Extracting Predictive Representations from Hundreds of Millions of Molecules

Dong Chen
School of Advanced Materials, Peking University, Shenzhen Graduate School https://orcid.org/0000-0001-5397-0447

Guowei Wei (weig@msu.edu)
Michigan State University https://orcid.org/0000-0001-8132-5998

Feng Pan
Peking University Shenzhen Graduate School https://orcid.org/0000-0002-8216-1339

Article

Keywords: Self-supervised learning, Pre-training, Virtual screening, Property prediction

Posted Date: August 10th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-745668/v1

Version of Record: A version of this preprint was published at The Journal of Physical Chemistry Letters on November 1st, 2021. See the published version at https://doi.org/10.1021/acs.jpclett.1c03058.

Extracting Predictive Representations from Hundreds of Millions of Molecules

Dong Chen[1,2], Guo-Wei Wei *[2,3,4] and Feng Pan †[1]

[1] *School of Advanced Materials, Peking University, Shenzhen Graduate School, Shenzhen 518055, China*
[2] *Department of Mathematics, Michigan State University, MI, 48824, USA*
[3] *Department of Electrical and Computer Engineering, Michigan State University, MI 48824, USA*
[4] *Department of Biochemistry and Molecular Biology, Michigan State University, MI 48824, USA*

Abstract Although deep learning can automatically extract features in relatively simple tasks such as image analysis, the construction of appropriate representations remains essential for molecular predictions due to intricate molecular complexity. Additionally, it is often expensive, time-consuming, and ethically constrained to generate labeled data for supervised learning in molecular sciences, leading to challenging small and diverse datasets. In this work, we develop a self-supervised learning approach via a masking strategy to pre-train transformer models from over 700 million unlabeled molecules in multiple databases. The intrinsic chemical logic learned from this approach enables the extraction of predictive representations from task-specific molecular sequences in a fine-tuned process. To understand the importance of self-supervised learning from unlabeled molecules, we assemble three models with different combinations of databases. Moreover, we propose a new protocol based on data traits to automatically select the optimal model for a specific predictive task. To validate the proposed representation and protocol, we consider 10 benchmark datasets in addition to 38 ligand-based virtual screening datasets. Extensive validation indicates that the proposed representation and protocol show superb performance.

Keywords Self-supervised learning, Pre-training, Virtual screening, Property prediction

*Corresponding author: weig@msu.edu
†Corresponding author: panfeng@pkusz.edu.cn

Contents

1 Introduction

In the past few years, machine learning (ML) has had profoundly changed the landscape of science, engineering, technology, finance, industry, defense, and society in general. It becomes a new approach for scientific discovery, following traditional experiments, theories, and simulations. In image analysis, deep learning algorithms, such as convolutional neural networks (CCN), can automatically extract image features without resorting to hand-crafted descriptors. However, for molecular predictions, due to the internal complexity of molecules, generating molecular representations or descriptors is an essential issue that is as important as data and algorithm in determining ML performance [1, 2]. It is a procedure that translates the chemical information in a molecule into a set of "machine" understandable features. Although many molecular descriptors of macroscopic physicochemical properties are obtained via experimental measurements, a wide variety of others has been extracted from molecular microscopic information, i.e., atomic constitution, electron density, molecular structures, etc. Various fingerprints have been developed in the past few decades.[3, 4] Two-dimensional (2D) fingerprints, such as ECFP, MACCS, Estate1, Daylight, etc. [3, 5] are a class of commonly used molecular representations and can be extracted from molecular connection tables without three-dimensional (3D) structural information. Many popular software packages generate 2D fingerprints.[6, 7] However, 2D fingerprints lack the 3D structural information of molecules, such as stereochemical information.

In recent years, molecular fingerprints based on 3D structures have been developed to capture the 3D spatial information of molecules.[8] However, the complexity and elemental diversity of molecular structures are major obstacles to the design of 3D fingerprints.[1] A variety of advanced mathematics-based 3D molecular representations, including algebraic topology[9], differential geometry[10], and algebraic graph-based methods[11, 12, 13], were devised to generate molecular fingerprints aimed at encoding 3D and elemental information of molecules by mathematical abstraction. These methods have been highly successful in the classification of proteins and ligands, as well as in the prediction of solubility, solubility free energy, protein-ligand binding affinity, protein folding stability changes after mutations, and mutation-induced protein-protein binding affinity changes [1]. However, these approaches rely on high-quality 3D molecular structures, which limits their applications.

Deep learning (DL) has been a successful and powerful tool in various fields, such as natural language processing[14], image classification[15], and bioinformatics[16, 17]. Conventional deep learning methods are constructed based on deep neural networks (DNN). In molecular sciences, the input to these models is usually a pre-extracted molecular descriptor, e.g., ECFP, MACCS. However, this type of input may not preserve certain molecular information and thus compromise the performance of downstream predictive tasks.[18, 19] Additionally, DNN usually requires a large amount of data for training, which constrains the application of some supervised learning DL methods in molecular sciences. To address these issues, data-driven unsupervised learning methods have been developed in recent years, such as recurrent neural network (RNN)-based autoencoder model[20] and variational autoencoder model[21]. These models are trained directly by a set of large and low-level molecular representations, i.e., the simplified molecular input line entry specification (SMILES) representation.[22] Some publicly available datasets, such as ChEMBL[23], PubChem[24], and ZINC[25], provide a large amount of unlabeled SMILES data, which allows DNN autoencoder to be better trained. Typically, an autoencoder consists of two neural networks, encoder, and decoder. The encoder converts the input, for example, the SMILES of a molecule, into a continuous (latent space) representation of a fixed size. The decoder, on the other hand, takes the latent space representation as input and aims to convert it into the probability distribution of the design target of interest, which can be translated into a molecule, the next possible word, or some predicted event. The entire autoencoder network is trained to minimize the error of predicting the target of the interest. The latent representation in the model is often used as a molecular fingerprint for other tasks, such as molecular property prediction[26], or virtual screening. In these tasks, the decoder is only used as a training device and does not contribute to the final prediction. The training of the decoder also takes up a large amount of computer time and memory.

In this work, we develop a self-supervised learning (SSL) platform to pre-train DL networks with over 700 million unlabeled SMILES data. With the data-mask pairs constructed from the unlabeled data, the SSL approach allows the model to be trained in a supervised learning fashion.[27] In particular, for SMILES data, we construct pairs of real SMILES and masked SMILES by hiding a certain percentage of symbols that have a specific physicochemical meaning.[13] We use a transformer model based on an attention mechanism for SSL. [28] This model has higher parallelism capability and training efficiency compared to RNN-based models. Due to the advantage of SSL, we avoid the need to construct a complete encoder-decoder framework and achieve the encoding of SMILES using only the encoder, a bidirectional encoder transformer (BET). Similar to the cloze test practice in language learning, the model inferred the symbols of the masked part by learning the unprocessed symbols in SMILES during the pre-training process, so that the purpose of understanding SMILES language can be achieved. To investigate the benefit of excessively large training datasets, we constructed three models based on ChEMBL, the union of ChEMBL and PubChem, and the union of ChEMBL, PubChem, and ZINC, with data sizes ranging from over one million to over 700 million. We show that, for a given predictive task associated with a dataset, the model trained on the largest dataset is not necessarily the best one. To enable the automatic selection of the optimal model for a specific task, we construct a dataset analysis module based on the Wasserstein distance to characterize the similarity between dataset distributions. Using this module, the optimal pre-trained model can be selected for any customized dataset. Subsequently, the selected pre-trained model is fine-tuned using the corresponding dataset to obtain a task-specific molecular fingerprint. To investigate the accuracy, robustness, and usefulness of the proposed SSL platform, we consider a total of 48 datasets, including 5 regression datasets, 5 classification datasets, and two virtual screening tasks with 17 and 21 additional datasets. Extensive numerical experiments indicate the proposed platform is an accurate and robust strategy for generating molecular data representations and ML predictions in molecular sciences.

2 Results

In this section, we present the proposed self-supervised learning platform for molecular predictions. The combination of datasets, i.e., ChEMBL[23], PubChem[24], and ZINC[25], was used as the pre-training datasets, as listed in Table 1. For the evaluation of the proposed platform, we carried out 5 classification and 5 regression tasks, as listed in the Table 1. Two VS experiments were performed on overall 38 datasets, including 21 targets of the Directory of Useful Decoys (DUD) and 17 targets of the Maximum Unbiased Validation (MUV) datasets.[29, 30]

Table 1: Three pre-training datasets and ten datsets used for benchmarking our platform.

Datasets	Task type	Compounds	Split	Metric
ChEMBL(C)[23]	Pre-train	1,941,410	-	Accuracy
ChEMBL and PubChem(CP)[24]	Pre-train	103,395,400	-	Accuracy
ChEMBL, PubChem, and ZINC(CPZ)[25]	Pre-train	775,007,514	-	Accuracy
Ames mutagenicity (Ames)[31]	Classification	6512	Random, 8:1:1	ROC-AUC
β-Secretase 1 inhibition (bace)[32]	Classification	1513	Random, 8:1:1	ROC-AUC
Blood–brain barrier penetration (bbbp)[33]	Classification	2039	Random, 8:1:1	ROC-AUC
Toxicity in honeybees (beet)[34]	Classification	254	Random, 8:1:1	ROC-AUC
ClinTox (Clinical trial results)[35]	Classification	1478	Random, 8:1:1	ROC-AUC
Aqueous Solubility (ESOL)[36]	Regression	1128	Random, 8:1:1	R^2
Lipophilicity (Lipop)[23]	Regression	4200	Random, 8:1:1	R^2
Free Solvation Database (FreeSolv)[37]	Regression	642	Random, 8:1:1	R^2
LogS[38]	Regression	4801	Random, 8:1:1	R^2
DPP-4 inhibitors (DPP4)[39]	Regression	3933	Random, 8:1:1	R^2

Self-supervised Learning Platform (SSLP). As shown in Figure 1, there are four main modules involved in the platform, which is pre-training datasets module (i.e., blue rectangle), dataset analysis module (i.e., purple rectangle), pre-trained model module (i.e., green rectangle), and fine-tune module (i.e., yellow rectangle). In the pre-training datasets module, the three pre-training datasets are obtained by combining three publicly available datasets, i.e., ChEMBL[23], PubChem[24], and ZINC[25]. Set C represents only ChEMBL and was used as the pre-training data. Set CP represents the combination of datasets ChEMBL and PubChem. And Set CPZ represents the union of all three datasets. For all three pre-training datasets, duplicated compounds were removed after the combination of datasets. In the pre-trained model module, we use a self-supervised learning strategy, especially the BET [28, 40], to obtain our pre-trained models. In the pre-training stage, we mask the unlabeled data in the dataset and then use the BET model to predict the masked parts of the SMILES for self-supervised learning. For the three pre-training datasets, i.e., set C, set CP, and set CPZ, we can obtain three pre-trained models correspond to model C, model CP, and model CPZ. Each pre-trained model can provide a self-supervised learning fingerprint (SSL-FP) for downstream tasks.

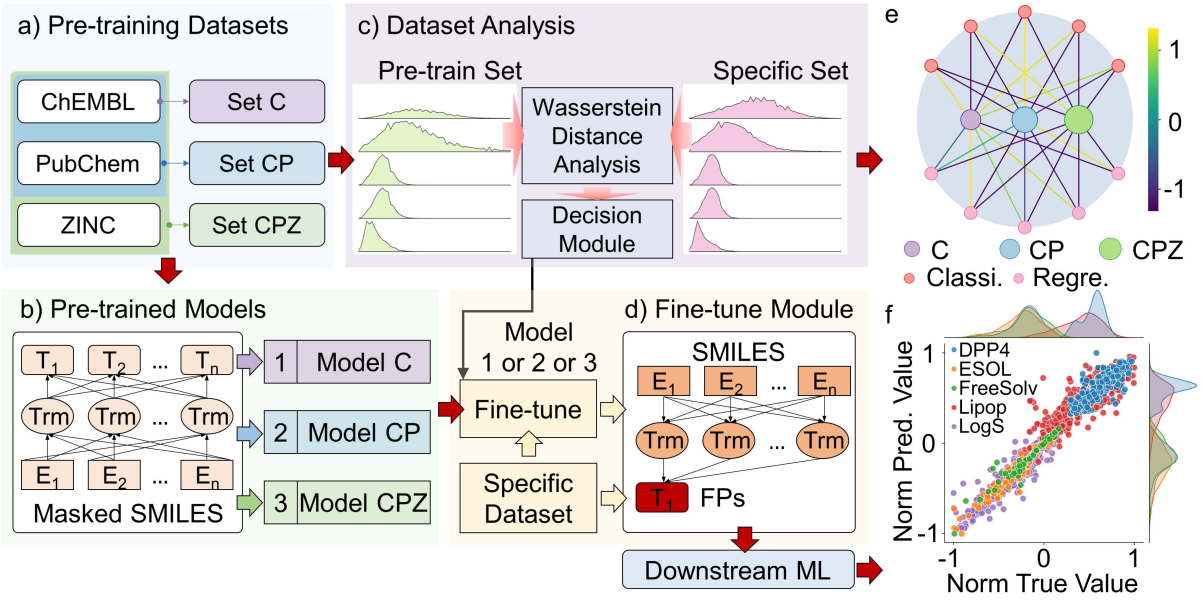

Figure 1: Illustration of the self-supervised learning platform. **a** Three public datasets are involved in the pre-training datasets module (blue rectangle). Set C only contains the ChEMBL dataset. Set CP consists of ChEMBL and PubChem datasets, and Set CPZ contains ChEMBL, PubChem, and ZINC datasets. **b** Based on those three datasets, three pre-trained models (green rectangle) are obtained by self-supervised learning, which is Model C, Model CP, and Model CPZ, respectively. **c** The dataset analysis module (purple rectangle) contains the Wasserstein distance analysis module and decision module. It will point to the best pre-trained model for a specific dataset. **d** The fine-tune module (yellow rectangle) fine-tunes the pre-trained model using a specific dataset. Finally, the fingerprints are generated from the fine-tuned model and used as input for the downstream machine learning tasks. **a**, The correlations between pre-training datasets and downstream datasets, including 5 classifications (Classif.) and 5 regressions (Regre.) datasets, and pre-trained datasets C, CP, and CPZ. **f** Normalized predicted results of the fingerprints from pre-trained model C for DPP4, ESOL, FreeSolv, Lipophilicity (Lipop), and LogS five regression datasets.

For the dataset analysis module, we use the Wasserstein distance analysis submodule and the decision submodule to decide the optimal model for the downstream task. Firstly, we generated the distribution of the proportion of each symbol in each SMILES in the dataset. The distribution of elemental symbols and the distribution of special symbols for the three pre-training datasets are shown in Figure 2a and b. It can

be found that commonly occurred elements, including carbon, oxygen, and nitrogen, in the molecule, have an abundant ratio in SMILES strings. Since some special symbols always appear in pairs in SMILES, such as '(' and ')', '[' and ']', the distribution of these symbols is the same. All 61 symbols used in this work are listed in Table S1, and the distributions of complete symbols are shown in Figure S1. Additionally, we also counted the distribution of the number of symbol types contained in each SMILES, as shown in Figure 2c. To analyze a specific dataset, we can also generate these distributions. In the second step, based on the various distributions obtained (63 in total), we use the Wasserstein distance analysis submodule to analyze the correlation between different datasets in several ways. Finally, using the decision submodule, a ridge linear regression model is used to determine the most suitable SSLP for a specific small dataset. Since the symbols in SMILES all have corresponding meanings, using the dataset analysis module, we can make a comprehensive comparison of the datasets from these distributions. In the SSLP, the fine-tune module is used to generate the task-specific fingerprints for the specific dataset. We can fine-tune the selected pre-trained model by using the specific dataset and generate the corresponding SSL-FPs for downstream machine learning tasks.

Evaluation on regression and classification tasks To evaluate the proposed platform, we performed 5 classification and 5 regression tasks, and all these datasets are listed in the Table 1. In addition, we compare three different fingerprints, which are circular fingerprints (circular 2D) with nine parameter settings, autoencoder-based fingerprints (auto-FPs)[20], and the fingerprints from our platform (SSL-FPs). In the downstream tasks, we carry out our evaluation by using some standard machine learning algorithms from the scikit-learn library, namely, gradient boosted decision tree (GBDT), random forest (RF), and support vector machine (SVM).[41] To avoid over-tuning the machine learning algorithm and to better compare the performance between fingerprints, we prefix a set of general machine learning parameters, as shown in Table S2. To reduce the systematic errors in the machine learning process, we applied for different random numbers and split all the datasets into training, validation, and test sets 10 times in the ratio of 8: 1: 1. For the split datasets, we repeated the computation 5 times for each machine learning model. The best-performing model of the three models was used for the final results. In this work, the squared Pearson correlation coefficient (R^2) is used to assess the performance of the regression task. For classification tasks, we use the area under the receiver operating characteristic convex hull (AUC-ROC) as the metric. The definitions of metrics are given in section S1.

Figure 3a and b show the results of the three types of fingerprints on 10 tasks. The toxicity in the honey bees (beet) dataset locates positive compounds above the threshold and negative below the selected threshold based on selected toxicity thresholds. In this work, 100 μ/bees were selected as the threshold. SSL-FPs, auto-FPs, and circular 2D FPs achieved the best results in 3, 4, and 3 tasks, respectively. For circular 2D fingerprints, in each task, we pick the best fingerprint from nine parameter settings for comparison. Although our SSL-FPs did not achieve the best performance on all tasks, it still performs on a comparable level for most prediction tasks. For the five regression tasks (except for the DPP4 dataset), deep learning-based fingerprints, including our fingerprints and autoencoder-based fingerprints, have a better performance than the circular 2D fingerprints. The complete results with multiple metrics are listed in the Table S3. We also compared the fingerprints generated by different pre-trained models, as shown in Figure 3c and d. It is interesting to see that model C achieved the best performance in 7 of the 10 tasks. For pre-trained model C, we only applied about 1.9 million unlabeled data for pre-training, while models CP and CPZ were pre-trained with over 103 million and 700 million data, respectively. It indicates that the performance of downstream molecular fingerprinting is not entirely determined by the size of the amount of pre-trained data. In summary, for molecular property predictions with data sizes ranging from 254 to 6512, our SSL-FPs achieve comparable performance in most cases, indicating their robustness. We also found that the choice of a particular pre-trained model is not absolutely correlated with the size of pre-trained data. For all molecular property prediction tasks in this work, we performed 50 calculations, i.e., 10 random splits of data, and 5 replicate machine learning experiments for each data split. Error bars are given in Figure 3.

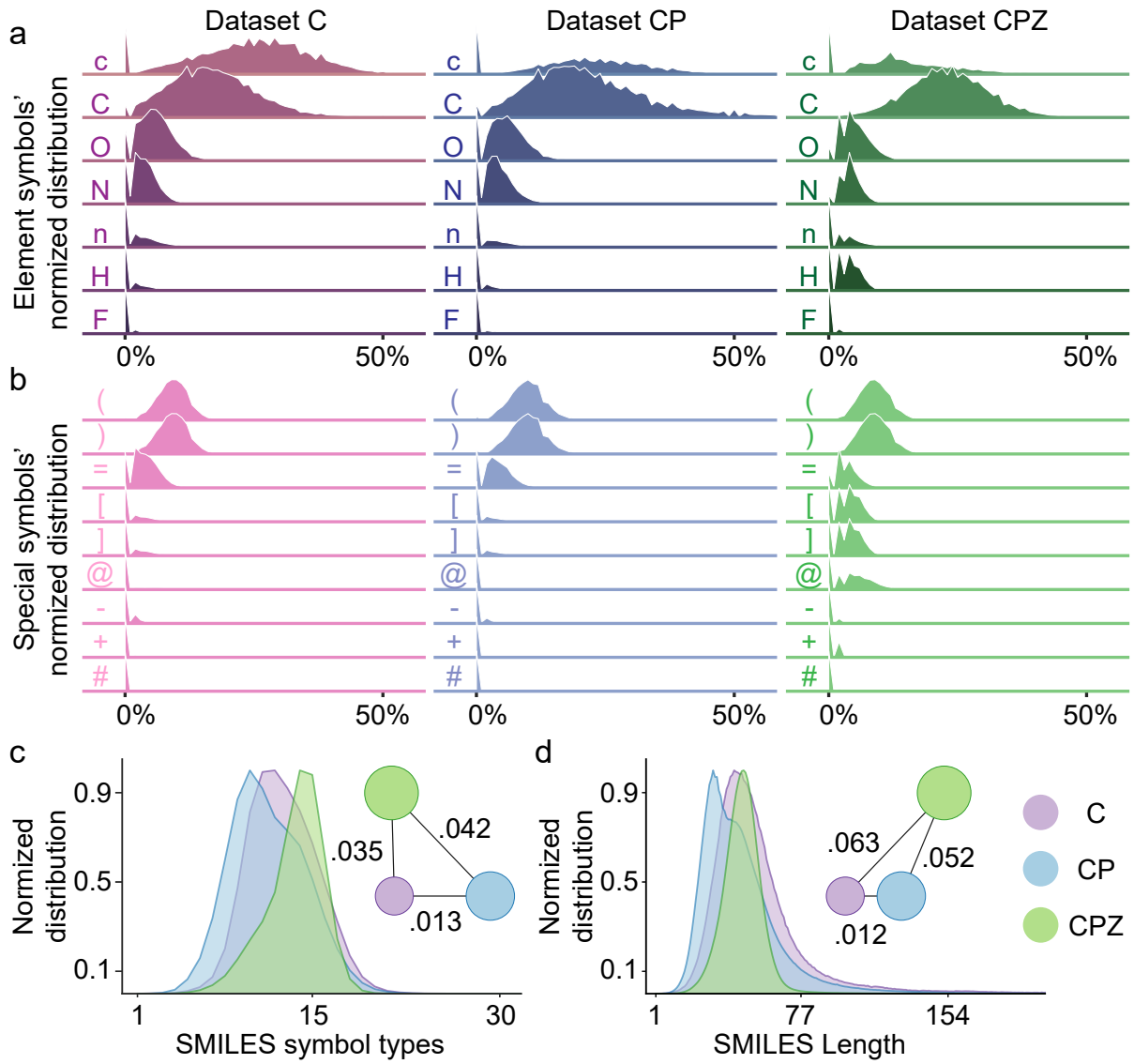

Figure 2: Datasets analysis for the pre-training datasets. **a** and **b**, The normalized distributions of elements and special symbols within SMILES in dataset ChEMBL (C), the concatenation of ChEMBL, and PubChem (CP), and the concatenation of ChEMBL, PubChem, and ZINC (CPZ), respectively. The x-axis represents the proportion of each symbol in a SMILES string, and the y-axis represents the proportion of SMILES in the dataset. **c** and **d**, The normalized distributions of SMILES symbol types within SMILES and SMILES length in dataset C, CP, and CPZ. The x-axis represents the number of different symbols per SMILES. The circles represent three datasets, and the circle size corresponds to the dataset size. The correlation of each pair of datasets is determined by the Wasserstein distance.

Ligand-based virtual screening experiments The basic idea of ligand-based virtual screening methods is to use existing information, e.g., similarity, in known active ligands to rank a large set of compounds of their activity on certain targets. It is based on the assumption that similar compounds have similar biological activity. To estimate the performance of our SSL-FPs on VS experiments, we followed the benchmark protocol of Riniker et al.[42], and 28 2D molecular fingerprints were used in the comparison as well as the auto-FPs. Since for VS experiments, there is no corresponding label for each compound, our SSL-FPs are directly derived from pre-trained models. For each target in the DUD and MUV databases, five active compounds in the corresponding dataset were randomly selected and the remaining molecules in the dataset were ranked by their average similarity to the selected active compounds. For the molecular fingerprints

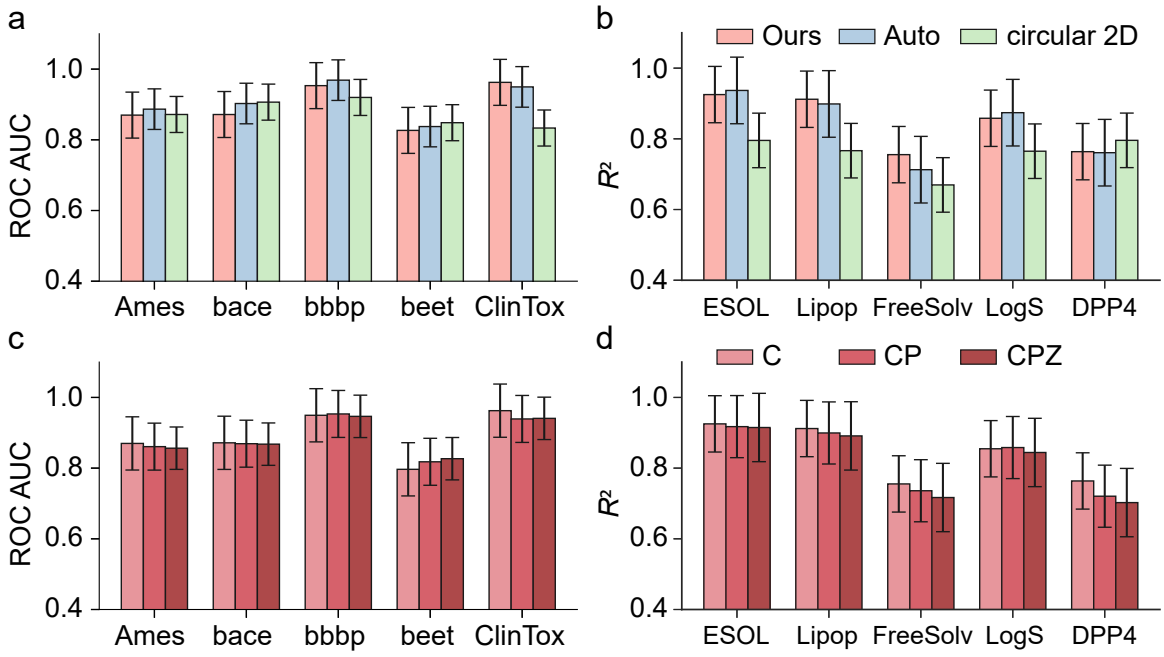

Figure 3: Results of the 5 classification and 5 regression tasks. **a** and **b**, The comparison between our FPs (Ours), auto-encoder FPs (auto), and circular 2D FPs. For circular 2D fingerprints, we choose the best fingerprint among the 9 parameter settings for each task as the final result. These three fingerprints achieved the best results in 3, 4, and 3 tasks, respectively. **c** and **d** The comparison between the fingerprints from pre-trained model C, model CP, and model CPZ. These three fingerprints produced the best performance on 7, 2, and 1 tasks of 10 tasks, respectively. All the results were generated by the best machine learning model among GBDT, RF, and SVM.

defined in the discrete spaces, i.e., 28 molecular 2D fingerprints, the Tanimoto similarity was used as the metric. The cosine similarity is used for the molecular fingerprints defined in the continuous space, such as SSL-FPs and auto-FPs. To eliminate the effect of randomness on the VS experiments, we repeated the experiment 50 times for all fingerprints. The performance of VS experiments was evaluated by the mean ROC-AUC over 50 repetitions for each dataset in DUD and MUV databases.

The results of the VS experiments for each target in the DUD database are shown in Figure 4a, and Figure 4b shows the results of the MUV database. The red diamond represents our SSL-FPs, specially generated from model C and the green circle represents the results of auto-FPs. The blue triangle is the 2D laval fingerprint, which is the best performing fingerprint among all 28 2D fingerprints. The color of the background in the figure represents is the same as the color of the best performing FPs in this target. The lightly colored scattered dots in the figure indicate 50 independent experiments. In the DUD datasets, our SSL-FPs have a smaller variance. Figure 4c shows the summary of all VS experiments, where our SSL-FPs obtained the best results in 18 out of 38 datasets. auto-FPs, on the other hand, obtained the best results in 7 datasets, while laval obtained the best performance in only 6 tasks. The fingerprint ap obtained the best performance in 7 tasks, of which six were in the MUV. Although our SSL-FPs do not achieve the best results on all datasets, it is easy to notice that our molecular fingerprinting can still show close performance on those tasks that fail to achieve the best result, such as the set ache and set ar datasets in DUD. To further measure the superiority of individual molecular fingerprints across all datasets, we calculated the average superiority, which is the average of the percentage of each molecular fingerprint outperforming the next best performing molecular fingerprint in the respective best performing dataset. The average superiority aims to measure the superiority of molecular fingerprints across datasets. As listed in Table 2, our SSL-FPs showed the highest average superiority in both DUD and MUV, with 8.02% and 5.41%, respectively. Although fingerprint ap obtained the best performance in MUV across 6 datasets, his average superiority was only 3.76%, lower

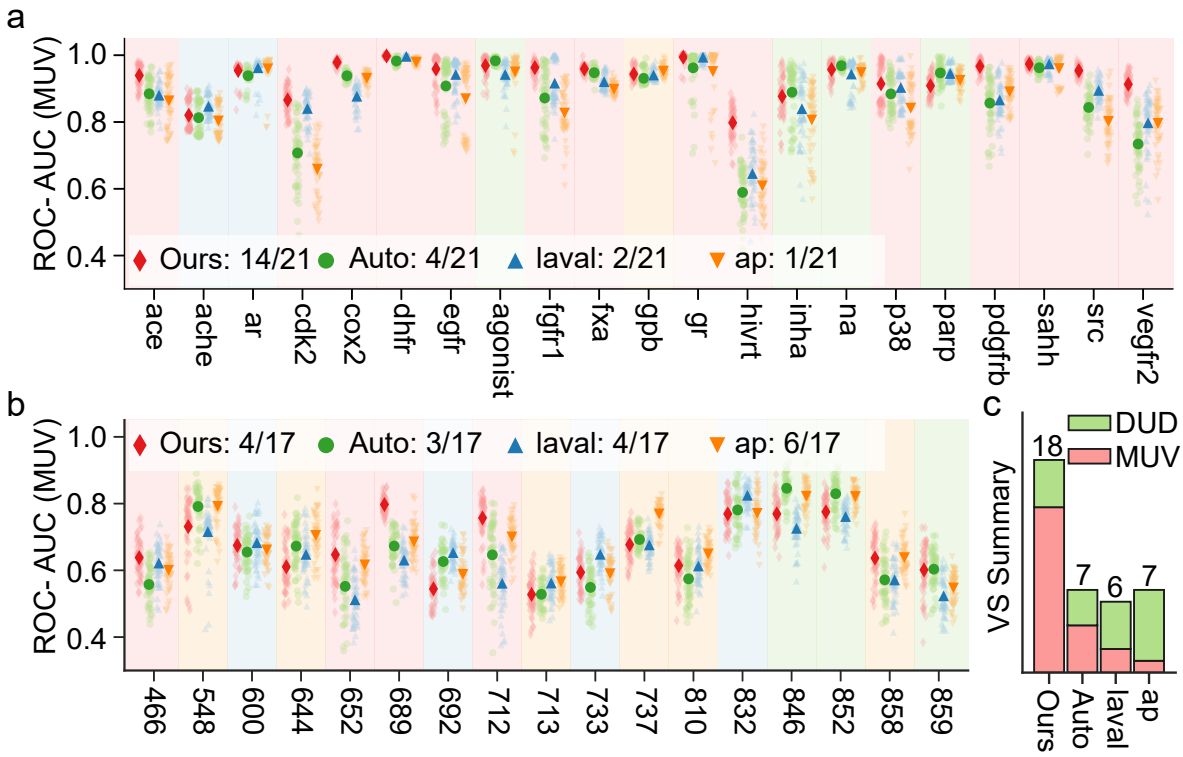

Figure 4: The results of the VS experiments for each target for the overall best fingerprints. **a** and **b**, ROC-AUC of the VS experiments for DUD and MUV datasets for the fingerprint from Model C (ours, red diamond), autoencoder fingerprint (auto, green circle), 2D fingerprint laval (laval, blue triangle), and 2D fingerprint ap (ap, orange triangle). The light-colored scattered dots indicate 50 independent experiments, while the highlighted data points indicate the average value. For each dataset, the background color in the figure corresponds to the best performing fingerprint color. In the VS experiment, for the DUD database, fingerprint laval was the best performing fingerprint among 28 2D fingerprints, and in the MUV database, fingerprint ap was the best performing one among all the 2D fingerprints. **c**, a summary of the VS experiments concluded that the four fingerprints, ours, auto, laval, and ap, obtained the best performance on 18, 7, 6, and 7 data sets, respectively.

than that of the SSL-FPs and laval which indicates that other molecular fingerprints can also obtain very close performance in these datasets. In summary, our SSL-FPs showed stable and higher superiority in VS experiments. The complete results for all fingerprints are listed in Table S4 and Table S5.

3 Discussion

In this section, we describe the application of self-supervised learning methods in molecular sciences, as well as the scalability and mobility of pre-trained models in our platform. Additionally, the significance and usage of our dataset analysis module are discussed.

Apply self-supervised learning (SSL) in the pre-training Self-supervised learning has been used in different fields, such as representation learning and natural language processing. It provides an efficient strategy to utilize large unlabeled data. In particular, this strategy learns from unlabeled data by constructing data-mask pairs, and this is called self-supervised learning. In molecular sciences, creating property labels through experiments or first-principle calculations can consume much time and resources. Therefore, it is often difficult to obtain a large number of property labels for deep neural network-based supervised learning. It is worth noting that the availability of large public chemical databases such ChEMBL, PubChem,

Table 2: The average superiority of the four molecular fingerprints, SSL-FPs, auto-FPs, laval, and ap, in the two databases DUD (21 targets) and MUV (17 targets).

MUV				
Descriptor	SSL-FPs	auto-FPs	laval	ap
Average superiority	8.02%	1.51%	5.01%	3.76%
DUD				
Descriptor	SSL-FPs	auto-FPs	laval	ap
Average superiority	5.41%	1.07%	1.80%	0.85%

and ZINC have provided massive unlabeled molecules. Additionally, the sequence-based representations of molecules such as SMILES strings have enabled the application of some techniques in the field of natural language processing to molecular sciences. Specifically, in the pre-training stage, we select a certain percentage of SMILES symbols and process them in three ways: masking, random replacement, or leaving them unchanged. The model is then trained to predict pre-selected symbols based on unprocessed SMILES information. This process is an unlabeled data-enabled supervised learning. The detailed description of the pre-trained model and the parameter settings are given in section 4.

In this work, we applied the SSL strategy to train different BET-based models using three datasets, i.e., Set C, Set CP, and Set CPZ (listed in Table 1). For a specific downstream task, such as a regression task, we simply use the task-specific dataset as input data to fine-tune the model so that task-specific molecular fingerprints can be generated from the fine-tuned model. We carry out the fine-tuning process to adapt the model to a specific task, allowing the resulting molecular descriptors to focus on relevant task-based information, thereby improving the accuracy of downstream tasks. Figure 1f shows the comparison of normalized predicted values with true values obtained by the downstream machine learning algorithm using SSL-FPs generated from model C. The majority of experimental points are on the diagonal, and the average R^2 for the five regression tasks is 0.955, indicating that the predictions based on SSL-FPs are in high agreement with the experimental values. Figure S3 shows the comparison between true values and the predicted results obtained from SSL-FPs generated by model CP and model CPZ with an average R^2 of 0.953 and 0.952, respectively. On the other hand, the pre-trained model itself is obtained based on the reconstructed molecular information from SMILES and thus can also be used directly to generate molecular descriptors. In the present work, all molecular fingerprints used in VS experiments are obtained directly from the pre-trained model. As shown in Figure 4, the SSL-FPs from pre-trained model C can also achieve 18 best results over 38 tasks.

In contrast to the encoder-decoder structure of traditional autoencoder models, in this work, we utilize the encoder-based BET, which greatly improves the efficiency of model training. For some downstream machine learning tasks, such as the molecular property prediction tasks and VS experiments discussed in this work, our self-supervised learning-based pre-trained encoder alone can be used to achieve excellent performance. Moreover, the parallel computing capability of the transformer was a crucial element for us to engage over 700 million molecules in our training.[28] The structure of the BET is shown in Figure S4.

Cross dataset analysis Three models, i.e., model C, model CP, and model CPZ, are trained respectively from Set C with about 1.9 million data, Set CP with over 103 million data, and Set CPZ with over 775 million data. Interestingly, the performance of the molecular descriptors generated from these models did not improve proportionally with the size of pre-trained data. On the contrary, the model with the smallest pre-training dataset (model C) gives the best overall result as shown in Figure 3c and d. For the VS experiments, a similar observation can be drawn from Table S4 and Table S5, where model C can perform even better. However, on some datasets, such as the LogS dataset in the regression tasks, the best performance is obtained by model CP. Based on this observation, we hypothesize that the performance of a model for a task depends on the correlation of the task-specific dataset with the pre-training dataset. To verify our hypothesis, we

developed a dataset analysis module in our self-supervised learning platform, which aims to identify pre-trained models that can provide the best performance.

Based on the composition of symbols in SMILES strings, we counted 61 common symbols and all the symbols listed in Table S1. For each type of symbol, we calculated its percentage in each SMILES. Therefore, for each dataset, we can obtain a distribution from 0% to 100% for each symbol, as shown in Figure 2. For the organic small molecule database, we can see that the distribution of carbon, oxygen, and nitrogen are the widest in each dataset, which indicates high diversity of these essential elements. In addition, the symbol 'c' represents the carbon element in the ring structure. As shown in Figure 2, it can be obtained the ring structure of dataset C has a higher diversity compared to the dataset CP and dataset CPZ For the special symbols, it can be noted that dataset CPZ has higher diversity for symbol '[', symbol ']', and symbol '+' which indicates that there is a more charged atom in the dataset. In addition to the symbolic analysis, we also statistic the distribution of SMILES lengths and the distribution of the number of element types contained in SMILES in each dataset, as shown in Figure 2c and d.

After collecting the various distributions of the dataset, the Wasserstein distance is employed to count the distance between the corresponding statistical distributions of the dataset. As shown in Figure 2d, the circles represent the three pre-training datasets, the size of each circle corresponds to the number of SMILES in the dataset, and the lines between the circles represent the Wasserstein distance between the SMILES length distributions of the corresponding datasets. For the SMILES length distribution, the distance between dataset C and dataset CP is the closest. Similarly, Wasserstein distance analysis can perform for the SMILES symbol type analysis. Then, based on 63 distributions, we can obtain 63 Wasserstein distances between every two datasets. In this work, we conducted experiments on a total of 48 downstream datasets, including 5 classification datasets, 5 regression datasets, 21 DUD virtual screening datasets, and 17 MUV virtual screening datasets. By analyzing the correlation of these datasets with three pre-training datasets, we constructed a 189-dimensional feature vector based on Wasserstein distance for each small dataset pair of the pre-training dataset. In this work, a total of 48 data points is considered. Based on these data points, we further constructed a linear classification model. With this model, a customized dataset can be analyzed to point to the most suitable pre-trained model. As shown in Figure 1e and Figure S2, with our decision module, each downstream dataset can get its confidence score to the pre-training dataset, and the value is indicated by colored line segments in the figure.

4 Methods

Data processing for Self-supervised learning To enable the self-supervised learning, in this work, we pre-process the input SMILES. A total of 51 symbols, as listed in the Table S1, are used to split these SMILES strings. '$< s >$' and '$< \backslash s >$' two special symbols were added to the beginning and end of each input. Since the length of SMILES varies from molecule to molecule, the '$< pad >$' is used as a padding symbol to fill in short inputs to reach the preset length. In the masking process, 15% symbol of the SMILES will be operated. Among these 15% symbols, 80% of symbols were masked, 10% of the symbols were unchanged, and the remaining 10% were randomly replaced. The strategy of dynamically changing the masking pattern was applied to the pre-training data.[43]

Bidirectional encoders of transformer for molecular representation Unlike sequences learning models such as RNN-based models, transformer is based on an attention mechanism[28], which is a kind of scaled dot-product attention,

$$\text{Attention}(Q, K, V) = \text{Softmax}(\frac{QK^T}{\sqrt{d_k}})V. \tag{1}$$

The Q, K, and V, namely query matrix, key matrix, and value matrix, are mapping from input data. The dot product of the query matrix and the key matrix is divided by the scaling factor $\sqrt{d_k}$, where the d_k

is the embedding dimension. The design of independent positional embedding allows the transformer to have better parallelism which dramatically reduces the training time for massive data. This feature also makes training of Set CPZ (over 700 million data) possible. Inspired by the representation model for natural language processing called BERT introduced by Devlin et al.[40] In the present work, only encoder of the transformer is applied. The input to BET is a SMILES string. Unlike the sentences in a traditional BERT for natural language processing, the SMILES strings of different molecules are not logically linked. Therefore, we only keep the masked learning task in the pre-learning process, which is to mask part of the input SMILES symbols during the training process and then recover the masked symbols by training. Specifically, our BET contains 8 encoder layers, the embedding dimension is set to 512, the number of the self-attention header is 8, and the embedding size of fully connected feed-forward layers is 1024. The maximum sequence length is set to 256, including the start and terminate symbols. The Adam optimizer is used in both pre-training and fine-tuning stages, the weight decay is set to 0.1. In addition, a warming-up strategy is applied for the first 5000 updates, maximum learning rate is set to 0.0001. To ensure the model fully converge, each pre-trained model is updated over 200 million times. The loss is defined by cross-entropy, which was applied to measure the difference between the predicted symbols and the real symbols at the masked position. The model is trained on six Tesla V100-SXM2 GPUs and the maximum sequence number in each GPU is set to 64. The structure of BET is shown in Figure S4.

For a specific downstream task, we use supervised learning to fine-tune the pre-trained model. There is no additional pre-processing for the input SMILES. The Adam optimizer is set as the same as that of pre-training. The warm-up strategy is used for the first 2 epochs, and a total of 50 epochs are trained for each dataset. The mean square error and cross-entropy are used in the fine-tuning stage for the regression task and classification task, respectively. The process of fine-tuning is shown in Figure S5. The molecular representation was generated from the last encoder layer's embedding vector of the first symbol, i.e. '$< s >$'.

Wassertein distance analysis of datasets In this work, the Wasserstein distance is used to measure the correlation between two distributions. Mathematically, the Wasserstein distance is a distance function defined between probability distributions on a given metric space M. For $p \geq 1$, the collection of all probability distribution on M with finite p^{th} moment is denoted as $P_p(M)$. And the p^{th} Wasserstein distance between two probability distributions μ and v in $P_p(M)$ is defined as

$$W_p(\mu, \nu) := \left(\inf_{\gamma \in \Gamma(\mu,\nu)} \int_{M \times M} d(x,y)^p \, \mathrm{d}\gamma(x,y) \right)^{1/p}, \tag{2}$$

where $\Gamma(\mu, v)$ denotes the collection of all distributions on $M \times M$ with marginals μ and v on the first an second factors respectively. Also, the Wasserstein metric is equivalently defined by

$$W_p(\mu, \nu) = \left(\inf \mathrm{E}\left[d(X,Y)^p \right] \right)^{1/p}, \tag{3}$$

where $\mathbf{E}$ represents the expected value and the infimum is taken over all joint distributions of the random variables X and Y with marginals μ and ν respectively.

For a downstream dataset, a set of distributions, including the distributions of 61 symbols, SMILES length distribution, and the distribution of SMILES symbol types can be generated. Thus, the similarity of the customized dataset to each pre-training set is determined by 63 Wasserstein distances. Since we have three pre-training datasets, a vector length of 189 features, denoted as X, will be used to determine the most appropriate pre-training model. Specifically, a ridge model was introduced to calculate the coefficient score for each pre-training dataset. The ridge coefficients minimize a penalized residual sum of squares:

$$\min_w ||Xw - y||_2^2 + \alpha ||w||_2^2, \tag{4}$$

where $\alpha > 0$ is the complexity parameter and it controls the amount of shrinkage.[44] And y here corresponds to the index of three pre-training models, i.e., 0, 1, and 2. Additionally, considering the influence of feature

dimensionality on the accuracy of the least squares, we use the principal component analysis (PCA) method to downscale the feature X. Figure S6 shows the accuracy of the model in selecting the best model as the feature dimension increases.

Downstream machine learning and evaluation metrics For the downstream prediction tasks, three machine learning algorithms are used in this work, namely, GBDT, RF, and SVM.[41] To better compare the performance of molecular fingerprints, we did not over search for the best machine learning model hyperparameters. Therefore, for these three machine learning methods, we simply set universal parameters based on the amount of data in the training set for the downstream task, as shown in Table S2. The predictions from the model with the best performance were chosen as the final results. In this study, the squared Pearson correlation coefficient (R^2) is used in regression tasks. The area under the receiver operating characteristic convex hull (AUC-ROC) is used to evaluate the performance of the model on classification tasks. All the definitions of related metrics are given in section S1.

5 Data Availability

The pre-training used in this work is the combination of ChEMBL27, PubChem, and ZINC13 3D datasets, which is publicly available at https://ftp.ebi.ac.uk/pub/databases/chembl/ChEMBLdb/releases/chembl_27/, https://ftp.ncbi.nih.gov/pubchem/Compound/, and http://files.docking.org/3D/, respectively. To ensure the reproducibility of this work, the 10 datasets used in this work, including 5 classification datasets (Ames, bace, bbbp, beet, ClinTox), and 5 regression datasets (ESOL, Lipophilicity, FreeSolv, LogS, and DPP4) are available at https://weilab.math.msu.edu/DataLibrary/2D/.

6 Code Availability

The overall models and related code have been released as an open-source code and is also available in the Github repository: https://github.com/ChenDdon/AGBTcode.

7 Acknowledgments

The research was financially supported by the Shenzhen Science and Technology Research Grant (No. JCYJ20200109140416788), the Chemistry and Chemical Engineering Guangdong Laboratory (No.1922018), and the Soft Science Research Project of Guangdong Province (No. 2017B030301013). The work of Wei was supported in partial by NSF Grants DMS-2052983, DMS1761320, IIS1900473, NIH grant GM126189, Bristol-Myers Squibb, and Pfizer. Chen was also supported by Michigan State University.

8 Author Contributions

Dong Chen designed the project, performed computational studies, analyzed data, wrote the first draft, and revised the manuscript. Guo-Wei Wei conceptualized and supervised the project, revised the manuscript and acquired funding. Feng Pan supervised the project and acquired funding.

9 Competing Interests

The authors declare no competing interests.

10 Supporting Information

The Supporting Information is available on the website at xxxxxx

References

[1] Duc Duy Nguyen, Zixuan Cang, and Guo-Wei Wei. A review of mathematical representations of biomolecular data. *Physical Chemistry Chemical Physics*, 22(8):4343–4367, 2020.

[2] Roberto Todeschini and Viviana Consonni. *Handbook of molecular descriptors*, volume 11. John Wiley & Sons, 2008.

[3] David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of chemical information and modeling*, 50(5):742–754, 2010.

[4] Kaifu Gao, Duc Duy Nguyen, Vishnu Sresht, Alan M Mathiowetz, Meihua Tu, and Guo-Wei Wei. Are 2d fingerprints still valuable for drug discovery? *Physical chemistry chemical physics*, 22(16):8373–8390, 2020.

[5] CA James, D Weininger, and J Delany. Daylight theory manual. daylight chemical information systems. *Inc., Irvine, CA*, 1995.

[6] Greg Landrum et al. Rdkit: Open-source cheminformatics. 2006.

[7] Noel M O'Boyle, Michael Banck, Craig A James, Chris Morley, Tim Vandermeersch, and Geoffrey R Hutchison. Open babel: An open chemical toolbox. *Journal of cheminformatics*, 3(1):1–14, 2011.

[8] Jitender Verma, Vijay M Khedkar, and Evans C Coutinho. 3d-qsar in drug design-a review. *Current topics in medicinal chemistry*, 10(1):95–115, 2010.

[9] Zixuan Cang, Lin Mu, Kedi Wu, Kristopher Opron, Kelin Xia, and Guo-Wei Wei. A topological approach for protein classification. *Computational and Mathematical Biophysics*, 1(open-issue), 2015.

[10] Duc Duy Nguyen and Guo-Wei Wei. Dg-gl: Differential geometry-based geometric learning of molecular datasets. *International journal for numerical methods in biomedical engineering*, 35(3):e3179, 2019.

[11] Zhenyu Meng and Kelin Xia. Persistent spectral based machine learning (perspect ml) for drug design. *arXiv preprint arXiv:2002.00582*, 2020.

[12] Rui Wang, Rundong Zhao, Emily Ribando-Gros, Jiahui Chen, Yiying Tong, and Guo-Wei Wei. Hermes: Persistent spectral graph software. *arXiv preprint arXiv:2012.11065*, 2020.

[13] Dong Chen, Kaifu Gao, Duc Duy Nguyen, Xin Chen, Yi Jiang, Guo-Wei Wei, and Feng Pan. Algebraic graph-assisted bidirectional transformers for molecular property prediction. *Nature Communications*, 12(1):1–9, 2021.

[14] Hang Li. Deep learning for natural language processing: advantages and challenges. *National Science Review*, 2017.

[15] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zihang Jiang, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. *arXiv preprint arXiv:2101.11986*, 2021.

[16] Maciej Wójcikowski, Pedro J Ballester, and Pawel Siedlecki. Performance of machine-learning scoring functions in structure-based virtual screening. *Scientific Reports*, 7(1):1–10, 2017.

[17] Jaswinder Singh, Kuldip Paliwal, Jaspreet Singh, and Yaoqi Zhou. Rna backbone torsion and pseudo-torsion angle prediction using dilated convolutional neural networks. *Journal of Chemical Information and Modeling*, 2021.

[18] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[19] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[20] Robin Winter, Floriane Montanari, Frank Noé, and Djork-Arné Clevert. Learning continuous and data-driven molecular descriptors by translating equivalent chemical representations. *Chemical science*, 10 (6):1692–1701, 2019.

[21] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[22] David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36, 1988.

[23] Anna Gaulton, Anne Hersey, Michał Nowotka, A Patricia Bento, Jon Chambers, David Mendez, Prudence Mutowo, Francis Atkinson, Louisa J Bellis, Elena Cibrián-Uhalte, et al. The chembl database in 2017. *Nucleic acids research*, 45(D1):D945–D954, 2017.

[24] Sunghwan Kim, Paul A Thiessen, Evan E Bolton, Jie Chen, Gang Fu, Asta Gindulyte, Lianyi Han, Jane He, Siqian He, Benjamin A Shoemaker, et al. Pubchem substance and compound databases. *Nucleic acids research*, 44(D1):D1202–D1213, 2016.

[25] John J Irwin and Brian K Shoichet. Zinc- a free database of commercially available compounds for virtual screening. *Journal of chemical information and modeling*, 45(1):177–182, 2005.

[26] Jure Zupan and Johann Gasteiger. *Neural networks in chemistry and drug design*. John Wiley & Sons, Inc., 1999.

[27] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[28] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

[29] Niu Huang, Brian K Shoichet, and John J Irwin. Benchmarking sets for molecular docking. *Journal of medicinal chemistry*, 49(23):6789–6801, 2006.

[30] Sebastian G Rohrer and Knut Baumann. Maximum unbiased validation (muv) data sets for virtual screening based on pubchem bioactivity data. *Journal of chemical information and modeling*, 49(2): 169–184, 2009.

[31] Katja Hansen, Sebastian Mika, Timon Schroeter, Andreas Sutter, Antonius Ter Laak, Thomas Steger-Hartmann, Nikolaus Heinrich, and Klaus-Robert Muller. Benchmark data set for in silico prediction of ames mutagenicity. *Journal of chemical information and modeling*, 49(9):2077–2081, 2009.

[32] Govindan Subramanian, Bharath Ramsundar, Vijay Pande, and Rajiah Aldrin Denny. Computational modeling of β-secretase 1 (bace-1) inhibitors using ligand based approaches. *Journal of chemical information and modeling*, 56(10):1936–1949, 2016.

[33] Ines Filipa Martins, Ana L Teixeira, Luis Pinheiro, and Andre O Falcao. A bayesian approach to in silico blood-brain barrier penetration modeling. *Journal of chemical information and modeling*, 52(6): 1686–1697, 2012.

[34] K Venko, V Drgan, and M Novič. Classification models for identifying substances exhibiting acute contact toxicity in honeybees (apis mellifera). *SAR and QSAR in Environmental Research*, 29(9): 743–754, 2018.

[35] Kaitlyn M Gayvert, Neel S Madhukar, and Olivier Elemento. A data-driven approach to predicting successes and failures of clinical trials. *Cell chemical biology*, 23(10):1294–1301, 2016.

[36] John S Delaney. Esol: estimating aqueous solubility directly from molecular structure. *Journal of chemical information and computer sciences*, 44(3):1000–1005, 2004.

[37] David L Mobley and J Peter Guthrie. Freesolv: a database of experimental and calculated hydration free energies, with input files. *Journal of computer-aided molecular design*, 28(7):711–720, 2014.

[38] Guoli Xiong, Zhenxing Wu, Jiacai Yi, Li Fu, Zhijiang Yang, Changyu Hsieh, Mingzhu Yin, Xiangxiang Zeng, Chengkun Wu, Aiping Lu, et al. Admetlab 2.0: an integrated online platform for accurate and comprehensive predictions of admet properties. *Nucleic Acids Research*, 2021.

[39] Oky Hermansyah, Alhadi Bustamam, and Arry Yanuar. Virtual screening of dpp-4 inhibitors using qsar-based artificial intelligence and molecular docking of hit compounds to dpp-8 and dpp-9 enzymes. 2020.

[40] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[41] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830, 2011.

[42] Sereina Riniker and Gregory A Landrum. Open-source platform to benchmark fingerprints for ligand-based virtual screening. *Journal of cheminformatics*, 5(1):1–17, 2013.

[43] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[44] Donald E Hilt and Donald W Seegrist. *Ridge, a computer program for calculating ridge regression estimates*, volume 236. Department of Agriculture, Forest Service, Northeastern Forest Experiment ..., 1977.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- SupportingInformation.pdf