論 文 の 内 容 の 要 旨

論文題目　**Integration of Molecular Data with Preference Learning**
（比較学習による分子データの統合）

氏　　名　Sun Xiaolin　(孫　暁琳)

Introduction

Machine-learning-based design of molecules and materials is increasingly common in recent years. Despite progress in biological and materials informatics, machine learning often yields poor results due to shortage of experimental data. The problem may be solved with data augmentation by leveraging legacy datasets in public databases or private repositories. However, due to different properties and unknown experimental conditions, it is hard to integrate external datasets with current dataset straightforwardly.

In this research, we employ preference learning models to integrate and derive valid information from external datasets. The learning process is based on pairwise comparison of target values in the same dataset. Figure 1 shows data integration with preference learning. Two datasets are related but incompatible due to different experimental methods. Each dataset is separately converted to pairwise preference relations. A Gaussian process-based preference learning model is trained from all pairs and yields probability distributions of latent values at all points in the descriptor space, followed with Bayesian optimization to search for optimum samples[1]. Neural network-based preference learning model is also used for processing large-scale datasets[2] . Figure 2 illustrates the process, consisting of three major steps: (i) generating pairwise preference, (ii) training preference learning neural network, and (iii) predicting ranking of candidate molecules.

In benchmarking our method, we first search for organic molecules with longer absorption wavelength[3]. By integrating external dataset of HOMO-LUMO gap, we found significant acceleration in Bayesian optimization search process. We integrated 129 different ChEMBL datasets measuring efficacy of drug molecules for inhibiting factor Xa (fXa)[4]. The average prediction and extrapolation performance improved significantly compared to those before integration.

Method

A set of candidate materials is represented as $\{z_i\}_{i=1,\dots,N}$, where $z_i \in \mathcal{R}^d$ is a vector of descriptors. The corresponding values of target property are represented as $\{y_i\}_{i=1,\dots,N}$. They are initially unknown and revealed by observation. Let us assume that k observations are already made $Z = \{(z_i, y_i)\}_{i=1,\dots,k}$. For any two observed materials $\{(z_i, y_i), (z_j, y_j)\} \in Z$, if $y_i > y_j$, we denote $z_i \succ z_j$, i.e., z_i is preferred over z_j. For an external dataset $Z' = \{(z_i', y_i')\}_{i=1,\dots,k'}$, a new preference list is generated. Comparisons are performed with two observed materials from the same dataset. After comparing all pairs, Z and Z' are converted to preference sets of size $\frac{k(k-1)}{2}$ and $\frac{k'(k'-1)}{2}$, respectively.

Preference learning with Gaussian Process and Bayesian Optimization

For notational simplicity, all descriptor vectors in $Z \cup Z'$ are redefined as $X = \{x_i\}_{i=1,\dots,n}$. Let D denote the merged preference set, $D = \{v_i \succ u_i\}_{i=1,2,\dots,m}$, where v_i,u_i are taken from X. Gaussian process is able to assign a latent value $f(x)$ to any vector $x \in \mathcal{R}^d$. The prior probability of $f(x_i)$ is defined as

$$P(\boldsymbol{f}) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}\boldsymbol{f}^T\Sigma^{-1}\boldsymbol{f}\right),$$

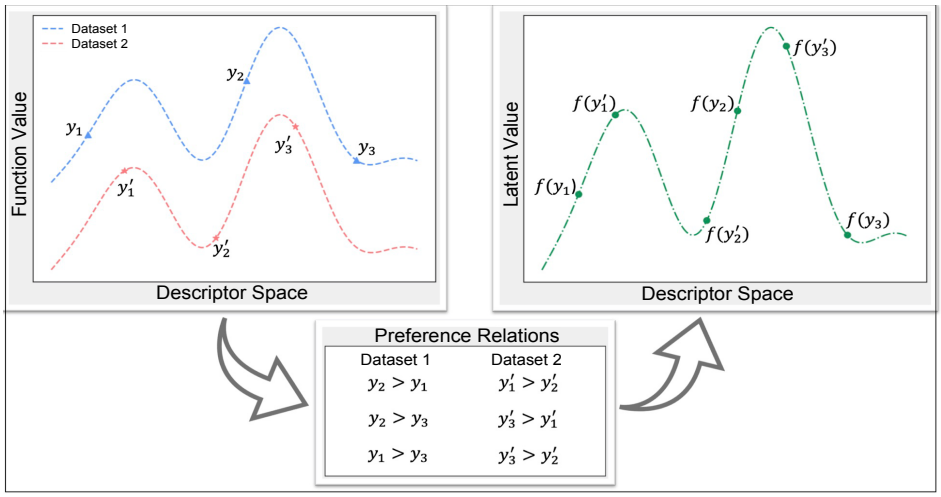

Figure 1 Data integration with preference learning.

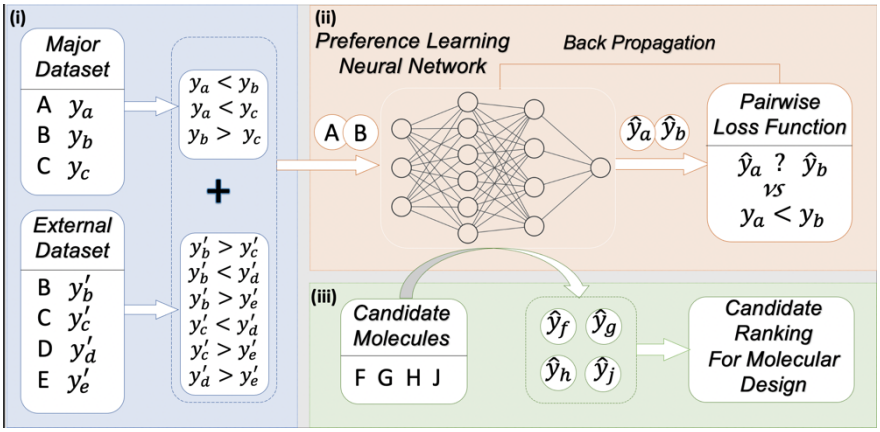

Figure 1 Data integration with preference learning neural network.

where $\boldsymbol{f} = [f(x_1), f(x_2), \ldots, f(x_n)]^T$, and Σ is the covariance matrix defined by a radial basis function kernel. Using Gaussian noise variables $\delta \sim \mathcal{N}(\delta; 0, \sigma^2)$, the probability of preference $v_k \succ u_k$ is described as

$$P\big(v_k \succ u_k | f(v_k), f(u_k)\big) = \int\int P(v_k \succ u_k | f(v_k) + \delta_v \; > \; f(u_k) + \delta_u)\, \mathcal{N}(\delta_v; 0{,}1)\mathcal{N}(\delta_u; 0{,}1) d\delta_v d\delta_u.$$

By using Bayes' theorem, we can arrive at the posterior probability,

$$P(\boldsymbol{f}|D) = \frac{P(\boldsymbol{f})P(D|\boldsymbol{f})}{P(D)} = \frac{P(\boldsymbol{f})}{P(D)} \prod_{k=1}^{m} P\big(v_k \succ u_k | f(v_k), f(u_k)\big).$$

The solution is obtained by maximizing a posteriori estimate (MAP). To make a prediction at a new sample point x^*, we infer the probability distribution of its latent value as

$$P(f^*|D) = \int P(f^*|\boldsymbol{f}) P(\boldsymbol{f}|D) d\boldsymbol{f} \sim N(f^*; \mathbf{K}^{*T}\Sigma^{-1}\boldsymbol{f}_{\text{MAP}}, K^{**} - \mathbf{K}^{*T}(\Sigma + \Lambda_{\text{MAP}}^{-1})^{-1}\mathbf{K}^*)$$

where $\mathbf{K}^* = [K(x^*, x_1), K(x^*, x_2), \ldots, K(x^*, x_n)]^T$, $K^{**} = K(x^*, x^*)$ and Λ_{MAP} is the Hessian matrix $\frac{\partial^2 S(f)}{\partial f \partial f^T} - \Sigma^{-1}$ at $\boldsymbol{f} = \boldsymbol{f}_{\text{MAP}}$. The predicted mean and variance of the latent value at x^* are $\mu^* = \mathbf{K}^{*T}\Sigma^{-1}\boldsymbol{f}_{\text{MAP}}$ and $\sigma^{*2} = K^{**} - \mathbf{K}^{*T}(\Sigma + \Lambda_{\text{MAP}}^{-1})^{-1}\mathbf{K}^*$, respectively.

In Bayesian optimization, the mean latent value μ^* and standard deviation σ^* are computed for all remaining candidates. Let μ_{max} denote the maximum value observed so far. The expected improvement of

a candidate x^* is described as follows.

$$\mathrm{EI}(x^*) = (\mu_{\max} - \mu^*)\Phi\left(\frac{\mu_{\max} - \mu^*}{\sigma^*}\right) + \sigma^*\varphi\left(\frac{\mu_{\max} - \mu^*}{\sigma^*}\right)$$

where Φ and φ represent the cumulative distribution function and the probability density function of standard normal distribution, respectively.

Preference Learning Neural Network with Gradient Descent

The neural network consists of two full-connected layers. It maps feature descriptors of a molecule to its surrogate target value. For a molecule z_i with features $x_i \in \mathcal{R}^d$, the input dimension is d. One output node predicts the surrogate target value $\hat{y}_i$. Forward propagation is performed separately for two examples in one pair. Pairwise probabilistic loss function is the key for learning pairwise information. Binary cross entropy function is used to estimate the divergence between true relations and predicted relations formulated as:

$$Loss = -\sum_{(v_i,u_i)\in D} plogP(v_i \succ u_i) + (1-p)logP(u_i \succ v_i)$$

True relation between v_i and u_i is represented using $p \in \{0,0.5,1\}$ with 0 for $u_i \succ v_i$, 1 for $v_i \succ u_i$ and 0.5 for being equal. Predicted probability P is determined by the output $\{\hat{y}_v, \hat{y}_u\}$ of the neural network given samples$\{v_i, u_i\}$, defined as:

$$P(v \succ u) = \frac{exp(\hat{y}_v)}{exp(\hat{y}_v) + exp(\hat{y}_u)}$$

For any pair $\{v_i, u_i\}$ in D, activation and gradient values are stored for updating a weight parameter $w_k \in \mathcal{R}$ using gradient descent:

$$w_k \equiv w_k - \alpha\left(\frac{\partial L}{\partial \hat{y}_v}\frac{\partial \hat{y}_v}{\partial w_k} + \frac{\partial L}{\partial \hat{y}_u}\frac{\partial \hat{y}_u}{\partial w_k}\right)$$

where L denotes the loss function as previously defined and α is learning rate. The back propagation is hereby accomplished. After sufficient training, the neural network is ready for predicting new samples. We performed pre-experiments on major dataset to determine the number of nodes in each layer. Datasets were divided into training, validation and test set in a ratio of 3:1:1. Hyperparameters were determined by the performance of validation set. Final result shows the ranking accuracy of test set.

Result

We created our own small database of 94 organic molecules with their absorption spectra and HOMO-LUMO gaps computed via TD-DFT. For each candidate set *C*, we created five types of external datasets, each consists of 50 molecules. For q=0, 25, 50, 75 and 100, the *q%-overlap dataset* consists of $\lfloor qN/100 \rfloor$ molecules in *C*, $50 - \lfloor qN/100 \rfloor$ molecules not in *C,* and their HOMO-LUMO gaps. Molecules in *C* are divided into 80% training set and 20% test set. Normalized Discounted Cumulative Gain (NDCG) is calculated on test set to evaluate the ranking accuracy. In Bayesian optimization, the success rate at iteration *j* is defined as the fraction of runs where the best molecule was found within *j* selections of molecules. The results are shown in figure 3. The overall process has been improved or accelerated indicating that an external dataset with a related but non-identical property can still improve our preference learning model. With more molecules to be observed included in external samples, the dataset would provide more information about experimental design. With small overlap, it may be difficult to accelerate the search.

ChEMBL datasets contains a total of 2959 molecules with IC50 values from 129 different sources. Each dataset containing 2 to 85 molecules, were labeled with a ChEMBL number according to its source. We chose one of them as major dataset with other 128 datasets as external datasets. Figure 4 (a) shows the result where training and test set were divided randomly. Since ChEMBL dataset contains data of the same

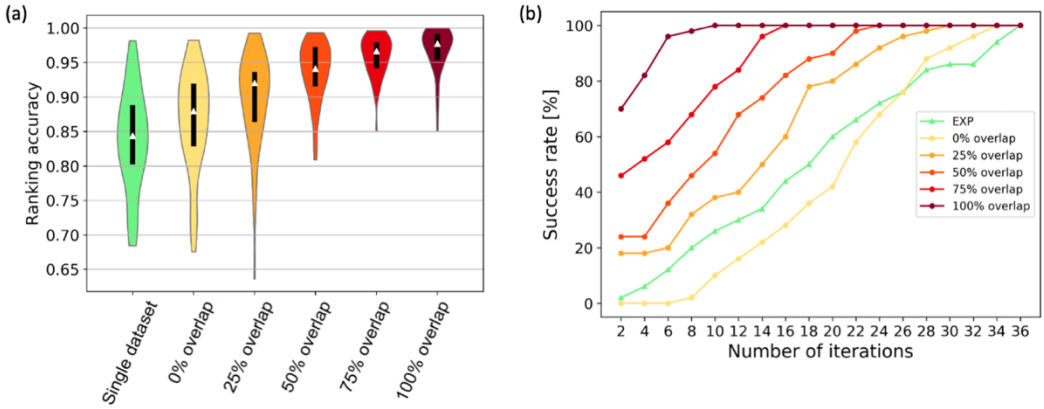

Figure 3 (a) Ranking accuracy by Gaussian process with preference learning. (b) Success rate of Bayesian optimization with preference learning against the number of iterations.

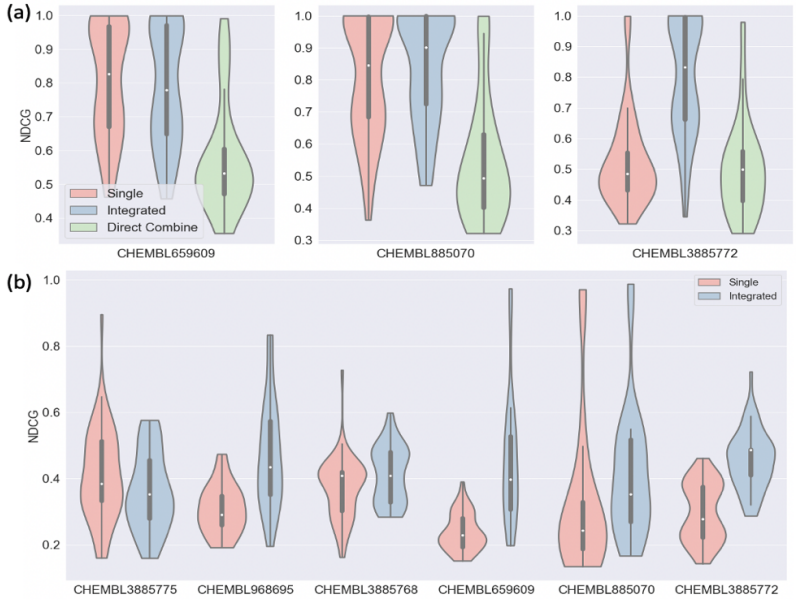

Figure 4 Results for ChEMBL molecules. (a) Ranking accuracy of random integration. (b) Ranking accuracy of extrapolation.

target property, we further created a group that directly combine all the datasets using their target values. The lower average NDCG score indicates that even measuring the same property, a direct integration among different sources is unadvisable. Figure 4 (b) shows the result of extrapolation where molecules in test set have larger values out of the training set observed range. Lack of information in target range makes the prediction more difficult. The overall accuracy decreased compared to random separation group, but the integrated group shows higher scores than the single group.

Reference

[1] Chu, Wei, and Zoubin Ghahramani. "Preference learning with Gaussian processes." *Proceedings of the 22nd international conference on Machine learning*. 2005.

[2] Burges, Chris, et al. "Learning to rank using gradient descent." *Proceedings of the 22nd international conference on Machine learning*. 2005.

[3] Sumita, Masato, et al. "Hunting for organic molecules with artificial intelligence: molecules optimized for desired excitation energies." *ACS central science* 4.9 (2018): 1126-1133.

[4] Ishihara, Tsukasa, et al. "A Novel Fragment Recommendation Workflow using Direct and Indirect Transfer of SAR According to Integrated Similarities of Scaffold Motifs and SAR Trends: Application to Identifying Factor Xa Inhibitors." *Chem-Bio Informatics Journal* 17 (2017): 1-18.