanca, Spain) has been employed for the neural network design and its optimization.

RESULTS AND DISCUSSION

Unsaturated Polyesters Synthesis

Fifteen unsaturated polyesters have been synthesized from the monomers conventionally used in industry. The stoichiometric ratio between the reagents called *r* corresponds to the initial molar amount of carboxylic acid groups on the initial molar amount of alcohol groups provided by the diacids and glycols of the polycondensation reaction. These different structures are listed in **Table 1**. They were characterized initially by DSC and SEC in order to obtain the glass transition temperature T_g, the number average molecular weight M_n and the dispersity index *Đ*.

During the reaction, the maleate units are isomerized into fumarate units. However, the isomerization rate depends mainly on the monomer composition of the resin (Curtis et al., 1964).

TABLE 1 | Structures of the unsaturated polyesters synthesized.

Polyester	Monomer 1 (mol%)	Monomer 2 (mol%)	Monomer 3 (mol%)	Monomer 4 (mol%)	Monomer 5 (mol%)	*r*	T_g(°C)	M_n (g/mol)	*Đ*
1	PG 80%	DPG 20%	AM 67%	AP 27%	AA 6%	0.97[a]	3.9	1,880	3.90
2	NPG 70%	PG 30%	–	AF 100%	/	0.93	9.1	2,678	2.23
3	NPG 70%	PG 30%	–	AM 60%	AP 40%	0.90	16.3	1,560	2.13
4	NPG 70%	PG 30%	–	AM 50%	AP 50%	0.90	24.4	1,652	2.51
5	NPG 70%	PG 30%	–	AM 70%	AP 30%	0.90	11.7	1,640	1.90
6	NPG 70%	PG 30%	–	AM 60%	AP 40%	0.91	16.0	1,780	1.87
7	NPG 50%	PG 50%	–	AM 60%	AP 40%	0.96	20.1	2,530	2.90
8	NPG 70%	PG 30%	–	IT 60%	AP 40%	0.98	12.1	1,205	2.64
9	PG 100%	–	–	AM 60%	AP 40%	0.90	22.0	1,610	3.69
10	PG 100%	–	–	AM 60%	AP 40%	0.91	23.6	1,760	1.5
11	NPG 70%	PG 30%	EH 5%	AM 60%	AP 40%	0.94	2.4	1,220	2.09
12	NPG 70%	PG 30%	–	AM 60%	AP 40%	0.96	21.2	1,960	2.59
13	CHDM 70%	PG 30%	–	AF 60%	AP 40%	0.92	22.6	2,420	1.84
14	DPG 100%	–	–	AF 60%	AP 40%	0.92	−6.5	1,409	2.47
15	DPG 50%	NPG 50%	–	AF 60%	AP 40%	0.91	1	1,330	2.20
16	NPG 70%	PG 30%	–	AM 60%	AP 40%	0.75	−2.5	950	1.84
17	NPG 70%	PG 30%	–	AM 60%	AP 40%	0.93	20.9	2,090	2.70
18	CHDM 100%	–	–	AF 60%	AP 40%	0.92	29.4	1,995	2.21
19	NPG 70%	PG 30%	AB 5%	AM 60%	AP 40%	0.94	11.2	1,410	2.23
20	NPG 30%	CHDM 70%	/	AF 60%	AP 40%	0.92	23.5	1,760	2.16
21	NPG 70%	PG 30%	/	AF 60%	AA 40%	0.9	−20.7	1,350	2.38

[a] *Final acid number = 50 mgKOH/g (instead of 30 mgKOH/g).*

Diols with secondary alcohols such as propylene glycol promote isomerization in contrast to diols having only primary alcohols. The presence of phthalic anhydride also promotes isomerization. Maleate units (*Z*-double bond) do not have the same properties as fumarate units (*E*-double bond) (Ebewele, 2000; Krevelen and Nijenhuis, 2009). In order to minimize the presence of maleates in the reaction, fumaric acid has been used in syntheses with primary diols or without phthalic anhydride.

The glass transition temperature T_g of the polyesters depends on the structure of the monomers used during the synthesis as well as the final average molecular weight obtained. The introduction of monomers comprising ether bridges such as dipropylene glycol or diethylene glycol allows the flexibilization of the polyester chains and therefore the lowering of the glass transition temperature of the polyesters (Young and Lovell, 1996; Zaske and Goodman, 1998; Ebewele, 2000). In order to be able to compare the impact of these monomers on the glass transition temperature, the acid monomer composition as well as the targeted degree of polymerization was fixed for polyesters described in polyesters **3**, **14,** and **15**. The polyester **4** composed solely of dipropylene glycol has a T_g of −6.5°C whereas the polyester **15** comprising 50% of neopentyl glycol and 50% of dipropylene glycol has a T_g of 1°C. A polyester without ethers monomers such as the one described in polyester **3** has a higher T_g of 16.3°C. The use of aromatic monomers such as orthophthalic anhydride also modulate the glass transition temperature of the unsaturated polyesters (Zaske and Goodman, 1998; Ebewele, 2000). The degree of polymerization as well as the glycol composition of the polyesters described in polyester **4**-**6** are similar while the ratio of maleic anhydride to orthophthalic anhydride has been varied. The increase in the ratio in favor of orthophthalic anhydride within the polyester induces an increase in the glass transition temperature. On the contrary, the introduction of long aliphatic chain within the polyester has a plasticizing action and thus induces a decrease in the glass transition temperature (Young and Lovell, 1996; Zaske and Goodman, 1998; Ebewele, 2000). When the orthophthalic anhydride is replaced by adipic acid, which has an aliphatic chain, the glass transition temperature drastically decreases (polyester **21**: $T_g = -20.7°\text{C}$ vs. polyester **3**: $T_g = 16.3°\text{C}$). In the same way, the incorporation of a mono-functional aliphatic alcohol such as 2-ethylhexanol has a plasticizing action and a decrease in the glass transition temperature is observed (polyester **11**: $T_g = 2.4°\text{C}$ vs. polyester **3**: $T_g = 16.3°\text{C}$).

The use of branched monomers such as neopentyl glycol or propylene glycol induces a steric hindrance and thus restricts the polymer chain rotation (Young and Lovell, 1996; Ebewele, 2000). Neopentyl glycol also has a symmetry with its two CH_3 groups in comparison to propylene glycol which has only one CH_3 group. Despite a larger steric hindrance, this symmetry induces a drop in the glass transition temperature (Mark, 2007). Moreover, neopentyl glycol has an additional CH_2 group relative to propylene glycol which makes the polyester more flexible. The polyester **9** composed solely of propylene glycol for the glycol portion has a glass transition temperature of 22.0°C. When 70 mol% of propylene glycol is replaced by neopentyl glycol (polyester **3**), the glass transition temperature decreases to 16.3°C. The introduction

TABLE 2 | Hansen solubility parameter of the synthesized unsaturated polyesters.

Polyester	δ_d	δ_p	δ_h	δ	R_0
1	16.6	14.2	3.9	22.1	13.1
2	19.0	9.2	8.5	21.0	6.0
3	17.8	13.4	4.4	22.7	12.7
4	18.7	14.6	5.1	24.3	13.6
5	17.8	13.5	4.4	22.7	12.7
6	18.8	12.8	5.8	23.5	12.1
7	18.8	13.7	5.4	23.9	12.9
8	17.7	13.5	4.4	22.7	12.7
9	17.5	13.7	4.5	22.7	12.5
10	18.0	13.2	5.9	23.1	11.6
11	17.5	13.8	4.4	22.6	12.6
12	18.7	14.6	5.1	24.2	13.5
13	19.4	7.0	7.8	22.1	8.6
14	17.3	13.6	4.0	22.4	12.9
15	17.7	13.5	4.4	22.7	12.7
16	17.2	11.7	6.9	21.9	11.5
17	18.1	13.2	5.1	23.0	12.3
18	19.1	6.7	7.4	21.5	6.6
19	17.4	13.8	4.4	22.6	12.6
20	17.9	8.0	8.5	21.4	8.7
21	18.7	13.4	5.1	23.6	12.6
Average	18.1	12.4	5.5	22.7	11.6
Standard deviation	0.7	2.4	1.4	0.9	2.14

of cycloaliphatic monomers such as cyclohexanedimethanol, for example, stiffens the polyester chains (Turner et al., 2001). The replacement of propylene glycol of polyester **3** by cyclohexanedimethanol involves an increase in the glass transition temperature (**20** $T_g = 23.5°\text{C}$ vs. **3** $T_g = 16.3$ °C). The polyester **18** containing only cyclohexanedimethanol has a glass transition temperature of 29.4°C. The influence of the number average molecular weight of the polyester was also studied. The monomer composition of the polyesters **3**, **16**, **17** was kept constant while varying the molecular weight. Obviously, the glass transition temperature increases as the average molecular weight of the polymer increases (Ebewele, 2000; Mark, 2007).

Hansen Solubility Parameter Experimental Determination

In order to predict the solution viscosity of a polyester, it is important to know its compatibility with different types of solvent (Flory, 1942; Hillyer and Leonard, 1973; Young and Lovell, 1996). Indeed, a polyester containing a large number of polar groups adopt a different behavior in an apolar solvent (i.e., xylene) or in a polar solvent (i.e., water or ethanol). The Hansen solubility parameters (Krevelen and Nijenhuis, 2009) were therefore measured in order to be able to compare them with the solubility parameters of the various solvents subsequently tested for the prediction of viscosities. The measured parameters are listed in **Table 2**.

The δ_d of the 21 unsaturated polyesters synthesized, does not seem to be influenced by the variation of the monomers used. The standard deviation is low compared to the average of measured δ_d. Polyesters **13** and **18** have the highest δ_d (19.4 and 19.1 $\text{MPa}^{1/2}$). Both of these polyesters have cyclohexanedimethanol units within their chains. The polyester **13** has 70 mol% of cyclohexanedimethanol relative to total glycols while polyester **18** is composed of 100% cyclohexanedimethanol. These cycloaliphatic units have a high density of carbon relative to other glycols which induces the high value of δ_d. The number average molecular weight of polyesters has an influence on δ_d. The higher the number average molecular weight, the more δ_d increases. This can be explained by the fact that an increase in the number of average units in the polyester gives rise to a lesser importance of the functions allowing the hydrogen bonds (alcohols or terminal acids) with respect to the aliphatic functions.

The different δ_p measured have an average of 12.4 $\text{MPa}^{1/2}$ with a standard deviation of 2.4 $\text{MPa}^{1/2}$. There is therefore a greater variation compared to the δ_d of the different polyesters. Polyesters **1**, **4**, **12** have the highest δ_p values with respective values of 14.2, 14.6, 14.6 $\text{MPa}^{1/2}$. They also have the greatest number of functional groups CH and quaternary C compared to other polyesters. These two types of groups induce asymmetries as well as an increase of the rigidity of the polyesters. These functional groups prevent the packing of the polyester chains by the irregularities they create within the polyester chain (Ebewele, 2000).

Polyesters **2**, **13**, **18,** and **20** have the lowest δ_p. Firstly polyester **2** has a structure composed only of maleate/fumarate units for the acid part. This singularity increases the regularity of the polyester chain with respect to a maleate/aromatic mixture. This regularity brings the polyester chains closer together. It is also composed mainly of neopentyl glycol which does not have asymmetric carbons. The polyesters **13**, **18,** and **20** have a high content of cyclohexanedimethanol at the origin of the low δ_p. The cyclohexanedimethanol do not have asymmetry centers and are therefore more regular than typical propylene glycol units (Turner et al., 2001).

The variation of δ_h is more important. It has indeed a significant standard deviation (1.4) with respect to its average of 5.5 for the 21 unsaturated polyesters. Polyesters **2**, **13**, **18,** and **20** which have structures without asymmetric functions also have the highest values of δ_h. However, these four resins also have the lowest R_0 of all the polyesters. They have the spheres of the smallest solubilities and are therefore soluble in less solvents than other polyesters (Krevelen and Nijenhuis, 2009). A small solubility radius indicates that the polyester prefers to create inter-molecular bonds instead of bonding with the solvent in which it is in solution. In order to be able to create inter-molecular bonds, however, the polyester must be regular and free of asymmetric functions so that the chains are close to one another (Young and Lovell, 1996; Ebewele, 2000; Delgove et al., 2017). This proximity allows the establishment of inter-molecular links. On the contrary, if the polyesters have many asymmetric functions, the polyester chains will not be able to get closer. Solvent molecules can thus more easily establish interactions with the polymer chains. The cyclohexanedimethanol unit does not have asymmetric functions. In polyester **13**, **18,** and **20** chains, it allows the packing of the chains and thus the lowering of the radius of the solubility sphere. Polyesters which possess a large number of asymmetric functions, such as in propylene glycol or dipropylene glycol, have their solubility ranges increased. Indeed, polyester **1**, composed of 80% propylene glycol and 20% dipropylene glycol, has a solubility radius of 13.1, which is above the average.

Unsaturated Polyesters Properties Prediction by QSPR Method

Manipulations to get Hansen solubility parameters of polyesters are repetitive and time-consuming. Each polyester should be diluted in 40 solvents for 24 h and the solubilization results should be interpreted for each solvent. Similarly, measurement of the glass transition temperature requires a DSC and may take more than 1 h for each polymer. It is therefore very useful to develop an easy method to predict these properties in order to save time. To provide a method without the need for extensive analyzes for determination of the glass transition temperature and Hansen parameters of unsaturated polyesters, it was chosen to rely on the initial experimental molar quantities of the monomers introduced into the reactor to calculate the QSPR input descriptors. In order to obtain the final conversion of the synthesized polyesters, the final acid number was recorded for each synthesis. To keep reliable predictions, this method of determination must therefore be limited to unsaturated polyesters with similar monomers and synthetic conditions to the study. Moreover, an additive method already used in literature methods has been chosen (Stefanis and Panayiotou, 2008; Krevelen and Nijenhuis, 2009) and each theoretical structure of polyesters as a function of simple functional groups were decomposed (-CH_2-, -CH_3, -COO-, -CH_2 =CH_2-, -orthophtalic-, etc. ...). In order to obtain the number of theoretical functional groups of a polyester, the Carothers equation on the average degree of polymerization of a step polymerization, nature and the quantity of the monomers introduced into the polycondensation reactor were coupled. In a first step, the stoichiometric ratio between the reagents called r was calculated between the initial molar amount of carboxylic acid groups on the initial molar amount of alcohol groups provided by the diacids and glycols of the polycondensation reaction. The conversion of the reaction called p was calculated by the ratio of the molar amount of carboxylic acids per gram of resin during the reaction to the initial molar amount per gram of resin. This conversion is followed by the acid number of the polycondensation reaction. The final conversion thus corresponds to the remaining amount of carboxylic acids per gram of resin over the initial amount per gram of resin. The average degree of polymerization is obtained thanks to the Carothers Equation (4).

$$DPn_{theo} = \frac{1+r}{1+r-2rp} \tag{4}$$

Once the average degree of polymerization is obtained, the polyester chain was divided into three distinct parts, the two terminal diols from one end to the other of the chain, the repeating units (diols + diacids) and finally a diacid unit binding one of the terminal diols with the first diol repeating unit. To simplify the calculation, the ester functions were integrated in the diacid patterns. The formula to calculate the number of theoretical functional groups is given by Equation (5).

$$\begin{aligned}
FG_{theo} &= (\textstyle\sum_{i=1}^{n} 2.0 \times FG_{endgroup-glycol_i} \times \%_{mol}glycol_i) \\
&+ (\textstyle\sum_{i=1}^{n} \frac{(DPn_{theo}-3.0)}{2} \times FG_{repetition\ unit-glycol_i} \times \%_{mol}glycol_i) \\
&+ \left(\textstyle\sum_{j=1}^{m} \frac{(DPn_{theo}-3.0)}{2} \times FG_{repetition\ unit-diacid_j} \times \%_{mol}diacid_j\right) \\
&+ \left(\textstyle\sum_{j}^{m} FG_{link-diacid_j} \times \%_{mol}diacid_j\right) + 2.0 \\
&\times \left(100.0 - \%_{mol}monoalcool\right) \times FG_{OH}
\end{aligned} \tag{5}$$

The value $\%_{mol}glycol_i$ corresponds to the molar part represented by one of the glycols on all the glycols used in the reaction. The value $\%_{mol}diacid_i$ is the equivalent for the diacid part of the synthesis. As an example for the number of functional groups in the diols, the propylene glycol comprises a –CH_3 group, a -CH_2- group and a -CH- group. The -OH end-of-chain groups must also be added. If the polycondensation reaction comprises monofunctional alcohols, these must be added to the terminal glycols in proportion to their molar ratios with respect to the total molar quantity of the glycols of the reaction. The addition of mono-alcohols also has an impact on the amount of alcohol functional groups at the end of the chain. As regards the diacids, itaconic acid comprises for example two -COO- groups, a -CH=CH_2 group and a -CH_2- group. The list of functional groups according to the different theoretical structures of the synthesized unsaturated polyesters is given in **Table S2**.

Hansen Solubility Parameter Prediction by QSPR Method

As for the determination of the glass transition temperature, a QSPR method was also applied for the prediction of the δ_d, δ_p, and δ_h components of the Hansen solubility parameters. The values of the coefficients of the functional groups obtained by the QSPR method are listed in **Table 3**.

The coefficients obtained for the δ_h prediction of unsaturated polyesters confirm the hypotheses depicted in section Unsaturated polyesters properties prediction by QSPR method. Indeed, each -CH- and -C- group within the polyester chain, respectively, decreases the δ_h of −52.9 and −80.5. These groups decrease the linearity of the polyester chains and inhibit the creation of hydrogen bonds between the chains. On the other hand, the other groups such as -CH_3, -cyclohexane-, -OH, and -O- are the groups which bring the most regularity to the polyester chains and thus increase the creation of polyester bonds.

Unlike the Stephanis-panayiotou or Hoftyzer-Van Krevelen methods, the QSPR method effectively predicts whether a polyester can be soluble in a wide range of solvents or not

TABLE 3 | Coefficients of the linear regression for HSP prediction.

Functional Group	δ_d	δ_p	δ_h	*Ra*
-CH_3	12.8	−22.4	26.5	−21.5
-CH_2-	0.4	−0.25	0.7	−0.2
-CH-	−26.0	44.56	−52.9	42.8
-C-	−39.2	67.4	−80.5	64.9
-Cyclohexane-	37.7	−67.4	77.4	−64.36
-CH=CH-	0.4	−0.83	1.2	−1.1
-CH=CH_2	0.2	−0.79	0.95	−1.1
-O-	12.8	−21.7	25.1	−20.8
-COO-	6	−10.4	11.8	−10.0
-OH	21.8	−34.8	43.2	−32.9
-Ortho-	1.5	−1.0	3.2	−1.0

via the determination of R_0. This possibility of prediction is critical in the industrial world in order to save handling time and to be able to quickly develop new resins. Indeed, it will be possible to know in advance the solubility or otherwise of an unsaturated polyester in a new solvent whose Hansen parameters are known. The influence of each functional group on the solubility radius of the unsaturated polyesters is obtained by means of the coefficients of the multiple linear equation. The groups -CH- and -C- have positive coefficients, respectively, of 42.8 and 64.9. They therefore have a positive influence on the solubility radius and allow solubilization of the polyesters in more solvents. As stated in section Unsaturated Polyesters Properties Prediction by QSPR Method, these groups introduce rigidity and asymmetries into the polyester chain. This prevents the polyester chains from associating and favors the polymer-solvent bonds. On the contrary, the -cyclohexane-, -CH_2-, and -CH_3- type units favor the association of the chains by their regularity. The groups -O-, -COO-, and -OH are groups allowing the hydrogen bonds. When the polyester is solubilized in a solvent which does not have the capacity to form hydrogen bonds, the polyester will therefore tend to form these hydrogen bonds interchain way and thus promote the association and non-solubilization.

Two techniques described in the literature on the prediction of Hansen solubility parameters of polymers, namely the Hoftyzer—Van Krevelen (Krevelen and Nijenhuis, 2009) and Stefanis—Panayiotou (Stefanis and Panayiotou, 2008) methods, allow to obtain the coefficient of each functional group to use them next in a multilinear equation. The division of the structure of the synthesized polyesters into simple functional groups has been resumed to perform the parameters calculation for the three methods. The comparison of the mean absolute error (MAE) and correlation coefficient R² of the calculation compared to the experimental values of these three methods is made in **Table 4**.

The MAE of the three prediction methods for δ_d are almost equivalent. The QSPR method adapted to unsaturated polyesters therefore has a limited interest on this parameter. However, correlation coefficient for $\delta_d$is much better for the QSPR method. On the other hand, the QSPR method has a much lower absolute error on the δ_p parameter than the two other methods described in the literature as well as a better correlation coefficient than the methods found in literature. Mean absolute error for δ_h is the lowest with QSPR method but Stephanis-Panayiotou method has a slightly better R² for δ_h prediction than the QSPR method. Globally, the QSPR method is more accurate with unsaturated polyester HSP prediction. The prediction method Hoftyzer-Van krevelen is particularly suitable for high molecular weight polymers of different natures which is not the case for oligomeric unsaturated polyesters. The Stephanis-Panayiotou method is also more reliable for this kind of polymers. Our QSPR method which has been developed specifically on unsaturated polyester proved to be more reliable than the two other models for prediction of the Hansen solubility parameters of the same polymers.

TABLE 4 | Comparison of the MAE and correlation coefficient *R*² for the three methods of HSP prediction.

	δ_d		δ_p		δ_h	
Methods	**MAE**	R^2	**MAE**	R^2	**MAE**	R^2
Hoftyzer—Van Krevelen	0.7	0.08	10.2	0.00	5.5	0.49
Stephanis—Panayiotou	0.7	0.00	1.9	0.74	1.1	0.89
QSPR method (This work)	0.5	0.55	0.3	0.96	0.4	0.85

TABLE 5 | Evolution of the correlation coefficient (*R*²) depending of the descriptors used for T_g modeling.

Descriptor(s) used	R^2 prediction vs. experimental
-Ortho-	0.37
-Ortho-, -CH_3	0.56
-Ortho-, -CH_3, -O-	0.67
-Ortho-, -CH_3-, -O-, -CH-	0.72
-Ortho-, -CH_3-, -O-, -CH-, -CH_2-	0.74
-Ortho-, -CH_3-, -O-, -CH-, -CH_2-, -C-	0.93

Glass Transition Temperature Prediction by QSPR Method

Methods of predicting the glass transition temperature already exist in the literature (Katritzky et al., 1996; Bicerano, 2002; Krevelen and Nijenhuis, 2009). However, in the same way as for the prediction of Hansen parameters, these are optimal for high molecular weight polymers. Thus, a QSPR method applied to unsaturated polyesters may also be particularly suitable to predict T_g. In order to correlate the impact of each functional group on the glass transition temperature of the polyesters, a multiple linear regression is set up again in order to obtain the best coefficient of correlation R^2. The evolution of the correlation coefficient as a function of the functional groups introduced into the equation is described in **Table 5**.

With the six descriptors which are the -Ortho-, -CH_3-, -O-, -CH-, -CH_2-, and -C- groups, the prediction of the glass transition temperature of the synthesized unsaturated polyesters is effective (**Figure 1**) and practical.

The mean absolute error is 2.7°C. The list of coefficients of each descriptor with respect to the regression equation is given in **Table 6**.

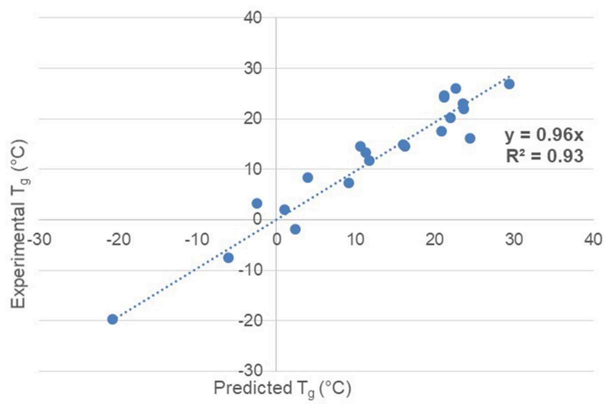


FIGURE 1 | T_g prediction accuracy vs. experimental via QSPR method.

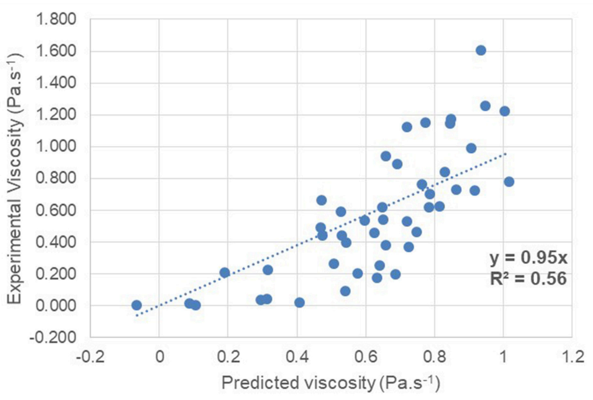


FIGURE 2 | Prediction accuracy of UP viscosity in solution according to the QSPR method.

TABLE 6 | Coefficients of the linear regression for T_g prediction.

Descriptor(s)	Coefficient
Intercept	−5.44
-CH_2-	−4.60
-CH_3	−4.03
-CH-	11.54
-C-	18.79
-O-	−7.92
-Ortho-	6.91

In addition to provide a linear equation allowing the extrapolation of the glass transition temperature of unsaturated polyesters with structures which are different from those already tested, these coefficients validate the concepts stated in part 3.1. The group -CH_2- having a coefficient of −4.60, the aliphatic chains such as adipic acid or 2-ethylhexanol do indeed have a plasticizer effect within the polyester chains. It is the same for the ether groups with, for example, dipropylene glycol or diethylene glycol. The introduction of -CH_3 groups within the polyester also has a negative effect on the glass transition temperature of the polyester (coefficient at −4.03) by the introduction of free volume between the chains. The groups -CH- and -C- by their steric hindrance have a mobility much smaller than a -CH_2- group or a -CH_3 group. In the polyester chain, they induce additional rigidity which results in an increase in the glass transition temperature. The same principle also applies when aromatic groups are introduced within the polyester chains. Until now, this prediction model is suitable for unsaturated polyesters with alcohol endings as well as aromatic groups based on orthophthalic anhydride. In fact, polyesters with acid terminations do not have the same hydrogen bonding capacity as the alcohol chain-ends. This difference must certainly play a role in establishing the glass transition temperature of polyesters. On the other hand, the impact on the glass transition temperature of the type of introduced aromatic acid, namely ortho-, iso-, tere-phthalate, within the polyester is significant because of the difference in steric hindrance. This prediction model does not take into account these constraints. These should be studied in a future work.

Unsaturated Polyester (UP) Viscosity Prediction by QSPR Method

A QSPR method was also applied to see if it was effective in predicting the viscosity of the unsaturated polyester resins in the database. The input data correspond to the number average molecular weight of the polyester M_n (obtained by SEC), its dispersity index $Đ$ (obtained by SEC), its glass transition temperature T_g (obtained by DSC), the *RED* polymer-solvent compatibility (obtained via HSPiP), the molar volume of the solvent M_{vol} (obtained via HSPiP), the concentration of the polyester in the solution. The target property of the QSPR method is the measured viscosity of each entry in the database. The coefficients of the multiple linear equation obtained from the 80% of the database were used to predict the viscosities of the remaining 20% of the database. The comparison between predicted and experimental viscosities is shown in **Figure 2**.

The prediction accuracy of the solution viscosity of polyesters by QSPR is low. The coefficient of correlation R^2 obtained by QSPR is 0.56. The mean absolute error (MAE) is 0.22 Pa.s^{-1}. This inefficiency of the prediction is explained by the limitation of the QSPR model to linear phenomena. However, the descriptors used maybe have a non-linear influence. Neural networks are therefore of great interest in this type of application and were tried in the next step.

Setup of the Neural Network

Inputs Selection

In order to set up a neural network allowing the future prediction of unsaturated polyesters viscosities in solution, several descriptors have been chosen as factors having potentially an impact on the viscosity. Seven descriptors were chosen, namely the number average molecular weight M_n (polystyrene equivalent) of the polyester (Ebewele, 2000; Mark, 2007), its dispersity index $Đ$ (Lundberg et al., 1960; Cross, 1969), its glass transition temperature T_g (Young and Lovell, 1996;

TABLE 7 | Linear correlation coefficient R^2 of each descriptor one by one on unsaturated polyester viscosity in solution.

Descriptor	R^2
Concentration	0.495
T_g	0.383
M_n	0.328
M_{vol}	0.289
δ_H	0.097
$Đ$	0.049
RED	0.037

Ebewele, 2000; Mark, 2007), the polyester-solvent compatibility denoted *RED* (Flory, 1942; Hillyer and Leonard, 1973; Krevelen and Nijenhuis, 2009), the molar volume of the solvent M_{vol} (Louwerse et al., 2017), the δ_h of the solvent (Krevelen and Nijenhuis, 2009) and the concentration of the polyester in the resin (Hillyer and Leonard, 1973; Louwerse et al., 2017). This choice was based on the existing literature describing the physical chemistry of polymers. However, it is important to check that these factors really have an impact and that they allow the neural network to build a reliable model based on these factors. In a first step, the impact of each descriptor is tested by calculating the linear correlation coefficients of each descriptor one by one on the measured viscosities (**Table 7**).

In order to test the quality of each descriptors in the neural network, 80% of the database was used for the training of the neural network and the remaining 20% to test the impact of the number of descriptors used on the normalized squared error obtained between the predicted viscosity and the experimental viscosity. Firstly, the neural network is trained only with the descriptor with the most important linear correlation coefficient R^2 (**Table 7**). The normalized squared error (NSE) following the training is calculated on both the training and test values. Then the second descriptor with the most important R^2 was added to the first one to see if it reduces the NSE. The third descriptor was then added to see again if the NSE still improve. This procedure was repeated until the integration of all the descriptors of the database. This test proved that there were no useless descriptors or no over-fitting during the test phase of the neural network. The results of these trainings are shown in **Figure 3**.

The evolution of the NSE according to the descriptors added for the training of the neural network makes it possible to see that there are two descriptors which do not improve the performances of the neural network. These two descriptors are the number average molecular weight M_n (descriptor 3) and the dispersity index Đ (descriptor 6). In order to check the performance of the neural network without these two descriptors, a new test was launched only with the remaining five descriptors (**Figure 4**).

Without the number average molecular weight M_n and the dispersity index *Đ*, the decrease in NSE is much more regular. In addition, the neural network goes from 7 descriptors in input to only 5 while keeping identical performances. The reduction of the descriptors number is beneficial for the neural network since this may avoid over-fitting phenomena when there are too many descriptors. In addition, from a practical point of view, the limitation of the number of descriptors required allows to set up and enrich an important database by reducing the number of information required for each manipulation.

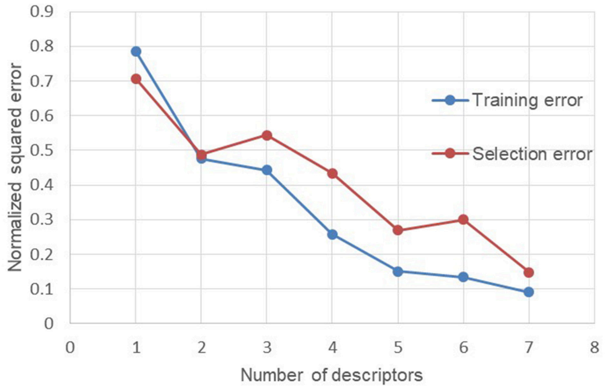


FIGURE 3 | Evolution of the normalized squared error depending of descriptors used for training.

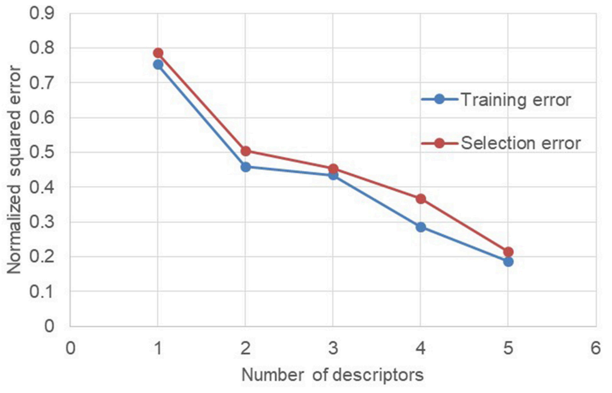


FIGURE 4 | Evolution of the normalized squared error depending of descriptors (without M_n and Đ) used for training.

Optimization of the Number of Neurons

The number of neurons in the hidden learning layer is an important parameter to optimize (Díaz-Rodríguez et al., 2014). Indeed, if there are too few neurons in relation to the complexity of the problem, there is a risk of under-fitting due to a lack of parameters. On the other hand, if there are too many neurons hidden in the learning layer, there is a risk of over-fitting during the prediction phase of the target property. In order to have a correct number of learning neurons, the database was randomly divided again with 80% of the inputs intended for learning and 20% for the test. Then the neural network was trained and then tested with a growing number of learning neurons. Three training and selection tests per number of neurons were performed to obtain the lowest normalized squared errors. The results are shown in **Figure 5**.

Between 0 and 3 learning neurons, under-fitting problems occur because the errors found are the highest in the range of the

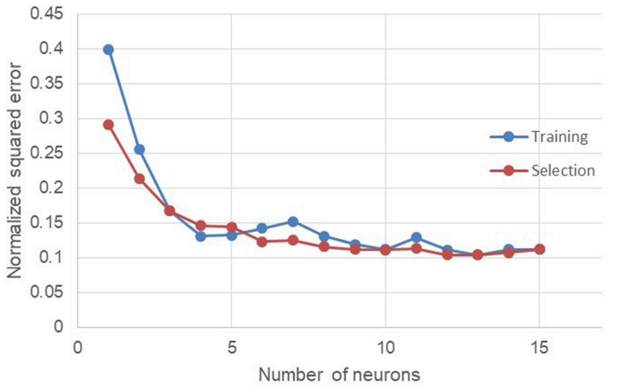


FIGURE 5 | Evolution of the normalized squared error depending of the number of neurons.

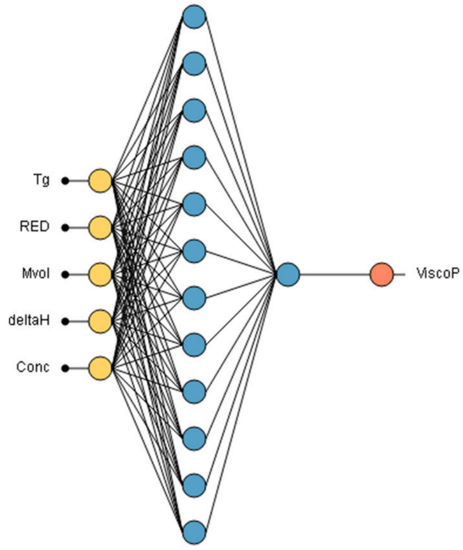


FIGURE 6 | Neural network used for unsaturated polyester resin viscosity prediction Input data are introduced through yellow neurons, the 12 learning neurons are represented in blue. One neuron in a second layer sum up linearly the outputs of the first layer. The orange neuron is the viscosity output neuron.

number of neurons tested. As the number of neurons increases, the errors decrease until they become stable. Similar tests have been conducted up to 40 hidden learning neurons without errors in the learning or testing phases indicating the occurrence of an over-fitting phenomenon. However, the multiplication of the number of neurons also implies the increase of the number of calculations and therefore a greater need for computation needs. As part of this work, the number of neurons was set at 12.

Training of the Neural Network

The neural network was trained with 80% randomly selected from the database created. The neural network is composed of 5 inputs, namely the glass transition temperature T_g of the polyester, the *RED* (polymer-solvent compatibility), the δ_h of the solvent, its molar volume M_{vol} and the concentration of the polyester in the solvent. The hidden learning layer has 12 neurons and consequently 85 synapses. The neural network used in this study is illustrated in **Figure 6**.

The activations functions used are the hyperbolic tangents. The training algorithm chosen is the quasi-Newton method (Setiono and Hui, 1995) with a normalized squared error. This algorithm is based on Newton's method but does not require the computation of the second derivative to find the local minimum of the error. Instead, the quasi-newton method computes an approximation of the inverse Hessian matrix at each iteration of the algorithm, by only using gradient information. A regularization coefficient of 0.01 was applied in order to have a better generalization of the model.

Influence of Each Descriptor on Viscosity

Once the neural network is trained, it is possible to isolate the influence of each descriptor on the viscosity by fixing the others by their average. This provides valuable information for understanding the phenomena influencing unsaturated polyester viscosity in concentrated solution. The results are shown in **Figure 7**.

The evolution of the viscosity as a function of the polyester concentration in the solution is represented by **Figure 7A**. This model obtained via the neural network corresponds to the models conventionally described in the literature (Yang, 1996). Indeed, taking into account other fixed descriptors, the viscosity of the polyester in solution slowly changes to 58.5% by weight of the polyester and the slope increases substantially thereafter. This phenomenon is due to the overrun of the critical concentration of the polyester in a solvent (Takahashi et al., 1985). At a concentration below the critical concentration, the number of chain entanglements of polymers is low with respect to concentration. While this number of entanglements increases drastically above the maximum critical concentration which causes the increase in the viscosity slope after 58.5% by weight of polyester in the solution.

The influence of polymer-solvent compatibility (*RED*) on viscosity is shown in **Figure 7B**. The viscosity of the polyester decreases progressively when the *RED* goes from 0.2 to 0.7 and then increases again from 0.7 to 1. This evolution of the viscosity can be explained from the point of view of the hydrodynamic volume occupied by the polymer in solution. When it is a dilute solution of polymer, the more it will be compatible with its solvent, the higher its hydrodynamic volume will be. Indeed, the number of polymer-solvent interactions being de facto high, the polymer chains will be relaxed. The entanglements of chains in the solution will therefore be maximized and the viscosity of the polymer in solution will increase. On the contrary, if the solvent is very poor compatible with the polymer, it will minimize these interactions with the solvent. It will shrink in the form of a globule, reduce its hydrodynamic volume, generate less entombment and thus reduce the viscosity in solution (Hillyer and Leonard, 1973). It is this phenomenon which explains the decrease of the viscosity for the *RED* from 0.2 to 0.7. However, in the case of unsaturated polyester resins, the polymer concentrations are

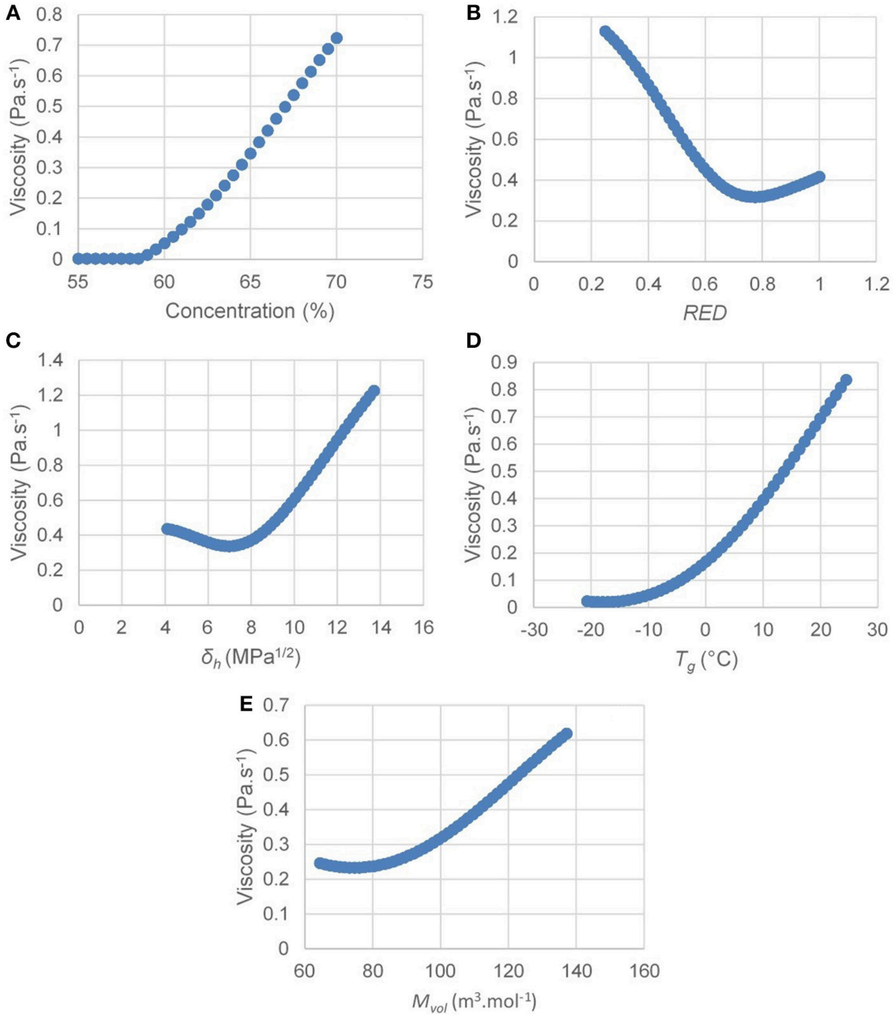


FIGURE 7 | Influence of each descriptor used in the neural network on the unsaturated polyester viscosity in solution [**(A)** influence of concentration; **(B)** influence of *RED*; **(C)** influence of δ_h; **(D)** influence of T_g; **(E)** influence of M_{vol}].

high. When the solvent become incompatible, the globule-like polymer chains will agglomerate to further minimize interactions with the solvent. This agglomerate of globule therefore has a larger hydrodynamic volume than the isolated globule, which implies a slight increase in viscosity from 0.7 in *RED* up to 1. This phenomenon has already been described in the literature (Burrell, 1973; Hillyer and Leonard, 1973) but the use of a neural network allows to find this result thanks to the processing of the data obtained.

Regarding the influence of δ_h on the polyester viscosity in solution represented in **Figure 7C**, the viscosity decreases between 4.1 and 7.0 MPa$^{1/2}$ and then increases significantly between 7.0 and 13.7 MPa$^{1/2}$. This phenomenon has already been described in the literature by Nelson who has taken over the classification of solvents from Pimentel and McClellan (Burrell, 1973). The solvents are classified in four categories namely: (a) proton donors (chloroform for example), (b) proton acceptors (ketones, esters, ethers, aromatic hydrocarbons for example), (c) proton donors and acceptors (alcohols, carboxylic acids, water for example), and (d) absence of hydrogen bonds (such as aliphatic hydrocarbons). The solvents used in the database of polyesters in solution with δ_h values between 4.1 and 7.0 MPa$^{1/2}$ are in category (b) some non-exhaustive examplesof which are styrene (δ_h = 4.1 MPa$^{1/2}$), cyclohexanone (δ_h = 5.1 MPa$^{1/2}$), methyl methacrylate (δ_h = 5.4 MPa$^{1/2}$), acetone (δ_h = 7.0 MPa$^{1/2}$). Since the polyesters are acceptors and donors of hydrogen bonds (terminal alcohol functions and ester functions), the proton acceptor solvents allow the hydrogen bonds between the polyester chains to be broken. The slight decrease in the viscosity between 4.1 and 7.0 MPa$^{1/2}$ is due to the greater capacity of solvents such as ketones, esters or ethers (δ_h = 5.0–7.0 MPa$^{1/2}$) to accept hydrogen bonds with respect to typical solvents such as aromatic hydrocarbons (δ_h = 4.0–5.0 MPa$^{1/2}$). On the contrary, the solvents possessing the higher δ_h belong to category (c) and are both acceptors and proton donors (acetic acid δ_h = 13.5 MPa$^{1/2}$, benzyl alcohol δ_h = 13.7 MPa$^{1/2}$). Before they can break the established hydrogen bonds between the polyester chains, the solvents with high δ_h must first break their own hydrogen bonds. This phenomenon leads for the polyester a longer and harder dissolution in these kind of solvents. In addition, there is also formation of a denser network of hydrogen bonds between the polyester chains and the solvent molecules. This network is at the origin of the drastic increase in viscosity for solvents with δ_h between 7 and 14 MPa$^{1/2}$.

The glass transition temperature of the polyester also influences the viscosity of the unsaturated polyester in solution (**Figure 7D**). Indeed, the higher the glass transition temperature (constant molecular weight), the higher the viscosity. The glass transition temperature is directly related to the rigidity of the chain. Thus, when the polyester chains are in solution at high concentration, the energy required for the mobility of the rigid chains will be greater compared to flexible chains. Rigid chains will therefore have a higher viscosity with respect to these flexible chains (Berry and Fox, 1968).

Regarding the influence of the molar volume of the solvent (**Figure 7E**), the viscosity increases as the molar volume of the solvent increases (Flory, 1942; Louwerse et al., 2017). This evolution can be explained by the entropy of mixing (solvent + polymer) (Equation 6).

$$\Delta S_{mix} = -R \times (xlnx + (1-x)\ln(1-x)) \qquad (6)$$

The value x is the molar fraction of the polymer and R is the ideal gas constant. Solvents with small molar volumes give a greater entropy of mixture per liter of solvent. They are therefore better solvents.

Prediction of Unsaturated Polyester Viscosity in Solution With Neural Network

In order to compare the prediction efficiency of the neural network with the QSPR method, the neural network was trained with the same 80% of the database used for the QSPR method. The remaining 20% of the database was tested to compare the predicted viscosity with the experimental viscosity. The prediction accuracy of the trained neural network is represented in **Figure 8**.

A correlation coefficient $R^2 = 0.88$ was obtained thanks to the trained neural network. The mean absolute error is 0.115 Pa.s^{-1}. The prediction efficiency is much higher with the neural network compared to the QSPR method. This method is therefore particularly suitable for this type of application.

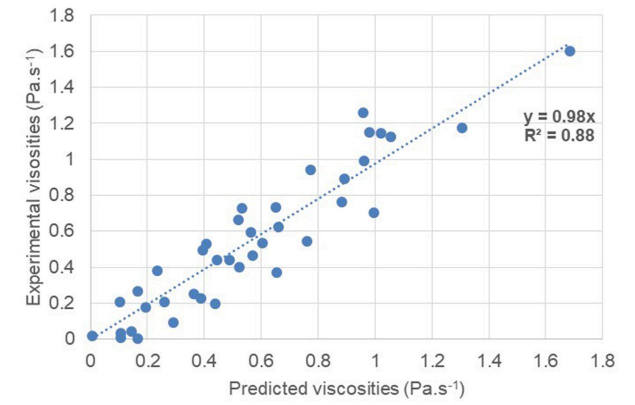


FIGURE 8 | Prediction accuracy of UP viscosity in solution according to the trained Neural Network.

TABLE 8 | Results of the K-fold cross validation ($K = 5$) method for the viscosities prediction.

Viscosities range (Pa.s^{-1})	R^2	MAE (Pa.s^{-1})
0.003–1.889	0.85	0.116

K-Fold Cross-Validation

K-Fold cross-validation is a method of validating the neural network to determine predictability. Indeed, all entries in the database are used to check the model. The database is divided into K fractions. In this work, the database was divided into 5 fractions ($K = 5$). The neural network was initially trained with 4 fractions of the database. The fifth fraction, which was not used for training, was used for the neural network prediction test. This operation was repeated 5 times with a different K fraction each time for the test phase. The averages of the correlation coefficients R^2 and the mean average error (MAE) obtained are given in **Table 8**.

The R^2 and MAE values obtained by the K-fold cross validation method allow the validation of the neural network stability as well as its ability to effectively predict the viscosity of unsaturated polyester resins. The current database includes 220 entries divided between 179 entries for training and 41 entries for testing the trained neural network. The latter has already shown to be very effective compared to a QSPR model. It might be interesting to extend this comparison by expanding the database. To do this, other polyester resins can be synthesized to teach the neural network new structures and new solvents can also be added.

CONCLUSION

The viscosity of unsaturated polyester resins is a very important criterion in the industrial field. Indeed, a viscosity out of specifications can interfere with the handling of the resin and make it impossible to process. This viscosity depends on the chemical structure of the polyester, the nature of the solvent and the concentration of the polyester in solution. The great diversity of existing diols and diacids as well as the current growth of the number of reactive diluents therefore implies a variation of the viscosity which is extremely difficult to predict simply by mathematical or physical laws.

Firstly, in order to avoid experimental input descriptors and to be able to predict the viscosity of polyester resins from theoretical and easily accessible values, a QSPR method has been applied to predict Hansen parameters as well as temperature of glass transition of unsaturated polyesters. This method has proved to be particularly effective compared to other existing methods in the literature because these described methods are based on high molecular weight polymers. However, the QSPR method has proved ineffective for predicting the viscosity of unsaturated polyesters in solution. A classical linear prediction method does not allow non-linear phenomena to be taken into account. It is therefore wise to use machine learning tools.

In this work, a neural network has been set up to verify the ability of such a machine learning process to predict the viscosity of these resins from 21 unsaturated polyesters and 220 mixtures with solvents. This network composed of five descriptors and 12 learning neurons allowed the successful prediction of the viscosity of 41 test resins with an R^2 correlation coefficient of 0.88 and an MAE of 0.116 $\mathrm{Pa.s}^{-1}$. These results are very promising given the amount of data available to date. The regular update of the database with the manipulations carried out over time will undoubtedly allow the improvement of the prediction.

DATA AVAILABILITY

All datasets generated for this study are included in the manuscript and/or the **Supplementary Files**.

AUTHOR CONTRIBUTIONS

JM wrote the manuscript. JM and AL worked on the Neural Network optimization and testing. JM and AL did the polyester syntheses and manipulations. J-VR designed the input descriptors for the neural network. A-SS and CR supervised the study and revised the manuscript.

ACKNOWLEDGMENTS

The authors also wish to thank Pr. Véronique Rataj and the CISCO laboratory from the Université de Lille for the support provided regarding the use of the software HSPiP.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fchem.2019.00375/full#supplementary-material

REFERENCES

Abbott, S. (2013). *Hansen Solubility Parameters in Practice (HSPiP)*. Available online at: https://hansen-solubility.com (accessed January 2019).

Behler, J. (2011). Neural network potential-energy surfaces in chemistry: a tool for large-scale simulations. *Phys. Chem. Chem. Phys.* 13:17930. doi: 10.1039/c1cp21668f

Bengough, W. I., Goldrich, D., and Young, R. A. (1967). The copolymerizations of methyl methacrylate with diethyl maleate and diethyl fumarate. *Eur. Polym. J.* 3, 117–123. doi: 10.1016/0014-3057(67)90088-2

Berry, G. C., and Fox, T. (1968). "The viscosity of polymers and their concentrated solutions," in *Fortschritte der Hochpolymeren-Forschung* eds H.-J. Cantow, G. Dall'Asta, J. D. Ferry, W. Kern, G. Natta, S. Okamura, C. G. Overberger, W. Prins, G. V. Schulz, W. P. Slichter, A. J. Staverman, J. K. Stille and H. A. Stuart (Berlin; Heidelberg: Springer-Verlag), 261–357.

Bicerano, J. (2002). *Prediction of Polymer Properties*. New York, NY: CRC Press.

Biron, M. (2013). *Thermosets and Composites: Material Selection, Applications, Manufacturing and Cost Analysis*. Oxford: Elsevier.

Burrell, H. (1973). "Trends in Solvent Science and Technology," in *Solvents Theory and Practice*, ed. R. W. Tess (Washington, DC: American Chemical Society), 1–10.

Camacho-Zuñiga, C., and Ruiz-Treviño, F. A. (2003). A new group contribution scheme to estimate the glass transition temperature for polymers and diluents. *Ind. Eng. Chem. Res.* 42, 1530–1534. doi: 10.1021/ie0205389

Cancilla, J. C., Aroca-Santos, R., Wierzchoś, K., and Torrecilla, J. S. (2016). Hazardous aromatic VOC quantification through spectroscopic analysis and intelligent modeling to assess drinking water quality. *Chemometr. Intell. Lab. Syst.* 156, 102–107. doi: 10.1016/j.chemolab.2016.05.008

Cancilla, J. C., Díaz-Rodríguez, P., Izquierdo, J. G., Bañares, L., and Torrecilla, J. S. (2014a). Artificial neural networks applied to fluorescence studies for accurate determination of N-butylpyridinium chloride concentration in aqueous solution. *Sensors Actuators B* 198, 173–179. doi: 10.1016/j.snb.2014.02.097

Cancilla, J. C., Wang, S. C., Díaz-Rodríguez, P., Matute, G., Cancilla, J. D., Flynn, D., et al. (2014b). Linking chemical parameters to sensory panel results through neural networks to distinguish olive oil quality. *J. Agric. Food Chem.* 62, 10661–10665. doi: 10.1021/jf503482h

Chen, X., Sztandera, L., and Cartwright, H. M. (2008). A neural network approach to prediction of glass transition temperature of polymers. *Int. J. Intell. Syst.* 23, 22–32. doi: 10.1002/int.20256

Cousinet, S., Ghadban, A., Allaoua, I., Lortie, F., Portinha, D., Drockenmuller, E., et al. (2014). Biobased vinyl levulinate as styrene replacement for unsaturated polyester resins. *J. Polym. Sci. Part A.* 52, 3356–3364. doi: 10.1002/pola.27397

Cousinet, S., Ghadban, A., Fleury, E., Lortie, F., Pascault, J.-P., and Portinha, D. (2015). Toward replacement of styrene by bio-based methacrylates in unsaturated polyester resins. *Eur. Polym. J.* 67, 539–550. doi: 10.1016/j.eurpolymj.2015.02.016

Cross, M. M. (1969). Polymer rheology: influence of molecular weight and polydispersity. *J. Appl. Polym. Sci.* 13, 765–774. doi: 10.1002/app.1969.070130415

Curtis, L. G., Edwards, D. L., Simons, R. M., Trent, P. J., and Von Bramer, P. T. (1964). Investigation of maleate-fumarate isomerization in unsaturated polyesters by nuclear magnetic resonance. *Indus. Eng. Chem. Prod. Res. Dev.* 3, 218–221. doi: 10.1021/i360011a011

Dagher, H. J., Iqbal, A., and Bogner, B. (2004). Durability of isophthalic polyester composites used in civil engineering applications. *Polym. Polym. Compos.* 12, 169–182. doi: 10.1177/096739110401200302

Dai, J., Ma, S., Teng, N., Dai, X., Shen, X., Wang, S., et al. (2017). 2,5-furandicarboxylic acid- and itaconic acid-derived fully biobased unsaturated polyesters and their cross-linked networks. *Indus. Eng. Chem. Res.* 56, 2650–2657. doi: 10.1021/acs.iecr.7b00049

Delgove, M. A. F., Luchies, J., Wauters, I., Deroover, G. G. P., De Wildeman, S. M. A., and Bernaerts, K. V. (2017). Increasing the solubility range of polyesters by tuning their microstructure with comonomers. *Polym. Chem.* 8, 4696–4706. doi: 10.1039/C7PY00976C

Díaz-Rodríguez, P., Cancilla, J. C., Plechkova, N. V., Matute, G., Seddon, K. R., and Torrecilla, J. S. (2014). Estimation of the refractive indices of imidazolium-based ionic liquids using their polarisability values. *Phys. Chem. Chem. Phys.* 16, 128–134. doi: 10.1039/C3CP53685H

Díaz-Rodríguez, P., Cancilla, J. C., Wierzcho,ś, K., and Torrecilla, J. S. (2015). Non-linear models applied to experimental spectroscopical quantitative analysis of aqueous ternary mixtures of imidazolium and pyridinium-based ionic liquids. *Sens. Actuators B* 206, 139–145. doi: 10.1016/j.snb.2014.09.037

Dong, D., McAvoy, T. J., and Zafiriou, E. (1996). Batch-to-batch optimization using neural network models. *Indus. Eng. Chem. Res.* 35, 2269–2276.

Ebewele, R. O. (2000). *Polymer Science and Technology*. Boca Raton, FL: CRC Press.

Fink, J. K. (2013). "Unsaturated Polyester Resins," in *Reactive Polymers Fundamentals and Applications* (New York, NY: Elsevier), 1–48.

Flory, P. J. (1942). Thermodynamics of high polymer solutions. *J. Chem. Phys.* 10, 51–61. doi: 10.1063/1.1723621

Gasteiger, J., and Zupan, J. (1993). Neural networks in chemistry. *Angewand. Chem. Int. Ed. Eng.* 32, 503–527. doi: 10.1002/anie.199305031

Gharagheizi F. (2007b). A new accurate neural network quantitative structure-property relationship for prediction of θ (Lower Critical Solution Temperature) of polymer solutions. *Epoly* 7:1314. doi: 10.1515/epoly.2007.7.1.1314

Gharagheizi, F. (2007a). QSPR analysis for intrinsic viscosity of polymer solutions by means of GA-MLR and RBFNN. *Comput. Mater. Sci.* 40, 159–167. doi: 10.1016/j.commatsci.2006.11.010

Hansen, C. M. (2002). *Hansen Solubility Parameters: A User's Handbook*. Boca Raton, FL: CRC Press.

Hillyer, M. J., and Leonard, W. J. (1973). "Calculation of concentrated polymer solution viscosities: a new approach," in *Solvents Theory and Practice*, ed R. W. Tess (Washington, DC: American Chemical Society), 31–47.

Joyce, S. J., Osguthorpe, D. J., Padgett, J. A., and Price, G. J. (1995). Neural network prediction of glass-transition temperatures from monomer structure. *J. Chem. Soc. Faraday Trans.* 91, 2491–2496. doi: 10.1039/FT9959102491

Katritzky, A. R., Rachwal, P., Law, K. W., Karelson, M., and Lobanov, V. S. (1996). Prediction of polymer glass transition temperatures using a general quantitative structure–property relationship treatment. *J. Chem. Inform. Comp. Sci.* 36, 879–884. doi: 10.1021/ci950156w

Krevelen, D. W., and Nijenhuis, K. (2009). *Properties of Polymers: Their Correlation With Chemical Structure: Their Numerical Estimation And Prediction From Additive Group Contributions*, *4th Edn*. Amsterdam: Elsevier.

Lewis, F. M., and Mayo, F. R. (1948). Copolymerization. IX. a comparison of some cis and trans isomers1,2. *J. Am. Chem. Soc.* 70, 1533–1536. doi: 10.1021/ja01184a071

Li, S., Yang, X., Huang, K., Li, M., and Xia, J. (2014). Design, preparation and properties of novel renewable UV-curable copolymers based on cardanol and dimer fatty acids. *Prog. Org. Coat.* 77, 388–394. doi: 10.1016/j.porgcoat.2013.11.011

Liu, W., and Cao, C. (2009). Artificial neural network prediction of glass transition temperature of polymers. *Colloid Polym. Sci.* 287, 811–818. doi: 10.1007/s00396-009-2035-y

Loschen, C., and Klamt, A. (2012). COSMOquick: a novel interface for fast σ-profile composition and its application to COSMO-RS solvent screening using multiple reference solvents. *Indus. Eng. Chem. Res.* 51, 14303–14308. doi: 10.1021/ie3023675

Louwerse, M. J., Maldonado, A., Rousseau, S., Moreau-Masselon, C., Roux, B., and Rothenberg, G. (2017). Revisiting hansen solubility parameters by including thermodynamics. *ChemPhysChem* 18, 2999–3006. doi: 10.1002/cphc.201700408

Lundberg, J. L., Hellman, M. Y., and Frisch, H. L. (1960). The study of the polydispersity of polymers by viscometry. *J. Polym. Sci.* 46, 3–17. doi: 10.1002/pol.1960.1204614702

Marengo, E., Bobba, M., Robotti, E., and Lenti, M. (2004). Hydroxyl and acid number prediction in polyester resins by near infrared spectroscopy and artificial neural networks. *Analy. Chim. Acta* 511, 313–322. doi: 10.1016/j.aca.2004.01.053

Mark, J. E. (2007). *Physical Properties of Polymers Handbook*, *2 Edn*. New York, NY: Springer.

Mattioni, B. E., and Jurs, P. C. (2002). Prediction of glass transition temperatures from monomer and repeat unit structure using computational neural networks. *J. Chem. Inf. Comput. Sci.* 42, 232–240. doi: 10.1021/ci010062o

Mishra, S., Mohanty, A ., Drzal, L., Misra, M., Parija, S., Nayak, S. et al. (2003). Studies on mechanical performance of biofibre/glass reinforced polyester hybrid composites. *Compo. Sci. Technol.* 63, 1377–1385. doi: 10.1016/S0266-3538(03)00084-8

Panic, V. V., Seslija, S. I., Popovic, I. G., Spasojevic, V. D., Popovic, A. R., Nikolic, V. B., et al. (2017). Simple one-pot synthesis of fully biobased unsaturated polyester resins based on itaconic acid. *Biomacromolecules* 18, 3881–3891. doi: 10.1021/acs.biomac.7b00840

Sadler, J. M., Nguyen, A.-P., Greer, S. M., Palmese, G. R., and La Scala, J. J. (2012). Synthesis and characterization of a novel bio-based reactive diluent as a styrene replacement. *J. Biobased Mater. Bioener.* 6, 86–93. doi: 10.1166/jbmb.2012.1193

Setiono R, and Hui LCK. (1995). Use of a quasi-Newton method in a feedforward neural network construction algorithm. *IEEE Trans. Neural Netw.* 6, 273–277. doi: 10.1109/72.363426

Sheela, K. G., and Deepa, S. N. (2013). Review on methods to fix number of hidden neurons in neural networks. *Mathemat. Probl. Eng.* 2013, 1–11. doi: 10.1155/2013/425740

Song, Y. S., Youn, J. R., and Gutowski, T. G. (2009). Life cycle energy analysis of fiber-reinforced composites. *Compos. Part A* 40, 1257–1265. doi: 10.1016/j.compositesa.2009.05.020

Stefanis, E., and Panayiotou, C. (2008). Prediction of hansen solubility parameters with a new group-contribution method. *Int. J. Thermophys.* 29, 568–585. doi: 10.1007/s10765-008-0415-z

Svozil, D., Kvasnicka, V., and Pospichal, J. (1997). Introduction to multi-layer feed-forward neural networks. *Chemometr. Intell. Lab. Syst.* 39, 43–62. doi: 10.1016/S0169-7439(97)00061-0

Takahashi, Y., Isono, Y., Noda, I., and Nagasawa, M. (1985). Zero-shear viscosity of linear polymer solutions over a wide range of concentration. *Macromolecules* 18, 1002–1008. doi: 10.1021/ma00147a033

Torrecilla, J. S., Aragón, J. M., and Palancar, M. C. (2008). Optimization of an artificial neural network by selecting the training function. *Applicat. Olive Oil Mills Waste. Indus. Eng. Chem. Res.* 47, 7072–7080. doi: 10.1021/ie8001205

Torrecilla, J. S., Tortuero, C., Cancilla, J. C., and Díaz-Rodríguez, P. (2013). Estimation with neural networks of the water content in imidazolium-based ionic liquids using their experimental density and viscosity values. *Talanta* 113, 93–98. doi: 10.1016/j.talanta.2013.03.060

Turner, S. R., Seymour, R. W., and Smith, T. W. (2001). Cyclohexanedimethanol Polyesters. *Encyclopedia Polym. Sci. Technol.* 2, 127–134. doi: 10.1002/0471440264.pst257

Wei, J. N., Duvenaud, D., and Aspuru-Guzik, A. (2016). Neural networks for the prediction of organic chemistry reactions. *ACS Cent. Sci.* 2, 725–732. doi: 10.1021/acscentsci.6b00219

Xu, B., Wang, N., Chen, T., and Li, M. (2015). *Empirical Evaluation of Rectified Activations in Convolutional Network. arXiv:1505.00853 [cs, stat]*. Available online at: http://arxiv.org/abs/1505.00853 (accessed January 14, 2019).

Yadav, S. K., Schmalbach, K. M., Kinaci, E., Stanzione, J. F., and Palmese, G. R. (2018). Recent advances in plant-based vinyl ester resins and reactive diluents. *Eur. Polym. J.* 98, 199–215. doi: 10.1016/j.eurpolymj.2017.11.002

Yang, Y.-S. (1996). Viscosities of unsaturated polyester resins: combining effects of prepolymer structure, resin composition, and temperature. *J. Appl. Polym. Sci.* 60, 2387–2395. doi: 10.1002/(SICI)1097-4628(19960627)60:13<2387::AID-APP10>3.0.CO;2-2.

Young, R. J., and Lovell, P. A. (1996). *Introduction to Polymers*, *2nd Edn*. London: Chapman and Hall.

Zaske, O. C., and Goodman, S. H. (1998). "Unsaturated polyester and vinyl ester resins," in *Handbook of Thermoset Plastics* (New York, NY: Elsevier), 97–168.

Zhang, J., Morris, A., Martin, E., and Kiparissides, C. (1998). Prediction of polymer quality in batch polymerisation reactors using robust neural networks. *Chem. Eng. J.* 69, 135–143.

Conflict of Interest Statement: 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.

frontiers in Chemistry

ORIGINAL RESEARCH
published: 31 May 2019
doi: 10.3389/fchem.2019.00385

Prediction of the Antioxidant Response Elements' Response of Compound by Deep Learning

Fang Bai[1†], *Ding Hong*[2†], *Yingying Lu*[3], *Huanxiang Liu*[1*], *Cunlu Xu*[2*] *and Xiaojun Yao*[3]

[1] School of Pharmacy, Lanzhou University, Lanzhou, China, [2] School of Information Science and Engineering, Lanzhou University, Lanzhou, China, [3] State Key Laboratory of Applied Organic Chemistry, Department of Chemistry, Lanzhou University, Lanzhou, China

The antioxidant response elements (AREs) play a significant role in occurrence of oxidative stress and may cause multitudinous toxicity effects in the pathogenesis of a variety of diseases. Determining if one compound can activate AREs is crucial for the assessment of potential risk of compound. Here, a series of predictive models by applying multiple deep learning algorithms including deep neural networks (DNN), convolution neural networks (CNN), recurrent neural networks (RNN), and highway networks (HN) were constructed and validated based on Tox21 challenge dataset and applied to predict whether the compounds are the activators or inactivators of AREs. The built models were evaluated by various of statistical parameters, such as sensitivity, specificity, accuracy, Matthews correlation coefficient (MCC) and receiver operating characteristic (ROC) curve. The DNN prediction model based on fingerprint features has best prediction ability, with accuracy of 0.992, 0.914, and 0.917 for the training set, test set, and validation set, respectively. Consequently, these robust models can be adopted to predict the ARE response of molecules fast and accurately, which is of great significance for the evaluation of safety of compounds in the process of drug discovery and development.

Keywords: antioxidant response elements (AREs), deep learning, toxicity, prediction, machine learning

INTRODUCTION

Antioxidant response elements (AREs), a series of momentous regulators of redox homeostasis and activators of cytoprotection during oxidative stress, can be activated by the exogenous sources of oxidative stress to participate in a variety of diseases ranging from cancer to neurodegeneration diseases (Raghunath et al., 2018). AREs are crucial in a variety of physiological functions and interact with numerous transcription factors to arrange the expression of a batch of cytoprotective genes in a spatio-temporal manner (Ney et al., 1990). More specifically, AREs profoundly contribute to the pathogenesis and progression of carbohydrate metabolism, cognition, inflammation, iron metabolism, metastasis, reduced nicotinamide adenine dinucleotide phosphate (NADPH) regeneration, lipid metabolism, and tissue remodeling (Hayes and Dinkova-Kostova, 2014). As such, AREs are the vital targets of the signal transduction pathway in eukaryotic cells responded to oxidative stress and the prevention of potential chemical toxicity. Therefore, determining if one compound can activate AREs is crucial for the assessment of potential risk of compound.

OPEN ACCESS

Edited by:
José S. Torrecilla,
Complutense University of Madrid,
Spain

Reviewed by:
Yun Tang,
East China University of Science and Technology, China
Peifeng Su,
Xiamen University, China

***Correspondence:**
Huanxiang Liu
hxliu@lzu.edu.cn
Cunlu Xu
clxu@lzu.edu.cn

† These authors have contributed equally to this work

Specialty section:
This article was submitted to Theoretical and Computational Chemistry, a section of the journal Frontiers in Chemistry

Received: *14 December 2018*
Accepted: *14 May 2019*
Published: *31 May 2019*

Citation:
Bai F, Hong D, Lu Y, Liu H, Xu C and Yao X (2019) Prediction of the Antioxidant Response Elements' Response of Compound by Deep Learning. Front. Chem. 7:385. doi: 10.3389/fchem.2019.00385

Generally, the *in vitro* and *in vivo* evaluations of interactions between a large number of compounds and the AREs are expensive, time-consuming and labor intensive. Relatively, the *in silico* approaches can be used as an alternative way to predict if a compound can activate AREs with lower cost. Based on the advantages of *in silico* approaches, some machine learning-based methods have been proposed to predict the AREs activators in the environment (Huang et al., 2016). However, there are some problems to be solved in the development of prediction model, such as high false positive and low precision. Several model optimization strategies were also applied, such as bagging, consensus modeling, and feature selection (Drwal et al., 2015; Filip, 2015; Abdelaziz et al., 2016; Gergo, 2016; Yoshihiro, 2016). Although these strategies can be effective on some degree, the predictive performance of traditional machine learning-based methods still needs to be improved. Undoubtedly, the process of feature filtering avoids dimensional disasters, but results in the loss of relevant information. One of the most promising models for AREs' response prediction is the DeepTox developed by Mayr et al. (2016). Based on the Tox21 challenge data, they used deep neural network methods to predict AREs' response. The best model has the area under the Receiver Operating Characteristic (ROC) curve (ROC-AUC) with 0.840 and balanced accuracy with 0.677 on the validation set. Moreover, other models based on traditional machine learning methods, such as random forest (RF), support vector machine (SVM) and Naive Bayesian etc., displayed ROC-AUC ranging from 0.768 to 0.832 and the balanced accuracy ranging from 0.519 to 0.729 (Huang et al., 2016). From above all, the more reliable models for the prediction of AREs' response are still needed.

Recently, deep learning (Lecun et al., 2015), as a promising machine learning method, has been applied in a wide range of fields, such as physics, life science and medical science (Gulshan et al., 2016). There were also some researches in biology (Mamoshina et al., 2016; Dang et al., 2018; Hou et al., 2018) and drug design areas (Gawehn et al., 2016; Hughes and Swamidass, 2017). Furthermore, deep learning methods have been also applied in small molecule toxicity assessment (Blomme and Will, 2016). For example, deep neural networks (DNN) was applied to predict drug-induced liver injury (Xu et al., 2015; Fraser et al., 2018). Convolution neural networks (CNN) was applied to predict the acute oral toxicity (Xu et al., 2017). Relative to other machine learning methods, deep learning methods (Wu and Wei, 2018) have some special advantages. For example, deep learning does not require feature selection, which can make the maximum use of extracted molecular features. Secondly, deep learning integrates a multi-layered network that enables the integration and selective activation of molecular features to avoid overfitting problems. Thirdly, deep learning includes many different network structures and can analyze and classify the problems from different perspectives. All of these suggests that the emerging deep learning algorithms may help us build more reliable models to predict AREs' response of the studied compounds.

In this study, to build more reliable prediction model of AREs' response, a series of deep learning methods including deep neural networks (DNN), recurrent neural network (RNN), highway networks (HN), convolution neural networks (CNN) were applied on a large date set (Tox21 challenge data) including 8,630 compounds. For comparison, the traditional machine learning methods, random forest (RF) and support vector machine (SVM), were also applied to predict AREs' response.

MATERIALS AND METHODS

Data Collection and Preparation

Tox21 challenge data[1] (shown in **Supporting Information**) was used to build model. The structures of compounds was downloaded from PubChem[2] according to the SID of compound. The AREs' response of compound was detected by CellSensor ARE-bla HepG2 cell line (Invitrogen), which was widely used to analyze the Nrf2/antioxidant response signaling pathway. To get the reliable data, each compound was tested in parallel by measuring the cell viability using CellTiter-Glo assay (Promega, Madison, WI) in the same wells. According to the test results, the molecules were categorized as "active," "inactive," or "inconclusion." To keep the built models reliable, the molecules with label of "inconclusion" were removed. The three-dimensional conformations of molecules play a pivotal role in the development of prediction model (Foloppe and Chen, 2009). Therefore, all compounds used in this study were initially subjected to full geometry optimization in LigPrep (Schrödinger, 2015). During the geometry optimization, the energy minimization was carried out using OPLS2005 force field (Kaminski et al., 2001). The inorganic compounds, mixtures, counterions, tautomers, and the duplicates were removed to make sure each compound has only one optimized conformation. The ionizable groups were taken into consideration and the distinct conformations were produced with the pH window of 7.0 ± 0.2. In particular, the molecules were deleted if there were some unreasonable or improper structures. After these pretreatments, the remaining compounds include 1,136 active and 6,299 inactive compounds.

Molecular Representation

The conventional molecular descriptors and molecular fingerprint features calculated by DRAGON 7.0 software (Kode srl, 2017) were used to describe the structural features of studied compounds. The calculated molecular descriptors include 0D (constitutional descriptors), 1D (functional groups counts, atom-centered fragments), 2D, and 3D-descriptors. The descriptors with missing values were removed. After this procedure, the number of remained molecular descriptors was 5,024. In general, the chemical features shared with those most active samples would be recognized to develop prediction models in the construction phase, while other chemical features shared with the least active molecules would be removed in order to avoid the complexity and increase the efficiency of models. The most relevant descriptors correlated with ARE toxicity were selected by Gini Index[3].

[1] https://tripod.nih.gov/tox21/challenge/data.jsp

[2] https://pubchem.ncbi.nlm.nih.gov/

[3] https://en.wikipedia.org/wiki/Gini_coefficient

TABLE 1 | The statistical summary of the data sets.

	Training set	Test set	External validation set
Activation	756	190	190
Inactivation	4,199	1,050	1,050

Molecular fingerprints (FPs) encode the structural information of a molecule by exploding its structure in all the possible substructure patterns. By this method, a molecule is described as a binary string of substructure keys. Different substructure patterns with SMARTS lists are predefined in a dictionary, within which substructures are created as atom-centered fragments using a variant of Morgan's extended connectivity algorithm. For a SMARTS pattern, if a substructure was presented in the given molecule, the corresponding bit was set to "1" and otherwise set to "0." In this study, the 1,024 bits extended connectivity fingerprints (ECFP) (Rogers and Hahn, 2010) were calculated by the DRAGON 7.0 program (Kode srl, 2017).

Data Splitting

To build the reliable model, the representative data set should be selected to build and test model. For this aim, we divided the data set into training set, test set and validation set with the ratio of 4:1:1 by the Kennard and Stone algorithm (Kennard and Stone, 1969) by considering the structural features and activity of compound. The statistical summary of the data set was presented in **Table 1**. To show the distribution of compounds in training set and test set , principal component analysis (PCA)[4] was performed based on the fingerprint features of compounds and the obtained results were shown in **Figure 1**, indicating that the compounds in training set and test set are well-distributed in the whole compound space.

Machine Learning Methods

Recently, deep learning (Lecun et al., 2015) algorithms have been widely applied in a variety of areas and gave promising results (Mamoshina et al., 2016). Deep learning methods comprise a lot of architectures, such as deep neural networks (DNN), recurrent neural network (RNN), highway networks (HN), and convolution neural networks (CNN). The principle of the used deep learning methods was described as below. Due that the RF (Breiman, 2001) and SVM (Mavroforakis and Theodoridis, 2006) have been introduced elsewhere, here, their principle was not given again.

DNN Classifier

The DNNs (Lecun et al., 2015) are developed from the structure of artificial neural networks with a large number of hidden layers. In the canonical deployment, the data are fed into the input layer and then transformed in a non-linear way through multiple hidden layers, and the final results are calculated and produced to the output layer. Neurons of hidden and output layer are connected to all neurons of the previous layer's. Each neuron calculates a weighted sum of its inputs and applies a non-linear activation function to generate its output as shown in **Figure 2**.

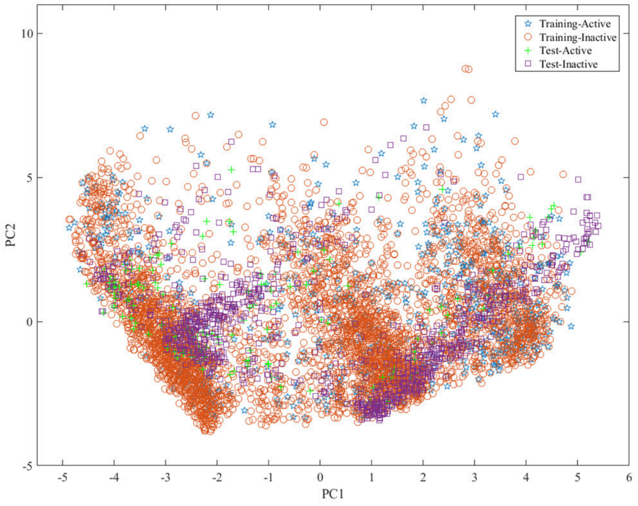


FIGURE 1 | The distribution of samples in the training set and test set by principle component analysis (PCA) based on the molecular fingerprint features.

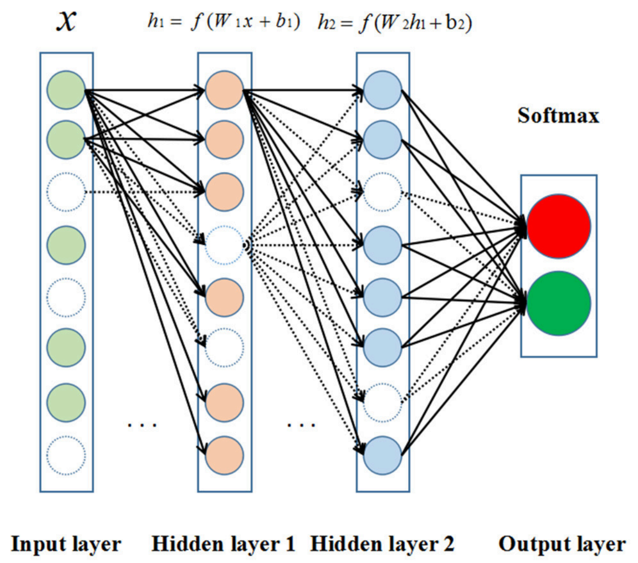


FIGURE 2 | The structure of deep neural network (DNN). Neurons are represented by circles. The colored circles indicate the activated neurons while the circles without color are inactivated neurons. In addition, the arrows represent heavy-weight transmissions between neurons, and the dashed arrows mean the invalid neuronal connections.

HN Classifier

The HNs (Srivastava et al., 2015) allows unimpeded information flow across several layers on information highways. The architecture is characterized by the use of gating units learning to regulate the flow of information through a network. HNs increases the possibility of studying extremely deep and efficient

[4] https://en.wikipedia.org/wiki/Principal_component_analysis

architectures for that it can be trained hundreds of layers directly with a variety of activation functions.

RNN Classifier

RNNs (Williams and Zipser, 1989) dedicates to process sequence data as it delivers state-of-the-art results in cursive handwriting and speech recognition. Its recent application in protein intrinsic disorder prediction demonstrated its significant ability to capture non-local interactions in protein sequences (Hanson et al., 2017). RNN processes an input sequence one element at a time, maintaining in its hidden units as a "state vector" that implicitly contains information about the history of all the past elements of the sequence. However, the training process becomes problematic for the backpropagated gradients either grow or shrink at each time step. After a batch of time steps they typically exploded or vanished (Hochreiter, 1991; Bengio et al., 2002). To solve the problem, a strategy was developed to augment the networks with an explicit memory-the long short-term memory (LSTM) networks. LSTM networks define special hidden units to remember the inputs for a long time (Hochreiter and Schmidhuber, 1997). A special unit called the memory cell acts like an accumulator or a gated leaky neuron. The cell has a connection to itself, so it copies its own real-valued state and it also accumulates the external signal at the same time. This self-connection mechanism decides whether to clear the content of the memory according to the other units states. LSTM networks have subsequently proved to be more effective than conventional RNNs, especially in several layers for each time step (Graves et al., 2013).

CNN Classifier

The CNNs (Krizhevsky et al., 2012) is a kind of multi-layer neural networks designed to process data fed in the form of multiple arrays. CNNs can exploit the property of many compositional hierarchies natural signals, owing to its ability of extracting higher-level features from lower-level ones. The architecture of typical CNN consists of three types of layers, which are convolutional, pooling, and fully-connected layers. Units in a convolutional layer are organized in feature maps. Each unit is connected to local patches of feature maps as well as previous layer through a set of weights called filter bank. After the process of convolutional layer, the new feature maps are obtained by applying a non-linear activation function, such as ReLU. The pooling layer is utilized to create an invariance filter to get small shifts and distortions by reducing the dimension of the feature maps. Each feature map of a pooling layer is connected to its preceding corresponding convolutional layers. The pooling layer computes the maximum of local patch of units in each feature map. And then the convolution and pooling layers are stacked by one or more fully-connected layers aiming to perform high-level reasoning feature generation (Hinton et al., 2012; Zeiler and Fergus, 2014).

The Implementations of Machine Learning Methods

For deep learning methods, the MinMaxScaler was utilized to transform features, by which each feature was scaled into a given range between zero and one. The nodes in the network used both rectified linear units (ReLUs) and tanh functions as activation functions. The dropout algorithm (Hinton et al., 2012; Dahl et al., 2014) and L2 regularization were used to prevent overfitting. The model was trained using Adam (Adaptive Moment Estimation) optimizer (Tieleman and Hinton, 2012). Xaiver initialization was applied to initialize the parameters (Glorot and Bengio, 2010; He et al., 2015). Grid search method was employed to search the best hyperparameters. It should be noted that CNN model was built based on fingerprint features but not the descriptors, for the reason that CNN could only process highly correlated local regions of input sequences (Lecun et al., 2015). The other models were constructed based on both fingerprints and descriptors. All Deep Learning methods were implemented in Deep Learning framework of Tensorflow (version 1.5.0). All deep learning methods had 3 layers and with dropout rate of 0.1. The loss function was cross entropy. The other hyperparameters of the deep learning methods are listed in **Table 2**. The RF and SVM proceeded in Python scikit-learn (version 0.19.0) (Pedregosa et al., 2011). There were 80 trees in RF models. For SVM models, the kernel function was set as polynomial with gamma 0.1.

The Evaluation of Model Performance

The performance of generated models was evaluated by several statistic metrics, such as sensitivity (SE), specificity (SP), accuracy (ACC), Matthews correlation coefficient (MCC) (Fang et al., 2013), F_1-score, and Precision. The formulas are shown as below:

$$\begin{aligned}
SE &= \frac{TP}{TP+FN}\\
SP &= \frac{TN}{TN+FP}\\
ACC &= \frac{TP+TN}{TP+TN+FP+FN}\\
MCC &= \frac{TP\times TN-FN\times FP}{\sqrt{(TP+FN)(TP+FP)(TN+FN)(TN+FP)}}\\
F_1 &= \frac{2TP}{2TP+FP+FN}\\
Precision &= \frac{TP}{TP+FP}
\end{aligned}$$

Where TP, TN, FP, and FN refer to the numbers of true positives, true negatives, false positives, and false negatives, respectively. All these various validation requirements have been suggested to evaluate the model performance. The Receiver Operating Characteristic (ROC) curve and the area under ROC curve (ROC-AUC) were also calculated to evaluate the predictive ability of built model.

RESULTS AND DISCUSSION

Performance Evaluation of Descriptors-Based Classification Models

In this study, firstly, we employed various algorithms to build classification models based on molecular descriptors. The statistical evaluation of these models on the training set, test set and validation set are summarized in **Table 3**. For clarity, we have

TABLE 2 | The hyperparameters of deep learning methods.

Models	Activation_function	Number of hidden units	Learning rate	Dropout rate	L2 weight decay	Epoches
DNN	Tanh, relu, softmax	5,024, 32, 32	0.00001	0.1	0.01	30,000
HN	Tanh, relu, softmax	5,024, 32, 32	0.0001	0.1	0.01	3,000
RNN	Tanh, relu, softmax	5,024, 32, 32	0.0001	0.1	0.01	3,000
CNN	Relu, relu, softmax	Patch size 10*10	0.0001	0.1	none	2,000

TABLE 3 | The performance of constructed models based on the general molecular descriptors.

Methods	Group	TP	TN	FP	FN	SE	SP	MCC	F1	Precision	ACC	ROC_AUC
RF	Tr	723	4,186	13	33	0.9563	0.9969	0.9638	0.9692	0.9823	0.9907	–
	Tst	75	1,050	0	115	0.3947	1.0000	0.5965	0.5660	1.0000	0.9073	0.8055
	Val	73	1,049	1	117	0.3842	0.9990	0.5828	0.5530	0.9865	0.9048	0.8298
SVM	Tr	751	3,689	510	5	0.9934	0.8785	0.7198	0.7447	0.5956	0.8961	–
	Tst	93	933	117	97	0.4895	0.8886	0.3631	0.4650	0.4429	0.8274	0.7755
	Val	98	958	92	92	0.5158	0.9124	0.4282	0.5158	0.5158	0.8516	0.7659
DNN	Tr	525	4,161	38	231	0.6944	0.9910	0.7766	0.7961	0.9325	0.9457	–
	Tst	75	1,046	4	115	0.3947	0.9962	0.5766	0.5576	0.9494	0.9040	0.8281
	Val	64	1,047	3	126	0.3368	0.9971	0.5321	0.4981	0.9552	0.8960	0.8573
HN	Tr	704	4,158	41	52	0.9312	0.9902	0.9270	0.9380	0.9450	0.9812	–
	Tst	99	1,043	7	91	0.5211	0.9933	0.6627	0.6689	0.9340	0.9210	0.7942
	Val	95	1,031	19	95	0.5000	0.9819	0.6008	0.6250	0.8333	0.9081	0.8267
RNN	Tr	693	4,151	48	63	0.9167	0.9886	0.9127	0.9259	0.9352	0.9776	–
	Tst	120	964	86	70	0.6316	0.9181	0.5320	0.6061	0.5825	0.8742	0.8287
	Val	110	949	101	80	0.5789	0.9038	0.4628	0.5486	0.5213	0.8540	0.8122

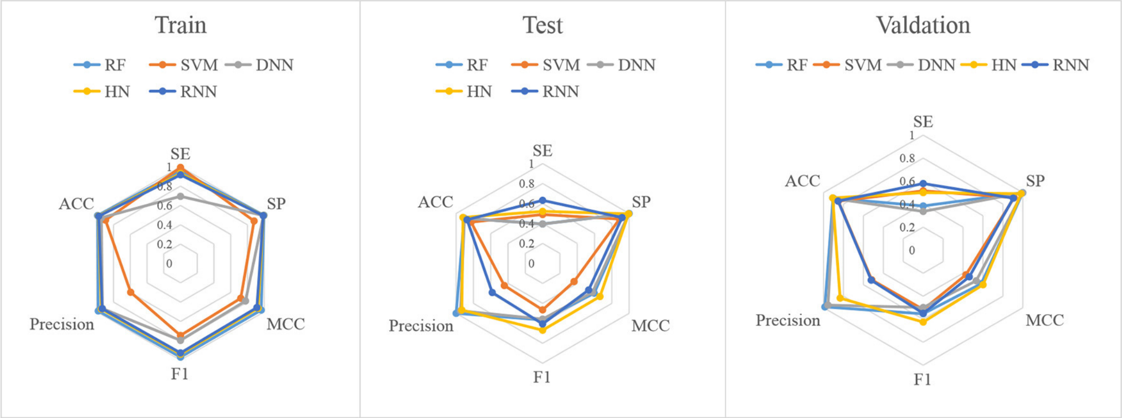


FIGURE 3 | Radar plot of the descriptors-based classification models.

grouped all the metrics by training, test and validation sets and presented them as radar plots. A perfect score on all metrics would be represented by a circle the size of the complete plot. The shape of the plots can also be indicative of the quality of the models. The larger the circle is, the better the model is. The radar plots of ARE toxicity model based on the structural descriptors are shown in **Figure 3**.

For the training set, all models gave very good SE, SP, MCC, F1-score, Precision, and ACC values. It should be noted that the SVM model showed lowest precision while DNN model exhibited lowest SE level. For the test and validation set, the indexes of all models exhibited a similar tendency, which tends to predict the compounds as inactivation due to the imbalanced distribution of active and inactive compounds. Among these models, the RNN model gave the highest SE value, while other indicators were not so well. It is worth noting that all indexes of the HN model were better than other models. In addition, the ROC-AUC is critical index for models performance and the ROC

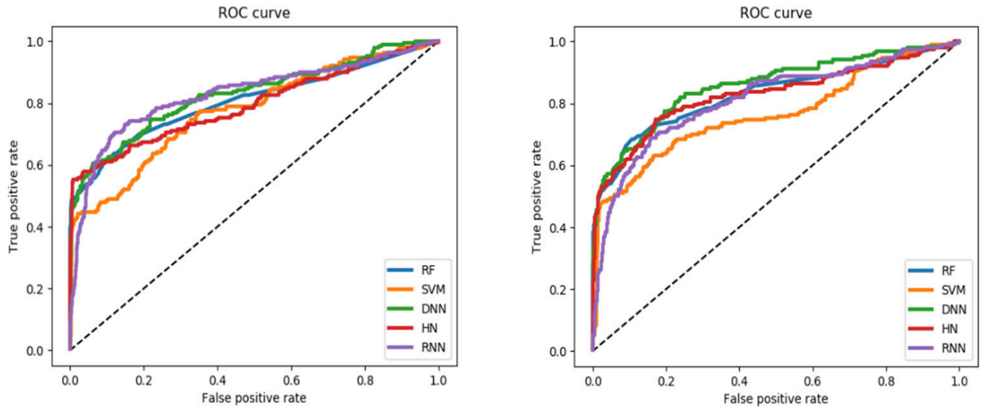


FIGURE 4 | ROC curve of descriptors-based model (the left one is test set, the right one is validation set).

TABLE 4 | 20 molecular descriptors selected by the RF method and Gini index analysis.

Name	Meaning	Bolck	Sub-block
TPC	Total path count	Walk and path counts	ID numbers
piPC09	Molecular multiple path count of order 9	Walk and path counts	Multiple path counts
PCR	Ratio of multiple path count over path count	Walk and path counts	ID numbers
ChiA_G	Average Randic-like index from geometrical matrix	3D matrix-based descriptors	Geometrical distance matrix (G)
Eig02_EA (bo)	Eigenvalue n. 2 from edge adjacency mat. weighted by bond order	Edge adjacency indices	Eigenvalues
StCH	Sum of tCH E-states	Atom-type E-state indices	E-State sums
piPC08	Molecular multiple path count of order 8	Walk and path counts	Multiple path counts
SM12_AEA (ri)	Spectral moment of order 12 from augmented edge adjacency mat. weighted by resonance integral	Edge adjacency indices	Spectral moments
SpDiam_B (m)	Spectral diameter from Burden matrix weighted by mass	2D matrix-based descriptors	Burden matrix weighted by mass (B (m))
SM13_AEA (ri)	Spectral moment of order 13 from augmented edge adjacency mat. weighted by resonance integral	Edge adjacency indices	Spectral moments
P_VSA_e_1	P_VSA-like on Sanderson electronegativity, bin 1	P_VSA-like descriptors	Sanderson electronegativity
GATS4s	Geary autocorrelation of lag 4 weighted by I-state	2D autocorrelations	Geary autocorrelations
SM02_AEA (bo)	Spectral moment of order 2 from augmented edge adjacency mat. weighted by bond order	Edge adjacency indices	Spectral moments
SM5_B (e)	Spectral moment of order 5 from Burden matrix weighted by Sanderson electronegativity	2D matrix-based descriptors	Burden matrix weighted by Sanderson electronegativity (B (e))
TDB01i	3D Topological distance based descriptors—lag 1 weighted by ionization potential	3D autocorrelations	TDB autocorrelations
Eta_betaS_A	Eta sigma average VEM coun	ETA indices	Basic descriptors
P_VSA_ppp_ar	P_VSA-like on potential pharmacophore points, ar—aromatic atoms	P_VSA-like descriptors	Potential Pharmacophore Points
SM5_B (i)	Spectral moment of order 5 from Burden matrix weighted by ionization potential	2D matrix-based descriptors	Burden matrix weighted by ionization potential (B (i))
SM4_B (v)	Spectral moment of order 4 from Burden matrix weighted by van der Waals volume	2D matrix-based descriptors	Burden matrix weighted by Van der Waals volume (B (v))
piPC02	Molecular multiple path count of order 2	Walk and path counts	Multiple path counts

of all models are shown in **Figure 4**. For the test set, the RNN exhibited highest ROC-AUC (0.829), while for the validation set, DNN gave the highest ROC-AUC value of 0.857. Compared with the previous models, our models displayed a higher ROC value and ACC values.

In general, the DNN model performed well for the external validation set predictions from the ROC-AUC metric, while the HN exhibited the higher ACC (0.908) than DNN as well as the MCC and F_1 with 0.601 and 0.625, respectively. The RF model gave higher SP (0.999) and Precision (0.986). On the contrary, the RNN method gave higher SE value (0.579) than other models.

We further analyzed what kinds of molecular properties will affect the ARE toxicity of compounds. The Gini index was applied to sort the importance of molecular descriptors. The

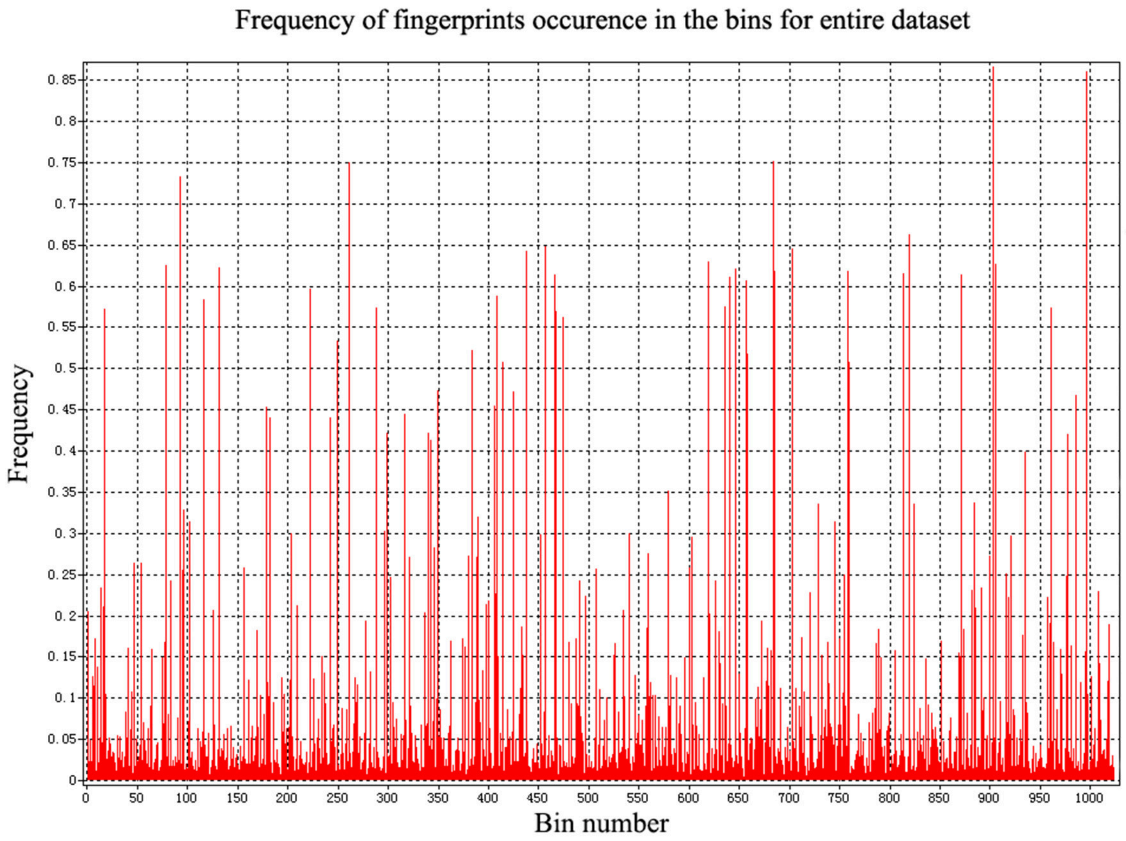


FIGURE 5 | The frequency of fingerprints occurred in compounds.

TABLE 5 | The performance of constructed models based on the fingerprints.

Methods	Group	TP	TN	FP	FN	SE	SP	MCC	F1	Precision	ACC	ROC_AUC
RF	Tr	723	4,191	8	33	0.9563	0.9981	0.9678	0.9724	0.9891	0.9917	–
	Tst	88	1,045	5	102	0.4632	0.9952	0.6269	0.6219	0.9462	0.9137	0.9613
	Val	86	1,047	3	104	0.4526	0.9971	0.6277	0.6165	0.9663	0.9137	0.9241
SVM	Tr	756	4,159	40	0	1.0000	0.9905	0.9699	0.9742	0.9497	0.9919	–
	Tst	55	1,040	10	135	0.2895	0.9905	0.4525	0.4314	0.8462	0.8831	0.8967
	Val	61	1,036	14	129	0.3211	0.9867	0.4650	0.4604	0.8133	0.8847	0.9049
DNN	Tr	725	4,190	9	31	0.9590	0.9979	0.9686	0.9732	0.9877	0.9919	–
	Tst	107	1,044	6	83	0.5632	0.9943	0.6977	0.7063	0.9469	0.9282	0.9607
	Val	106	1,046	4	84	0.5579	0.9962	0.7020	0.7067	0.9636	0.9290	0.9167
HN	Tr	743	4,172	27	13	0.9828	0.9936	0.9691	0.9738	0.9649	0.9919	–
	Tst	116	1,017	33	74	0.6105	0.9686	0.6415	0.6844	0.7785	0.9137	0.9329
	Val	119	1,021	29	71	0.6263	0.9724	0.6652	0.7041	0.8041	0.9194	0.8794
RNN	Tr	670	4,157	42	86	0.8862	0.9900	0.8982	0.9128	0.9410	0.9742	–
	Tst	106	1,026	24	84	0.5579	0.9771	0.6291	0.6625	0.8154	0.9129	0.9296
	Val	100	1,011	39	90	0.5263	0.9629	0.5585	0.6079	0.7194	0.8960	0.8534
CNN	Tr	746	4,169	30	10	0.9868	0.9929	0.9692	0.9739	0.9613	0.9919	–
	Tst	86	1,037	13	104	0.4526	0.9876	0.5851	0.5952	0.8687	0.9056	0.9329
	Val	92	1,032	18	98	0.4842	0.9829	0.5917	0.6133	0.8364	0.9065	0.8967

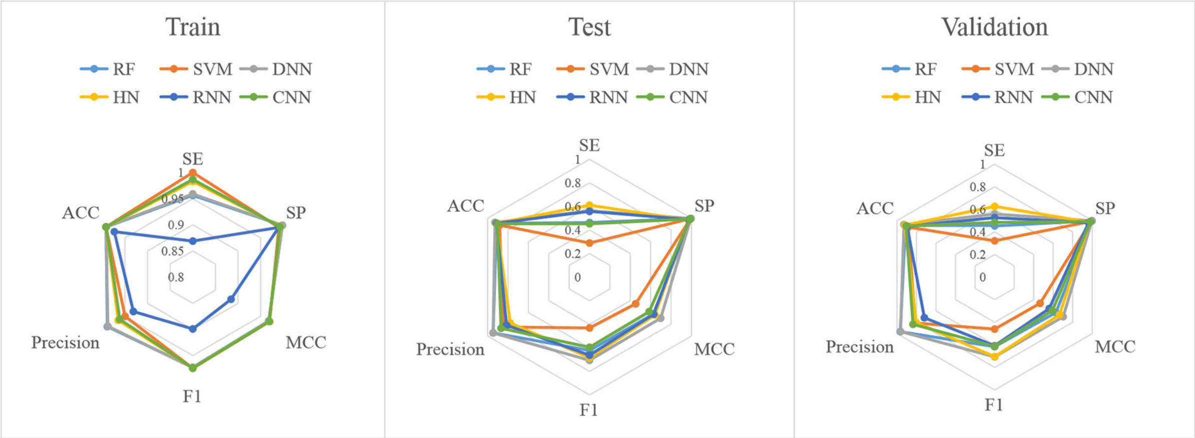


FIGURE 6 | Radar plot of the fingerprints-based classification model.

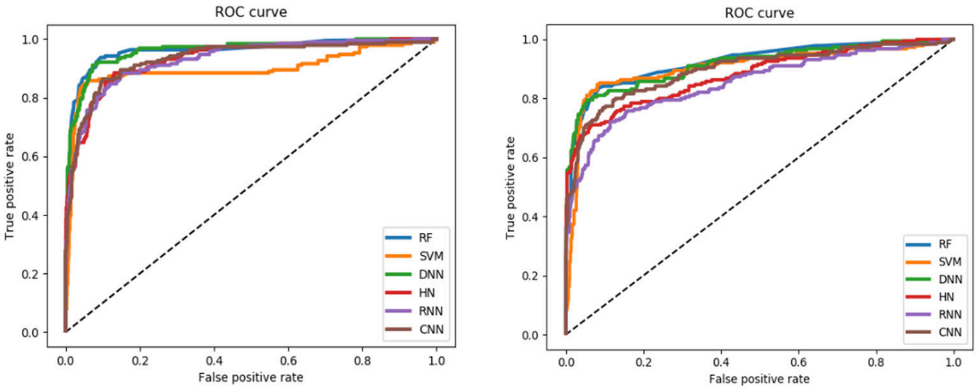


FIGURE 7 | ROC curve of fingerprints-based model (**left:** test set, **right:** validation set).

top 20 descriptors and their corresponding meanings are shown in **Table 4**. From the information of selected descriptors, it is clearly that the walk and path count descriptors hold a great impact on the ARE toxicity of compound. The 3D matrix-based descriptors, the edge adjacency indices as well as the atom-type E-state indices are also significant for the ARE toxicity of compound. Besides, the 2D matrix-based descriptors and 2D autocorrelations P_VSA-like descriptors also have a close correlation with ARE toxicity of compound.

Performance Evaluation of Fingerprints-Based Classification Models

In addition to the general molecular descriptors, the molecular fingerprint is another effective method to represent the structural features of molecules. A typical frequency of fingerprints occurred in the 1,024 bins of the compounds in the data set is shown in **Figure 5**. The fingerprints features were applied to build the six models including DNN, HN, RNN, CNN, RF, and SVM. The results are presented in **Table 5** and the radar plots are presented in **Figure 6**.

For the training set, 5 out of all 6 models performed very well, except for the RNN method. According to the prediction results for test set, the value of SP, ACC, and precision were relatively stable, while the SE, F_1-score and MCC showed different performance. The HN model exhibited the highest SE value while the SVM gave the lowest one. For the validation set, HN also performed better than other models on SE. As shown in **Figure 7**, all 6 models presented good ROC and large ROC-AUC, which were better than descriptor-based models. RF model has the highest ROC-AUC with 0.924 better than the DNN model with 0.917. However, the ACC of RF was lower than DNN model. But for the external validation set, Deep Learning methods had better generalization ability. Overall, the fingerprints-based models can give better prediction results than those based on molecular descriptors. The fingerprints of compounds were more useful than the descriptors for ARE toxicity prediction of compounds.

Compared with the traditional machine learning methods, deep learning methods had better learning ability and they could extract the inherent characteristics of the data. For the models based on the molecular descriptors, DNN showed highest ROC_AUC and ACC, while the HN exhibited the best SE performance. Considering the fingerprints features, the performance of DNN model was still well and HN showed higher SE than other models.

TABLE 6 | The reported top 10 prediction models of ARE toxicity prediction in Tox 21 challenge data set.

Methods	ROC-AUC	Balanced accuracy
DNN based on FP[a]	0.917	0.777
Bioinf@JKU	0.840	0.677
Bioinf@JKU-ensemble4	0.832	0.716
Bioinf@JKU-ensemble3	0.832	0.650
Bioinf@JKU-ensemble2	0.830	0.729
Bioinf@JKU-ensemble1	0.827	0.650
AMAZIZ	0.805	0.715
Microsomes	0.804	0.605
T	0.801	0.696
NCI	0.783	0.711
dmlab	0.768	0.519

[a] *FP means Fingerprints.*

The Comparisons Between Our Models and Other Models

We also compared the performance of our models with other reported models[5]. For the ARE toxicity prediction of Tox21 challenge data, the deep neural network models developed by Mayr et al. (2016) gave the best prediction results compared with other models. The best results they obtained had ROC-AUC 0.840, Balanced Accuracy 0.677 for the validation set. Other models displayed ROC-AUC ranging from 0.768 to 0.832 with the balanced accuracy between 0.519 and 0.729 using traditional machine learning methods, such as RF, SVM, and Naive Bayesian (shown in **Table 6**). Compared to their models and other models, our prediction models can give better prediction results. For the validation set, our best DNN model had ROC-AUC 0.917 and Accuracy 0.929.

[5] https://tripod.nih.gov/tox21/challenge/leaderboard.jsp

CONCLUSIONS

In this study, multiple deep learning algorithms were used to predict the ARE toxicity of compounds based on two kinds of molecular features including the general molecular descriptors and fingerprints. The DNN model based on fingerprints had an outstanding performance with ROC-AUC 0.917 and ACC 0.929, while the DNN model based on the general molecular descriptors had relative lower predictive ability with ROC-AUC 0.857 and ACC 0.896, suggesting that the fingerprints can represent the structural features of compounds related to their ARE toxicity more comprehensively. Compared with the traditional machine learning model, the deep learning models had much better predictive ability. Our constructed accurate predictive models on ARE toxicity will be valuable to the assessment of toxicity of compounds.

AUTHOR CONTRIBUTIONS

HL and CX conceived and designed the study. FB, DH, YL, and XY performed the experiment, analyzed the data, and wrote the manuscript.

FUNDING

This work was supported by the National Natural Science Foundation of China (Grant No. 21775060).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fchem.2019.00385/full#supplementary-material

The ID of compounds used in our model is available in the **Supplementary Material**.

REFERENCES

Abdelaziz, A., Spahn-Langguth, H., Karl-Werner, S., and Tetko I. V. (2016). Consensus modeling for HTS assays using *in silico* descriptors calculates the best balanced accuracy in Tox21 challenge. *Front. Environ. Sci.* 4, 1–12. doi: 10.3389/fenvs.2016.00002

Bengio, Y., Simard, P., and Frasconi, P. (2002). Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. Neural Netw.* 5, 157–166. doi: 10.1109/72.279181

Blomme, E. A., and Will, Y. (2016). Toxicology strategies for drug discovery: present and future. *Chem. Res. Toxicol.* 29, 473–504. doi: 10.1021/acs.chemrestox.5b00407

Breiman, L. (2001). Random forests. *Mach. Learn.* 45, 5–32. doi: 10.1023/A:1010933404324

Dahl, G. E., Jaitly, N., and Salakhutdinov, R. (2014). Multi-task neural networks for QSAR predictions. *arXiv: 1406.1231*. Available online at: https://arxiv.org/pdf/1406.1231.pdf

Dang, N. L., Hughes, T. B., Miller, G. P., and Swamidass, S. J. (2018). Computationally assessing the bioactivation of drugs by N-dealkylation. *Chem. Res. Toxicol.* 31, 68–80. doi: 10.1021/acs.chemrestox.7b00191

Drwal, M. N., Siramshetty, V. B., Banerjee, P., Goede, A., Preissner, R., and Dunkel, M. (2015). Molecular similarity-based predictions of the Tox21 screening outcome. *Front. Environ. Sci.* 3:54. doi: 10.3389/fenvs.2015.00054

Fang, J., Yang, R., Gao, L., Zhou, D., Yang, S., Liu, A. L., et al. (2013). "Predictions of BuchE inhibitors using support vector machine (SVM) and naive Bayesian classification techniques," in *The 12th Meeting of The Asia Pacific Federation of Pharmacologists* (Beijing), 3009–3020. doi: 10.1021/ci400331p

Filip, S. (2015). Prediction of compounds activity in nuclear receptor signaling and stress pathway assays using machine learning algorithms and low-dimensional molecular descriptors. *Front. Environ. Sci.* 3:77. doi: 10.3389/fenvs.2015.00077

Foloppe, N., and Chen, I. J. (2009). Conformational sampling and energetics of drug-like molecules. *Curr. Med. Chem.* 16, 3381–3413. doi: 10.2174/092986709789057680

Fraser, K., Bruckner, D. M., and Dordick, J. S. (2018). Advancing predictive hepatotoxicity at the intersection of experimental, *in silico*, and artificial intelligence technologies. *Chem. Res. Toxicol.* 31, 412–430. doi: 10.1021/acs.chemrestox.8b00054

Gawehn, E., Hiss, J. A., and Schneider, G. (2016). Deep learning in drug discovery. *Mol. Inform.* 35, 3–14. doi: 10.1002/minf.201501008

Gergo, B. (2016). Identifying biological pathway interrupting toxins using multi-tree ensembles. *Front. Environ. Sci.* 4:52. doi: 10.3389/fenvs.2016.00052

Glorot, X., and Bengio, Y. (2010). "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the 13 th International Conference on Artificial Intelligence and Statistics (AISTATS)*

(Sardinia), 249–256. Available online at: http://proceedings.mlr.press/v9/glorot10a/glorot10a.pdf?hc_location=ufi

Graves, A., Mohamed, A.-R., and Hinton G. (2013). "Speech recognition with deep recurrent neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing* (Toronto, ON: IEEE).

Gulshan, V., Peng, L., Coram, M., Stumpe, M. C., Wu, D., Narayanaswamy, A., et al. (2016). Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *JAMA* 316, 2402–2410. doi: 10.1001/jama.2016.17216

Hanson, J., Yang, Y., Paliwal, K., and Zhou, Y. (2017). Improving protein disorder prediction by deep bidirectional long short-term memory recurrent neural networks. *Bioinformatics* 33, 685–692. doi: 10.1093/bioinformatics/btw678

Hayes, J. D., and Dinkova-Kostova, A. T. (2014). The Nrf2 regulatory network provides an interface between redox and intermediary metabolism. *Trends Biochem. Sci.* 39, 199–218. doi: 10.1016/j.tibs.2014.02.002

He, K., Zhang, X., Ren, S. G., and Sun, J. (2015). "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition* (Boston, MA), 770–778. doi: 10.1109/CVPR.2016.90

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *Comput. Sci.* 3, 212–223. Available online at: https://arxiv.org/pdf/1207.0580.pdf

Hochreiter, S. (1991). *Untersuchungen zu Dynamischen Neuronalen Netzen*. [Master's Thesis], Institut Fur Informatik, Technische Universitat, Munchen.

Hochreiter, S., and Schmidhuber, J. (1997). Long short-term memory. *Neural Comput.* 9, 1735–1780. doi: 10.1162/neco.1997.9.8.1735

Hou, T.-Y., Weng, C.-F., and Leong, M. K. (2018). Insight analysis of promiscuous estrogen receptor α-ligand binding by a novel machine learning scheme. *Chem. Res. Toxicol.* 31, 799–813. doi: 10.1021/acs.chemrestox.8b00130

Huang, R., Xia, M., Nguyen, D.-T., Zhao, T., Sakamuru, S., Zhao, J., et al. (2016). Tox21 challenge to build predictive models of nuclear receptor and stress response pathways as mediated by exposure to environmental chemicals and drugs. *Front. Environ. Sci.* 3:85. doi: 10.3389/fenvs.2015.00085

Hughes, T. B., and Swamidass, S. J. (2017). Deep learning to predict the formation of quinone species in drug metabolism. *Chem. Res. Toxicol.* 30, 642–656. doi: 10.1021/acs.chemrestox.6b00385

Kaminski, G. A., Friesner, R. A., Tirado-Rives, J., and Jorgensen, W. L. (2001). Evaluation and reparametrization of the OPLS-AA force field for proteins via comparison with accurate quantum chemical calculations on peptides. *J. Phys. Chem. B* 105, 6474–6487. doi: 10.1021/jp003919d

Kennard, R. W., and Stone, L. A. (1969). Computer aided design of experiments. *Technometrics* 11, 137–148. doi: 10.1080/00401706.1969.10490666

Kode srl. (2017). *Dragon (Software for Molecular Descriptor Calculation) Version 7.0.8*. Available online at: https://chm.kode-solutions.net

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25(NIPS2012)*, eds F. Pereira, C. Burges, L. Bottou, and K. Weinberger (Lake Tahoe), 1097–1105. Available online at: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

Lecun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature* 521:436. doi: 10.1038/nature14539

Mamoshina, P., Vieira, A., Putin, E., and Zhavoronkov, A. (2016). Applications of deep learning in biomedicine. *Mol. Pharm.* 13, 1445–1454. doi: 10.1021/acs.molpharmaceut.5b00982

Mavroforakis, M. E., and Theodoridis, S. (2006). A geometric approach to Support Vector Machine (SVM) classification. *IEEE Trans. Neural Netw.* 17, 671–682. doi: 10.1109/TNN.2006.873281

Mayr, A., Klambauer, G., Unterthiner, T., and Hochreiter, S. (2016). DeepTox: toxicity prediction using deep learning. *Front. Environ. Sci.* 3:80. doi: 10.3389/fenvs.2015.00080

Ney, P. A., Sorrentino, B. P., Lowrey, C. H, and Nienhuis, A. W. (1990). Inducibility of the HS II enhancer depends on binding of an erythroid specific nuclear protein. *Nucleic Acids Res.* 18, 6011–6017. doi: 10.1093/nar/18.20.6011

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., and Thirion, B. (2011). Scikit-learn: machine learning in python. *J. Mach. Learn. Res.* 12, 2825–2830. doi: 10.1145/2786984.2786995

Raghunath, A., Sundarraj, K., Nagarajan, R., Arfuso, F., Bian, J., Kumar, A. P., et al. (2018). Antioxidant response elements: discovery, classes, regulation and potential applications. *Redox Biol.* 17, 297–314. doi: 10.1016/j.redox.2018.05.002

Rogers, D., and Hahn, M. (2010). Extended-connectivity fingerprints. *J. Chem. Inf. Model.* 50, 742–754. doi: 10.1021/ci100050t

Schrödinger. (2015). *Schrödinger Release 2015-1: LigPrep*. New York, NY: Schrödinger, LLC.

Srivastava, R. K., Greff, K., and Schmidhuber, J. (2015). Highway Network. *arXiv:1505.00387v2*. Available online at: https://arxiv.org/pdf/1505.00387v2.pdf

Tieleman, T., and Hinton, G. (2012). *Lecture 6.5-RMSProp, COURSERA: Neural Networks for Machine Learning*. Technical Report.

Williams, R. J., and Zipser, D. (1989). A learning algorithm for continually running fully recurrent neural networks. *Neural Comp.* 1, 270–280. doi: 10.1162/neco.1989.1.2.270

Wu, K., and Wei, G. W. (2018). Quantitative toxicity prediction using topology based multitask deep neural networks. *J. Chem. Inf. Model.* 58, 520–531. doi: 10.1021/acs.jcim.7b00558

Xu, Y., Dai, Z., Chen, F., Gao, S., Pei, J., and Lai, L. (2015). Deep learning for drug-induced liver injury. *J. Chem. Inf. Model.* 55, 2085–2093. doi: 10.1021/acs.jcim.5b00238

Xu, Y., Pei, J., and Lai, L. (2017). Deep learning based regression and multiclass models for acute oral toxicity prediction with automatic chemical feature extraction. *J. Chem. Inf. Model.* 57, 2672–2685. doi: 10.1021/acs.jcim.7b00244

Yoshihiro, U. (2016). Rigorous selection of random forest models for identifying compounds that activate toxicity-related pathways. *Front. Environ. Sci.* 4:9. doi: 10.3389/fenvs.2016.00009

Zeiler, M. D., and Fergus, R. (2014). "Visualizing and understanding convolutional networks," in *European Conference on Computer Vision* (Cham: Springer), 818–833. doi: 10.1007/978-3-319-10590-1_53

Conflict of Interest Statement: 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.

frontiers in Chemistry

ORIGINAL RESEARCH
published: 25 November 2019
doi: 10.3389/fchem.2019.00779

Development of Predictive Models for Identifying Potential S100A9 Inhibitors Based on Machine Learning Methods

Jihyeun Lee[1†], *Surendra Kumar*[1†], *Sang-Yoon Lee*[2], *Sung Jean Park*[1] and *Mi-hyun Kim*[1*]

[1] Department of Pharmacy, Gachon Institute of Pharmaceutical Science, College of Pharmacy, Gachon University, Incheon, South Korea, [2] Gachon Advanced Institute for Health Science and Technology, Graduate School and Neuroscience Research Institute, Gachon University, Incheon, South Korea

S100A9 is a potential therapeutic target for various disease including prostate cancer, colorectal cancer, and Alzheimer's disease. However, the sparsity of atomic level data, such as protein-protein interaction of S100A9 with RAGE, TLR4/MD2, or CD147 (EMMPRIN) hinders the rational drug design of S100A9 inhibitors. Herein we first report predictive models of S100A9 inhibitory effect by applying machine learning classifiers on 2D-molecular descriptors. The models were optimized through feature selectors as well as classifiers to produce the top eight random forest models with robust predictability and high cost-effectiveness. Notably, optimal feature sets were obtained after the reduction of 2,798 features into dozens of features with the chopping of fingerprint bits. Moreover, the high efficiency of compact feature sets allowed us to further screen a large-scale dataset (over 6,000,000 compounds) within a week. Through a consensus vote of the top models, 46 hits (hit rate = 0.000713%) were identified as potential S100A9 inhibitors. We expect that our models will facilitate the drug discovery process by providing high predictive power as well as cost-reduction ability and give insights into designing novel drugs targeting S100A9.

Keywords: S100, machine learning, random forest, ligand-based virtual screening, feature selection, classification, consensus vote, Alzheimer's disease

OPEN ACCESS

Edited by:
John C. Cancilla,
Scintillon Institute, United States

Reviewed by:
Carles Curutchet,
University of Barcelona, Spain
Thomas Vogl,
Universitätsklinikum
Münster, Germany

***Correspondence:**
Mi-hyun Kim
kmh0515@gachon.ac.kr

†These authors have contributed equally to this work and share first authorship

Specialty section:
This article was submitted to Theoretical and Computational Chemistry,
a section of the journal Frontiers in Chemistry

Received: *10 February 2019*
Accepted: *29 October 2019*
Published: *25 November 2019*

Citation:
Lee J, Kumar S, Lee S-Y, Park SJ and Kim M (2019) Development of Predictive Models for Identifying Potential S100A9 Inhibitors Based on Machine Learning Methods.
Front. Chem. 7:779.
doi: 10.3389/fchem.2019.00779

INTRODUCTION

Drug R&D is currently facing a productivity crisis to overcome low productivity as well as high risk/high return in the context of economics (Scannell et al., 2012; Mullard, 2014; Mignani et al., 2016; Bendtsen et al., 2017). In order to develop an efficient and cost-effective R&D process (Bendtsen et al., 2017), computing and simulations have decreased the traditional resource demand for drug R&D (Kapetanovic, 2008; Bendtsen et al., 2017). In particular, an early stage of drug discovery involves virtual screening (VS) to identify therapeutic targets or hit compounds (Walters et al., 1998; Bajorath, 2002; Oprea and Matter, 2004; Shoichet, 2004). Successful VS depends on the predictive power of predictors and the quality of the virtual library and dataset used. When the 3D-structure of a molecular target is available, structure-based virtual screening (SBVS) is considered prior to ligand-based virtual screening (LBVS) or SBVS/LBVS in combination due to an easy understanding of the predictive (atomic level) model and empirical evidence on bioactive conformation as well as the activity resulting from interaction between a target and a compound

(Lavecchia and Di Giovanni, 2013; Sliwoski et al., 2014; Gadhe et al., 2015; Lavecchia, 2015; Jang et al., 2018; Lee et al., 2018; Yadav et al., 2018). Recently, the conceptual advance of drug targeting from "single target" to "protein-protein interactions (PPI)," it is unsatisfactory to obtain atomic level confidence of a novel druggable target with only partial structural information. It is therefore very difficult for researchers to propose a druggable binding site of a new molecular target for drug design without the background science or evidence. Therefore, when promising drug targets have insufficient information or when multiple targets need to be considered together, LBVS is commonly used, where the known active small molecules are used as screening templates. With improvement in the volume, quality, velocity, and accessibility of molecular data, versatile machine learning (ML) algorithms like support vector machine (SVM) (Cortes and Vapnik, 1995), Naïve Bayes (NB) (Domingos and Pazzani, 1997), decision tree (DT) (Breiman, 2017), and ensemble methods, such as random forest (RF) (Breiman, 2001) have contributed to the improvement of LBVS predictors (Geppert et al., 2010; Lo et al., 2018). With these advances, we can expect a diversity of training data like the heterogeneous property of activity index (or assay methods) and structural diversity beyond the congenericity of active compounds. In the case of classification models, selection methods for molecular descriptors (selectors) as well as classification algorithms (classifiers) decide the predictive power and coverage of models (Melville et al., 2009). Therefore, it is natural that multiple trials on various combinations of learning methods and feature sets coupled with raw dataset can facilitate the best performance of classifiers (Stahura and Bajorath, 2005; Domingos, 2012).

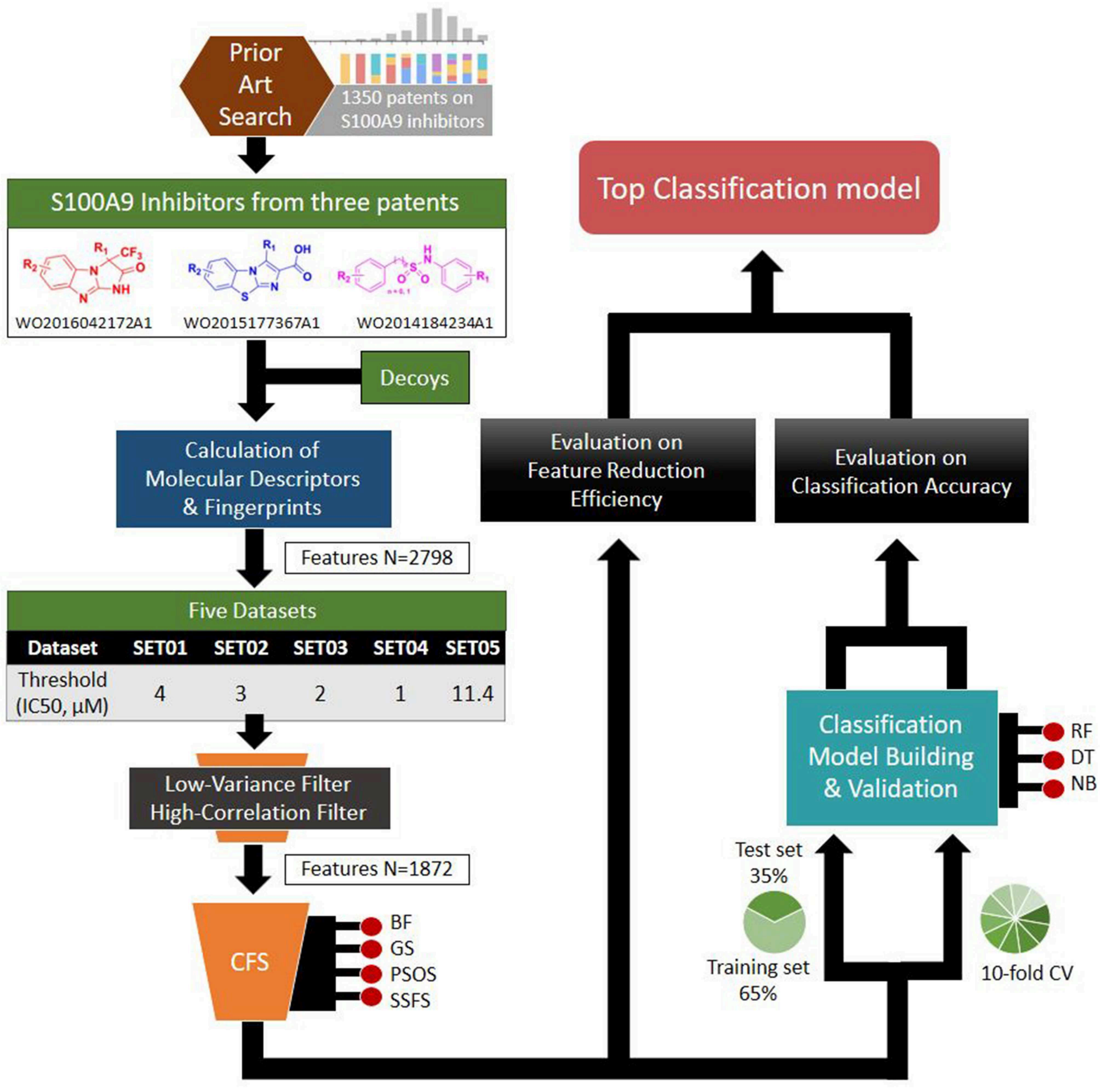
The S100 protein family is one of the challengeable drug target candidates (Donato, 2001; Ryckman et al., 2003). They are low molecular weight (ca. 100 amino acids) proteins with high similarity within the subfamily, and comprise two metal-binding EF-hands and a hinge. Due to their biophysical properties, they tend to form protein complexes (e.g., heterodimer like S100A8/S100A9, homodimer like S100B/S100B Donato, 1999), ligand-protein complex like S100A/RAGE complex (Yatime et al., 2016) rather than remaining as a single protein in a cell. Therefore, in the spite of many biological and pathological studies on several S100A9-mediated diseases, such as prostate cancer (Hermani et al., 2005), colorectal cancer (Kim et al., 2009), Alzheimer's disease (Horvath et al., 2015), and other neurodegenerative disorders (Gruden et al., 2017; Iashchishyn et al., 2018), atomic level knowledge is limited for SBVS or structure-based drug design of S100A9 inhibitors. Notably, the characterization of S100A9 complex has been updated, such as the hydrophobic binding of V-RAGE domain into S100A9 homodimer (Chang et al., 2016), V-RAGE domain into S100A9/S100A12 heterodimer (Katte and Yu, 2018) following the first X-ray report (Itou et al., 2002). However, the small molecule, *CHAPS* of the reports is a detergent (for protein stabilization or solubilizing) rather than a drug inducing functional change of S100A9. In addition, the SPR measurement of Q-compounds recently produces the question, whether the inhibition of Q-compounds is non-specific or specific (Björk et al., 2009; Yoshioka et al., 2016; Pelletier et al., 2018). Therefore, a ligand-based model can is required to compensate current insufficient characterization for targeting S100A9. For the purpose, maximum collection of the available data and selection of the most relevant features should be considered. Very delightfully, competitive inhibitors binding to S100A9 in the presence of the target receptors, such as RAGE, TLR4/MD2, and EMMPRIN (CD147) were reported in three patents (Fritzson et al., 2014; Wellmar et al., 2015, 2016). However, the patents proposed neither a druggable binding site nor different interaction mode between the target receptors. In other words, despite the presence of the inhibitors, no reliable predictive model has been reported to identify novel S100A9 inhibitors.

Based on the S100A9 competitive inhibitors of the patents, we present herein, the first predictive models using multi-scaffolds of competitive inhibitors (binding to the complex of S100A9 with rhRAGE/Fc, TLR4/MD2, or rhCD147/Fc) as a training set. For the purpose, highly efficient feature sets was considered in this study. Even though the input data matrix consisting of a low number of rows (data points/compounds) and a large number of columns (features) is never special in 2D/3D-QSAR or classification models built from limited and insufficient biological data (Guyon and Elisseeff, 2003; Muegge and Oloff, 2006), data processing (filtering, suitability, scaling) and feature selection were considered to remove irrelevant and redundant data (Liu, 2004; Yu and Liu, 2004). Adding a few other features to a sufficient number of features often leads to an exponential increase in prediction time and expense (Koller and Sahami, 1996; Liu and Yu, 2005), and whenever a large screening library is generated, feature generation of the library can be a practical burden. Further, because more irrelevant features hinder classifiers from identifying a correct classifying function (Dash and Liu, 1997), the feature optimization process is essential to increase the learning accuracy of the classifier and to escape the curse of dimensionality that emerge in a consequence of high dimensionality (Bellman, 1966). In addition, versatile machine learning models were built resulting from 5 × 4 × 3 trials: (1) five IC_{50} thresholds between activeness and inactiveness, (2) four feature selectors, and (3) three classifiers, thereby resulting in comprehensive validation of 60 models. The overall workflow depicted in **Figure 1** was designed to select the optimal classification models with the best predictive ability and efficiency. In particular, we tried to gain a golden triangle between cost-effectiveness, speed, and accuracy. For this purpose, compact feature selection was critical for more than six million library screening showing the original data matrix of six million compounds (rows) × ca. 3,000 features (columns).

ALGORITHMS AND METHODS

Datasets

Through patent searching, S100 inhibitors and their respective IC50 values were collected from three different patents. In the patents, even though the inhibitory effect on every complex (the binding complex of S100A9 with hRAGE/Fc, TLR4/MD2, or hCD147/Fc) was measured through the change of resonance units (RU) in surface plasmon resonance (SPR) (Fritzson et al., 2014), IC50 was calculated through the AlphaScreen assay of


Prior Art Search
1350 patents on S100A9 inhibitors
S100A9 Inhibitors from three patents
WO2016042172A1
WO2015177367A1
WO2014184234A1
Decoys
Calculation of Molecular Descriptors & Fingerprints
Features N=2798
Five Datasets
Low-Variance Filter High-Correlation Filter
Features N=1872
CFS
BF
GS
PSOS
SSFS
Top Classification model
Evaluation on Feature Reduction Efficiency
Evaluation on Classification Accuracy
Classification Model Building & Validation
RF
DT
NB
Test set 35%
Training set 65%
10-fold CV

Dataset	SET01	SET02	SET03	SET04	SET05
Threshold (IC50, μM)	4	3	2	1	11.4

FIGURE 1 | Workflow depicting the process of the top classification model development.

several concentrations in only biotinylated hS100A9 complex with rhRAGE-Fc (Fritzson et al., 2014; Wellmar et al., 2015, 2016). Therefore, the predicted inhibitory effect of our model means competitive inhibition of S100A9-RAGE in this study. The assay method for IC50 was identical in the three patents. The total number of molecules collected was 266: 115 compounds from WO2011184234A1, 97 compounds from WO2011177367A1, and 54 compounds from WO2012042172A1. The three distinct scaffolds led to the structural diversity of the dataset which was confirmed through the principal component analysis (PCA) of patent molecules (**Figure 2**). To investigate a more reasonable decision boundary between the activity and inactivity of the inhibitory effect on S100A9, five datasets (SET01, SET02, SET03, SET04, and SET05) were generated with different thresholds of activity (respectively 4, 3, 2, 1, and 11.4 μM of IC50). Insufficient numbers of inactive molecules were compensated by decoys from the DUD-E database (Mysinger et al., 2012), in order to obtain the same size for each dataset ($N = 402$), with a ratio of 66.17% ($N = 266$) patent molecules and 33.83% ($N = 136$) inactive decoy

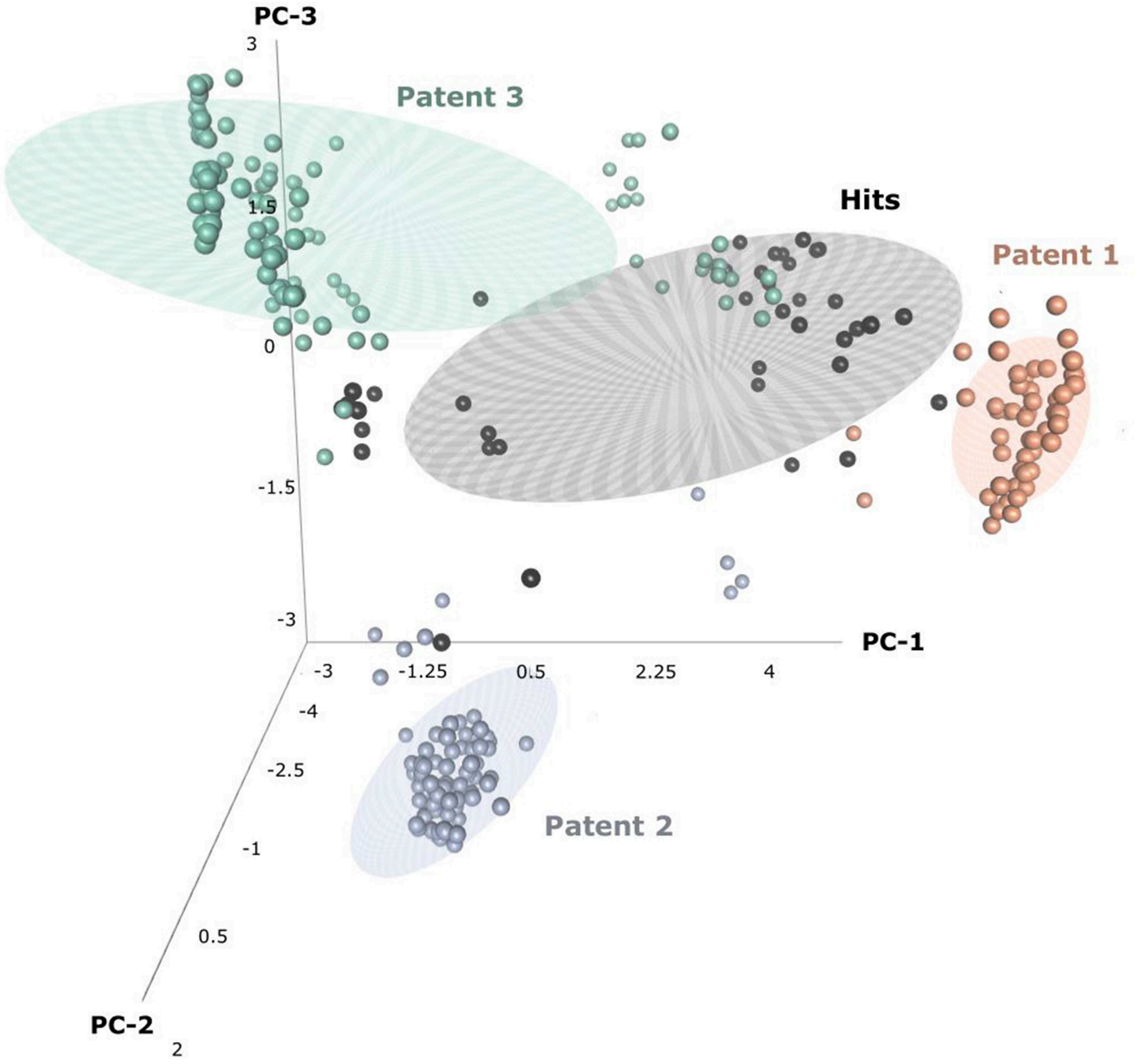


FIGURE 2 | Three-dimensional principal component analysis(PCA) of hits and patent molecules. Patent 1, Patent 2, and Patent 3 refers to WO2015177367A1, WO2014184234A1, and WO2016042172A1.

molecules (see **Table S1** and **Datasheet 1** in Supplementary Materials for SMILES information of the dataset). The activity property was converted to a binominal value according to the threshold of each set for a dichotomous classification. In particular, the activity threshold of 11.4 μM in SET05 is the highest IC50 value among patent molecules, thus making every patent molecule active, and every decoy molecule inactive in SET05.

Descriptor and Fingerprint Calculation

Useful descriptors provide a better understanding of the molecules, and are widely used to construct models to predict certain molecular properties (Glover and Kochenberger, 2006). In our study, 2,798 features were generated using PaDEL-Descriptor ver. 2.21 (PaDEL-Descriptor, Pharmaceutical Data Exploration Laboratory) (Yap, 2011). All kinds of 1D and 2D descriptors were calculated to produce 1,444 features. The remaining features are from three kinds of fingerprints: MACCSFP, 166 bits; PubChemFP, 881 bits; SubstructureFP, 307 bits.

Dimensionality Reduction

To avoid the curse of dimensionality and to enhance the efficiency of the overall predicting process, we applied several

TABLE 1 | The number of features.

	Initial features	Low-variance filter	Low-variance filter and high-correlation filter
1D/2D descriptors	1,444	1,218	1,017
Fingerprints	1,354	855	855
MACCSFP	166	147	147
PubChemFP	881	598	598
SubstructureFP	307	110	110
Total	2,798	2,073	1,872

The numbers of descriptors and bits of fingerprints generated initially, and the selected numbers of features after the removal of unnecessary features by certain filtration methods are listed. Note that each bit of a fingerprint was considered as a single feature.

strategies to greatly reduce the number of features. Notably, each bit of fingerprint was considered as a single feature in our study; thus, the optimal feature set comprises hybrid fingerprints and descriptors after the feature reduction process. By removing irrelevant bits from the original intact fingerprint, a hybrid fingerprint can achieve increased prediction accuracy as well as reduced computational cost (Williams, 2006; Nisius and Bajorath, 2009, 2010; Singla et al., 2013; Smieja and Warszycki, 2016; Warszycki et al., 2017).

Low-Variance Filter and High-Correlation Filter

In our pre-processing step, we applied two steps of filtering: the low-variance filter and the high-correlation filter. First, to avoid redundancy, features with low variance were removed after normalization. Among 2,797 features, 724 columns with zero variance were removed (**Table 1**) to obtain a small feature set without reducing the prediction performance. Second, the correlation between two random variables was ranked to obtain Kendall's Tau-a coefficient matrix. Features with strong dependency ($\tau > 0.9$) were removed to ensure maximum dissimilarity between features (Ding and Peng, 2005). Here, 201 columns were removed, leaving 1,872 independent features that were de-normalized for further processing (**Table 1**).

Correlation-Based Feature Subset Selection

In addition to the correlation filter, we used a correlation-based feature subset selection method (Hall, 1999) to obtain a compact number of features. Merit, composed of Pearson's correlation formula, is used to evaluate the correlation-based feature selection (CFS) algorithm. To determine subsets containing features that are highly correlated with the class but are uncorrelated with each other, the following merit is calculated along a search:

$$Merit_s = \frac{k\overline{r_{cf}}}{\sqrt{k + k(k-1)\overline{r_{ff}}}} \tag{1}$$

where $Merit_s$ is the heuristic merit of subset S containing k features, $\overline{r_{cf}}$ is the average correlation with the class, and $\overline{r_{ff}}$ is the average inter-correlation. The subset with the highest merit is selected to obtain features with high predictive ability and low redundancy. Various search algorithms are applicable for improving the efficiency of feature selection (FS) methods. Herein we applied four different search algorithms: best first, genetic search, particle swarm optimization search, and subset size forward selection. To assess the effectiveness of the FS methods, two measurable indexes were selected: the rate of feature reduction and the merit of the best subset found. All calculations were performed in Weka software packages (Weka Environment for Knowledge Analysis ver. 3.6, The University of Waikato, Hamilton, New Zealand) (Hall et al., 2009).

Best First (BF)

Best first search is one of general algorithms for exploiting heuristic information to reduce search times. The general strategy assesses the merit of every candidate feature set exposed during the search, and then continues exploration along the direction of the highest merit (Kohavi and John, 1997). In our study, the search was terminated when an improved node was not found in the last 5 expansions. Also, backtracking was applied to reduce the size of the search space and to allow the algorithm to move toward a more promising subset (Freuder, 1988). Because the running times for the backward search starting from the full set of features could render the approach infeasible, especially if there are many features, forward selection was applied here to achieve cost-effectiveness.

Genetic Search (GS)

The genetic algorithm was first introduced by John Holland (Holland, 1992), and David Goldberg presented an application in 1989 (Goldberg, 1989) that triggered a wide variety of modifications and developments to genetic algorithms (Glover and Kochenberger, 2006). Genetic algorithms derive their name from the fact that they are inspired by the mechanism of natural selection, where the fittest individuals survive to the following generations (Man et al., 1996). Although the search method using genetic theory may result in higher computational costs than other methods, such as best first, it remains popular, because it is relatively insensitive to noise and is well-suited for problems where little knowledge is provided (Vafaie and De Jong, 1992). In this study, the total number of generations was 20, with 20 feature subsets in each generation. The probability of crossover and the mutation rate were set to 0.6 and 0.33, respectively.

Particle Swarm Optimization Search (PSO)

Particle swarm optimization (PSO), suggested by Kennedy and Eberhart in 1995 (Eberhart and Kennedy, 1995), is based on social-psychological principles. Because only a few lines of code and primitive mathematical operators are required, this method has been proved to be highly efficient for application to numerous areas (Shi, 2001). Herein we utilized the geometric particle swarm optimization (GPSO) (Moraglio et al., 2007), where a convex combination was applied to update the positions of particles. In GPSO, three convex weights w_1, w_2, and w_3 are employed, where $w_1, w_2, w_3 > 0$ and $w_1 + w_2 + w_3 = 1$. The function of GPSO can be defined as:

$$x_i = CX\left((x_i, w_1), (\hat{g}, w_2), (\hat{x}_i, w_3)\right)$$

where $\hat{g}$ is the global optimum and $\hat{x_i}$ is the local optimum. Each convex weight represents the inertia weight (w_1), social weight (w_2), and individual weight (w_3), which were set to 0.33, 0.33, and 0.34, respectively in our study. The number of particles in the search space and the number of populations in each generation were both set to 20.

Subset Size Forward Selection (SSFS)

Subset size forward selection (SSFS) is an extension of linear forward selection. Through this method, a compact feature set can be obtained from large-scale features with a relatively small number of instances (Gutlein et al., 2009). The optimal size was determined through 5-fold cross-validation with fixed-set linear forward selection, resulting in a reduced error compared to searching in a single training and test set. The number of top-ranked features forming a search space was set to 50.

Machine Learning Classifiers

After selecting the optimal feature sets, three different classifiers (decision tee, random forest, and naïve Bayes) were applied to develop and determine the best classification model for S100 inhibitors. All ML processes and calculations were performed using the KNIME software.

Decision Tree (DT)

The decision tree classifier is a simple and widely comprehensive method that can be constructed relatively quickly compared to other well-known classifiers (Kotsiantis et al., 2007). The scalable parallelizable induction of decision trees (SPRINT) (Shafer et al., 1996), a modified form of the well-known C4.5 (Quinlan, 2014), was applied in this study so that the model can take a large-scale database as an input. The Gini index was measured to determine the root node, which is the best feature that divides the dataset. To avoid overfitting problems, we applied both pre-pruning and post-pruning strategies. For post-pruning process, minimum descriptor length (MDL) pruning was applied here (Rissanen, 1978).

Random Forest (RF)

Random forest (Breiman, 2001) was developed by introducing bootstrap aggregating to decision tree. Trees are built with randomly sampled features to form a forest, and the most voted tree is selected as the optimal classifier. This ensemble learning method can handle high-dimensional data with numerous features. In addition, it is less susceptible to noise and builds a robust model, often outperforming other classifiers (Verikas et al., 2011; Khuri et al., 2017). In this study, features were evaluated based on the information gain ratio to obtain the best splits.

Naïve Bayes

Along with decision trees, naïve Bayes is one of the most popular machine learning methods for classification models. Unlike the canonical Bayesian method, naïve Bayes assumes that all features are independent of each other. Although this "naïve" assumption rarely fits in practice, it has been verified to perform reasonably well in various situations, without the requirement of independence between features (Domingos and Pazzani, 1997). This occurs because the strong false assumption may lead to reduced overfitting (Domingos, 2012). Another advantage of this method is its simplicity and low computational cost (Kotsiantis et al., 2007), which allows one to search in very large databases with high efficiency.

TABLE 2 | The optimized parameters for random forest models and the AUC of ROC values of test set prediction.

Applied feature selector	Dataset	maxDepth[a]	numTrees[b]	AUC of ROC
BF	SET01	5	52	0.971
	SET02	6	42	0.961
	SET03	10	215	0.956
	SET04	10	203	0.912
	SET05	2	15	1
GS	SET01	3	82	0.932
	SET02	10	164	0.935
	SET03	10	112	0.948
	SET04	6	105	0.867
	SET05	3	36	1
PSOS	SET01	8	76	0.952
	SET02	5	45	0.915
	SET03	9	185	0.952
	SET04	10	84	0.882
	SET05	4	13	1
SSFS	SET01	7	97	0.967
	SET02	6	59	0.966
	SET03	9	236	0.963
	SET04	8	245	0.896
	SET05	2	50	1
None	SET01	5	25	0.954
	SET02	7	96	0.941
	SET03	7	124	0.949
	SET04	7	60	0.872
	SET05	5	79	1

Parameter Optimization

To achieve the best performance, several parameters were optimized prior to the development of predictive models. For each optimization, 10-fold cross-validation was performed with the training set (65% of the original dataset), where the optimal parameters exhibiting the largest area under the curve (AUC) of receiver operating characteristics (ROC) curves were exported to construct the model. The optimized parameters and AUC values are listed in **Table 2**.

Model Validation

To assess the prediction performance of the models, two validation methods were employed: (i) evaluation by test set (35% of the original dataset) and (ii) 10-fold cross-validation of the training set. The AUC of ROC curve and the Matthews correlation coefficient (MCC) were calculated to obtain the top models among every combination of feature selectors and ML

TABLE 3 | The number of molecules from eMolecules database in each subset.

Subset	Number of molecules
Subset01–subset09	100,000
Subset10–subset18	200,000
Subset19–subset32	250,000
Subset33	247,184
Total	6,447,184

TABLE 4 | The number of selected features after each FS method.

	BF	GS	PSO	SSFS
SET05	51	852	591	47
SET01	37	940	552	29
SET02	50	751	602	23
SET03	66	741	600	24
SET04	70	667	610	28

classifiers. The MCC value can be defined as:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)\,(TP + FN)\,(TN + FP)\,(TN + FN)}},$$

where TP, TN, FP, and FN refers to true positive, true negative, false positive, and false negative.

In silico Screening of the eMolecules Database

Over past decade, large drug discovery companies have been actively applying high-throughput screening (HTS) to search potent hit molecules (Stahura and Bajorath, 2004; Reddy et al., 2007). However, HTS often demands prior validation and preparation time as well as great expense and facilities. To aid or complement HTS, the VS method should have the ability to select only a small number of potent molecules from a huge database. Thus, the top models that we have chosen previously were further evaluated by *in silico* screening of a large-scale dataset ($N = 6{,}447{,}184$) from the eMolecules database (http://www.emolecules.com/).

Screening Library

eMolecules provides almost eight million unique compound structures along with the information of vendors of the respective molecules assembled from more than 150 suppliers and manufacturers (Williams, 2008). Many studies have successfully discovered potential hits by screening molecules from this database (Bisignano et al., 2015; Lenselink et al., 2016; Shehata et al., 2016). In our study, 6,447,184 molecules were collected for screening, and split into 33 subsets in order to reduce computing (memory) burdens (**Table 3**). Features of each subset molecules could be generated based on the optimal features of the top models. First, the upper class of necessary descriptors were calculated, because only the upper class of descriptors can be selected rather than each single feature in PaDEL-Descriptor. Then, using KNIME software, the features needed were chosen to generate the exact same kind of feature set, which was used in top model building.

Prediction and Identification of Hits

Each subset with each respective feature set was then used as an input to the random forest predictor, which was built through the learning of patent molecules. Then, the predictor was used to assign possibility as S100A9 inhibitors among the screened molecules. Only molecules with a higher probability than 0.9 of being active than were selected. Overlapped molecules from the consensus of eight top models were collected to obtain the final hits.

Prediction of ADME Properties

Since poor pharmacokinetic profiles and high potential of toxicity are one of the main reasons of failure in drug development, it is crucial to consider such absorption, distribution, metabolism, excretion (ADME) properties in advance to encourage further assays and clinical trials of final hits. Thus, we predicted several drug-likeness and ADME properties of hit molecules using the QikProp module of Maestro 11.4 (Schrodinger Release 2017-4: QikProp, Schrödinger, LLC, New York, NY, 2017). QikProp computes pharmaceutically relevant properties of molecules to help eliminate those with unsatisfactory ADMET profiles. Here, we generated computational properties to ensure the drug-likeness of hits, including molecular weight (MW), LogP, hydrogen bond donor, hydrogen bond acceptor, number of N and O, polar surface area (PSA), and violation of Lipinski's rule of five as well as Jorgensen's rule of three. Also, the apparent Caco-2 cell permeability and MDCK cell permeability was also calculated to investigate intestinal absorption and oral absorption abilities.

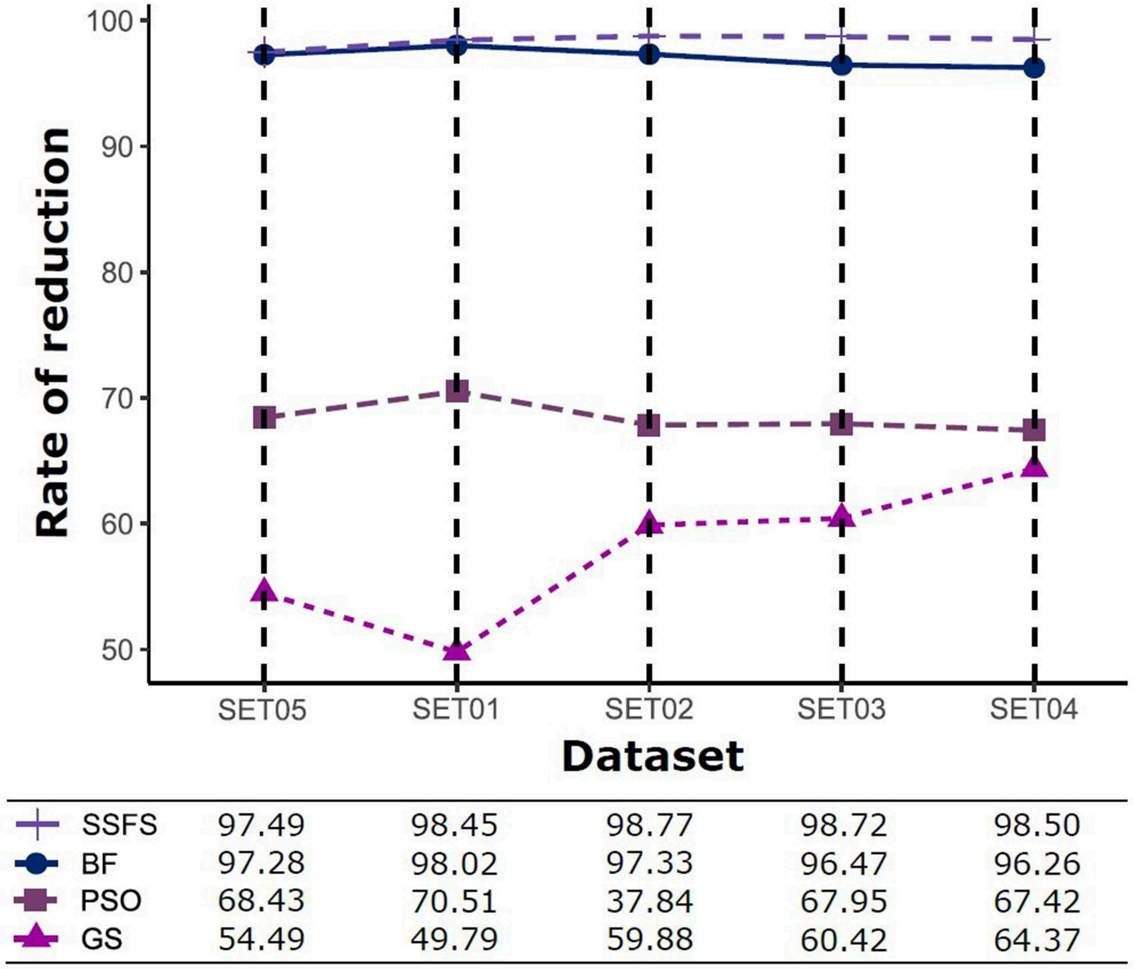
RESULTS AND DISCUSSION

Reasonable Compression of Features for Predictive Models

In order to compare the performances before and after FS, we could consider predictive power and cost-effectiveness. The efficiency of each feature selection method was evaluated by calculating two measurements: the rate of feature reduction, and the merit.

Feature Reduction

Feature reduction can play an important role in model building due to its ability to greatly reduce computational burden and to increase classification accuracy. Herein the cost-reducing effect of each FS method was evaluated through feature reduction ability. After two serial filtrations which removed 926 features from 2,798 original features, we applied CFS with four different search methods to further obtain a compact and optimal feature sets. The reduction ability of each FS method was evaluated and compared to determine optimal approaches. The selected number of features after each FS method is presented in **Table 4**. The rates of feature reduction are also shown in **Figure 3**, which are the number of excluded features divided by the number of features before CFS.



	SET05	SET01	SET02	SET03	SET04
+ SSFS	97.49	98.45	98.77	98.72	98.50
● BF	97.28	98.02	97.33	96.47	96.26
■ PSO	68.43	70.51	37.84	67.95	67.42
▲ GS	54.49	49.79	59.88	60.42	64.37

FIGURE 3 | The rates of feature reduction. The reduction rate is the ratio between the number of features removed after FS method and the original number of features before FS method.

As shown in **Table 4** and **Figure 3**, BF and SSFS excluded most of the features with over 96% removal in all five datasets. However, a relatively high number of features remained after GS or PSO, and especially, GS showed the least consistency between subsets (49.786%∼64.37%). When comparing between BF and SSFS, the actual number of features is less through SSFS than BF, yet the rates of reduction are similar. In SET03, the number of features remaining after SSFS was 36.36% ($N = 24$) of that after BF ($N = 66$). Thus, SSFS is expected to achieve the greatest effectiveness regarding cost reduction, and since the number of features selected is also small enough in BF, it is also expected to have a high efficiency similar to SSFS. The composition of each feature set is shown in **Figure 4**. See **Table S2** in Supplementary Materials for detailed information of the selected features. Due to the large number of original features, autocorrelation (e.g., ATS, AATS, ATSC), Pubchem fingerprint (e.g., PubchemFPxxx), and atom type electrotopological state (e.g., SpMax1_Bhm) could also show the highest relative frequency ratio among 63 descriptor types of 2,798 original features. In addition, with the three type descriptors, burden modified eigenvalues, molecular linear free energy relation, path count, MACCS fingerprint, and substructure fingerprint were commonly chosen through four FS methods. Because fragmented fingerprints and burden modified eigenvalues have relatively large number of original features (96–489 features), molecular linear free energy relation (with 6 features) and path count (with 22 features) are more impact per feature than other descriptors but the descriptors could not exist in every subset (5 subsets × 4 FS method).

Merit

The predictive performance of a model strongly depends on the usefulness of the features. After feature selection, the remaining features may not fully represent the original features. Therefore, the merit of a feature set is measured as shown in **Figure 5** to determine which FS method produce the best discriminative ability for model building. Despite this ability, the merit value itself does not consider the size of the dataset and a standard of a "high enough" merit value cannot be defined. Only a comparison

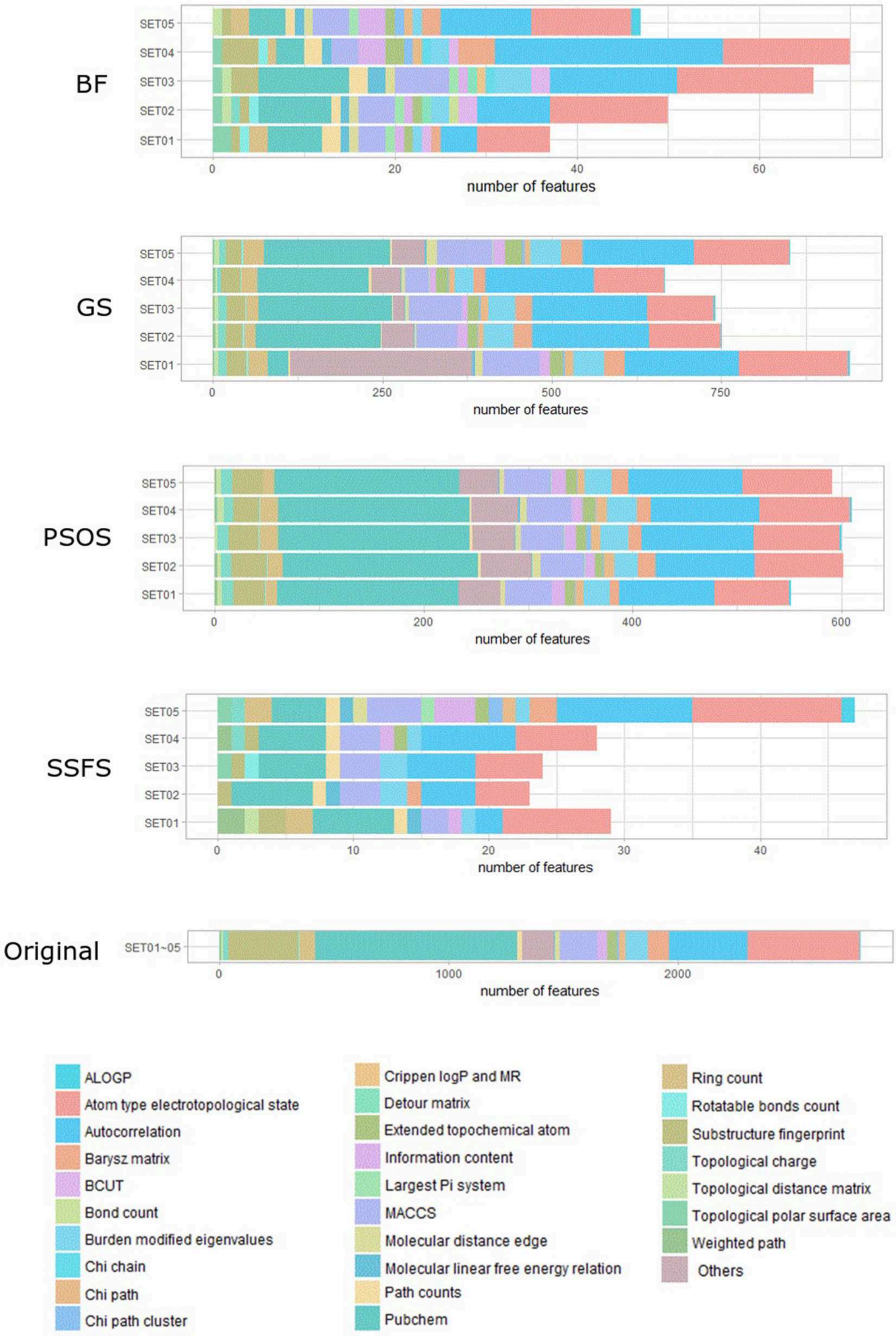


FIGURE 4 | The composition of each feature set. The number of each kind of descriptor and fingerprint bit after each FS method is shown here. SET0N refers to the different IC50 threshold (SET01:4 µM; SET02:3 µM; SET03:2 µM; SET04:1 µM; SET05:11.4 µM). Note that the maximum value of horizontal axis of the graph differs between each FS method.

between methods with the same dataset is valid and therefore, as described later, we further examined the effects on classification accuracy. A general observation is that the merit improved with an increase in the activity threshold. When every compound from a same resource is classified into the same class, it seems that the merit value tends to be enhanced, as shown in **Figure 5**, where the merit was the highest in SET05 among all datasets. The merits of BF and SSFS were higher than those of GS and PSO in every dataset, although they decreased rapidly (0.917–0.395 and 0.903–0.31, respectively) as the range of activity narrowed. GS and PSO selected feature sets with relatively poor merits, lower than 0.3 in every dataset, and almost near to zero in SET04. The results indicate that BF and SSFS achieve efficiency as well as enhance the predictive ability of the model, whereas GS and PSO barely improve the prediction ability.

Evaluation of Classification Performance

To assess the performance of the classification models, two validation approaches, external validation using the test set (35% of initial dataset) and internal 10-fold cross-validation, were used to acquire the AUC of ROC curve and MCC. Effectiveness of 5 × 3 × 4 models: (1) five type activity thresholds between activeness and inactiveness, (2) three FS methods (selectors), and (3) four ML methods (classifiers) were evaluated. The 60 models were also compared with the models without a CFS process as control groups to evaluate the effectiveness of FS on the classification performance. Mean values of measurements in each dataset were calculated to better focus on the comparison between FS methods. The control group, where FS was not treated, is labeled as "none." Every dataset used in all models contain identical molecules but differently assigned activity.

AUC of ROC

The AUC values of ROC curve of each model are illustrated in **Figure 6**. Generally, AUC declined as the IC50 threshold narrowed. Nevertheless, the RF models produced the highest AUC values in all combinations of activity thresholds and the FS methods in both external test set validation and 10-fold cross-validation. On the other hand, the AUC values were dramatically reduced as the activity threshold narrowed in NB or DT models, especially when built without feature selection process. This indicates that RF models have the most robust predictive ability among classifiers, showing a constantly high AUC ranging from 0.859 to 1 and from 0.839 to 1 in test set validation and cross-validation, respectively. Regarding FS methods, BF or SSFS exhibited relatively higher AUC than PSO or GS, as well as none (without CFS methodology). In addition, they produced the highest AUC when built with the RF classifier. The NB models appears to get the largest benefit from BF and SSFS methods, achieving substantial increase compared to the model without CFS process. However, GS or PSO methods could not greatly enhance the AUC values of NB models, producing only a slight increase compared to the model built without them, especially when the activity threshold was low. This suggests that RF models built with BF or SSFS feature selection methods have strong possibility to be the optimal model and exhibit the greatest robustness.

MCC

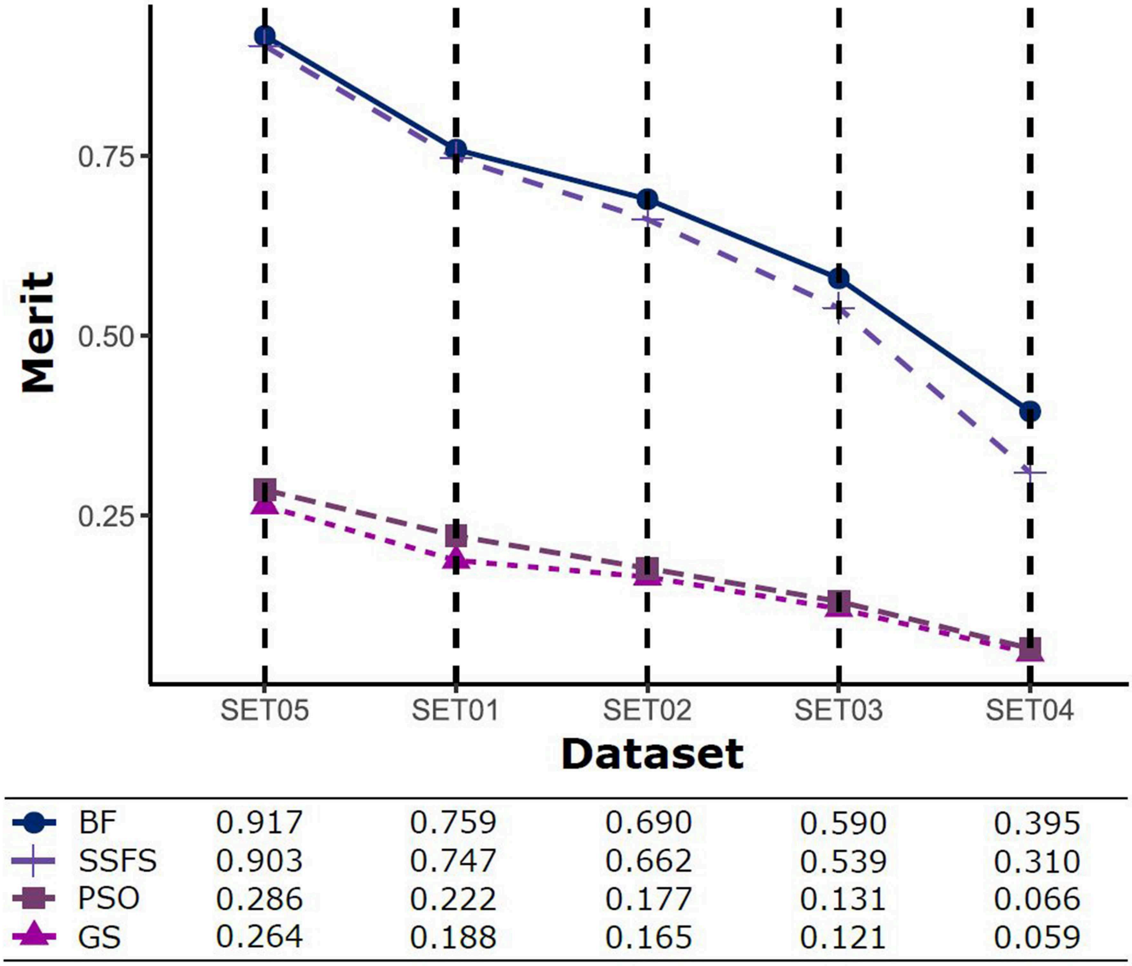
In general, the MCC values exhibited similar tendencies to the AUC (**Figure 7**). Here also, RF achieved the highest MCC for every combination except for the cross-validation result of models applying GS or no feature selector with SET04. The overall MCC values of RF classifier with other datasets except for SET04 were reasonably high, ranging from 0.693 to 0.984 in external test set validation, and from 0.721 to 0.994 in 10-fold cross-validation. Among FS methods, BF and SSFS also achieved the best performance for all combinations. In particular, they exhibited enhanced MCC values when combined with the RF classifier. On the other hand, the NB classifier with the GS or PSO feature selector exhibited considerably lower values compared to other methods, and a rapid decline could be seen as the IC50 threshold narrowed. Even when combined with BF or SSFS, the NB models resulted in relatively low MCC compared to the RF or DT models.

In summary, "RF classifier + BF selector" or "RF classifier + SSFS selector" under their optimal hyperparameters presented the best predictive ability. Obviously, RF was more distinguished than other classifiers with a robust performance in all IC50 thresholds. BF and SSFS enhanced the classification performance, obtaining higher AUC and MCC values than other selectors. It is thus observed that the IC50 activity threshold has non-negligible influences on prediction performance. As the threshold narrowed, the accuracy and MCC values declined without any exception, implying the toughness of distinguishing between patent molecules with low IC50 values. Nevertheless, models built with low activity threshold may lead to the discovery of highly potent molecules selectively. Among all IC50 thresholds (SET01 to SET05), 1 μM (SET04) was excluded to generate the Top models: four IC50 activity thresholds (11.4, 4, 3, and 2 μM) and two feature selectors (BF and SSFS) under the optimal RF classifier. Through the consensus vote of the top 8 models, potential S100A9 inhibitors could be obtained.

Quality, Cost, and Effectiveness of Screening Hits

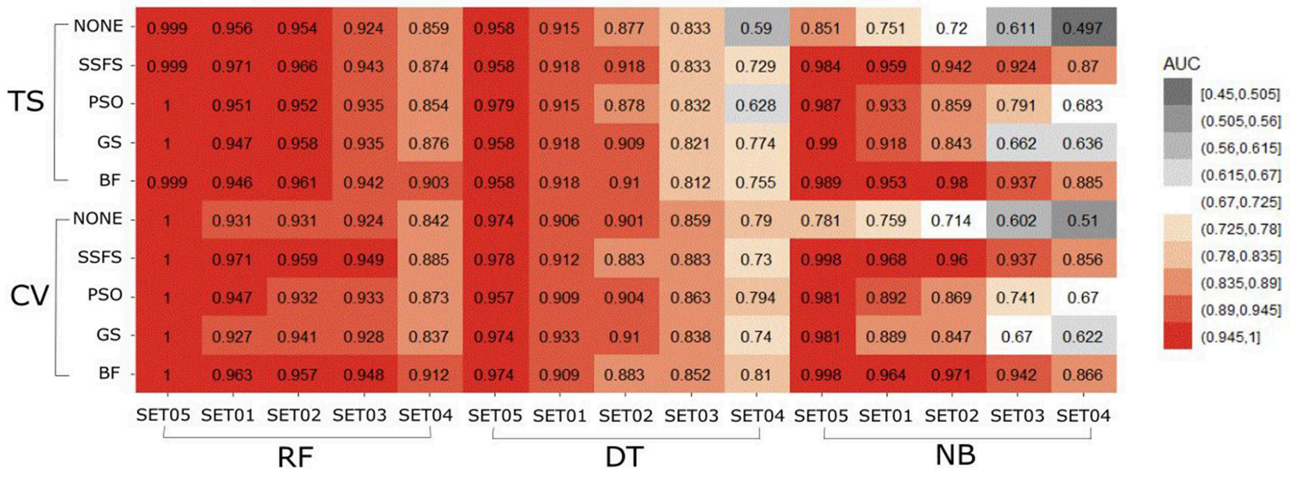
Ligand-based virtual screening was performed using a large-scale dataset ($N = 6,447,184$) derived from the eMolecules database. We finally obtained 46 potential S100A9 inhibitors through unanimous votes from top models (hit rate = 0.000713%). The 2D structures of hits are presented in **Table S3**. Notably, the prediction probabilities of selected hits were similarly high compared with patent molecules, ranging from 0.902 to 1 with little differential between models (**Figure 8**). In order to qualify the hit compounds, their structure novelty also was evaluated. For this purpose, the Tanimoto similarity between each hit compound and the nearest neighbor was presented (**Table 5**).

In the view of structural novelty, our virtual screening could certainly guarantee similarity, such as the level of recent generative model-based *de novo* design (Popova et al., 2018). Our hits not only retain the structural diversity of active molecules, but also exhibit differentiation from patents, thereby suggesting our models' ability to elicit novel S100A9 inhibitors (**Figure 2**). Furthermore, our model is economical in the view of cost. The overall screening process including feature generation of the 6 M size library took ca. 161 h under 1 CPU and 8 GB memory



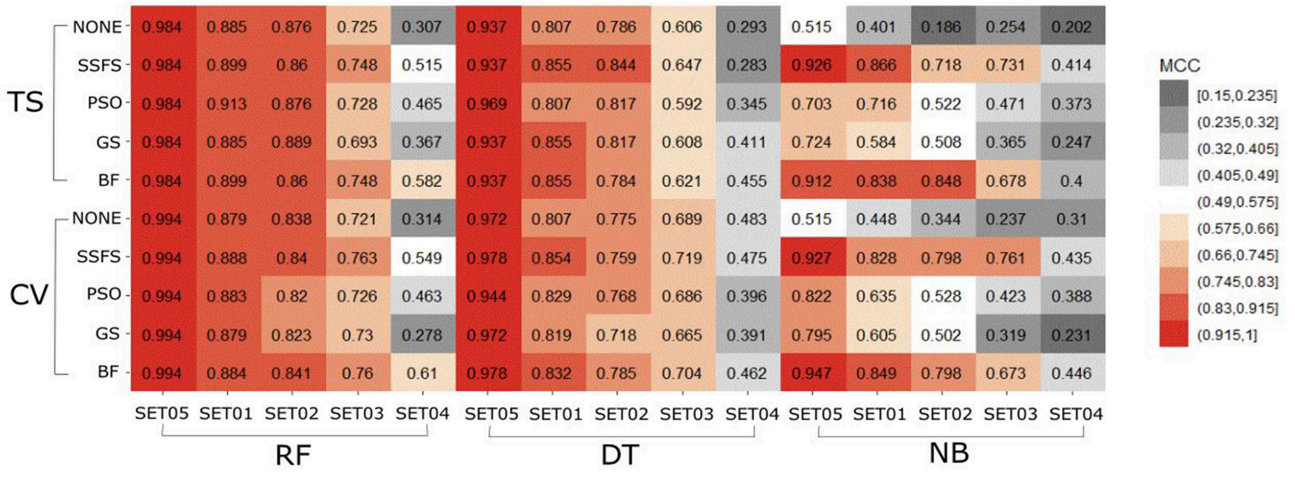
	SET05	SET01	SET02	SET03	SET04
BF	0.917	0.759	0.690	0.590	0.395
SSFS	0.903	0.747	0.662	0.539	0.310
PSO	0.286	0.222	0.177	0.131	0.066
GS	0.264	0.188	0.165	0.121	0.059

FIGURE 5 | The merits of feature sets after feature selection method in each dataset with different IC50 threshold.



		RF					DT					NB				
		SET05	SET01	SET02	SET03	SET04	SET05	SET01	SET02	SET03	SET04	SET05	SET01	SET02	SET03	SET04
TS	NONE	0.999	0.956	0.954	0.924	0.859	0.958	0.915	0.877	0.833	0.59	0.851	0.751	0.72	0.611	0.497
	SSFS	0.999	0.971	0.966	0.943	0.874	0.958	0.918	0.918	0.833	0.729	0.984	0.959	0.942	0.924	0.87
	PSO	1	0.951	0.952	0.935	0.854	0.979	0.915	0.878	0.832	0.628	0.987	0.933	0.859	0.791	0.683
	GS	1	0.947	0.958	0.935	0.876	0.958	0.918	0.909	0.821	0.774	0.99	0.918	0.843	0.662	0.636
	BF	0.999	0.946	0.961	0.942	0.903	0.958	0.918	0.91	0.812	0.755	0.989	0.953	0.98	0.937	0.885
CV	NONE	1	0.931	0.931	0.924	0.842	0.974	0.906	0.901	0.859	0.79	0.781	0.759	0.714	0.602	0.51
	SSFS	1	0.971	0.959	0.949	0.885	0.978	0.912	0.883	0.883	0.73	0.998	0.968	0.96	0.937	0.856
	PSO	1	0.947	0.932	0.933	0.873	0.957	0.909	0.904	0.863	0.794	0.981	0.892	0.869	0.741	0.67
	GS	1	0.927	0.941	0.928	0.837	0.974	0.933	0.91	0.838	0.74	0.981	0.889	0.847	0.67	0.622
	BF	1	0.963	0.957	0.948	0.912	0.974	0.909	0.883	0.852	0.81	0.998	0.964	0.971	0.942	0.866

FIGURE 6 | Heat-map depicting the AUC of ROC curve of the classification models.



		RF					DT					NB				
		SET05	SET01	SET02	SET03	SET04	SET05	SET01	SET02	SET03	SET04	SET05	SET01	SET02	SET03	SET04
TS	NONE	0.984	0.885	0.876	0.725	0.307	0.937	0.807	0.786	0.606	0.293	0.515	0.401	0.186	0.254	0.202
	SSFS	0.984	0.899	0.86	0.748	0.515	0.937	0.855	0.844	0.647	0.283	0.926	0.866	0.718	0.731	0.414
	PSO	0.984	0.913	0.876	0.728	0.465	0.969	0.807	0.817	0.592	0.345	0.703	0.716	0.522	0.471	0.373
	GS	0.984	0.885	0.889	0.693	0.367	0.937	0.855	0.817	0.608	0.411	0.724	0.584	0.508	0.365	0.247
	BF	0.984	0.899	0.86	0.748	0.582	0.937	0.855	0.784	0.621	0.455	0.912	0.838	0.848	0.678	0.4
CV	NONE	0.994	0.879	0.838	0.721	0.314	0.972	0.807	0.775	0.689	0.483	0.515	0.448	0.344	0.237	0.31
	SSFS	0.994	0.888	0.84	0.763	0.549	0.978	0.854	0.759	0.719	0.475	0.927	0.828	0.798	0.761	0.435
	PSO	0.994	0.883	0.82	0.726	0.463	0.944	0.829	0.768	0.686	0.396	0.822	0.635	0.528	0.423	0.388
	GS	0.994	0.879	0.823	0.73	0.278	0.972	0.819	0.718	0.665	0.391	0.795	0.605	0.502	0.319	0.231
	BF	0.994	0.884	0.841	0.76	0.61	0.978	0.832	0.785	0.704	0.462	0.947	0.849	0.798	0.673	0.446

FIGURE 7 | Heat-map depicting the MCC of the classification models.

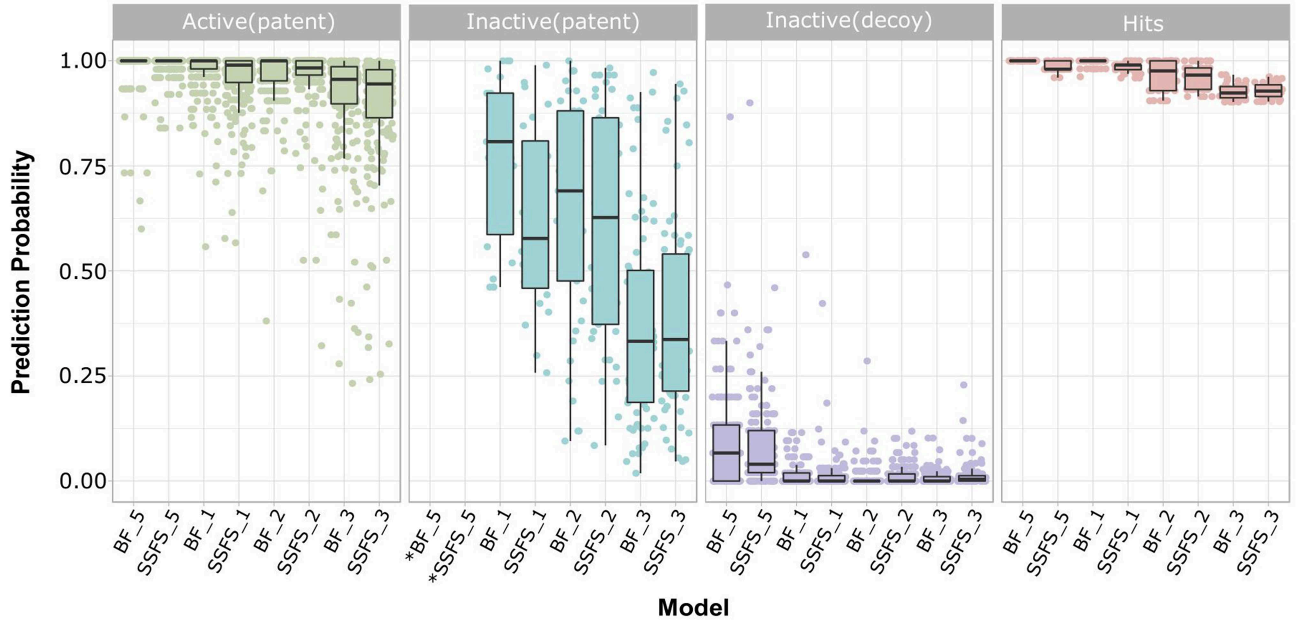


FIGURE 8 | Prediction probability of top 8 models on patent molecules, decoy molecules, and hits. Each label of horizontal axis represents each top random forest model. The FS method and the dataset (SET0N) used in the model is indicated as "FS method_N". For example, SSFS_3 refers to the random forest model built with the feature set chosen by SSFS with dataset SET03. *Note that every patent molecules were considered as active in BF_5 and SSFS_5.

condition for being to show 40 times faster than the screening using S100A9 docking models in the same computing resource. It proved strong cost-reduction ability and efficiency enough to apply to the real-world drug R&D.

In sequence, binding mode of the hit compounds was compared with known S100A9 inhibitors, 266 dataset under in-house docking model. For the docking simulations, homodimer of the mutant S100A9 (C3S) was gain from PDB 5I8N code (Chang et al., 2016). The S100A9 inhibitors were docked to S100A9-RAGE V dining domain to share the common region surrounded by Glu52 (at the hinge between H2 and H3), Arg85 (at H4), and Trp88 (at H4) in **Figures S1–S3**. 46 hit compounds also presented similar binding modes: (1) pi-pi or pi-cation interaction with residues at H4 (e.g., Trp88, Arg85) or (2) hydrogen bonding with hinge (e.g., Glu52 or Asn55) in **Figures S4, S5** to add promising evidence of the hit compounds. Finally, since poor pharmacokinetic profiles and high potential toxicity are likely to fail in clinical trials, it is also crucial to predict such properties in advance to encourage further *in vivo* validation of hit molecules. We calculated the molecular

TABLE 5 | Drug-likeness, ADME parameters prediction for 46 hits using QikProp and their Tanimoto similarity between the nearest neighbor.

Molecule index	MW[a]	LogPo/w[b]	dHB[c]	aHB[d]	No. N&O[e]	PSA[f]	Lo5[g]	Jo3[h]	Caco2[i]	MDCK[j]	Sim[k]
1	438.81	4.64	2	5.25	6	83.28	0	1	883.32	6500.64	0.723
2	447.52	1.88	1	10.25	8	110.03	0	0	169.49	217.42	0.733
3	475.58	2.45	1	10.25	8	107.08	0	0	212.80	254.04	0.725
4	459.55	3.35	2	9	7	109.44	0	1	252.94	277.07	0.709
5	357.35	2.21	1	6.5	6	79.76	0	0	662.83	1251.4	0.803
6	475.54	2.06	2	12	10	152.60	0	0	74.61	50.07	0.686
7	369.37	2.66	2	5.5	7	112.07	0	0	123.62	120.63	0.823
8	489.56	2.35	2	12	10	152.60	0	1	74.59	50.05	0.685
9	463.57	2.06	3	9.5	9	138.29	0	0	31.69	26.88	0.763
10	399.25	2.08	1.25	7.75	7	102.73	0	0	310.52	651.41	0.831
11	394.81	2.23	1.25	7.75	7	102.66	0	0	338.55	628.45	0.757
12	376.82	1.93	1.25	7.75	7	103.25	0	0	371.75	355.25	0.767
13	394.81	2.10	1.25	7.75	7	103.88	0	0	336.23	499.24	0.757
14	449.54	2.32	3	9.5	9	134.36	0	0	78.22	41.92	0.711
15	463.57	2.57	2	9.75	8	110.80	0	0	245.98	297.80	0.708
16	396.39	3.56	1.25	5.25	6	93.24	0	1	414.29	1915.26	0.727
17	378.42	0.73	3	10	8	136.31	0	0	55.49	35.38	0.747
18	408.45	0.61	2	11.75	10	140.34	0	0	89.51	46.28	0.738
19	388.46	2.35	2	6.5	7	120.67	0	0	135.00	195.05	0.633
20	392.47	3.46	2	5.25	6	88.54	0	0	274.13	938.46	0.663
21	379.41	0.81	3	9.25	8	135.31	0	0	48.47	31.86	0.833
22	488.49	4.53	1	8.7	8	90.37	0	2	1243.59	3476.5	0.697
23	374.43	3.01	2.25	5.75	6	96.24	0	0	317.78	825.75	0.718
24	376.86	2.93	2.25	5.75	6	95.44	0	0	309.55	1326.69	0.721
25	376.86	2.92	2.25	5.75	6	95.45	0	0	344.43	1241.09	0.721
26	360.41	2.61	2.25	5.75	6	94.32	0	0	309.07	1006.01	0.721
27	424.44	3.73	2.25	5.75	6	98.39	0	0	240.41	1826.54	0.704
28	410.41	3.34	2.25	5.75	6	94.30	0	0	309.14	2445.21	0.706
29	394.81	1.97	1.25	7.75	7	101.41	0	0	336.23	484.57	0.757
30	397.52	2.51	2	8	6	90.69	0	0	823.95	875.59	0.753
31	378.46	2.15	2	7.7	7	110.06	0	0	286.99	254.97	0.776
32	382.82	2.67	1	6.75	8	111.06	0	0	226.09	236.12	0.783
33	427.46	0.74	1	11	10	129.95	0	0	146.33	120.44	0.747
34	410.41	2.18	2	9	7	115.79	0	0	223.42	411.72	0.744
35	410.41	2.15	2	9	7	116.40	0	0	182.76	361.39	0.744
36	357.79	1.93	2	7	6	92.65	0	0	276.14	510.66	0.828
37	357.79	1.93	2	7	6	93.28	0	0	259.64	489.39	0.828
38	357.79	1.86	2	7	6	93.79	0	0	240.46	428.65	0.828
39	371.81	2.37	1	7.5	6	81.06	0	0	544.41	1089.44	0.780
40	374.43	3.67	1.25	5.75	6	84.30	0	1	922.94	2328.24	0.750
41	370.79	1.97	2	6.5	7	111.17	0	0	122.26	206.45	0.759
42	399.87	3.14	1	7.5	6	82.38	0	0	625.21	1265.93	0.791
43	412.80	2.35	1.25	7.75	7	101.92	0	0	323.45	793.95	0.757
44	398.40	4.57	2	4.5	6	80.90	0	1	1275.28	3912.14	0.671
45	379.42	2.24	2	8.5	6	101.90	0	0	319.42	418.38	0.759
46	348.39	0.82	2	9.5	7	114.57	0	0	141.61	101.70	0.783
Standard value[l]	130.0 –725.0	–2.0 –6.5	0.0 –6.0	2.0 –20.0	2–15	7.0 –200.0	Maximum is 4	Maximum is 3	<25 poor, >500 great	<25 poor, >500 great	

[a] *Molecular weight.*

[b] *Octanol/water partition coefficient.*

[c] *Number of HB donors.*

[d] *Number of HB acceptors.*

[e] *Number of N and O atoms.*

[f] *Polar surface area.*

[g] *Number of violation of Lipinski's rule of five.*

[h] *Number of violation of Jorgensen's rule of five.*

[i] *Apparent Caco-2 cell permeability (nm/s).*

[j] *Apparent MDCK cell permeability (nm/s).*

[k] *Tanimoto coefficient of the entry between the nearest neighbor among 266 active molecules from patents.*

[l] *Standard values from 95% of known drugs based on results of Qikprop.*

parameters regarding drug-likeness and ADME properties to ensure that the hit compounds are suitable for further drug development processes (**Table 5**). Hopefully, all predicted values of 46 molecules are within the acceptable range. Neither Lipinski's rule of five nor Jorgensen's rule of three was violated by almost all hits. Even though we did not implement any physicochemical predictor into our model, the physicochemical property of the dataset could be transferred into screening hits through a structure-property relationship. If our model can be linked with a powerful inverse design model, we can expect our model can also provide powerful predictability with a physicochemical property range.

CONCLUSION

In summary, through extensive validation of 60 models built from multi-scaffold ligand information, we optimized the machine learning classifier as well as the feature selector to obtain highly predictive classification models for identifying S100A9 inhibitors. Unlike many other reports employing only several kinds of descriptors or a whole bits of fingerprint, we combined various kinds of descriptors with a hybrid fingerprint to generate a compact and effective feature set. Ultimately, this high efficiency allowed us to further obtain 47 hits from over six million compounds through the consensus vote of models within a week, indicating the high cost-reduction ability of the models. In addition, our study is the first example of reasonable classification models for S100A9 inhibitors. Regarding the clinical importance of S100A9, as well as the difficulty of generating models for its unique characteristics, we expect that our study will further aid in developing the first S100A9 agents and guide new paths of curing diverse diseases, including Alzheimer's disease and other neurodegenerative diseases.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this manuscript will be made available by the authors, without undue reservation, to any qualified researcher.

AUTHOR CONTRIBUTIONS

MK and S-YL conceived and designed the study at their grant based research projects. Under the designed study, SK and JL built their models, validated them and acquired *in-silico* hits through their models. MK and JL wrote the manuscript. MK and SP revised the manuscript. All the authors read and approved the final manuscript.

FUNDING

This study was supported by the Basic Science Research Program of the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science, and Technology (No. 2017R1E1A1A01076642), by a grant of the Korea Health Technology R&D Project through the Korea Health Industry Development Institute (KHIDI), funded by the Ministry for Health and Welfare, Korea (HI14C1135).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fchem.2019.00779/full#supplementary-material

DataSheet 1 | The 2D-structure of Dataset in **Table S1**.

REFERENCES

Bajorath, J. (2002). Integration of virtual and high-throughput screening. *Nat. Rev. Drug Discov.* 1:882. doi: 10.1038/nrd941

Bellman, R. (1966). Dynamic programming. *Science* 153, 34–37. doi: 10.1126/science.153.3731.34

Bendtsen, C., Degasperi, A., Ahlberg, E., and Carlsson, L. (2017). Improving machine learning in early drug discovery. *Ann. Math. Artif. Intell.* 81, 155–166. doi: 10.1007/s10472-017-9541-2

Bisignano, P., Burford, N. T., Shang, Y., Marlow, B., Livingston, K. E., Fenton, A. M., et al. (2015). Ligand-based discovery of a new scaffold for allosteric modulation of the μ-opioid receptor. *J. Chem. Inf. Model.* 55, 1836–1843. doi: 10.1021/acs.jcim.5b00388

Björk, P., Björk, A., Vogl, T., Stenström, M., Liberg, D., Olsson, A., et al. (2009). Identification of human S100A9 as a novel target for treatment of autoimmune disease via binding to quinoline-3-carboxamides. *PLoS Biol.* 7:e1000097. doi: 10.1371/journal.pbio.1000097

Breiman, L. (2001). Random forests. *Mach. Learn.* 45, 5–32. doi: 10.1023/A:1010933404324

Breiman, L. (2017). *Classification and Regression Trees*. Ogden, UT: Routledge.

Chang, C. C., Khan, I., Tsai, K. L., Li, H., Yang, L. W., Chou, R. H., et al. (2016). Blocking the interaction between S100A9 and RAGE V domain using CHAPS molecule: a novel route to drug development against cell proliferation. *Biochim. Biophys. Acta* 1864:1558. doi: 10.1016/j.bbapap.2016.08.008

Cortes, C., and Vapnik, V. (1995). Support-vector networks. *Mach. Learn.* 20, 273–297. doi: 10.1007/BF00994018

Dash, M., and Liu, H. (1997). Feature selection for classification. *Intell. Data Anal.* 1, 131–156. doi: 10.3233/IDA-1997-1302

Ding, C., and Peng, H. (2005). Minimum redundancy feature selection from microarray gene expression data. *J. Bioinform. Comput. Biol.* 3, 185–205. doi: 10.1142/S0219720005001004

Domingos, P., and Pazzani, M. (1997). On the optimality of the simple Bayesian classifier under zero-one loss. *Mach. Learn.* 29, 103–130. doi: 10.1023/A:1007413511361

Domingos, P. M. (2012). A few useful things to know about machine learning. *Commun. ACM* 55, 78–87. doi: 10.1145/2347736.2347755

Donato, R. (1999). Functional roles of S100 proteins, calcium-binding proteins of the EF-hand type. *Biochim. Biophys. Acta* 1450, 191–231. doi: 10.1016/S0167-4889(99)00058-0

Donato, R. (2001). S100: a multigenic family of calcium-modulated proteins of the EF-hand type with intracellular and extracellular functional roles. *Int. J. Biochem. Cell Biol.* 33, 637–668. doi: 10.1016/S1357-2725(01)00046-2

Eberhart, R., and Kennedy, J. (1995). "A new optimizer using particle swarm theory," in *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science* (Nagoya: IEEE), 39–43.

Freuder, E. C. (1988). "Backtrack-free and backtrack-bounded search," in *Search in Artificial Intelligence*, eds L. Kanal and V. Kumar (New York, NY: Springer), 343–369. doi: 10.1007/978-1-4613-8788-6_10

Fritzson, I., Liberg, D., East, S., Mackinnon, C., and Prevost, N. (2014). *N-(heteroaryl)-Sulfonamide Derivatives Useful as S100-Inhibitors*. U.S. Patent No 9,873,687,2018.

Gadhe, C. G., Lee, E., and Kim, M.-H. (2015). Finding new scaffolds of JAK3 inhibitors in public database: 3D-QSAR models & shape-based screening. *Arch. Pharm. Res.* 38, 2008–2019. doi: 10.1007/s12272-015-0607-6

Geppert, H., Vogt, M., and Bajorath, J. (2010). Current trends in ligand-based virtual screening: molecular representations, data mining methods, new application areas, and performance evaluation. *J. Chem. Inf. Model.* 50, 205–216. doi: 10.1021/ci900419k

Glover, F. W., and Kochenberger, G. A. (2006). *Handbook of Metaheuristics.* Springer US.

Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Boston, MA: Addison-Wesley Longman Publishing Co., Inc.

Gruden, M. A., Davydova, T. V., Kudrin, V. S., Wang, C., Narkevich, V. B., Morozova-Roche, L. A., et al. (2017). S100A9 protein aggregates boost hippocampal glutamate modifying monoaminergic neurochemistry: a glutamate antibody sensitive outcome on Alzheimer-like memory decline. *ACS Chem. Neurosci.* 9, 568–577. doi: 10.1021/acschemneuro.7b00379

Gutlein, M., Frank, E., Hall, M., and Karwath, A. (2009). "Large-scale attribute selection using wrappers," in *2009 IEEE Symposium on Computational Intelligence and Data Mining* (Nashville, TN: IEEE), 332–339. doi: 10.1109/CIDM.2009.4938668

Guyon, I., and Elisseeff, A. (2003). An introduction to variable and feature selection. *J. Mach. Learn. Res.* 3, 1157–1182. doi: 10.1162/153244303322753616

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The WEKA data mining software: an update. *ACM SIGKDD Explor. Newsl.* 11, 10–18. doi: 10.1145/1656274.1656278

Hall, M. A. (1999). *Correlation-Based Feature Selection for Machine Learning* (Ph.D. thesis). The University of Waikato. Hamilton, New Zealand.

Hermani, A., Hess, J., De Servi, B., Medunjanin, S., Grobholz, R., Trojan, L., et al. (2005). Calcium-binding proteins S100A8 and S100A9 as novel diagnostic markers in human prostate cancer. *Clin. Cancer Res.* 11, 5146–5152. doi: 10.1158/1078-0432.CCR-05-0352

Holland, J. H. (1992). *Adaptation in Natural and Artificial Systems: An Introductory Analysis With Applications to Biology, Control, and Artificial Intelligence*. Cambridge, MA: MIT Press.

Horvath, I., Jia, X., Johansson, P., Wang, C., Moskalenko, R., Steinau, A., et al. (2015). Pro-inflammatory S100A9 protein as a robust biomarker differentiating early stages of cognitive impairment in Alzheimer's disease. *ACS Chem. Neurosci.* 7, 34–39. doi: 10.1021/acschemneuro.5b00265

Iashchishyn, I. A., Gruden, M. A., Moskalenko, R. A., Davydova, T. V., Wang, C., Sewell, R. D., et al. (2018). Intranasally administered S100A9 amyloids induced cellular stress, amyloid seeding, and behavioral impairment in aged mice. *ACS Chem. Neurosci.* 9, 1338–1348. doi: 10.1021/acschemneuro.7b00512

Itou, H., Yao, M., Fujita, I., Watanabe, N., Suzuki, M., Nishihira, J., et al. (2002). The crystal structure of human MRP14 (S100A9), a Ca^{2+}-dependent regulator protein in inflammatory process. *J. Mol. Biol.* 316:265. doi: 10.1006/jmbi.2001.5340

Jang, C., Yadav, D. K., Subedi, L., Venkatesan, R., Venkanna, A., Afzal, S., et al. (2018). Identification of novel acetylcholinesterase inhibitors designed by pharmacophore-based virtual screening, molecular docking and bioassay. *Sci. Rep.* 8:14921. doi: 10.1038/s41598-018-33354-6

Kapetanovic, I. (2008). Computer-aided drug discovery and development (CADDD): *in silico*-chemico-biological approach. *Chem. Biol. Interact.* 171, 165–176. doi: 10.1016/j.cbi.2006.12.006

Katte, R., and Yu, C. (2018). Blocking the interaction between S100A9 protein and RAGE V domain using S100A12 protein. *PLoS ONE* 13:e0198767. doi: 10.1371/journal.pone.0198767

Khuri, N., Zur, A. A., Wittwer, M. B., Lin, L., Yee, S. W., Sali, A., et al. (2017). Computational discovery and experimental validation of inhibitors of the human intestinal transporter OATP2B1. *J. Chem. Inf. Model.* 57, 1402–1413. doi: 10.1021/acs.jcim.6b00720

Kim, H.-J., Kang, H. J., Lee, H., Lee, S.-T., Yu, M.-H., Kim, H., et al. (2009). Identification of S100A8 and S100A9 as serological markers for colorectal cancer. *J. Proteome Res.* 8, 1368–1379. doi: 10.1021/pr8007573

Kohavi, R., and John, G. H. (1997). Wrappers for feature subset selection. *Artif. Intell.* 97, 273–324. doi: 10.1016/S0004-3702(97)00043-X

Koller, D., and Sahami, M. (1996). *Toward Optimal Feature Selection*. Stanford, CA: Stanford InfoLab.

Kotsiantis, S. B., Zaharakis, I., and Pintelas, P. (2007). Supervised machine learning: A review of classification techniques. *Emerg. Artif. Intell. Appl. Comput. Eng.* 160, 3–24. doi: 10.1007/s10462-007-9052-3

Lavecchia, A. (2015). Machine-learning approaches in drug discovery: methods and applications. *Drug Discov. Today* 20, 318–331. doi: 10.1016/j.drudis.2014.10.012

Lavecchia, A., and Di Giovanni, C. (2013). Virtual screening strategies in drug discovery: a critical review. *Curr. Med. Chem.* 20, 2839–2860. doi: 10.2174/09298673113209990001

Lee, J., Cho, S., and Kim, M.-H. (2018). Discovery of CNS-like D3R-selective antagonists using 3D pharmacophore guided virtual screening. *Molecules* 23:2452. doi: 10.3390/molecules23102452

Lenselink, E. B., Beuming, T., van Veen, C., Massink, A., Sherman, W., van Vlijmen, H. W., et al. (2016). In search of novel ligands using a structure-based approach: a case study on the adenosine A 2A receptor. *J. Comput. Aided Mol. Des.* 30, 863–874. doi: 10.1007/s10822-016-9963-7

Liu, H., and Yu, L. (2005). Toward integrating feature selection algorithms for classification and clustering. *IEEE Trans. Knowl. Data Eng.* 17, 491–502. doi: 10.1109/TKDE.2005.66

Liu, Y. (2004). A comparative study on feature selection methods for drug discovery. *J. Chem. Inf. Comput. Sci.* 44, 1823–1828. doi: 10.1021/ci049875d

Lo, Y.-C., Rensi, S. E., Torng, W., and Altman, R. B. (2018). Machine learning in chemoinformatics and drug discovery. *Drug Discov. Today* 23, 1538–1546. doi: 10.1016/j.drudis.2018.05.010

Man, K.-F., Tang, K.-S., and Kwong, S. (1996). Genetic algorithms: concepts and applications [in engineering design]. *IEEE Trans. Ind. Electron.* 43, 519–534. doi: 10.1109/41.538609

Melville, J. L., Burke, E. K., and Hirst, J. D. (2009). Machine learning in virtual screening. *Comb. Chem. High Throughput Screen.* 12, 332–343. doi: 10.2174/138620709788167980

Mignani, S., Huber, S., Tomas, H., Rodrigues, J., and Majoral, J.-P. (2016). Why and how have drug discovery strategies in pharma changed? What are the new mindsets? *Drug Discov. Today* 21, 239–249. doi: 10.1016/j.drudis.2015.09.007

Moraglio, A., Di Chio, C., and Poli, R. (2007). "Geometric particle swarm optimisation," in *European Conference on Genetic Programming* (Springer), 125–136. doi: 10.1007/978-3-540-71605-1_12

Muegge, I., and Oloff, S. (2006). Advances in virtual screening. *Drug Discov. Today* 3, 405–411. doi: 10.1016/j.ddtec.2006.12.002

Mullard, A. (2014). New drugs cost US $2.6 billion to develop. *Nat. Rev. Drug Discov.* 13:877. doi: 10.1038/nrd4507

Mysinger, M. M., Carchia, M., Irwin, J. J., and Shoichet, B. K. (2012). Directory of useful decoys, enhanced (DUD-E): better ligands and decoys for better benchmarking. *J. Med. Chem.* 55, 6582–6594. doi: 10.1021/jm300687e

Nisius, B., and Bajorath, J. (2009). Molecular fingerprint recombination: generating hybrid fingerprints for similarity searching from different fingerprint types. *ChemMedChem* 4, 1859–1863. doi: 10.1002/cmdc.200900243

Nisius, B., and Bajorath, J. (2010). Reduction and recombination of fingerprints of different design increase compound recall and the structural diversity of hits. *Chem. Biol. Drug Des.* 75, 152–160. doi: 10.1111/j.1747-0285.2009.00930.x

Oprea, T. I., and Matter, H. (2004). Integrating virtual screening in lead discovery. *Curr. Opin. Chem. Biol.* 8, 349–358. doi: 10.1016/j.cbpa.2004.06.008

Pelletier, M., Simard, J. C., Girard, D., and Tessier, P. A. (2018). Quinoline-3-carboxamides such as tasquinimod are not specific inhibitors of S100A9. *Blood Adv.* 2:1170. doi: 10.1182/bloodadvances.2018016667

Popova, M., Isayev, O., and Tropsha, A. (2018). Deep reinforcement learning for de novo drug design. *Sci. Adv.* 4:eaap7885. doi: 10.1126/sciadv.aap7885

Quinlan, J. R. (2014). *C4.5: Programs for Machine Learning.* San Mateo, CA: Elsevier.

Reddy, A. S., Pati, S. P., Kumar, P. P., Pradeep, H., and Sastry, G. N. (2007). Virtual screening in drug discovery-a computational perspective. *Curr. Protein Peptide Sci.* 8, 329–351. doi: 10.2174/138920307781369427

Rissanen, J. (1978). Modeling by shortest data description. *Automatica* 14, 465–471. doi: 10.1016/0005-1098(78)90005-5

Ryckman, C., Vandal, K., Rouleau, P., Talbot, M., and Tessier, P. A. (2003). Proinflammatory activities of S100: proteins S100A8, S100A9, and S100A8/A9 induce neutrophil chemotaxis and adhesion. *J. Immunol.* 170, 3233–3242. doi: 10.4049/jimmunol.170.6.3233

Scannell, J. W., Blanckley, A., Boldon, H., and Warrington, B. (2012). Diagnosing the decline in pharmaceutical R&D efficiency. *Nat. Rev. Drug Discov.* 11:191. doi: 10.1038/nrd3681

Shafer, J., Agrawal, R., and Mehta, M. (1996). "SPRINT: a scalable parallel classifier for data mining," in *VLDB* (Citeseer), 544–555.

Shehata, M. A., Nøhr, A. C., Lissa, D., Bisig, C., Isberg, V., Andersen, K. B., et al. (2016). Novel agonist bioisosteres and common structure-activity relationships for the orphan G protein-coupled receptor GPR139. *Sci. Rep.* 6:36681. doi: 10.1038/srep36681

Shi, Y. (2001). "Particle swarm optimization: developments, applications and resources," in *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No. 01TH8546)* (Seoul: IEEE), 81–86.

Shoichet, B. K. (2004). Virtual screening of chemical libraries. *Nature* 432:862. doi: 10.1038/nature03197

Singla, D., Tewari, R., Kumar, A., and Raghava, G. P. (2013). Designing of inhibitors against drug tolerant Mycobacterium tuberculosis (H37Rv). *Chem. Cent. J.* 7:49. doi: 10.1186/1752-153X-7-49

Sliwoski, G., Kothiwale, S., Meiler, J., and Lowe, E. W. (2014). Computational methods in drug discovery. *Pharmacol. Rev.* 66, 334–395. doi: 10.1124/pr.112.007336

Smieja, M., and Warszycki, D. (2016). Average information content maximization—a new approach for fingerprint hybridization and reduction. *PLoS ONE* 11:e0146666. doi: 10.1371/journal.pone.0146666

Stahura, F. L., and Bajorath, J. (2004). Virtual screening methods that complement HTS. *Comb. Chem. High Throughput Screen.* 7, 259–269. doi: 10.2174/1386207043328706

Stahura, F. L., and Bajorath, J. (2005). New methodologies for ligand-based virtual screening. *Curr. Pharm. Des.* 11, 1189–1202. doi: 10.2174/1381612053507549

Vafaie, H., and De Jong, K. (1992). "Genetic algorithms as a tool for feature selection in machine learning," in *Proceedings Fourth International Conference on Tools with Artificial Intelligence TAI'92* (Arlington, VA: IEEE), 200–203.

Verikas, A., Gelzinis, A., and Bacauskiene, M. (2011). Mining data with random forests: a survey and results of new tests. *Pattern Recognit.* 44, 330–349. doi: 10.1016/j.patcog.2010.08.011

Walters, W. P., Stahl, M. T., and Murcko, M. A. (1998). Virtual screening—an overview. *Drug Discov. Today* 3, 160–178. doi: 10.1016/S1359-6446(97)01163-X

Warszycki, D., Smieja, M., and Kafel, R. (2017). Practical application of the average information content maximization (AIC-MAX) algorithm: selection of the most important structural features for serotonin receptor ligands. *Mol. Divers.* 21, 407–412. doi: 10.1007/s11030-017-9729-8

Wellmar, U., East, S., Bainbridge, M., Mackinnon, C., Carr, J., and Hargrave, J. (2016). *Imidazo [2, 1-b] thiazole and 5, 6-Dihydroimidazo [2, 1-b] thiazole Derivatives Useful as S100-Inhibitors*. U.S. Patent Application No 15/545,573, 2018.

Wellmar, U., Liberg, D., Ekblad, M., Bainbridge, M., East, S., Hargrave, J., et al. (2015). *Compounds Useful as S100-Inhibitors*. U.S. Patent No 9,771,372,2017.

Williams, A. J. (2008). Public chemical compound databases. *Curr. Opin. Drug Discov. Dev.* 11:393. Available online at: https://www.researchgate.net/publication/5424985_Public_chemical_compound_databases

Williams, C. (2006). Reverse fingerprinting, similarity searching by group fusion and fingerprint bit importance. *Mol. Divers.* 10, 311–332. doi: 10.1007/s11030-006-9039-z

Yadav, D. K., Sharma, P., Misra, S., Singh, H., Mancera, R. L., Kim, K., et al. (2018). Studies of the benzopyran class of selective COX-2 inhibitors using 3D-QSAR and molecular docking. *Arch. Pharm. Res.* 41, 1178–1189. doi: 10.1007/s12272-017-0945-7

Yap, C. W. (2011). PaDEL-descriptor: an open source software to calculate molecular descriptors and fingerprints. *J. Comput. Chem.* 32, 1466–1474. doi: 10.1002/jcc.21707

Yatime, L., Betzer, C., Jensen, R. K., Mortensen, S., Jensen, P. H., and Andersen, G. R. (2016). The structure of the RAGE: S100A6 complex reveals a unique mode of homodimerization for S100 proteins. *Structure* 24, 2043–2052. doi: 10.1016/j.str.2016.09.011

Yoshioka, Y., Mizutani, T., Mizuta, S., Miyamoto, A., Murata, S., Ano, T., et al. (2016). Neutrophils and the S100A9 protein critically regulate granuloma formation. *Blood Adv.* 1:184. doi: 10.1182/bloodadvances.2016000497

Yu, L., and Liu, H. (2004). Efficient feature selection via analysis of relevance and redundancy. *J. Mach. Learn. Res.* 5, 1205–1224.

Conflict 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.

frontiers in Chemistry

ORIGINAL RESEARCH
published: 17 January 2020
doi: 10.3389/fchem.2019.00929

Deep Learning Techniques to Improve the Performance of Olive Oil Classification

Belén Vega-Márquez [1]*, *Isabel Nepomuceno-Chamorro* [1], *Natividad Jurado-Campos* [2] and *Cristina Rubio-Escudero* [1]

[1] Department of Computer Languages and Systems, University of Sevilla, Sevilla, Spain, [2] Department of Analytical Chemistry, Institute of Fine Chemistry and Nanochemistry, International Agrifood Campus of Excellence (ceiA3), University of Córdoba, Córdoba, Spain

OPEN ACCESS

Edited by:
John C. Cancilla,
Scintillon Institute, United States

Reviewed by:
Ricard Boqué,
University of Rovira i Virgili, Spain
Ramesh L. Gardas,
Indian Institute of Technology
Madras, India

***Correspondence:**
Belén Vega-Márquez
bvega@us.es

Specialty section:
This article was submitted to
Theoretical and Computational
Chemistry,
a section of the journal
Frontiers in Chemistry

Received: *15 February 2019*
Accepted: *20 December 2019*
Published: *17 January 2020*

Citation:
Vega-Márquez B,
Nepomuceno-Chamorro I,
Jurado-Campos N and
Rubio-Escudero C (2020) Deep
Learning Techniques to Improve the
Performance of Olive Oil Classification.
Front. Chem. 7:929.
doi: 10.3389/fchem.2019.00929

The olive oil assessment involves the use of a standardized sensory analysis according to the "panel test" method. However, there is an important interest to design novel strategies based on the use of Gas Chromatography (GC) coupled to mass spectrometry (MS), or ion mobility spectrometry (IMS) together with a chemometric data treatment for olive oil classification. It is an essential task in an attempt to get the most robust model over time and, both to avoid fraud in the price and to know whether it is suitable for consumption or not. The aim of this paper is to combine chemical techniques and Deep Learning approaches to automatically classify olive oil samples from two different harvests in their three corresponding classes: extra virgin olive oil (EVOO), virgin olive oil (VOO), and lampante olive oil (LOO). Our Deep Learning model is built with 701 samples, which were obtained from two olive oil campaigns (2014–2015 and 2015–2016). The data from the two harvests are built from the selection of specific olive oil markers from the whole spectral fingerprint obtained with GC-IMS method. In order to obtain the best results we have configured the parameters of our model according to the nature of the data. The results obtained show that a deep learning approach applied to data obtained from chemical instrumental techniques is a good method when classifying oil samples in their corresponding categories, with higher success rates than those obtained in previous works.

Keywords: olive oil classification, chemometric approaches, GC-IMS method, machine learning, deep learning, feed-forward neural network

1. INTRODUCTION

Olive oil is a fatty substance which is obtained from the fruit of the olive tree *Olea europea L.*. There are three different olive oil categories that in descending order of quality are named as extra virgin olive oil (EVOO), virgin olive oil (VOO), and *lampante* olive oil (LOO). The first two are edible while the last one should be refined prior to be consumed. The EVOO flavor is characterized by a pleasant balanced flavor of green and fruity sensory characteristics. In the VOO and LOO, some negative attributes (chemical compounds associated to defects) can be detected in different proportions. The EVOO is the only non-defective olive oil and therefore it is the most appreciated and expensive. Moreover, selling lower quality olive oils as EVOO is one of the most common olive oil commercial frauds. The classification of olive oil depends on (i) chemical parameters such as free acidity, peroxide value and absorbance (K270 and K232) defined by the current European Union Regulation (EEC, 1991) and (ii) a sensory assessment by trained tasters. The sensory assessment methodology is slow and expensive. Consequently, instrumental analytical measurements used in conjunction with chemometric methodologies represent an alternative for reducing costs in the task of differentiating between olive oil categories.

Few studies (Borràs et al., 2015; Borràs et al., 2016; Garrido-Delgado et al., 2015; Sales et al., 2017; Contreras et al., 2019b) can be found to demonstrate the potential of analytical instruments in order to complement the sensorial analysis to classify olive oil samples as EVOO, VOO, and LOO. To demonstrate the usefulness of these methods, the amount of analyzed samples of different harvests should be high in order to obtain representative conclusions. Also, the accuracy of the classification models could be assessed by splitting the total number of analyzed samples in training and testing sets. And finally, the selection of the correct chemometric approaches would be a key point to offer a method which could classify olive oil with guarantee.

Machine learning algorithms have been used in chemistry for several decades obtaining successful results (Svetnik et al., 2003; Du et al., 2008). The massive use of these algorithms has been due to the fact that they create intuitive models which transform complex input chemical data to an explainable output. However, in more sophisticated chemical problems, the relationships between input data and output solutions are not so easy to identify. Apart from that, some machine learning algorithms are not efficient enough in dealing with high-dimensional data when no dimension reduction is performed. Neural networks solve most of the problems that arise with the use of machine learning algorithms: firstly, they solve the problem of searching and identifying existing relationships, resulting black-box models that are not so interpretable, but with a high level of accuracy. Lastly, there is no problem with the amount of data, that's why they can work efficiently with high-dimensional data.

The use of artificial intelligence to detect the quality of gastronomic and agricultural products is not a new research field. In particular, Deep Learning techniques are being used for similar classification tasks with promising results, for example in the detection of different types of wine using taste sensors and neural networks (Riul et al., 2004) and in food classification (Dębska and Guzowska-Świder, 2011). There are also several works with the objective of determining the quality of olive oil with artificial neural networks, as expressed in the review from Gonzalez-Fernandez et al. (2019), however, none of them distinguishes among the three currently existing categories (EVOO, VOO, and LOO), they only distinguish between two (EVOO/non EVOO, LOO/non LOO).

Our aim in this study has been the application of Deep Learning techniques to a group of significant markers obtained by analytical instrumentation, specifically based on gas chromatography coupled to ion mobility spectrometry (GC-IMS). This approach has been applied to 701 samples of the categories EVOO, VOO, and LOO, from two different olive oil harvests (2014–2015 and 2015–2016). The study has been divided in two parts: on the one hand we have studied the two crops covering the years 2014–2016 with the aim of improving the results obtained in a work related to the same dataset (Contreras et al., 2019b) and on the other hand we have applied well known algorithms in the literature to these same harvests in order to compare them with our methodology.

The article is organized as follows: section 2 provides a detailed description about the technique used to obtain the data and the algorithm and methodologies applied to carry out the classification task. Section 3 shows the results obtained with the previous techniques, and finally, section 4 samples the conclusions that have been obtained after the study.

2. MATERIALS AND METHODS

In this study, we aimed at providing a data mining approach based on Deep Learning techniques to classify olive oil samples based on chemical data. The main goal is to provide a computational methodology to help and complement the standardized sensory analysis according to the panel test method (Circi et al., 2017). The process followed is known as Knowledge Discovery in Databases (KDD). According to Lara Torralbo (2014), the KDD process pursues the automated extraction of non-trivial, implicit, previously unknown and potentially useful knowledge from large volumes of data. In summary, it can be said that KDD is a term that refers to the whole process of knowledge extraction encompassing certain phases or stages as can be seen in **Figure 1**.

The stages can be summarized as follows:

- Data acquisition and selection: In this phase, data from different sources are integrated into a single data repository, creating a target dataset with interesting variables or data samples, on which discovery is to be performed.
- Preprocessing: It might not be possible to perform data mining on the data collected in the dataset, because the data may not be clean, may contain irrelevant attributes, etc. Different types of data selection, cleaning and transformation techniques are applied in this phase, e.g., feature selection, data cleaning.
- Transformation: The data mining algorithms that will be used in the later phase sometimes need to have a specific data input format. The transformation phase is in charge of this task, with techniques such as normalization or auto-scaling.
- Data Mining: this part of the process is in charge of solving the main problem presented, using classification, regression, among others.
- Evaluation: After obtaining the data mining models, the last step of the KDD process consists of evaluating the quality of these models and interpreting them to obtain the desired knowledge. In general, in order to evaluate a model, a small subset of the data (test set) is reserved and used to validate the model built with the rest of the data (training set). This approach is known as simple validation.

This process is not static, that is, it can vary depending on the problem, taking into account the nature of the data chosen to decide whether to follow all phases, add extra phases or just follow some of them. We have mainly carried out four stages: data acquisition, data visualization techniques, data

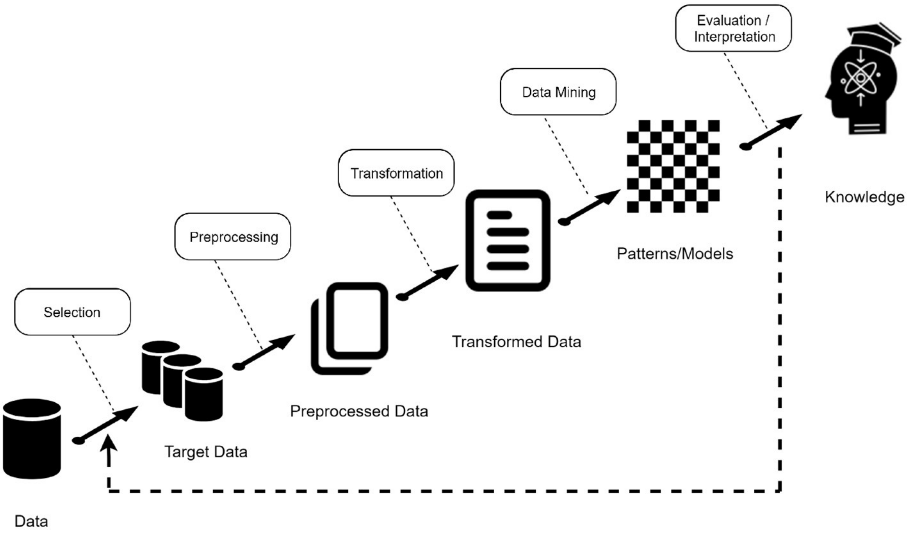


FIGURE 1 | Steps in the data analysis methodology.

preprocessing, classification models and finally a validation stage of the proposed model.

2.1. Data Acquisition

2.1.1. GC-IMS Analysis

Analyses of olive oil samples were carried out with a GC-IMS commercial instrument (FlavourSpec®). The IMS module was equipped with a tritium radioactive ionization source of 6.5 KeV and a drift tube of 5 cm long (Gesellschaft für Analytische Sensorsysteme mbH, G.A.S., Dortmund, Germany). A non-polar column (94% methyl-5% phenyl-1% vinylsilicone) with 30 m of length, an internal diameter of 0.32 mm and 0.25 μm of film thickness (SE-54-CB of CS-Chromatographie Service GmbH, Düren, Germany) was coupled to the IMS device. In addition, an automatic sampler unit (CTC-PAL, CTC Analytics AG, Zwingen, Switzerland) was employed to improve the reproducibility of measurements. The GC-IMS method for olive oil analysis was obtained from a previous work by Contreras et al. (2019b). The sample introduction system employed was a headspace generated in a 20 mL glass vial closed with magnetic cap and silicone septum. Then, 1 g of olive oil was placed in that vial and the sample was heated at 60°C for 8 min. The automatic injection of 200 μL of headspace was carried out with a heated syringe (80°C) into the heated injector (80°C). The injected headspace was driven into the GC column by using nitrogen 5.0 as carrier gas at 5 mL min^{-1} the first 6 min and then it was increased to 25 mL min^{-1} until the end of the analysis (23 min). Neutral analytes were separated at 40°C. Later, this neutral volatiles were introduced into the IMS ionization chamber to generate their corresponding ions. The generation of ions of this IMS device takes place due to the presence of an excess reagent whose signal is called reactant ion peak (RIP) which is always registered in the measurements. In positive polarity, the RIP consist on hydrated protons generated due to the collision of primary electrons emitted by the tritium source with nitrogen, and a subsequent series of reactions. When one analyte (M) enters into the ionization chamber, the corresponding ion is formed due to the association of M to this hydrated proton resulting in the displacement of water molecules (Jurado-Campos et al., 2018). Then, the ions were separated in the drift tube working at a constant temperature and voltage of 55°C and 400 V cm^{-1}, respectively. A counter-current gas flow of nitrogen was also used (drift gas) at a 250 mL min^{-1} rate. This flow is necessary to eliminate neutral molecules in the drift tube and influences the separation of ions in it. The values of different IMS parameters were set at: 32 for average of scans for each spectrum acquired, 100 μs for grid pulse width, 21 ms for repetition rate and 150 kHz for sampling frequency. Finally, two-dimensional GC-IMS data were acquired in positive mode, represented as topographic plots in LAV software (version 2.0.0) from G.A.S. So that, each individual signal or marker included in these 2D maps is characterized by the retention time of the neutral compound in the GC column, the drift time of the ion generated in the IMS (the time that the swarm of ions spend traveling along the drift tube) and its intensity value which depends on the concentration. The intensity of each marker can be automatically obtained from the topographic plots using LAV quantification module tool of the software.

2.1.2. Datasets

We analyzed 292 olive oil samples from the 2014–2015 harvest and 409 samples from the 2015-2016 harvest, henceforth named datasets D1, and D2. For D1, the 292 olive oil samples are divided in 98 EVOO, 159 VOO and 35 LOO samples. D2 harvest was composed by 92 EVOO examples, 196 VOO and 121 LOO.

The structure of the dataset for harvest D1 and D2 is the same, i.e., the datasets have a total of 118 attributes, with 113 being intensity of the markers (Contreras et al., 2019b) and the remaining others indicate the identifier of the sample

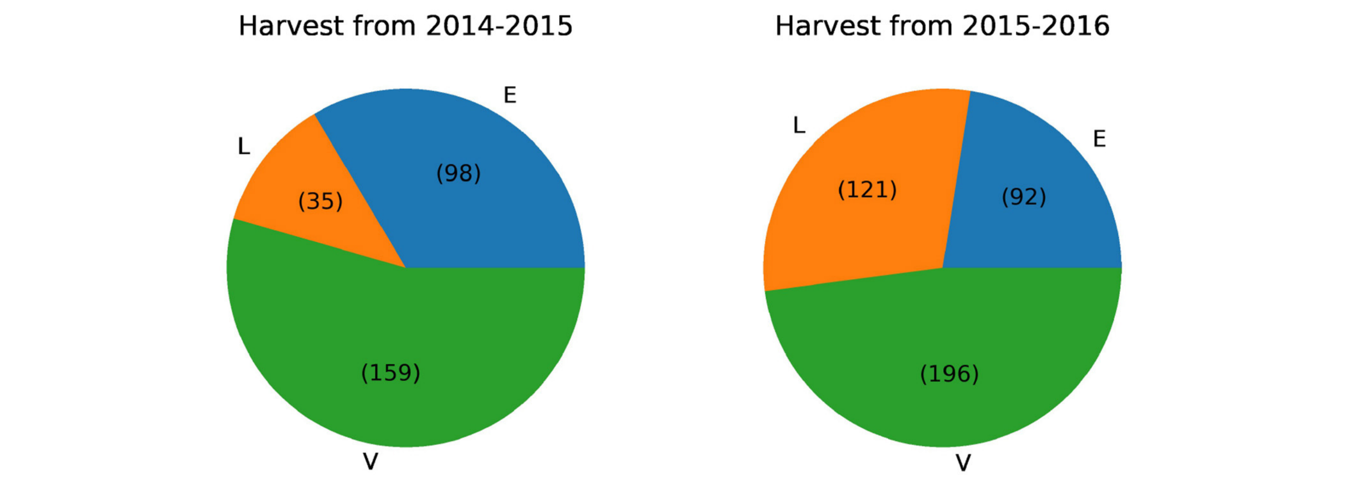


FIGURE 2 | Number of instances for each olive oil class in harvests from 2014 to 2016.

("Name"), the class (EVOO, LOO, VOO) to which it belongs ("Class"), the base value ("Baseline"), the position of the RIP ("RIP Position") and the maximum intensity of the RIP ("RIP Height") respectively.

2.2. Visualization

Before applying data analysis techniques it is important to know the nature of the data. The stage of visualization undertakes this task. In this section we provide some graphical information about the dataset analyzed. In particular, two different visualizations have been carried out: first, we show the proportion of each type of olive oil sample using pie charts and second, we reported results from principal component analysis to describe possible partitions in the dataset.

Figure 2 reports the proportion of each type of olive oil in the different harvests using a pie plot graphic. It can be seen that the two harvests have very few instances of EVOO compared to the last. For this reason we decided to merge these harvests into one. This union serves to improve the classification algorithm results since the training set will have more instances. After this union the distribution of instances is 190 EVOO, 355 VOO, and 156 LOO.

Furthermore, a principal component analysis (PCA) has been carried out. This study aims at a priori determination of the number of possible existing partitions. **Figure 3** illustrates data distribution into the first two components of PCA-analysis for 2014–2016 harvest. According to these figures there is not an a priori clear separation among classes, and therefore we decide to apply Deep Learning techniques to this problem. Deep learning techniques are able to learn a meaningful latent space, i.e., find and represent relationships among attributes that are not known a priori and are suitable for the olive oil classification problem.

2.2.1. Preprocessing

Two fundamental tasks were carried out in the preprocessing phase: the normalization of samples with respect to RIP Height in order to reduce potential instrumental variations and auto-scaling of markers that may improve the results obtained in the classification task. First, the normalization is made by dividing each of the values of markers for the maximum value of the RIP, in order to work with more homogeneous data. Second, after carrying out several tests, we found out that the auto-scaling (sometimes also called, standardization, or z-transformation) of markers resulted in slightly improved classification results. Thus, each column of the dataset was auto-scaled, i.e., numeric columns will have zero mean and unit variance. The equation used to do this task is the following:

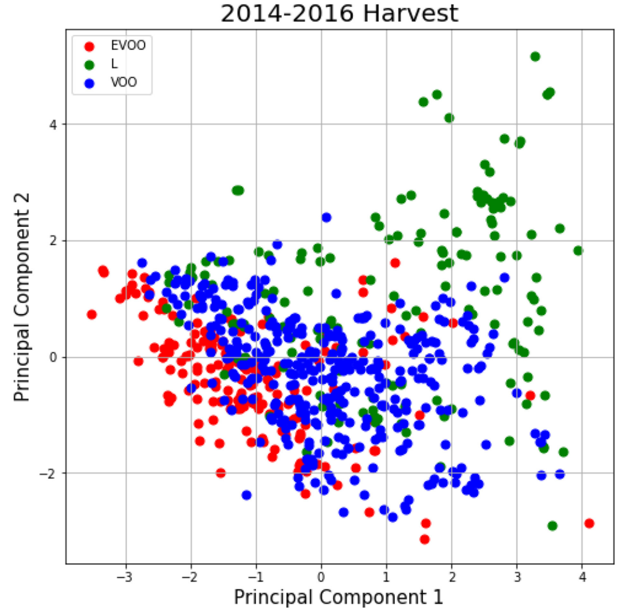


FIGURE 3 | PCA for the 2014–2016 harvest.

$$z_1 = \frac{x_i - \bar{x}}{s} \tag{1}$$

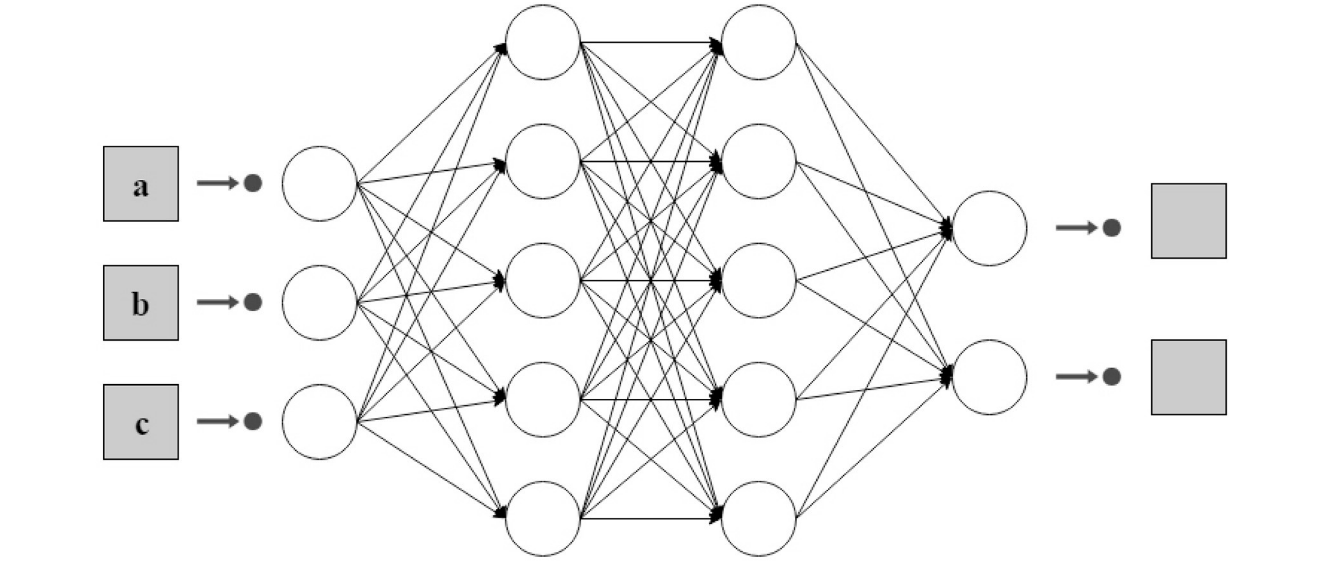


FIGURE 4 | An example of and architecture of two hidden layers for a dataset with three attributes and two possible classes.

where:

- z_1: marker auto-scaled,
- x_1: marker we want to auto-scale,
- $\bar{x}$: mean of the values for the marker,
- s: sample standard deviation.

2.2.2. Classification Task

For the classification task, a feed forward artificial neural network was used. An artificial neural network is a computational learning algorithm based on the architecture of the biological neural networks of the brain (Gibson and Patterson, 2016). These networks seek at finding a function that approximates data input into a desired output (DeepAI contributors, 2018). The architecture of an artificial neural network is determined by three main elements, nodes, connections between nodes and layers. Nodes are elements that try to model the neurons of the biological brains. The connections between nodes, such as synapses in brains, allow signals to be transmitted from one node to another. The combination of neurons are called a layer, the set of one or more layers constitutes the neural network. There are three types of layers: input, hidden and output. The input layer is composed of neurons that receive data of the problem that is under study. In this case, the input layer obtains the data of each of the features of the dataset, in our problem markers of the harvests. The hidden layers are those between the input and the output, so they do not have a direct connection to the environment. The output layer is the one that is responsible for providing the classification result obtained after applying the learning algorithm. Depending on the number of layers and the direction in which the information flows, several types of neural networks can be distinguished (Larranaga et al., 2019). A multilayer feed forward network (Gibson and Patterson, 2016) has been used to classify olive oils in our study. A multilayer feed-forward network is a neural network with an input layer, one or more hidden layers, and an output layer where each layer has one or more artificial neurons as can be seen in **Figure 4**.

Input layer. This is the first layer of a feed forward neural network. It receives the information of the problem, i.e., the input dataset. The number of neurons in this layer is usually the same as the number of attributes of the problem under study. Input layers in classical feed-forward neural networks are fully connected to the next hidden layer.

Hidden layer. The number of hidden layer in a feed forward neural network depends on the problem. Hidden layers are in charge of encoding and transporting the information extracted from the dataset to the following layers. These layers are also the key that allow neural networks to model non-linear functions.

Output layer. This layer is the one that allows to obtain the prediction of the model on the data. Depending on the nature of the problem, this prediction can be a real value (regression) or a set of probabilities (classification). To obtain these values, the corresponding activation function is chosen. In our case we have chosen the *softmax* function that represents the distribution of probability over *K* different outputs. In our example, the output is a vector with three values (or two values depending on whether the model is ternary or binary) that indicates the probability that an example belongs to one class or another.

2.2.3. Validation

The previous study carried out on this same dataset (Contreras et al., 2019b) used the accuracy as the validation metric. In order to compare with the previous results we decided to take this measure to validate the generated model. Accuracy is defined as the percentage of correctly classified examples from the dataset. To calculate it, it is necessary to take a look at the confusion matrix. If we define two variables, P for the positive instances

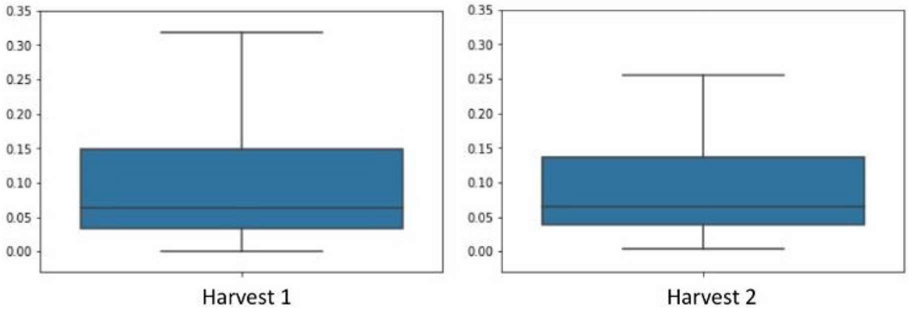


FIGURE 5 | Distribution of averages of each marker for each of the harvests.

and N for the negative ones, a confusion matrix is a table that allows for the visualization of the performance of an algorithm, typically a supervised learning one. It is a table with four different combination values: the rows indicate the predicted values by model and the columns represent the actual value of the class.

Taking into account the values of the confusion matrix, the accuracy score can be defined as follows:

$$accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{2}$$

where:

- TP (True Positive): values correctly classified as positive.
- FP (False Positive): Predicted values with negative label but which actually belong to the positive class.
- FN (False Negative): incorrectly predicted as negative values because their real value is positive.
- TN (True Negative): correctly predicted values as negative since they actually belong to the negative class.

In multi-class classification with N classes, the confusion matrix has N*N different values and the accuracy score can be obtained in two different ways by the *one vs. all* approach or by the *one vs. one*. The one vs. all approach involves training a single classifier per class, with the samples of that class as positive samples and the remaining as negatives. Finally, accuracy is obtained as a mean of each of the accuracy obtained individually for each class. In the other hand, the one vs. one approach considers each binary pair of classes and trains the classifier on a subset of data containing those classes. During the classification task, each classifier predicts one class, and the class which has been predicted the most is the answer (voting scheme). In this case, one vs. all methodology was used.

Due to the imbalance between the classes, we have also decided to take into account other more appropriate measures: sensitivity and specificity. This measures can be defined as follows:

$$sensitivity = \frac{TP}{TP + FN} \tag{3}$$

$$specificity = \frac{TN}{FP + TN} \tag{4}$$

2.3. Software and Experimental Setting

The neural networks used in this study have been implemented with the Keras library (Chollet et al., 2015). Keras is a high-level neural networks API (application programming interface), written in Python and capable of running on top of Tensorflow. The standardization of the data as well as the division of the training set in train and test has been carried out with the scikit-learn library (Pedregosa et al., 2011). The selection of parameters of the model for each of the harvests involved executing the code as many times as the number of possible neurons in the hidden layer. Due to the large amount of data available, the executions were performed on an Intel machine, specifically Intel(R) Core(TM) i7-8700 CPU @ 3.20 GHz, with 64 GB of RAM and 12 cores. The source code with the different tests performed in this study can be found in Vega (2019).

3. RESULTS

3.1. Preprocessing of the Data

First, a preprocessing step was performed, this step includes two sub-processes: in the first place a normalization of the data with respect to the maximum height of the RIP was carried out, i.e., each one of the samples is divided by the maximum value of intensity found in each one of them, in order to avoid the variations that can be introduced by the instrumental equipment used. Second, an auto-scaling of the data was carried out since as a previous study (Han et al., 2003) showed that data auto-scaling is a necessary step to improve final classification results. Furthermore, LeCun et al. (2012) have shown that the convergence of Deep Learning models is usually greater if the mean of each of the variables of the training set is close to zero. Because of this, we have auto-scaled the data in order to obtain better results with Deep Learning techniques.

As we mentioned before, the chemical method used to obtain the data from D1 and D2 are the same being the number of markers equal for each case. Thus, after data auto-scaling, a union of the datasets D1 and D2 was carried out in order to study them as a whole, henceforth named D1–D2. We could observe in the **Figure 5** that the distribution of averages for each column of the dataset were very similar between D1 and D2, which is another motivation behind our decision to merge the two crops.

3.2. Use of Deep Learning Models for Classification of Olive Oil Samples

The capabilities of a neural network to make good predictions depends on its architecture and its parameters, it is an essential task to define a well structured network before implementing the model. Parameters which define the model architecture are known as hyperparameters and the process of assessing the best configuration for those parameters is called hyperparameter tuning (Diaz et al., 2017).

For the present study, multilayer and unidirectional (feed-forward) neural networks have been used, with an input layer, a hidden and an output layer, with a flow of information that run from the entrance to the exit, only in one direction.

The first step was to improve the classification of the model varying the values for the activation function and optimization algorithm. The best results were obtained with Rectified Linear Unit function (RELU) and Adam algorithm, respectively. The second step was to choose the optimal number of hidden layers for this particular problem. Finally, the number of the neurons in the hidden layers was optimized. Taking as a guide the rules of thumb (Heaton, 2008) and the geometric pyramid rule (Masters, 1993) that will be explained below, experiments were performed for datasets D1–D2 as a whole. In each of these experiments tests were made varying the number of neurons looking for the number that provided the best results.

3.2.1. Choosing the Number of Hidden Layers

The universal approximation theorem (Csáji, 2001) states that a feed-forward network with only a single hidden layer containing a finite number of neurons can approximate continuous functions as well as other interesting functions when appropriate parameters are given. The use of more than one hidden layers are better for complex datasets than involves time-series or computer vision. The dataset of this study does not belongs to any of these two categories, so we considered that one hidden layer is the best approach.

3.2.2. Choosing the Number of Neurons in the Hidden Layer

Deciding the correct number of hidden layers is only one part of the problem. The correctness of the model also depends on the number of the neurons in the hidden layers. There are a lot of theorems that provide a first approximation for this issue. The one selected for our research is called "geometric pyramid rule" proposed by Masters in Masters (1993). Basically, this rule asserts that there is no magic formula for selecting the optimum number of hidden neurons although it provides a rough approximation for different structure, e.g., for a three layer network with n input and m output neurons, the hidden layer would have $\sqrt{n \times m}$ neurons.

Besides the geometric pyramid rule, a few rules of thumb methods (Heaton, 2008) have been considered for determining an acceptable number like the following:

- The number of hidden neurons should be between the size of the input layer and the size of the output layer.
- The number of hidden neurons should be 2/3 the size of the input layer, plus the size of the output layer.
- The number of hidden neurons should be less than twice the size of the input layer.

TABLE 1 | Number of neurons chosen for the hidden layer.

	2014–2016 (D1-D2)	
	Non-standardized	**Standardized**
EVOO/VOO/LOO	32	40
EVOO/non-EVOO	10	3
LOO/non-LOO	68	53

Taking into account these previous rules, we decided to train the model for each of the possible combinations of neurons considering the inputs and outputs of the neural network according to the harvests under study: for D1-D2, tests were carried out varying the neurons from 2 to 3 to 113 depending on whether the model distinguishes between two classes (binary model), that is, between lampante and no lampante (LOO/non-LOO) or extra or no extra (EVOO/non-EVOO), or between three (EVOO/VOO/LOO). The **Table 1** shows among all the possible values of neurons, the one that maximizes the accuracy value for each of the tests. As it can be seen in this table, the number of neurons for each case is completely different, there is no single number that ensures the total quality of the model. Although this process has been very time consuming, it is totally necessary since it is the first time that Deep Learning techniques have been used with these specific data, so it was convenient to see each one of the cases. For future studies we propose training the neural network with many more examples in order to further homogenize the parameter selection of the model.

3.3. Training the Model

A train-test split method was used for the validation of the model. A training set containing 80% of the samples was used for the calibration of the models and the remaining 20% of the samples were used as a validation or blind test. The performance of the neural network was shown by the accuracy score.

A total of 6 tests with the data from D1 to D2 has been carried out. The models were tested with auto-scaled and non auto-scaled data, as well as the division of tests according to the type of oil. For each of the tests the optimum number of neurons in the hidden layer has been calculated, so that for each test a model has been made for each of the possible neurons in the hidden layer according to the rules described above, specifically 110 iterations for the model that discriminates between the 3 classes (the number of neurons must be between the output number and the number of input neurons) and 111 for those that distinguish between two classes.

3.3.1. Results Obtained for 2014–2016 Harvests

A total of 701 samples from 2014–2016 harvests were studied. The Deep Learning model was built using 80% of these samples (a total of 531 olive oil samples, of which 286 where VOO,

TABLE 2 | Results obtained for 2014–2016 harvests.

	Previous results	Our Results		Rate of increase (%)
		Non standardized	Standardized	
EVOO/VOO/LOO	74.29	80.71	81.42	9.59
EVOO/non-EVOO	85.72	88.57	90.00	4.99
LOO/non-LOO	90.71	94.28	95.00	4.72

149 EVOO, and 126 LOO) and the remaining 20% to evaluate the model (69 VOO, 41 EVOO, and 30 LOO). To compare our results to those obtained by Contreras et al. (2019b), we have replicated each of their tests, obtaining 3 different models: 2 binary models and 1 ternary model. The first binary model allows to differentiate between EVOO and non-EVOO examples, the second model discriminates between LOO and non LOO, and finally, the ternary model discriminates between all classes, i.e., among EVOO, LOO, and VOO. As mentioned above, auto-scaling seemed to be a good preprocessing task that should be carried out with this dataset, so the three models obtained have also been carried out in two different ways, first without auto-scaling the data and second with auto-scaled data.

Table 2 shows the comparison of results between our study and the existing previous study as well as the accuracy increase ratio. We can observe that our results improve the results obtained without auto-scaling (see column 2 and 3). Furthermore, the part in which our results are shown verifies that a previous preprocessing of the data is a good technique, as it improves the results in comparison to those obtained without this preprocessing. On the other hand, if we compare our results with the previous results, we see a significant improvement for each of the three models studied, with the rate of increase always positive.

3.4. Comparison to Other Methods

Our methodology has been compared to five different benchmark methods: K-Nearest Neighbors (Altman, 1992), Support Vector Machine (Boser et al., 1992), Decision Tree Classifier (Safavian and Landgrebe, 1991), Logistic Regression (Scott et al., 1991) and XGBoost (Chen and Guestrin, 2016). The data used for comparison are the auto-scaled data, since the objective of this comparison is to provide a comparative framework on the best results obtained with the proposed methodology.

We have evaluated these methods for D1-D2 harvest data. Firstly, for accuracy score (**Table 3**), in EVOO/VOO/LOO model, among the five models used for comparison, XGBoost offers the best performance. In the case of EVOO/non-EVOO model, XGBoost is as good as k-NN. Lastly, LOO/non-LOO model gets higher performance with Logistic Regression. Although the results are quite satisfactory with benchmark algorithms, none of the models achieves better results than our Deep Learning proposal if we take into account the average of the three models (last row). It can be seen that the best value (in bold) is always the one in the first column, which is the one corresponding to Deep Learning. Lastly, for sensitivity (**Table 4**) and specificity (**Table 5**) our proposal is the best if we also take into account the average of the three models.

TABLE 3 | Accuracy comparison with other methods for 2014–2016 (D1-D2) harvests.

	Deep learning	SVM	k-NN	Tree	Regressor	XGBoost
EVOO/VOO/LOO	81.42	73.57	77.14	68.57	77.85	80.71
EVOO/non-EVOO	90.00	85.71	85.71	82.14	85.71	86.42
LOO/non-LOO	95.00	90.00	90.71	84.28	92.85	90.00
	88.81	83.09	84.52	78.33	85.47	85.71

TABLE 4 | Sensitivity comparison with other methods for 2014–2016 (D1-D2) harvests.

	Deep learning	SVM	k-NN	Tree	Regressor	XGBoost
EVOO/VOO/LOO	63.47	55.82	59.33	49.76	61.52	64.11
EVOO/non-EVOO	68.29	68.29	63.41	60.97	68.29	68.29
LOO/non-LOO	80.00	56.66	63.33	60.00	76.66	63.33
	70.58	60.25	62.02	56.91	68.82	65.24

TABLE 5 | Specificity comparison with other methods for 2014–2016 (D1-D2) harvests.

	Deep learning	SVM	k-NN	Tree	Regressor	XGBoost
EVOO/VOO/LOO	87.55	83.57	85.45	80.00	86.58	87.81
EVOO/non-EVOO	93.93	92.92	94.94	90.90	92.92	93.93
LOO/non-LOO	98.18	99.09	98.18	90.09	97.27	97.27
	93.22	91.86	92.85	86.99	92.25	93.00

4. DISCUSSION

Deep Learning techniques are proving to be one of the best tools when performing complex tasks that require expert knowledge (Arel et al., 2010; LeCun et al., 2015). In this study we used Deep Learning techniques to provide an automatic complement to the panel test method. This is an essential task to avoid fraud in the price and to know whether the olive oil is suitable for consumption or not. This work has shown the feasibility of a feed forward artificial neural networks-based model as a classifier to differentiate EVOO, VOO, and LOO oil from GC-IMS spectroscopy data.

The preprocessing step should be highlighted since the auto-scaling of data has been a fundamental part of the study carried out. This step has meant an improvement in the classification algorithms as can be seen in **Table 2**.

This study also shows that the neural network architecture must be different for each of the potential models. The fact that the number of neurons in the hidden layer is different for each of the models (binary or ternary) is not surprising; indeed, we would even say it is necessary, due the fact that the network must be adapted to the input data.

Until now, the best works on oil classification (Contreras et al., 2019b; Gonzalez-Fernandez et al., 2019) worked in a similar way to our proposal: they first made a chemical treatment to obtain the data, and then applied some mathematical model to carry out the olive oil classification. The main advantage of our approach is that there is a searching for the most suitable parameters, thus achieving a better adaptation to the input data to achieve the most accurate results.

One of the objectives of this work has been trying to improve the results obtained by Contreras et al. (2019a) with D1 and D2 harvests. Considering that in that previous work they obtained an accuracy of the 74.29% using techniques such as PCA and OPLS-DA, our work, with an accuracy of 81.42%, has shown that Deep Learning techniques are a very useful tool to classify olive oil samples from GC-IMS data.

Additionally, regardless of the number of neurons used, the best results are obtained for binary models, especially the model that classify between LOO and non-LOO. This may be due to the fact that in the case of the ternary model, the elements are more difficult to split since the VOO is at the crossroads between EVOO and LOO, which means that the separation between classes is not so clear.

We have also studied the performance obtained by five different benchmark methods: k-Nearest Neighbors, Support Vector Machine, Logistic Regression, Decision Tree Classifier and XGBoost. Although the performance of these algorithms is satisfactory, in none of the cases they have improved our Deep Learning approach.

Finally, some limitations of our study should be noted and discussed. First, it is known that the success of a Deep Learning algorithm lies in the amount of data available to train. In this case, we have only a total of 701 examples. For further studies, we propose to create synthetic data with Conditional Generative Adversarial Networks as proposed in Vega-Márquez et al. (2020). Lastly, another major problem we have encountered is the imbalance between classes, in olive oil industry is common to have more instances from VOO that LOO and EVOO. In order to address this issue we propose to employ Machine Learning algorithms as SMOTE (Chawla et al., 2002) to balance classes.

DATA AVAILABILITY STATEMENT

The datasets for this manuscript are not publicly available because the data has been obtained and treated only for analysis as shown in this article. For any further use new access has to be granted. Requests to access the datasets should be directed to Cristina Rubio-Escudero, crubioescudero@us.es.

AUTHOR CONTRIBUTIONS

BV-M has contributed with the implementation, design of tests, and manuscript redaction. NJ-C has contributed with the data acquisition, first analysis, and manuscript redaction. IN-C and CR-E have contributed with the implementation and tests supervision and manuscript correction.

ACKNOWLEDGMENTS

The authors want to thank for the financial support given by the Spanish Ministry of Economy and Competitivity project TIN2017-88209-C2-2-R. NJ-C wishes to thank the Spanish Ministry of Education, Culture and Sport for award of a pre-doctoral grant. Lastly the authors want to thank to Spanish Olive Oil Interprofessional for providing the samples.

REFERENCES

Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *Ame. Statist*. 46, 175–85.

Arel, I., Rose, D., and Karnowski, T. (2010). Deep machine learning-A new frontier in artificial intelligence research. *IEEE Comput. Intell. Magaz*. 5, 13–18. doi: 10.1109/MCI.2010.938364

Borràs, E., Ferré, J., Boqué, R., Mestres, M., Aceña, L., Calvo, A., et al. (2016). Olive oil sensory defects classification with data fusion of instrumental techniques and multivariate analysis (PLS-DA). *Food Chem.* 203, 314–322. doi: 10.1016/j.foodchem.2016.02.038

Borràs, E., Mestres, M., Aceña, L., Busto, O., Ferré, J., Boqué, R., et al. (2015). Identification of olive oil sensory defects by multivariate analysis of mid infrared spectra. *Food Chem.* 187, 197–203. doi: 10.1016/j.foodchem.2015.04.030

Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). "Training algorithm for optimal margin classifiers," in *Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory* (New York, NY: ACM). doi: 10.1145/130385.130401

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* 16, 321–357. doi: 10.1613/jair.953

Chen, T. and Guestrin, C. (2016). "XGBoost: a scalable tree boosting system," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY: ACM).

Chollet, F., et al. (2015). Keras. Available online at: https://keras.io/getting-started/faq/#how-should-i-cite-keras

Circi, S., Capitani, D., Randazzo, A., Ingallina, C., Mannina, L., and Sobolev, A. P. (2017). Panel test and chemical analyses of commercial olive oils: a comparative study. *Chem. Biol. Technol. Agricult.* 4:18. doi: 10.1186/s40538-017-0101-0

Contreras, M. D. M., Arroyo-Manzanares, N., Arce, C., and Arce, L. (2019a). HS-GC-IMS and chemometric data treatment for food authenticity assessment: olive oil mapping and classification through two different devices as an example. *Food Control* 98, 82–93. doi: 10.1016/j.foodcont.2018.11.001

Contreras, M. D. M., Jurado-Campos, N., Arce, L., and Arroyo-Manzanares, N. (2019b). A robustness study of calibration models for olive oil classification: targeted and non-targeted fingerprint approaches based on GC-IMS. *Food Chem.* 288, 315–324. doi: 10.1016/j.foodchem.2019.02.104

Csáji, B. (2001). *Approximation with Artificial Neural Networks*. Faculty of Sciences, Etvs Lornd University.

Dębska, B. M., and Guzowska-Świder, B. (2011). Application of artificial neural network in food classification. *Anal. Chim. Acta* 705, 283–291. doi: 10.1016/j.aca.2011.06.033

DeepAI contributors (2018). *Neural network—DeepAI*. Available online at: https://deepai.org/machine-learning-glossary-and-terms/neural-network (accessed December 26, 2018).

Diaz, G. I., Fokoue-Nkoutche, A., Nannicini, G., and Samulowitz, H. (2017). An effective algorithm for hyperparameter optimization of neural networks. *IBM J. Res. Dev*. 61, 9:1–9:11. doi: 10.1147/JRD.2017.2709578

Du, H., Wang, J., Hu, Z., Yao, X., and Zhang, X. (2008). Prediction of fungicidal activities of rice blast disease based on least-squares support vector machines and project pursuit regression. *J. Agricul. Food Chem*. 56, 10785–10792. doi: 10.1021/jf8022194

EEC (1991). European Commission Regulation (EEC). European Commission Regulation EEC/2568/91 of 11 July on the characteristics of olive and pomace oils and on their analytical methods. *Off. J. Eur. Communit.* L248, 1–82.

Garrido-Delgado, R., del Mar Dobao-Prieto, M., Arce, L., and Valcárcel, M. (2015). Determination of volatile compounds by GC–IMS to assign the quality of virgin olive oil. *Food Chem.* 187, 572–579. doi: 10.1016/j.foodchem.2015.04.082

Gibson, A., and Patterson, J. (2016). *Deep Learning A Practitioner's Approach*. Sebastopol, CA: O'Reilly Media, Inc.

Gonzalez-Fernandez, I., Iglesias-Otero, M. A., Esteki, M., Moldes, O. A., Mejuto, J. C., and Simal-Gandara, J. (2019). A critical review on the use of artificial neural networks in olive oil production, characterization and authentication. *Crit. Rev. Food Sci. Nutrit.* 59, 1913–1926. doi: 10.1080/10408398.2018.1433628

Han, H., Giles, C. L., Manavoglu, E., Zha, H., Zhang, Z., and Fox, E. A. (2003). "Automatic document metadata extraction using support vector machines," in *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries* (Houston, TX). doi: 10.1109/JCDL.2003.1204842

Heaton, J. (2008). *Introduction to Neural Networks for Java, 2nd Edn*. St. Louis, MS: Heaton Research, Inc.).

Jurado-Campos, N., Garrido-Delgado, R., Martínez-Haya, B., Eiceman, G. A., and Arce, L. (2018). Stability of proton-bound clusters of alkyl alcohols, aldehydes and ketones in ion mobility spectrometry. *Talanta* 185, 299–308. doi: 10.1016/j.talanta.2018.03.030

Lara Torralbo, J. A. (2014). *Mineria de datos*. Madrid: Centro de Estudios Financieros.

Larranaga, P., Inza, I., and Moujahid, A. (2019). *Tema 8. Redes Neuronales*. Universidad del Pais Vasco, 12–17.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature* 521, 436–444. doi: 10.1038/nature14539

LeCun, Y. A., Bottou, L., Orr, G. B., and Müller, K. R. (2012). "Efficient backprop," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, eds G. Montavon, G. B. Orr, Müller K. R (Berlin; Heidelberg: Springer), 9–48. doi: 10.1007/978-3-642-35289-8_3

Masters, T. (1993). *Practical neural network recipes in C++*. San Diego, CA: Academic Press Professional, Inc. doi: 10.1016/B978-0-08-051433-8.50001-X

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* 12, 2825–2830. Retrieved from: https://dl.acm.org/journal/jmlr

Riul, A., de Sousa, H. C., Malmegrim, R. R., dos Santos, D. S., Carvalho, A. C., Fonseca, F. J., et al. (2004). Wine classification by taste sensors made from ultra-thin films and using neural networks. *Sensors Actuat. B Chem.* 98, 77–82. doi: 10.1016/j.snb.2003.09.025

Safavian, S. R., and Landgrebe, D. (1991). A survey of decision tree classifier methodology. *IEEE Trans. Syst. Man Cybernet*. 21, 660–674. doi: 10.1109/21.97458

Sales, C., Cervera, M. I., Gil, R., Portolés, T., Pitarch, E., and Beltran, J. (2017). Quality classification of Spanish olive oils by untargeted gas chromatography coupled to hybrid quadrupole-time of flight mass spectrometry with atmospheric pressure chemical ionization and metabolomics-based statistical approach. *Food Chem*. 216, 365-373. doi: 10.1016/j.foodchem.2016.08.033

Scott, A. J., Hosmer, D. W., and Lemeshow, S. (1991). Applied logistic regression. *Biometrics*. 47, 1632–1633. doi: 10.2307/2532419

Svetnik, V., Liaw, A., Tong, C., Christopher Culberson, J., Sheridan, R. P., and Feuston, B. P. (2003). Random forest: a classification and regression tool for compound classification and QSAR modeling. *J. Chem. Informat. Comput. Sci*. 43, 1947–1958. doi: 10.1021/ci034160g

Vega, B. (2019). *oliveoil*. Available online at: https://github.com/bvegaus/oliveOil

Vega-Márquez, B., Rubio-Escudero, C., Riquelme, J. C., and Nepomuceno-Chamorro, I. (2020). "Creation of synthetic data with conditional generative adversarial networks," in *14th International Conference on Soft Computing Models in Industrial and Environmental Applications (SOCO 2019)*, eds F. Martínez Álvarez, A. Troncoso Lora, J. A. Sáez Muñoz, H. Quintián, and E. Corchado (Cham), 231–240.

Conflict 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.

Advantages of publishing in Frontiers

OPEN ACCESS
Articles are free to read
for greatest visibility
and readership

FAST PUBLICATION
Around 90 days
from submission
to decision

HIGH QUALITY PEER-REVIEW
Rigorous, collaborative,
and constructive
peer-review

TRANSPARENT PEER-REVIEW
Editors and reviewers
acknowledged by name
on published articles

Frontiers
Avenue du Tribunal-Fédéral 34
1005 Lausanne | Switzerland

Visit us: www.frontiersin.org
Contact us: info@frontiersin.org | +41 21 510 17 00

REPRODUCIBILITY OF RESEARCH
Support open data
and methods to enhance
research reproducibility

DIGITAL PUBLISHING
Articles designed
for optimal readership
across devices

FOLLOW US
@frontiersin

IMPACT METRICS
Advanced article metrics
track visibility across
digital media

EXTENSIVE PROMOTION
Marketing
and promotion
of impactful research

LOOP RESEARCH NETWORK
Our network
increases your
article's readership