이학박사 학위논문

Deep Learning Approaches to Predictions of Liquid Properties

심층학습을 이용한 액체계의 성질 예측

2020년 2월

서울대학교 대학원

화학부 물리화학 전공

임 현 태

Deep Learning Approaches to Predictions of Liquid Properties

by

Hyuntae Lim

Supervised by

Professor **YounJoon Jung**

A Dissertation
Submitted to the Faculty of
Seoul National University
in Partial Fulfillment of
the Requirements for the Degree of
Doctor of Philosophy

February 2020

Department of Chemistry
Graduate School
Seoul National University

Abstract

Recent advances in machine learning technologies and their chemical applications lead to the developments of diverse structure-property relationship based prediction models for various chemical properties; the free energy of solvation is one of them and plays a dominant role as a fundamental measure of solvation chemistry. Here, we introduce a novel machine learning-based solvation model, which calculates the target solvation free energy from pairwise atomistic interactions. The novelty of our proposed solvation model involves rather simple architecture: two encoding function extracts vector representations of the atomic and the molecular features from the given chemical structure, while the inner product between two atomistic features calculates their interactions, instead of black-boxed perceptron networks. The cross-validation result on 6,493 experimental measurements for 952 organic solutes and 147 organic solvents achieves an outstanding performance, which is 0.2 kcal/mol in MUE. The scaffold-based split method exhibits 0.6 kcal/mol, which shows that the proposed model guarantees reasonable accuracy even for extrapolated cases. Moreover, the proposed model shows an excellent transferability for enlarging training data due to its solvent-non-specific nature. Analysis of the atomistic interaction map shows there is a great potential that our proposed model reproduces group contributions on the solvation energy, which makes us believe that the proposed model not only provides the predicted target property, but also gives us more detailed physicochemical insights.

Keywords: Deep learning, Structure-property relationship, Solvation free energy, Solubility, Liquid property, Liquid system

Student Number: 2010-23098

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The importance of solvation or hydration mechanism and its involved free energy change has made various *in silico* calculation methods for the solvation energy a major topic in computational chemistry.[1–22] The solvation free energy directly influences to many chemical properties in solution and plays a dominant role in various chemical reactions: drug delivery[4, 15, 17, 23], organic synthesis[24], electrochemical redox reactions[25–28], *et cetera*.

The realistic computer simulation approaches for the solvent and the solute molecules directly offer the microscopic structure of the solvation shell, which surrounds the solutes molecule.[9, 10, 13, 16, 17, 29] The solvation shell structure could provide us detailed physicochemical information like microscopic mechanisms on solvation or the interplay between the solvent and the solute molecules when we use an appropriate force field model and parameters. However, those *explicit solvation* methods we stated above need an extensive amount of numerical calculations since we have to simulate each individual molecule in the solvated system. Moreover, the free energy calculation procedure with an explicitly implemented solvent model necessarily involves rare-event sampling methods, which make the task even more computationally expensive. The realistic problems on the explicit solvation model restrict its applications to classical molecular mechanics simulations,[9, 10, 16] or a limited QM/MM approaches.[13, 29]

For classical mechanics approaches for macromolecules or calculations for small compounds at quantum-mechanical level, the idea of *implicit solvation* enables us to calculate solvation energy with feasible time and computational costs when one considers a given solvent as a continuous and isotropic medium in the Poisson-Boltzmann equation.[1–3, 5–8, 11] Many theoretical advances have introduced to construct the PB-based equation, which involves parameterized solvent properties: the polarizable continuum model (PCM),[11] the conductor-like screening model (COSMO),[3] generalized Born approximations like solvation model based on density (SMD)[7] or solvation model 6, 8, 12, ... (SMx).[1, 6] The conductor-like screening model for realistic solvents (COSMO-RS) is a noteworthy solvation model since it is believed to be the state-of-the-art method.[2] This is realized by statistical thermodynamics treatment on the polarization charge densities, which helps COSMO-RS with making successful predictions even in polar solvents where the fundamental idea of the dielectric continuum solvation collapses.[8]

The quantitative structure activity relationship (QSAR) or the quantitative structure property relationship (QSPR) is a rather new approach, which predicts the solvation free energy with a completely different point of view when compared to computer simulation approaches with precisely defined theoretical backgrounds[30, 31]. The underlying architecture of QSPR consists of two elementary mathematical functions[30]: one is the *encoding function*, which encodes the structural or chemical features of a given compound into a *molecular descriptor*. The other, the *mapping function*, predicts the target property (or activity) that we intend to find out using the descriptor from the encoding function. Although we cannot expect detailed chemical or physical insights other than the target property since the QSAR/QSPR is a regression analysis in its intrinsic nature, It has shown its advantages in terms of transferability and outstanding computational efficiency[20, 30, 31].

Recent successes in the machine learning (ML) technique[32] and their implementations in computational chemistry[20, 33] are promoting broad applications of QSAR/QSPR in numerous chemical studies[4, 18, 21, 23, 27, 34–43]. Those studies proved that ML guarantees faster calculations than computer simulations and more precise estimations than traditional

QSPR estimations; a decent number of models showed accuracies comparable to *ab initio* solvation models in the aqueous system[20].

In this thesis, we introduce a novel artificial neural-network-based ML model called *Delfos* that predicts free energies of solvation for generic organic solvents in the previous work[22]. The model not only has a great potential of showing an accuracy comparable to the state-of-the-art computational chemistry methods[1, 2] but offers information about which substructures play a dominant role in the solvation process. As a further development, we propose an improved ML model for the solvation energy estimation, which is based on the group-contribution method. The key idea of the proposed model is the calculation of pairwise atomic interactions by inner products of atomic feature vectors, while each encoder network for the solvent and the solute extracts such atomic features.

The outline of the rest of the present thesis is as follows: in Chapter 2, we mainly discuss the performance of Delfos, with both MD and *ab initio* simulation strategies[1, 2, 44, 45] and analyze database sensitivity using cluster cross-validation method. We also visualize important substructures in solvation via attention mechanism. In Chapter 3, we introduce a new ML model for the solvation energy prediction, which is based on pairwise atom-by-atom interactions. The chapter quantifies the proposed model's performance with 6,594 data points, mainly focused on group contributions and pairwise atomistic interactions. In the last chapter of the thesis, we summarize and conclude our work.

Chapter 2

Delfos: Deep Learning Model for Prediction of Solvation Free Energies in Generic Organic Solvents

2.1 Methods

2.1.1 Embedding of Chemical Contexts

Natural language processing (NLP) is one of the most cutting-edge subfields of computer science in varied applications of machine learning and neural networks[46–50]. To process human languages using computers, we need to encode words and sentences and extract their linguistic properties. The process is commonly implemented via *word embedding* method[46, 47]. To perform the task, unsupervised learning schemes such as skip-gram and continuous bag of words (CBOW) algorithms generate a vector representation of the given word in an arbitrary vector space[47, 51]. If the necessary vector space is well-defined, one can conjecture the semantic or syntactic features of the given word from the position of the embedded vector, and the inner product of two vectors corresponding to two different words provides information about their semantic similarity.

It is worthwhile to note that we can employ the embedding technique for chemical or biophysical processes if we consider an atom or a substructure as a word and a compound as a sentence[52–54]. In that case, positions of molecular substructures in the embedded vector space represent their chemical and physical properties, instead of linguistic information. Several models have already been developed along the line of this idea. For example, bio-vector models[52] that have been developed to encode sequences of proteins or DNAs, and atomic-vector embedding models have been introduced recently to encode structural features of chemical compounds[53, 54]. Mol2Vec is one of such embedding techniques, and it generates vector representations of a given molecule from the *molecular sentence*[54]. To make molecular sentences, Mol2Vec uses the Morgan algorithm[55] that assorts identical atoms in the molecule. The algorithm is commonly used to generate ECFP fingerprints[56], which are the *de facto* standard in cheminformatics[57], and they make identifiers of the given atom from the chemical environment where the atom is positioned. An atom may have multiple identifiers depending on the pre-set maximum value of *radius* $r_{\max}$, which denotes the maximum topological distance between the given atom and its neighboring atoms. The atom itself is identified

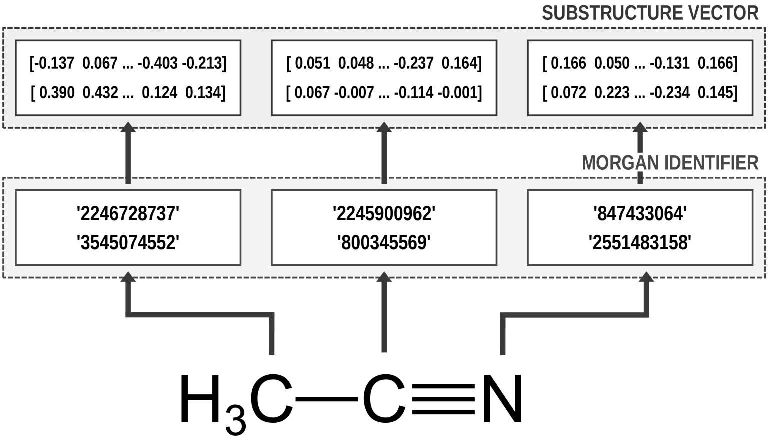


Figure 2.1: Schematic illustration of the molecular embedding process for acetonitrile (SMILES: CC#N) and $r_{\max} = 1$. The Morgan algorithm discriminates identifiers between two substructures: one is for itself ($r = 0$) and the other considers its nearest neighbor atoms ($r = 1$). Then the embedding layer calculates the vector representation from the given identifier.

by $r = 0$, and additional substructure identifiers for adjacent atoms are denoted by $r = 1$ (nearest neighbor), $r = 2$ (next nearest neighbor), and so on. Since Mol2Vec has demonstrated promising performances in several applications of QSAR/QSPR[54], Delfos uses Mol2Vec as the primary encoding means. We schematically illustrated embedding procedure for acetonitrile in Fig. 2.1.

2.1.2 Encoder-Predictor Network

As shown in Fig. 2.2, the fundamental architecture of Delfos involves three sub-neural networks: the solvent and the solute encoders extract dominant structural features of the given compound from SMILES strings, while the predictor calculates the solvation energy of the given solvent-solute pair from their encoded features.

The primary architecture of the encoder is based on two bidirectional recurrent neural networks (BiRNNs)[58]. The network is designed for handling sequential data and we consider the molecular sentence $[\mathbf{x}_1, \cdots, \mathbf{x}_N]$ as a sequence of embedded substructures, $\mathbf{x}_i$. RNNs may have a failure when input sequences are lengthy; gradients of the loss function can be diluted or amplified because of accumulated precision error from the back-propagation process[59]. The excessive or restrained gradient may cause a decline in learning performance, and we call these two problems as vanishing or exploding gradient. To overcome these limits which stem from lengthy input sequences, one may consider using both forward-directional RNN ($\overrightarrow{\mathrm{RNN}}$) and backward-directional RNN ($\overleftarrow{\mathrm{RNN}}$) within a single layer:

$$\overrightarrow{\mathrm{RNN}}([\mathbf{x}_1, \cdots, \mathbf{x}_N]) = [\overrightarrow{\mathbf{h}_1}, \cdots, \overrightarrow{\mathbf{h}_N}], \tag{2.1a}$$

$$\overleftarrow{\mathrm{RNN}}([\mathbf{x}_1, \cdots, \mathbf{x}_N]) = [\overleftarrow{\mathbf{h}_1}, \cdots, \overleftarrow{\mathbf{h}_N}], \tag{2.1b}$$

$$\overleftrightarrow{\mathrm{RNN}}([\mathbf{x}_1, \cdots, \mathbf{x}_N]) = [\mathbf{h}_1, \cdots, \mathbf{h}_N]. \tag{2.1c}$$

In Eqn. 2.1, $\mathbf{x}_i$ is the embedded atomic vector of a given molecule, $\overrightarrow{\mathbf{h}_i}$ and $\overleftarrow{\mathbf{h}_i}$ are hidden state outputs of each recurrent unit, and $\mathbf{h}_i = \overrightarrow{\mathbf{h}_i}; \overleftarrow{\mathbf{h}_i}$ means concatenation of two hidden states, respectively. The long-short term memory[60] (LSTM) and gated recurrent unit[61] (GRU) networks, which are modifications of RNN, are invented to handle lengthy input sequences. They introduce *gates* in each RNN cell state to memorize important information of the previous cell state and minimize vanishing and exploding gradient problem.

After RNN layers, the molecular sentences of both the solvent $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_N]$ and the solute $\mathbf{Y} = [\mathbf{y}_1, \cdots, \mathbf{y}_M]$ are converted to hidden states, $\mathbf{H} = [\mathbf{h}_1, \cdots \mathbf{h}_N]$ and $\mathbf{G} = [\mathbf{g}_1, \cdots, \mathbf{g}_M]$, respectively. Each hidden state is then put into the shared *attention* layer and weighted. The attention mechanism, which was originally proposed to enhance performances of machine translator[48], is an essential technique in diverse NLP applications nowadays[49, 50]. Principles of the attention start from the definition of the score function of hidden states and its normalization with the softmax function:

$$\alpha_{ij} = \frac{\exp(\mathrm{score}(\mathbf{h}_i, \mathbf{g}_j))}{\sum_k \exp(\mathrm{score}(\mathbf{h}_i, \mathbf{g}_k))}, \tag{2.2a}$$

$$\mathbf{p}_i = \sum_j^M \alpha_{ij} \mathbf{g}_j, \tag{2.2b}$$

$$\mathrm{score}(\mathbf{h}_i, \mathbf{g}_j) = \mathbf{h}_i \cdot \mathbf{g}_j. \tag{2.2c}$$

There are various score functions that have been introduced to achieve efficient predictions[48–50], and among them we use Luong's dot-product attention[50] in Eqn. 2.2c as a score function since it is computationally efficient. The solvent context, $\mathbf{P} = \alpha\mathbf{G}$ denotes an *emphasized* hidden state $\mathbf{H}$ with the attention alignment, α. We also get the solute context $\mathbf{Q}$ using the same procedure. The context weighted from the attention layer is an $L \times 2D$ matrix, where L is the sequence length and D is the dimension of two RNN hidden layers since we use bidirectional RNN (BiRNN). Two max-pooling layers, which is the last part of each encoder reduces contexts $\mathbf{H}$, $\mathbf{G}$, $\mathbf{P}$, and $\mathbf{Q}$ to $2D$-dimensional feature vectors $\mathbf{u}$ and $\mathbf{v}$[50]:

$$\mathbf{u} = \mathrm{MaxPooling}([\mathbf{h}_1; \mathbf{p}_1, \cdots, \mathbf{h}_N; \mathbf{p}_N]), \tag{2.3a}$$

$$\mathbf{v} = \mathrm{MaxPooling}([\mathbf{g}_1; \mathbf{q}_1, \cdots, \mathbf{g}_M; \mathbf{q}_M]). \tag{2.3b}$$

The predictor has a single fully-connected perceptron layer with rectifier unit (ReLU) and an output layer. It uses the concatenated feature of the solvent and solute $[\mathbf{u}; \mathbf{v}]$ as an input. The overall architecture of our model is shown in Figure 2.2. We also consider encoders without RNN and attention layers in order to quantify the impact of these layers on prediction performances of the network; each encoding network contains only the embedding layer and directly connected to the MLP layer. The solvent and solute features are simple summations of atomic vectors, $\mathbf{u} = \sum_i^N \mathbf{x}_i$ and $\mathbf{v} = \sum_i^M \mathbf{y}_i$, respectively. This model was initially used for gradient boost-

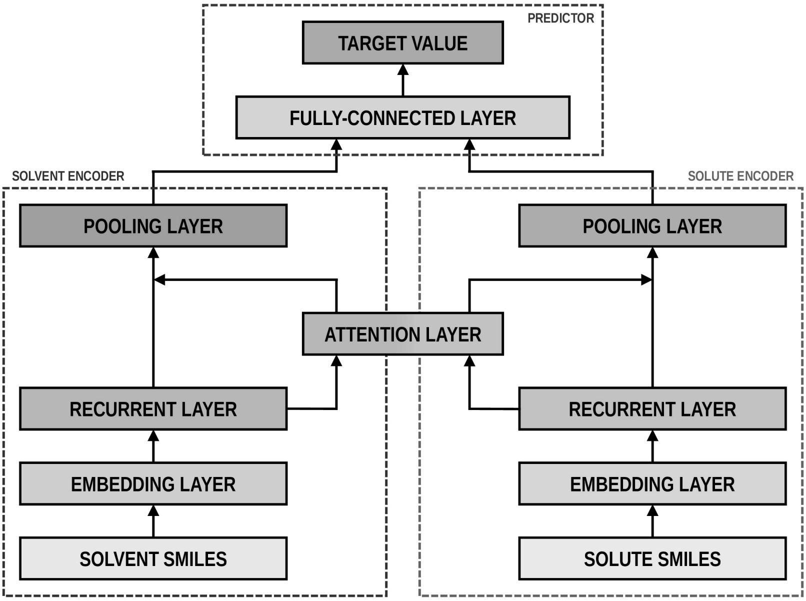


Figure 2.2: The fundamental architecture of Delfos. Each encoder network has one embedding and one recurrent layer, while the predictor has a fully-connected MLP layer. Two encoders share an attention layer, which weights outputs from recurrent layers. Black arrows indicate flow of input data.

ing (GBM) regression analysis for aqueous solubilities and toxicities[54].

2.2 Results and Discussions

2.2.1 Computational Setup and Results

We use the Minnesota solvation database[62] (MNSOL) as the dataset over which we train and test, and it provides 3,037 experimental measures of free energies of solvation and transfer energies for 790 unique solutes in 92 solvents. Because the MNSOL only contains common names of compounds, we perform an automated searching process using PubChemPy[63] script and receive SMILES strings of compounds from PubChem database. There are 363 results for charged solutes and 144 results for transfer free energies in the MNSOL which are excluded from machine learning dataset, and 35 results of solvent-solute combinations are not valid in PubChem. We finally prepare SMILES specifications of 2,495 solutions for 418 solutes and 91 solvents for the machine learning input.

For the implementation of the proposed neural networks, we use Keras 2.2.4 framework[64] with TensorFlow 1.12 backend[65]. At the very first stage, Morgan algorithm for $r = 0$ and $r = 1$ generates molecular sentences of the solvent and solute from their SMILES strings. Then the given molecular sentence is embedded to a sequence of 300-dimensional substructure vectors by the skip-gram pretrained Word2Vec model available at https://github.com/samoturk/mol2vec, which contains information of $\sim 20,000,000$ compounds and $\sim 20,000$ substructures from the ZINC15 database[54]. We consider BiLSTM and BiGRU layers in both solvent and solute encoders to compare their performances. Since our model is a regression problem, we use mean squared error (MSE) as the loss function.

We employ 10-fold cross-validation (CV) for secure representativeness of the test data because the dataset we use has a limited number of experimental measures; the total dataset is uniformly and randomly split into 10 subsets, and we iteratively choose one of the subsets as a test set and the training run uses the remaining 9 subsets. Consequentially, a 10-fold CV task performs 10 independent training and test runs, and relative sizes of the training and test sets are 9 to 1. We use Scikit-Learn library[66] to implement the CV task and perform an extensive grid search for tuning hyperparameters: learning algorithms, learning rates, and dimensions of hidden layers. We select the stochastic gradient descent (SGD) algorithm with Nesterov momentum, whose learning rate is 0.0002 and momentum is 0.9. Optimized hidden dimensions are 150 for recurrent layers and 2000 for the fully connected layer. To minimize the variance of the test run, we take averages for all results over 9 independent random CV, split from different random states.

Solvation free energies that we calculated from the MNSOL using attentive BiRNN encoders are exhibited in Fig. 2.3 and 2.4. Prediction errors for the BiLSTM model are ± 0.57 kcal/mol in RMSE, ± 0.30 kcal/mol in MAE, and the Pearson correlation coefficient is $R^2 = 0.96$ while results from the BiGRU model indicate there is no meaningful difference between the two recurrent models. The encoder without BiRNN and attention layers produces much less accurate results, whose error metrics are ± 0.77 kcal/mol in RMSE, ± 0.43 kcal/mol in MAE, and 0.92 in R^2 value, respectively.

We cannot directly compare our results with other ML models because Delfos is the first ML-based study using the MNSOL database. Nonetheless, several studies on aqueous system have previously calculated solubilities or hydration free energies using various ML techniques and molecular descriptors[4, 20, 53, 54, 67, 68]. For comparison, we have tested our neural network model for hydration free energy. A benchmark study of Wu et al. [20] provides hydration energies of 642 small molecules in a group of QSPR/ML models. Their RMSEs were up to $1.15\ \mathrm{kcal/mol}$ while our prediction from the BiLSTM encoder attains $1.19\ \mathrm{kcal/mol}$ for the same dataset and split method. This result suggests our neural network model guarantees considerably good performances even in a specific solvent of water.

Meanwhile, for studies which are not ML-based, there are several results from both classical and quantum-mechanical simulation studies that use the MNSOL as the reference data[1, 2, 44, 45, 69–71]. In Table 2.1, we choose two DFT studies which employ several widely-used QM solvation models[1, 2] for comparison with our proposed ML model: solvation model 8/12 (SM8/SM12), solvation model based on density (SMD), and full/direct conductor-like screening model for realistic solvation (COSMO-RS/D-COSMO-RS). Albeit all of those QM methods exhibited excellent performances given chemical accuracy $1.0\ \mathrm{kcal/mol}$, among the rest, full COSMO-RS is a noteworthy solvation model since it is believed to be the state-of-the-art method which shows the best accuracy[72]. This is realized by statistical thermodynamics treatment on the polarization charge densities, which helps COSMO-RS with making successful predictions even in polar solvents where the key idea of the dielectric continuum solvation collapses[8, 72, 73]. As a result, COSMO-RS calculations with BP86 functional and TZVP basis set achieved 0.52 $\mathrm{kcal/mol}$ for 274 aqueous, 0.41 $\mathrm{kcal/mol}$ for 2,072 organic solvents, and 0.43 $\mathrm{kcal/mol}$ for the full dataset in mean absolute error[2].

For the proposed ML models, Delfos with BiLSTM shows a comparable accuracy in water solvent, which MAE is 0.64 $\mathrm{kcal/mol}$. Delfos makes much better predictions in non-aqueous organic solvents; machine learning for 2121 non-aqueous systems result in 0.24 kcal$/$mol, which is 44% of SM12CM5 and 59% of COSMO-RS. However, one may argue that K-fold CV from random split does not produce the real prediction accuracy of the model. That is, the random-CV results only indicate the accuracy for *trained* or *practiced* chemical structures. Accordingly, one may ask the following questions. For example, will the ML model ensure the comparable prediction accuracy in “structurally” new compounds? What happens if the ML model couldn’t learn sufficiently varied chemical structures? We will discuss these questions in the next section.

2.2.2 Transferability of the Model for New Compounds

Since our study uses techniques of machine learning with empirical data from experimental measures, there is a likelihood that Delfos would not

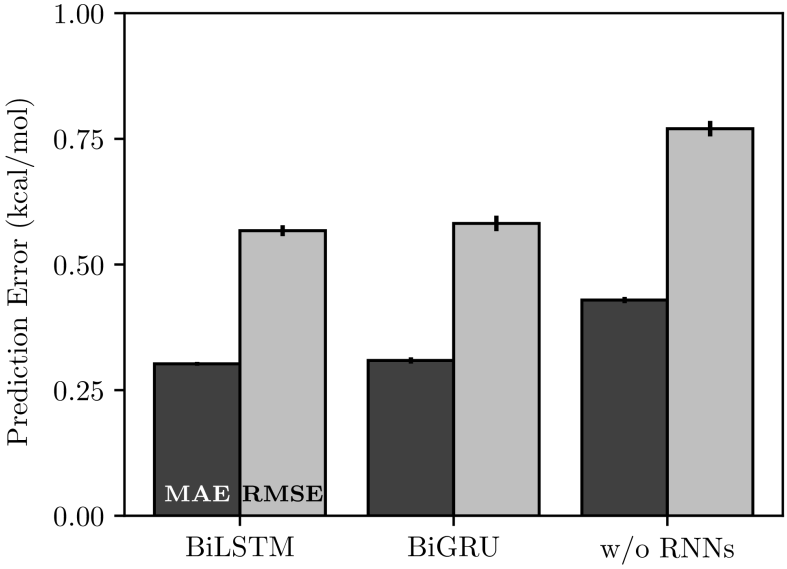


Figure 2.3: Benchmark chart for three kinds of encoder networks, for two metrics (MAE and RMSE). The BiLSTM and the BiGRU models show no significant differences, while it makes relatively inaccurate predictions without recurrent networks. All results are averaged over 9 independent test runs and black lines on tops of boxes denote variances.

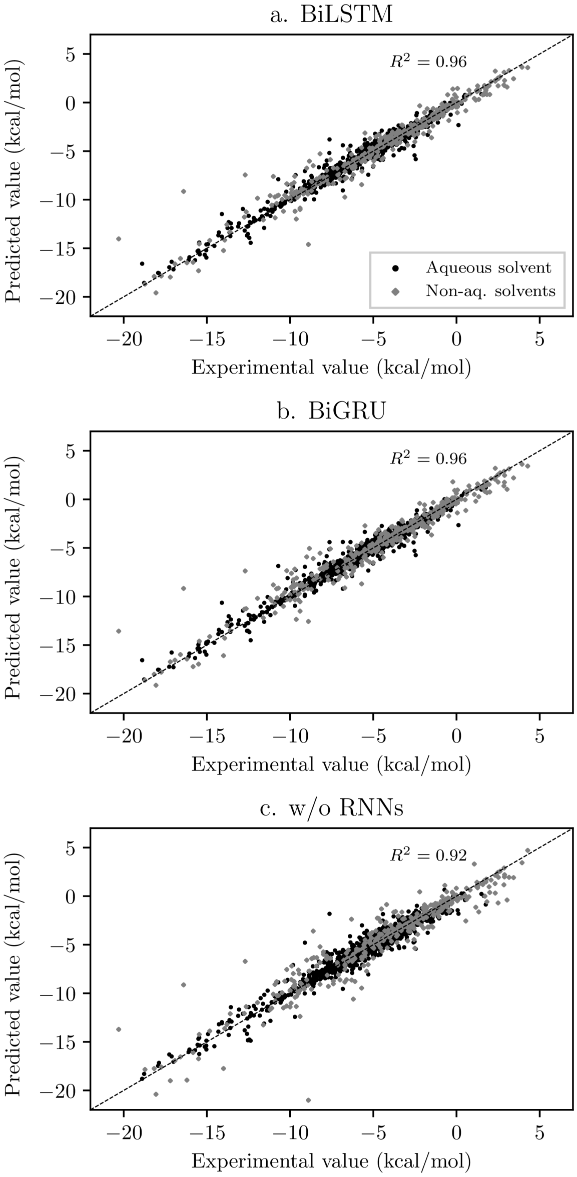


Figure 2.4: Scatter plot for true (x-axis) and ML predicted (y-axis) values of solvation energies in three different models: (a) BiLSTM, (b) BiGRU, and (c) without recurrent layers. All results are averaged over 9 independent 10-fold CV runs.

Solvent	Method	N_{data}	MAE	Ref
Aqueous	SM12CM5/B3LYP/MG3S	374	0.77	[1]
	SM8/M06-2X/6-31G(d)	366	0.89	[1]
	SMD/M05-2X/6-31G(d)	366	0.88	[1]
	COSMO-RS/BP86/TZVP	274	0.52	[2]
	D-COSMO-RS/BP86/TZVP	274	0.94	[2]
	Delfos/BiLSTM	**374**	**0.64**	
	Delfos/BiGRU	**374**	**0.68**	
	Delfos w/o RNNs	**374**	**0.90**	
Non-aqueous	SM12CM5/B3LYP/MG3S	2129	0.54	[1]
	SM8/M06-2X/6-31G(d)	2129	0.61	[1]
	SMD/M05-2X/6-31G(d)	2129	0.67	[1]
	COSMO-RS/BP86/TZVP	2072	0.41	[2]
	D-COSMO-RS/BP86/TZVP	2072	0.62	[2]
	Delfos/BiLSTM	**2121**	**0.24**	
	Delfos/BiGRU	**2121**	**0.24**	
	Delfos w/o RNNs	**2121**	**0.36**	

Table 2.1: Comparisons between encoder-predictor networks and various quantum-mechanical solvation models for aqueous and non-aqueous solutions. The error metric is MAE and $\mathrm{kcal/mol}$. Data in bold texts are our results, while QM results are taken from the work of Marenich et al. [1] and Klamt and Diedenhofen [2].

guarantee prediction accuracy for structurally new solvents or solutes which are not present in the dataset, although the MNSOL contains a considerable number commonly-used solvents and solutes.[62]. In order to investigate this potential issue, we perform another train and test runs with the *cluster cross-validation*[43, 74], instead of using the random-split CV. As a start, we individually obtain 10 clusters for solvents and solutes using the K-mean clustering algorithm and the molecular vector. The molecular vector is a simple summation of substructure vectors as we used for the simple MLP model without RNN encoders[54]: $\mathbf{u} = \sum_i^N \mathbf{x}_i$ for solvents and $\mathbf{v} = \sum_i^M \mathbf{y}_i$ for solutes, respectively. Then, we iteratively perform cross-validation process over each cluster. The size of each cluster is (422, 482, 186, 231, 443, 243, 143, 251, 15, 79) for solvents and (401, 672, 514, 75, 64, 6, 512, 54, 42, 155) for solutes, respectively.

Results from the solvent and the solute cluster CV tasks shown in Table 2.2 exhibit generalized expectation error ranges for new solvents or solutes which are not in the dataset. Winter et al. [43] reported that the split method based on the clustering brings an apparent degradation of prediction performances in various properties; we find that our proposed model exhibits a similar tendency as well. For the BiLSTM encoder model, increments of MAE are $0.52\ \mathrm{kcal/mol}$ for the solvent clustering and $0.69\ \mathrm{kcal/mol}$ for the solute clustering. The reason why the random K-fold CV exhibits superior performances is obvious; if we have a pair ($\mathcal{A}$, $\mathcal{B}$) of solvent $\mathcal{A}$ and solute $\mathcal{B}$ in the test set and the training set have $(\mathcal{A}, \mathcal{C})$ and $(\mathcal{D}, \mathcal{B})$ pairs, then both $(\mathcal{A}, \mathcal{C})$ and $(\mathcal{D}, \mathcal{B})$ could enhance prediction accuracy of ($\mathcal{A}$, $\mathcal{B}$). However, the clustering limits the location of a specific compound, and pairs of specific solvent or solute should be either in the test set or the train set.

For an additional comparison, Table 2.2 also contains results taken from SMD calculations with semi-empirical methods[45], COSMO, COSMO-RS[2], and classical molecular dynamics[44] for four small organic solvents: toluene ($C_6H_5CH_3$), chloroform ($CHCl_3$), acetonitrile (CH_3CN), and dimethyl sulfoxide ($(CH_3)_2SO$), respectively. Albeit MD is based on classical dynamics, the results of generalized amber force field (GAFF) tells us that an explicit solvation model with a suitable force field could make considerably good predictions. The bottom line of cluster CV is if the dataset for train contains at least one side of the solvent-solvent pair of which we want to estimate the solvation free energy, the expectation error of Delfos lies within chemical accuracy $1.0\ \mathrm{kcal/mol}$, which is the general error of computer simulation scheme. Also, results for four organic solvents demonstrate that predictions from the cluster CV have the accuracy that is comparable with MD simulations using AMOEBA polarizable force field[44].

Results from the cluster CV highlight the necessity for discussion on the importance of database preparation. As described earlier, the cluster CV causes a considerable increase in prediction error, and we suspect that the degradation mainly comes from the decline in the diversity of the training set. Namely, the number of substructures that the neural network learns in training process is not so many as the random CV if we use the cluster CV. To prove this speculation, we define *unique* substructures, which are substructures that only exists in the test cluster. As shown in Figure 2.5, in the solute cluster CV, MAE for 1,226 pairs which do not have any unique substructures in solutes is $0.54\ \mathrm{kcal/mol}$, while the prediction error for the rest 1,269 solutions is $1.64\ \mathrm{kcal/mol}$. The solvent cluster CV shows even more extreme results: the MAE for 374 aqueous solvents is $2.48\ \mathrm{kcal/mol}$, while non-aqueous solvents exhibit $0.52\ \mathrm{kcal/mol}$ in contrast. We believe that the outlying behavior of water is due to its distinctive nature. Water has only one, unique substructure since the oxygen atom does not have any neighbors. So the solvent clustering makes the network unable to learn the structure of water in indirect ways, results in a prediction failure. This logic tells us that the most critical thing in an ML prediction task is securement of the training dataset which contains as many as possible kinds of solvents and solutes. We believe that computational approaches would be as helpful as experimental measures for enriching structural diversity of the training data, given recent advances on QM solvation models[1, 2, 75] such as COSMO-RS. Furthermore, since there are 418 solutes and 91 solvents in the dataset we use[62], which make up 38,038 possible pairs, we expect Delfos and MNSOL would guarantee similar precision levels with the random CV for numerous systems.

Solvent	Method	N_{data}	MAE	RMSE	Ref
All	COSMO/TZVP	2346	2.15	2.57	[2]
	COSMO-RS/TZVP	2346	0.42	0.75	[2]
	SMD/PM6	2500	-	3.6	[45]
	Random CV	**2495**	**0.30**	**0.57**	
	Solvent Clustering	**2495**	**0.82**	**1.45**	
	Solute Clustering	**2495**	**0.99**	**1.61**	
Toluene	MD/GAFF	21	0.48	0.63	[44]
	MD/AMOEBA	21	0.92	1.18	[44]
	COSMO/TZVP	21	2.17	2.71	[2]
	COSMO-RS/TZVP	21	0.27	0.34	[2]
	Solvent Clustering	**21**	**0.66**	**1.10**	
	Solute Clustering	**21**	**0.93**	**1.46**	
Chloroform	MD/GAFF	21	0.92	1.11	[44]
	MD/AMOEBA	21	1.68	1.97	[44]
	COSMO/TZVP	21	1.76	2.12	[2]
	COSMO-RS/TZVP	21	0.50	0.66	[2]
	Solvent Clustering	**21**	**0.78**	**0.87**	
	Solute Clustering	**21**	**1.14**	**1.62**	
Acetonitrile	MD/GAFF	6	0.43	0.52	[44]
	MD/AMOEBA	6	0.73	0.77	[44]
	COSMO/TZVP	6	1.42	1.58	[2]
	COSMO-RS/TZVP	6	0.33	0.38	[2]
	Solvent Clustering	**6**	**0.74**	**0.82**	
	Solute Clustering	**6**	**0.80**	**0.94**	
DMSO	MD/GAFF	6	0.61	0.75	[44]
	MD/AMOEBA	6	1.12	1.21	[44]
	COSMO/TZVP	6	1.31	1.42	[2]
	COSMO-RS/TZVP	6	0.56	0.73	[2]
	Solvent Clustering	**6**	**0.93**	**1.19**	
	Solute Clustering	**6**	**0.91**	**1.11**	

Table 2.2: Prediction accuracy of the random-split CV, the solvent and solute cluster CVs using K-mean algorithm, and several theoretical solvation models for four different organic solvents: toluene ($C_6H_5CH_3$), chloroform ($CHCl_3$), acetonitrile (CH_3CN), and dimethyl sulfoxide ($(CH_3)_2SO$), respectively. Units of MAE and RMSE are $\mathrm{kcal/mol}$.

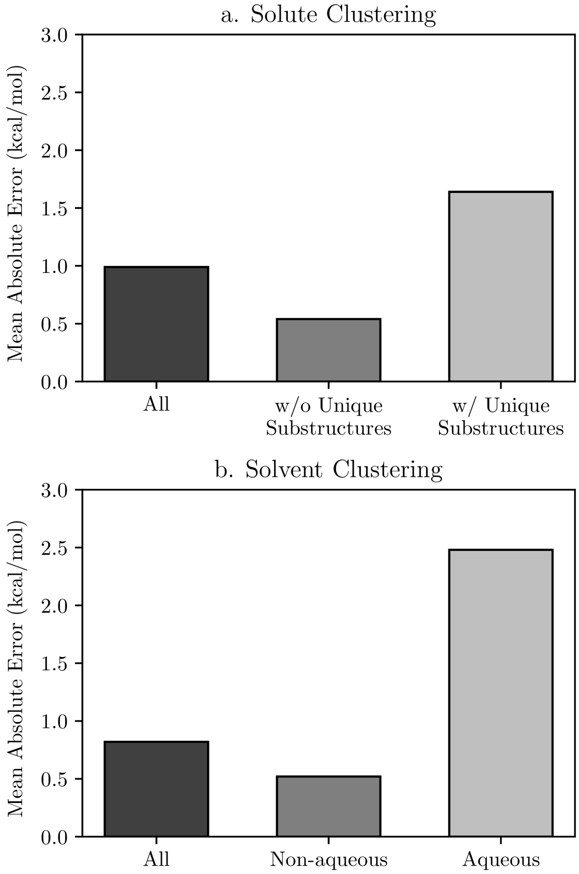


Figure 2.5: Results of cross-validation tasks using K-mean clustering algorithm for (a) solutes and (b) solvents. We conclude that unique substructures in the given compounds are the main cause of the decline in prediction accuracy. Each encoder network includes a BiLSTM layer and we use the same hyperparameters which are optimized in the random CV task.

2.2.3 Visualization of Attention Mechanism

A useful aspect of attention mechanism is that the model provides not only the prediction value of solvation energy of a given input but also a clue to why the neural network makes such a prediction based on the correlations between recurrent hidden states[49, 53, 76]. In this section, we visualize how the attention layer operates, and verify how well such correlations correspond to chemical intuitions for inter-molecular interactions. The matrix of attention alignments, α from Eqn. 2.2a indicates which substructures in the given solvent and solute are strongly correlated with each other so they play dominant roles in determining their solvation energy. In Figure 2.6, we demonstrate attention alignments of nitromethane (CH_3NO_2) solute in four different solvents: 1-octanol ($C_8H_{17}OH$, 3.51 kcal/mol), 1-butanol (C_4H_9OH, 3.93 kcal/mol), ethanol (C_2H_5OH, 4.34 kcal/mol), and acetonitrile (CH_3CN, 5.62 kcal/mol). The scheme for visualizing attention alignments is as follows: (i) first, we calculate the average alignment $\langle\alpha\rangle_j$ of each substructure j of the solute over the entire solvent structure $\{i\}$, $\langle\alpha\rangle_j = \sum_i^N \alpha_{ij}/N$. (ii) Then, we get relative amounts of averaged attention alignments $[\tilde{\alpha_1}, \cdots, \tilde{\alpha_M}]$ from dividing $\langle\alpha\rangle_j$ by the maximum value, $\tilde{\alpha}_j = \langle\alpha\rangle_j / \max(\langle\alpha\rangle_1, \cdots, \langle\alpha\rangle_M)$. (iii) Also, since the embedding algorithm which we use generates two substructure vectors per an atom, we individually visualize two alignments maps, $[\tilde{\alpha}_1, \tilde{\alpha}_3, \cdots, \tilde{\alpha}_{M-1}]$ (for $r = 0$)

and $[\tilde{\alpha}_2, \tilde{\alpha}_4, \cdots, \tilde{\alpha}_M]$ (for $r = 1$) for simpler and more intuitive illustration. (iv) Finally, the color representation of each atom in Fig. 2.6 denotes the amount of $\tilde{\alpha}_j$; the neural network judges that red-colored substructures (higher $\tilde{\alpha}_j$) in the solute are more "similar" to the solvent and the model puts more weights on them during the prediction task. In contrast, green-colored substructures have lower $\tilde{\alpha}_j$, which means they do not have similarity with the solvent molecule so much as red-colored one.

Overall results in Fig. 2.6 imply that the *chemical similarity* taken from the attention layer has a significant connection to fundamental knowledge of chemistry like polarity or hydrophilicity. Each alcoholic solvent has one hydrophilic $-$OH group, and it results in increasing contributions of the nitro group in the solute as hydrocarbon chains of alcohols shorten. For the acetonitrile-nitromethane solution, the attention mechanism reflects the highest contributions of $-NO_2$ groups due to strong polarity and aprotic nature of the solvent. Although the attention mechanism seems to reproduce molecular interactions in a faithful way, however, we find there is a defective prediction which does not agree with chemical knowledge. Two oxygen atoms $=$O and $-O^-$ in the nitro group are indistinguishable due to the resonance structure, thus they must have equivalent contributions in any solvents, but we find they show different attention scores in our model. We believe those problems happen because the SMILES string of nitromethane (C[N+](=O)[O-]) does not encode the resonance effect in the

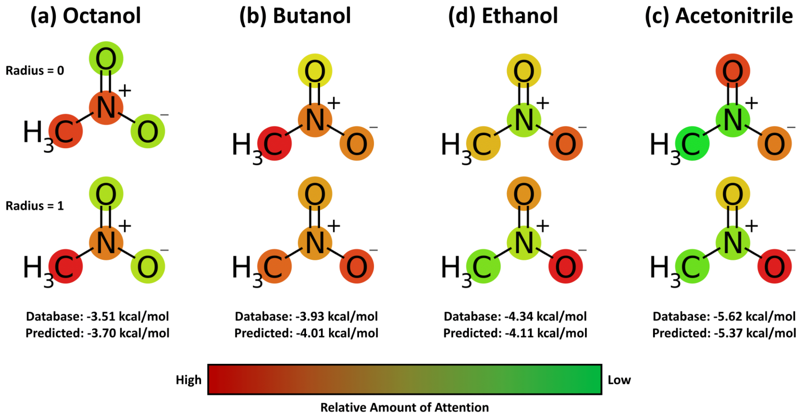


Figure 2.6: Relative and mean attention alignments map for nitromethane in four different solvents: (a) octanol, (b) butanol, (c) ethanol, (d) and acetonitrile, respectively. Color representations denote that the neural network invests more weights on red, while green substructures have relatively low contributions for the solvation energy.

nitro group. Indeed, the Morgan algorithm generates different identifiers for two oxygen atoms in the nitro group, [864942730, 2378779377] for $=$O and [864942795, 2378775366] for $-$O$^-$. The absence of resonance might be a problem worthwhile considering when one intends to use word embedding models with SMILES strings[43, 53, 54], although estimated solvation energies for nitromethane from the BiLSTM model are within a moderate error range as shown in Fig. 2.6.

Chapter 3

Group Contribution Method for the Solvation Energy Estimation with Vector Representations of Atom

3.1 Model Description

3.1.1 Word Embedding

In the proposed work, the primary strategy for the encoding of the input compound's structure is the *word embedding*, mainly inspired by Google's word2vec model[46, 51]. The first attempt of continuous vector representations of human vocabularies in arbitrary space introduced in the mid-1980s[51], however, the remarkable breakthrough has been made by developments of neural network language model (NNLM) and recurrent neural network language model[77] (RNNLM).

The general procedure of word embedding starts from the construction of a one-hot encoded vector $\mathbf{x}(I) = [x_1(I), \cdots, x_V(I)]$ of a given, tokenized input word I, where V is the vocabulary size[46]. By the nature of one-hot encoding, we know the vector $\mathbf{x}$ has only one non-zero element at the corresponding dimension to the given word, $x_I(I) = 1$ and the other elements are 0, in short, $x_i(I) = \delta_{i,I}$. Fig. 3.1 illustrates the embedding procedure when the input context has only one word.

$$\mathbf{h}(I) = \mathbf{x}(I)\mathbf{W}, \tag{3.1a}$$

$$\mathbf{y}(I) = \text{Softmax}(\mathbf{h}(I)\mathbf{W}^{'}). \tag{3.1b}$$

In Eqn. 3.1 and Fig. 3.1, the first fully-connected layer $\mathbf{W}$ forms a $V \times N$ matrix, and the second, $\mathbf{W}^{'}$ is $N \times V$. So the hidden layer (or the *projection layer*) $\mathbf{h}(I)$ has a shape of N-dimensional vector and is identical to the I-th row of $\mathbf{W}$, $\mathbf{w}_I$. The second FC layer calculates the output $\mathbf{y}(I)$, following the equations shown below:

$$\mathbf{h}(I)\mathbf{W}^{'} = \left[\mathbf{w}_1^{'} \cdot \mathbf{w}_I, \cdots, \mathbf{w}_V^{'} \cdot \mathbf{w}_I\right], \tag{3.2a}$$

$$y_i(I) = \frac{\exp(\mathbf{w}_i^{'} \cdot \mathbf{w}_I)}{\sum_{j=1}^{V} \exp(\mathbf{w}_j^{'} \cdot \mathbf{w}_I)}. \tag{3.2b}$$

Each projecting element for the second FC layer in Eqn. 3.2, $\mathbf{w}_j^{'}$ is the j-th column of $\mathbf{W}^{'}$. Both $\mathbf{w}$ and $\mathbf{w}^{'}$ have the same shape, and one can either use them as the N-dimensional embedded vector representation of the input word. Since we train the embedding model as classification tasks with a specific target word T, the conditional probability of finding T given an input I is:

$$P(T|I) = y_J(I). \tag{3.3}$$

The general optimization scheme for the classification model is logistic regression that is maximizing $P(T|I)$ and minimizing the binary cross-entropy loss function.

$$\begin{aligned} L &= -\mathbf{x}(T) \cdot \log \mathbf{y}(I) & \text{(3.4a)} \\ &= -\mathbf{w}_T^{'} \cdot \mathbf{w}_I + \log \sum_{j=1}^{V} \exp(\mathbf{w}_j^{'} \cdot \mathbf{w}_I). & \text{(3.4b)} \end{aligned}$$

Another essential feature of the word embedding is that both the input word and the target word are taken from a single context. That is to say, an embedding model calculates predictivity or co-occurrence between the target word and the input word in a single sentence. This strategy makes the embedding model as an unsupervised machine learning problem, so one can easily enlarge the size of the pre-training dataset. There are two models in Word2Vec: the continuous bag of words (CBOW) model and the skip-gram model. As shown in Fig. 3.2, the CBOW model predicts the central word from its neighboring words; the skip-gram model uses the central word as the input to predict its neighbors. The model complexity of a CBOW model,

Q is dependent on the embedding dimension D, the window length N and the vocabulary size V.

$$Q = D(N + \log_2 V), \tag{3.5}$$

and for a skip-gram model, Q is as follows:

$$Q = ND(1 + \log_2 V). \tag{3.6}$$

The logarithmic dependence on the vocabulary size $\log_2 V$ is originated from the *hierarchical softmax* activation function, which makes it unncessary for the model to update all weights in $\mathbf{W}$ and $\mathbf{W}'$[51].

A number of studies showed that the the unsupervised context learning in the word embedding scheme can also be a powerful tool for encoding structural features of chemical compounds[18, 23, 43, 54]. The idea is realized by the consideration of a given molecular structure as *chemical contexts* of atoms of substructure; positions of projected atomic feature vectors in the embedded vector space now represent their chemical or physical properties, instead of linguistic information. In the present study, we use Mol2Vec embedding model as the primary encoding means[54], which uses the Morgan algorithm to assort atoms in an identical chemical environment and generate the chemical context of a given compound[56].

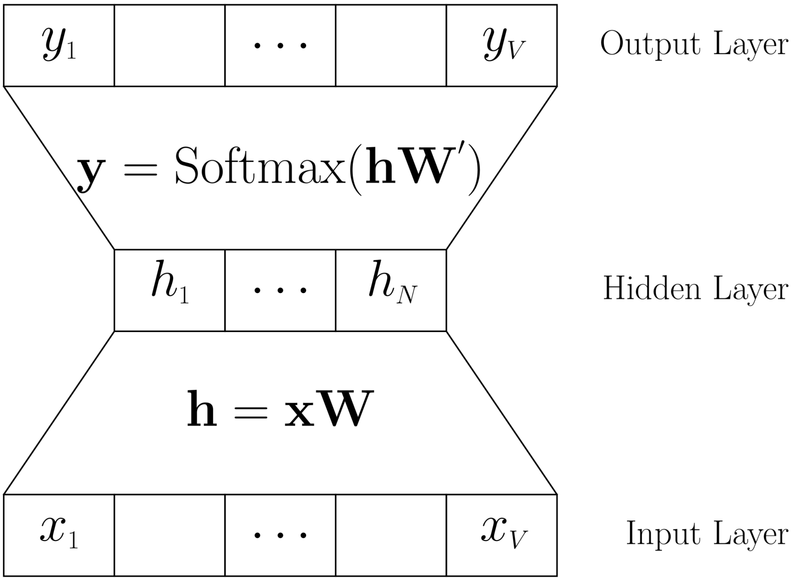


Figure 3.1: Embedding procedure for simple one-word context.

3.1.2 Network Architecture

In the proposed model, the linear regression task between the given chemical structures of the solvent and solute molecules and their solvation free energy starts with embedded vector representations of the given solvent $\mathbf{x}_\alpha$ and solute $\mathbf{y}_\gamma$, where α and γ are atom indices. The entire molecular structure is now can be expressed as a sequence of vectors or a matrix:

$$\mathbf{X} = \{\mathbf{x}_\alpha\}\,, \tag{3.7a}$$

$$\mathbf{Y} = \{\mathbf{y}_\gamma\}\,, \tag{3.7b}$$

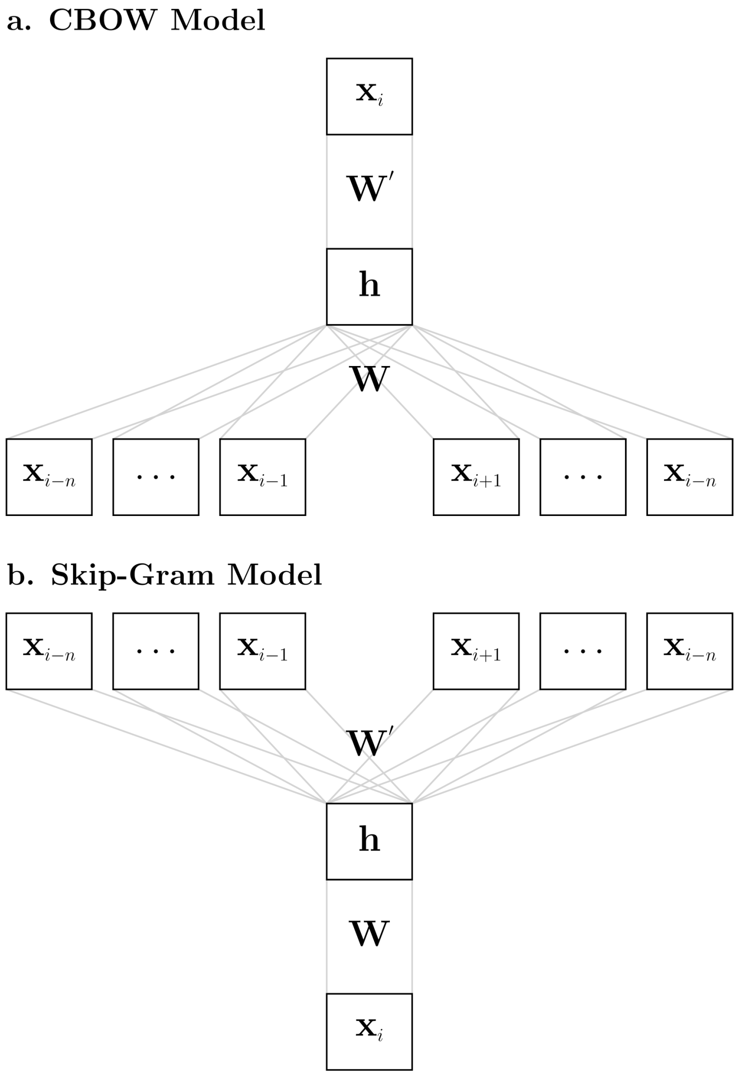


Figure 3.2: Model architecture diagrams for (a) the CBOW model and (b) the skip-gram model. The CBOW model predicts the current word based on neighboring words, while the skip-gram words predicts surrounding words from the current word.

so $\mathbf{x}_\alpha$ and $\mathbf{y}_\gamma$ are α-th row of $\mathbf{X}$ and γ-th row of $\mathbf{Y}$, respectively. Then the encoder function learns their chemical structures and extracts feature matrices for the solvent $\mathbf{P}$ and the solute $\mathbf{Q}$.

$$\mathbf{P} = \mathrm{Encoder}(\mathbf{X}), \tag{3.8a}$$

$$\mathbf{Q} = \mathrm{Encoder}(\mathbf{Y}). \tag{3.8b}$$

Columns of $\mathbf{P}$ and $\mathbf{Q}$, p_α and q_γ involve atomistic chemical features of atoms α and γ, which are directly related to the target property, the solvation free energy. We now calculate the un-normalized attention (or *chemical similarity*) between α and γ with on Luong's dot-product attention score function[50]:

$$I_{\alpha\gamma} = -\mathbf{p}_\alpha \cdot \mathbf{q}_\gamma. \tag{3.9}$$

Since our target quantity is the free energies of solvation, we expect such chemical similarity $I_{\alpha\gamma}$ to well correspond to atomistic interactions between α and γ, which involves both the energetic and the entropic contributions. Eventually, the free energy of solvation of the given pair, which is the final regression target, is given as a simple summation of atomistic interactions:

$$\Delta G^\circ_{sol} = \sum_{\alpha\gamma} I_{\alpha\gamma}. \tag{3.10}$$

Certainly, one can also calculate the free energies of solvation from two molecular feature vectors, those are representing the solvent properties $\mathbf{u}$ and the solute properties $\mathbf{v}$, respectively:

$$\Delta G^{\circ}_{sol} = \mathbf{u} \cdot \mathbf{v} = \left(\sum_{\alpha} \mathbf{p}_{\alpha}\right) \cdot \left(\sum_{\alpha} \mathbf{q}_{\alpha}\right). \tag{3.11}$$

The inner-product relation between molecular feature vectors $\mathbf{u}$ and $\mathbf{v}$ has a formal analogy with the solvent-gas partition coefficient calculation method via the solvation descriptor approach, which is founded by Abraham and Acree[78, 79]:

$$\log K = c + eE + sS + aA + bB + lL. \tag{3.12}$$

In Eqn. 3.12, the solute descriptor $(1, E, S, A, B, L)$ is determined from a series of experimental measures, and the solvent descriptor (c, e, s, a, b, l) is a fitted value. In our proposed model, both $\mathbf{u}$ and $\mathbf{v}$ are purely fitted quantities from the scratch, with the skip-gram pre-training and the linear regression analysis.

We choose and compare two different neural network models in order to encode the input molecular structure and extract important structural or chemical features which are strongly related to solvation behavior: one is bidirectional language model (BiLM)[80] based on the recurrent neural network (RNN), the other is the graph convolutional neural network (GCN)[81] which explicitly handles the connectivity (bonding) between atoms with the adjacency matrix.

The detailed mathematical expressions of the bidirectional language model are given below[80]:

$$\overrightarrow{\mathbf{H}}^{(i+1)} = \overrightarrow{\mathrm{RNN}}(\overrightarrow{\mathbf{H}}^{(i)}), \tag{3.13a}$$

$$\overleftarrow{\mathbf{H}}^{(i+1)} = \overleftarrow{\mathrm{RNN}}(\overleftarrow{\mathbf{H}}^{(i)}). \tag{3.13b}$$

In Eqn. 3.13, the right-headed arrow in $\overrightarrow{\mathrm{RNN}}$ denotes a forward-directed recurrent unit which propagates from the leftmost of the sequence to the rightmost one. The BiLM also involves the backward-directed recurrent neural network ($\overleftarrow{\mathrm{RNN}}$) and it propagates from the rightmost to the leftmost. The superscript (i) in hidden layers $\mathbf{H}^{(i)}$ denotes the position at the stacked configuration: at the first stack, both forward-directed and backward-directed RNN share the pre-trained sequence $\mathbf{X}$ as an input, $\overrightarrow{\mathbf{H}}^{(0)} = \overleftarrow{\mathbf{H}}^{(0)} = \mathbf{X}$. In addition, use of more improved versions of RNNs, e.g. the gated recurrent unit (GRU)[61] or the long-short term memory (LSTM)[60], are more suitable when one considers cumulated numerical errors due to the deep-structured nature of RNNs[59],

$$\mathbf{H}^{(i)} = \overrightarrow{\mathbf{H}}^{(i)} + \overleftarrow{\mathbf{H}}^{(i)}. \tag{3.14}$$

Hidden layers from the forward and backward RNNs are then merged into a single sequence, as described in Eq. 3.14. Finally, we obtain the sequence of chemical feature vectors of the α-th atom in the given solvent with weighted summation of rnn stacks,

$$\mathbf{P} = \sum_i c_i \mathbf{H}^{(i)}. \tag{3.15}$$

The encoder function for solutes has an identical neural network architecture, which converts the pre-trained solute sequence $\mathbf{Y}$ into the feature sequence $\mathbf{Q}$.

To sum up, the BiLM encoder considers a given molecule as just a simple sequence of atomic vector representations. The idea is quite clear and rather straightfoward for implementation of the neural network. However, this idea may causes "problems" in more complex compounds due to the lack of intramolecular bonding information between atoms. We also consider the graph convolutional neural network (GCN), which is one of the most well-known algorithms in chemical applications of neural networks[34, 81]. The GCN model represents the input molecule as a mathematical graph, instead of a simple sequence: each node corresponds to the atom, and each edge in the adjacency matrix $\mathbf{A}$ involves connectivity (or existence of bonding) between atoms:

$$\mathbf{H}^{(i+1)} = \mathrm{GCN}(\mathbf{H}^{(i)}, \mathbf{A}). \tag{3.16}$$

The role of adjacency matrix in the GCN constrains convolution filters to the node and its nearest neighbors. Eqn. 3.17 describes a more detailed mathematical expression of the skip-connected GCN[81]

$$\mathrm{GCN}(\mathbf{H}, \tilde{\mathbf{A}}) = \sigma(\tilde{\mathbf{A}}\mathbf{H}\mathbf{W}_1 + \mathbf{H}\mathbf{W}_2 + \mathbf{b}), \tag{3.17}$$

where $\mathbf{W}_1$ and $\mathbf{W}_2$ are convolution filters, $\mathbf{b}$ is the bias vector, and σ denotes the activation function - we choose the hyperbolic tangent in the proposed model. The GCN encoder also invloves stacked structure, and we can obtain the feature sequence for each molecule with the same manner as described in Eqn. 3.15.

3.2 Results and Discussions

3.2.1 Computational Details

For the training and test tasks of the proposed neural network, we prepare 6,594 experimental measures of free energies of solvation for 952 organic solvents and 147 organic solutes, including some inert gases. 642 experimental measures for free energies of hydration are taken from the FreeSolv

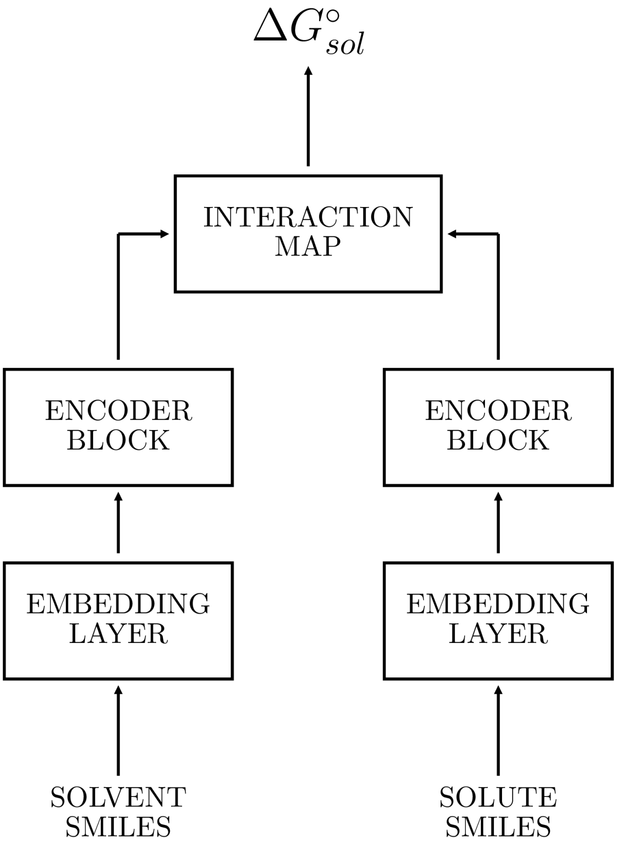


Figure 3.3: Architecture of the proposed model. Each encoder network extracts atomistic feature vectors given pre-trained vector representations, and the interaction map calculates pairwise atomistic interactions.

database[14],and 5,952 data points for non-aqueous solvents are collected with the Solv@TUM database version 1.0[78, 79], which is available at https://github.com/hille721/solvatum. Compounds in the dataset involves 10 kinds of atoms, which are commonly used in organic chemistry: hydrogen (H), carbon (C), oxygen (O), sulfur (S), nitrogen (N), phosphorus (P), fluorine (F), chlorine (Cl), bromine (Br), and iodine (I). The maximum heavy-atom count is 28 for solutes and 18 for solvents.

For the very first stage, we perform the skip-gram pre-training process for 10,229,472 organic compounds, which are collected from the ZINC15 database[82], using Gensim 3.8.1 and Mol2Vec skip-gram model to construct the 128-dimensional embedding lookup table[54]. For the implementation of the neural network model, we mainly use the Tensorflow 2.0 and Keras 2.3.1 frameworks[65]. To construct the BiLM encoder, we both consider CuDNN implementations[65] for the LSTM and the GRU, which are basic layers in the Tensorflow. For GCN encoder, we use codes taken from Spektral library version 0.1.1, which implements the skip-connected graph convolutional network. Each model has L2 regularization to prevent excessive changes on weights and minimize the variance and uses the RMSprop algorithm with 10^{-3} of learning rate and $\rho = 0.9$ for optimizing its loss function, the mean squared error (MSE).

We employ 5-fold cross-validation to evaluate the prediction accuracy of the chosen model; the entire dataset is randomly split into five uniform-sized subsets, and we iteratively choose one of the subsets as a test set, and the training run uses the remainder 4 subsets. Consequentially, a 5-fold CV task performs 5 independent training and test runs, and relative sizes of the training and test sets are 8 to 2. To minimize the variation of results from CV tasks, we take averages for all results over 9 independent random CV, split from different random states. The procedure for CV is implemented with the Scikit-Learn library version 0.2.2[66].

3.2.2 Prediction Accuracy

The selection of the optimized model for the target property is realized by an extensive grid-search task for tuning model hyperparameters. First, we choose 32 as the batch size, and RMSprop as an optimization algorithm with learning rate is 10^{-3}. It is generally known that the smaller batch size generates a better result; however, a too small batch size is computationally inefficient, so we take the value of 32 as the point of compromise between the prediction performance and the computational efficiency. Table 3.2.2 shows additional searching information for the optimized stack size of the encoder networks and maximum epochs are 50 for the BiLM model and 100 for the GCN model, respectively. Fig. 3.4 shows epoch-evolution of training and validation loss for both the BiLM/LSTM encoder and the GCN encoder, where optimized stack size is 3. BiLM encoder shows a much faster convergence behavior untill $\sim$ 50 epochs and overfitting appears, while the

GCN encoder exhibits minimum validation loss around ~ 100 epochs.

The results for test run using 5-fold CV tasks for the optimized models with grid search tasks are shown at Fig. 3.5. We found that the BiLM encoder with the LSTM layer performs slightly better than the GCN encoder, although their differences are not pronounced: the mean unsigned prediction error (MUE) for the BiLM/LSTM encoder model is 0.19 kcal/mol, while the GCN model results in 0.23 kcal/mol. Both MUE values show that the our proposed mechanism is actually working and guarentees excellent prediction accuracies for well-trained chemical structures. Moreover, since we use a simple version of the graph-based neural network as the encoder, we might expect the GCN-based model to perform better than a simple graph-based embedding model or more progressed version of graph neural networks to perform even better for chemical structures: such as the messege-passing neural network (MPNN)[35], the deep tensor neural network (DTNN)[36], and so on.

As the last of this section, we confirm whether or not the proposed neural network architecture is working as we designed. Fig. 3.6 presents t-SNE visualizations for pre-trained solute vectors $\mathbf{y}$ and encoded molecular feature $\mathbf{v}$[38]. Color codes denote predicted hydration free energies for 15,432 points, whose structures are randomly taken from the ZINC15[82]; red dots correpond to the compounds with low hydration free energies while the blue dots correspond to them with high hydration free energies. The correlation

Encoder	Stack	Training RMSE	Validation RMSE	Test RMSE
BiLM	1	0.29 ± 0.00	0.59 ± 0.04	
	2	0.24 ± 0.01	0.44 ± 0.04	
	3	$\mathbf{0.24 \pm 0.01}$	$\mathbf{0.43 \pm 0.02}$	$\mathbf{0.41 \pm 0.01}$
	4	0.23 ± 0.00	0.49 ± 0.03	
	5	0.20 ± 0.02	0.52 ± 0.02	
GCN	1	0.34 ± 0.00	0.73 ± 0.04	
	2	0.26 ± 0.00	0.70 ± 0.07	
	3	0.25 ± 0.00	0.51 ± 0.08	
	4	$\mathbf{0.26 \pm 0.01}$	$\mathbf{0.46 \pm 0.05}$	$\mathbf{0.44 \pm 0.01}$
	5	0.27 ± 0.01	0.77 ± 0.16	

Table 3.1: Error metrics for training, validation, and test runs with respects to the number of stacked encoder layers. The units of all errors are $\mathrm{kcal/mol}$.

between molecular features and predicted free energies is a clear clue that the model architecture can extract geometrical correlations and calculate free energy. Meanwhile, the pre-trained solute vectors from the skip-gram embedding model exhibit only weak correlations.

3.2.3 Model Transferability

Since our proposed neural network model is a solvent-non-specific one that considers both the solvent structure and the solute structure as seperate inputs, it has a distinct character when compared to the other solvent-specific ML models. The model can train with the structure of a single solute repeatedly when the solute has multiple solvation energy data for different kinds of solvents[22]; this logic is also valid for a single solvent. Therefore, one of the most useful advantages of our model is that we can easily enlarge the

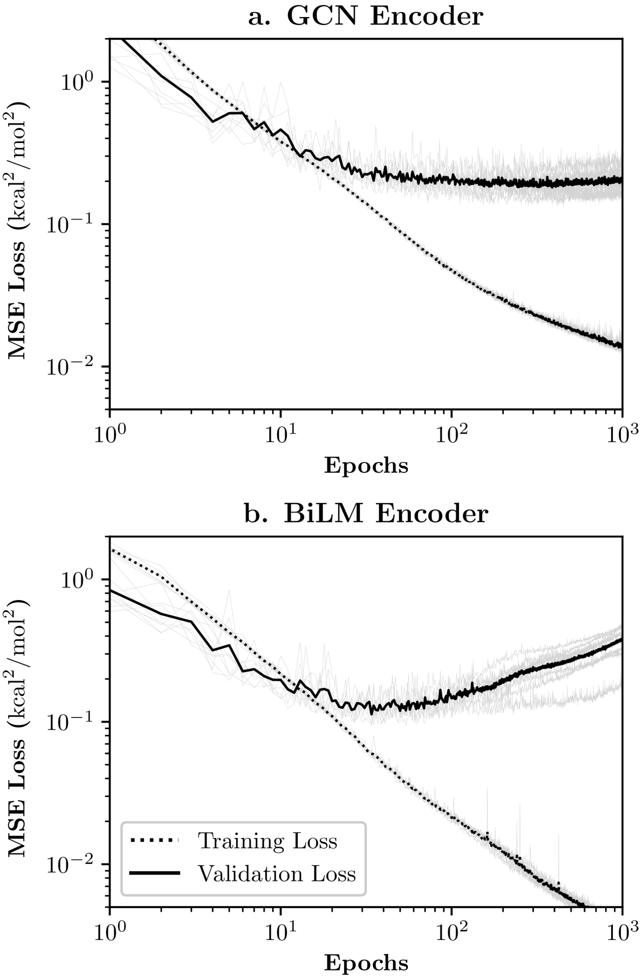


Figure 3.4: Epoch-evolution of mean squared loss functions (RMSE) for (a) the GCN encoder model and (b) the BiLM encoder model. Solid lines denote evolution of training losses while dotted lines denote validation losses. All results are averaged over 8 independent cross-validation runs.

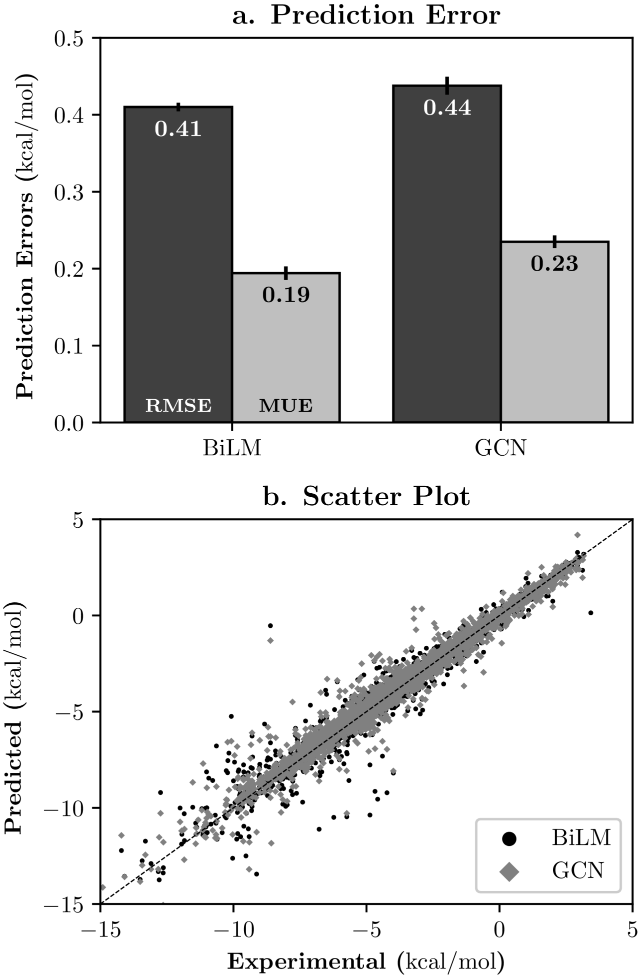


Figure 3.5: (a) Prediction erros for two models in kcal/mol, taken from 5-fold cross validation results. (b) Scatter plot between the experimental value and ML the ML predicted value. Black circles denote the BiLM model while the GCN results are shown in gray diamonds.

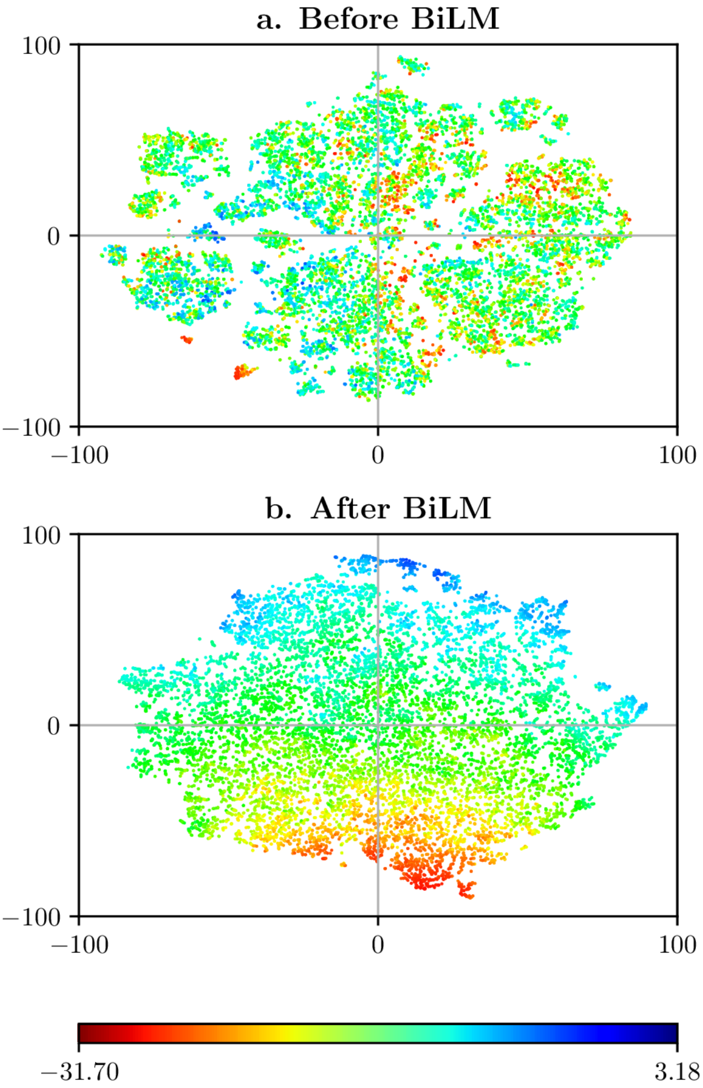


Figure 3.6: 2-dimensional visualizations on (a) the pre-trained vector $\sum_{\gamma} \mathbf{y}_{\gamma}$ and (b) the molecular feature vector $\mathbf{v}$ for 15,432 solutes. We reduce the dimension of each vector with the t-SNE algorithm. The color representation denotes the hydration energy of each point.

dataset for training, even in the scenario that we want to predict solvation free energies for a specific solvent. Fig. 3.7 shows 5-fold cv results for 642 hydration free energies (FreeSolv) from both the BiLM and the GCN models, in two different situations. One uses only the FreeSolv[14] database for train and test tasks, and the other additionally uses the Solv@TUM[78, 79]. Although the Solv@TUM database only involves non-aqueous data points, it enhances each model's accuracy by about 20% (BiLM) to 30% (GCN) in terms of mean unsigned errors. Those results imply that there are possible applications of the transfer learning to other solvation-related properties, like aqueous solubilities[4] or octanol-water partition coefficients.

However, in some other situations, the advantage we discussed above might be a downside: the repetitive training for a single compound may make the model tends to overfit, and they could weaken predictivity for the structurally new compound, which is considered as an extrapolation. We investigate the model's predictivity for extrapolation situations with the *scaffold-based* split[22, 35, 43]. Instead of the ordinary K-fold CV task with the random and uniform split method, the K-means clustering algorithm builds each fold with the MACCS substructural fingerprint. One can simulate an extreme extrapolation situation through CV tasks over the clustered fold. As shown in Fig. 3.8, albeit the scaffold-based split degrades MUEs by a factor of three, they are still within an acceptable error range $\sim 0.6 \text{ kcal/mol}$, given chemical accuracy 1.0 kcal/mol. Furthermore, we

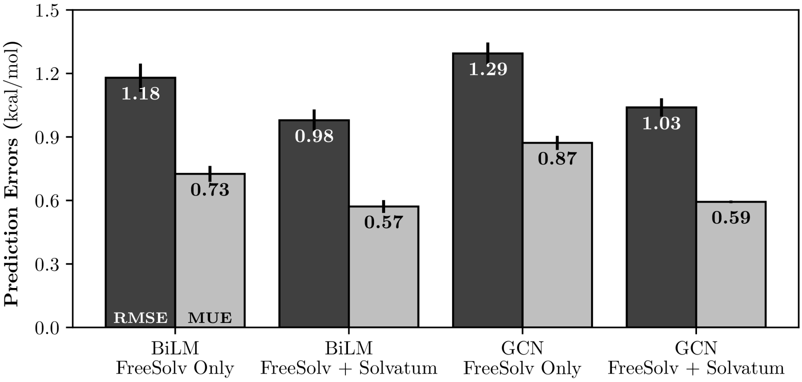


Figure 3.7: CV-results for FreeSolv hydration energies with two different training dataset selection. Deep-colored boxes denote CV results with the augmented dataset with the Solv@TUM database.

do not see any clear evidence that our model tends to overfit more than other solvent-specific models[35, 43].

3.2.4 Group Contributions of Solvation Energy

Although we showed that the proposed NN model guarantees an excellent predictivity for solvation energies of various solute and solvent pairs, the main objective of the present study is obtaining the solvation free energy as the sum of decomposed inter-atomic interactions, as we described at Eq. 3.9 and 3.10. In order to verify whether or not the the model's solvation energy estimation has correspondence to group-contribution based calculation, we define the sum of atomic interactions $\mathbf{I}_{\alpha\gamma}$ over the solvent indices γ as the

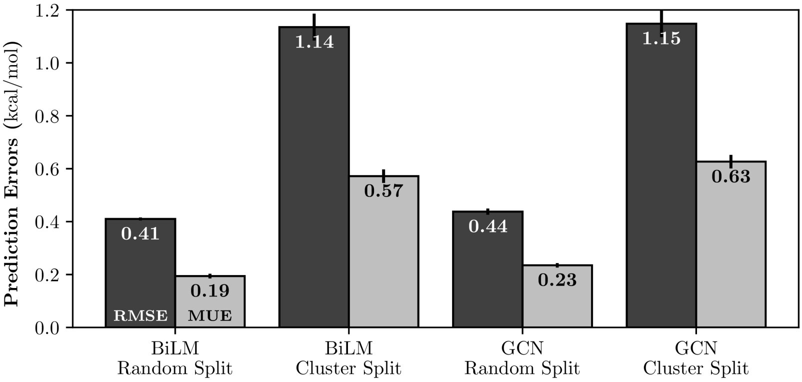


Figure 3.8: Comparison between CV results with the random-split and the scaffold-based split (or cluster split).

group contributions of the α-th solute atom:

$$\mathbf{I}_{\alpha} = \sum_{\gamma} \mathbf{I}_{\alpha\gamma}. \tag{3.18}$$

Figure 3.9 shows hydration free energy contributions for four linear and small organic solutes which have six heavy atoms: n-hexane (CCCCCC), 1-chloropentane (CCCCCCl), pentaldehyde (CCCCC=O), and 1-aminopentane (CCCCCN). As shown in Fig. 3.9, both the BiLM and the GCN model exhibit a resembling tendency in group contributions; the model estimates that atomic interactions between the solute atoms and water increases near the hydrophilic groups. Although the results show that we can find a significant correspondence to intuitive chemical knowledge, it might need further quantified analysis of computer simulation approaches. For example, molecular dynamics simulations with an appropriate explicit solvation model. The Kirkwood charging formula can give atomic free energy contributions with pairwise interactions $u(\mathbf{r}, \lambda)$ and the solvation shell structure $g(\mathbf{r}, \lambda)$[10]:

$$\mu = \rho \int_0^1 d\lambda \int d\mathbf{r} g(\mathbf{r}, \lambda) \frac{\partial u(\mathbf{r}, \lambda)}{\partial \lambda}. \tag{3.19}$$

However, there is an aspect that we can easily verify without quantitative computer simulations. It is obvious that each atom in cyclohexane and benzene must have identical contributions to the free energy, but the results in Fig. 3.10 clearly shows that the BiLM model makes faulty predictions while the GCN model works well as expected. We believe that this malfunctioning of the BiLM model originates from the sequential nature of the recurrent neural network. Since the RNN considers the input molecule is just a simple sequence of atomic vectors and there are no explicit statements that involve bonding information, the model could not be aware of the cyclic shape of the input compound[23, 34]. We conclude that it is inevitable to use explicitly bond (or connectivity) information when one constructs a group-contribution based ML model, although the RNN-based model well predicts in terms of their sum.

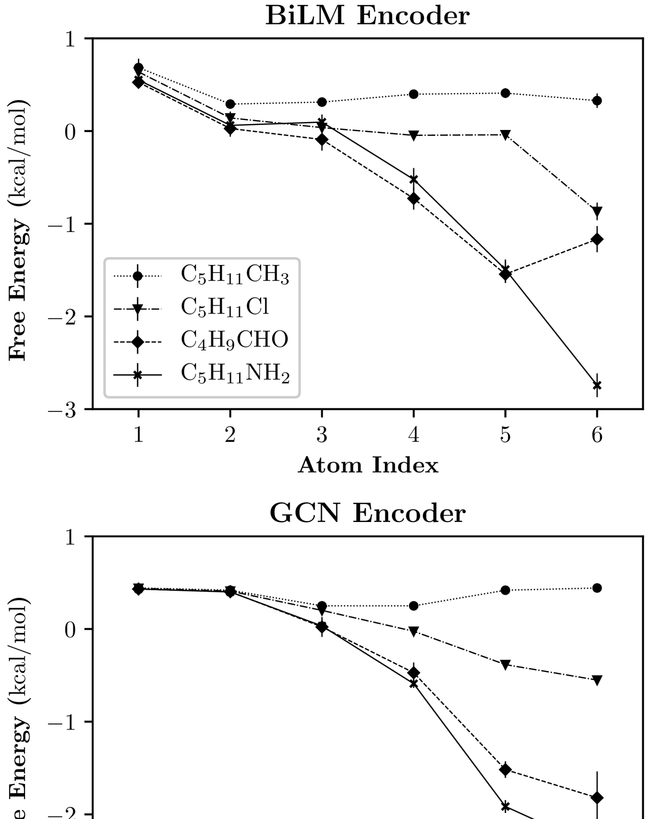


Figure 3.9: ML-calculated atomistic group contributions for four small, linear organic molecules which have six heavy atoms. The atom index starts from the leftmost of the given molecule and only counts heavy atoms.

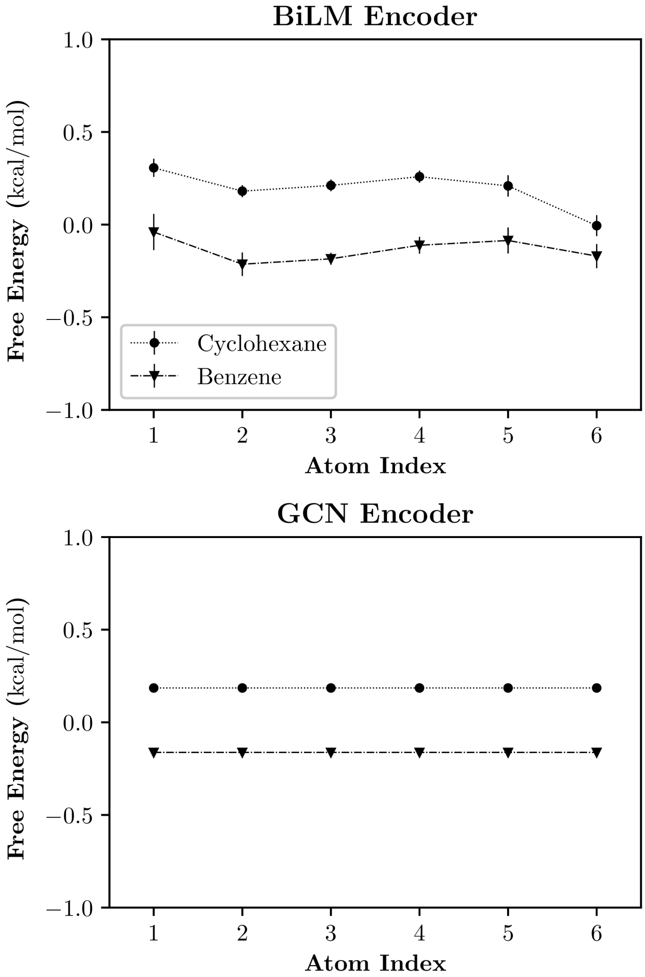


Figure 3.10: Group contributions for two simple cyclic compounds: cyclohexane and benzene.

Chapter 4

Empirical Structure-Property Relationship Model for Liquid Transport Properties

In this chapter, we present a simple structure-property relationship estimation procedure for two major transport properties of the liquid state: the dynamic viscosity (η) and the dielectric constant (ϵ).

Computer simulation approaches for the calculation of transport properties are not easily feasible since they are non-equilibrium measures which are depending on the external field: shear stress (viscosity) and electric field (dielectric constant). Generally, the calculation of transport property via equilibrium simulation needs to generate multiple molecular dynamics trajectories to evaluate the Green-Kubo relation, which is the exact mathematical expression for transport coefficients in the linear response regime[83]:

$$\gamma = \int_0^\infty d\tau \, \langle A(0)A(\tau) \rangle \, . \tag{4.1}$$

Eqn. 4.1 calculates the given transport coefficient γ with the time integration of a specific time correlation function. At high-viscous liquids, it is difficult to sample trajectories and calculate the Green-Kubo relation due to extremely slow relaxation of the liquid system.

In previous chapters, we showed that the structure-property relationship could be a powerful tool for the prediction of the free energy of solvation. Here, we seek another application of SPR estimation of non-equilibrium transport properties, which might be applicable in many systems - even in viscous liquids. The basis of the present SPR model is the decision-tree regression model; the model generates tree-like graphs of decision rules and learns the training database[84]. Also, we employ two *ensemble methods*, the *random forest*[85] (RF) and the *gradient boosting*[86] (GBM) algorithms to minimize bias and variance of the tree-based machine learning model.

The mathematical expression of the ensemble method starts with the mathematical function F of a regression model an input descriptor $\mathbf{x}$ to its label y[86]:

$$\hat{y}_i = F(\mathbf{x}_i; \mathbf{P}), \tag{4.2}$$

where $\mathbf{P}$ is the collection of trainable parameters of the function F and $\hat{y}$ is the predicted value of the model, given input descriptor $\mathbf{x}$. The linear regression task loss function $L(y_i, F(\mathbf{x}_i)) = (y_i - \hat{y}_i)^2$ by the least-square method.

$$\mathbf{P}^* = \arg\min_{\mathbf{P}} \sum_i L(y_i, F(\mathbf{x}_i; \mathbf{P})). \tag{4.3}$$

A random forest regression model involves a set of independent, randomly generated decision-tree *subpredictors* $\{F_1(\mathbf{x}; \mathbf{P}_1), \cdots, F_K(\mathbf{x}; \mathbf{P}_K)\}$, and one can get the optimized model from the ensemble average over K "weakly-optimized" subpredictors[85].

$$\mathbb{F}(\mathbf{x}_i) = \sum_{k=1}^{K} F_k(\mathbf{x}_i; \mathbf{P}_k^*). \tag{4.4}$$

If the model is a classification problem, each subpredictor casts a unit vote for the selection of the most popular class.

The gradient boosting algorithm takes a different approach to the RF model. It has an analogy with the RF that the model consists a set of subpredictors, however, instead of the ensemble average over subpredictors, a GBM model updates its prediction model F_k via the sequential iteration task and chooses the last model F_K^* as the optimized model[86]:

$$F_{k+1}^*(\mathbf{x}) = F_k^*(\mathbf{x}) + h_k(\mathbf{x}). \tag{4.5}$$

Here, we fit the *base learner* h_k with *pseudo-residuals* $\{r_{ik}\}$:

$$r_{ik} = -\left[\frac{\partial L(y_i, F(\mathbf{x}_i))}{\partial F(\mathbf{x}_i)}\right]_{F(\mathbf{x}) = F_{k-1}^*(\mathbf{x})}. \tag{4.6}$$

At the very first stage, the initial model F_1^* is equivalent with Eqn. 4.3.

We perform an extensive searching task over tens of elementary structural properties and choose the collection of 19 values, which are shown in Table. 4.1, as the optimized molecular descriptor for liquid transport properties. All properties are available in RDkit 2019.09 python module, and their evaluation process does not require additional simulations or theoretical calculations. For the train and validation tasks, we collect 1,375 experimental data for the liquid dynamic viscosity and the relative permittivity (the dielectric constant) from the web version of DIPPR 801 database[87]. The two decision-tree based ensemble models are implemented using Scikit-learn 0.22[66] and XGBoost 0.90 libraries.

We optimize the hyperparameters and evaluate the predictivity of two models for two transport properties using the 5-fold cross-validation task. The optimized RF model's maximum tree depth is 8, while the GBM model has 6 maximum nodes; both models have the same number of estimators, 100. Fig. 4.1 shows scatter plots between experimental values (x-axis) and predicted values (y-axis). We also specify the Pearson correlation coefficient in order to indicate the prediction accuracy of each model. The GBM model shows better accuracy: R^2 values are 0.91 for the dynamic viscosity and 0.81 for the dielectric constant in the logarithmic scale, respectively. while the RF model shows $R^2 = 0.89$ for the viscosity and 0.78 for the dielectric constant, respectively.

No.	Property	Unit
1	Molecular weight	A. U.
2	Heavy atom weight	A. U.
3	Maximum partial charge	e
4	Minimum partial charge	e
5	Fraction of sp^3 carbons	-
6	Labute accessible surface area	Å^2
7	Topological polar surface area	Å^2
8	Number of aliphatic carbocycles	-
9	Number of aliphatic heterocycles	-
10	Number of aromatic carbocycles	-
11	Number of aromatic heterocycles	-
12	Number of saturated carbocycles	-
13	Number of saturated heterocycles	-
14	Number of stereo centers	-
15	Number of hydrogen bond acceptor	-
16	Number of hydrogen bond donor	-
17	Number of Lipinski hydrogen bond acceptor	-
18	Number of Lipinski hydrogen bond donor	-
19	Number of heteroatoms	-

Table 4.1: Collection of 19 elementary structural properties for the description of a given organic molecule. All properties are available in RDKit python module.

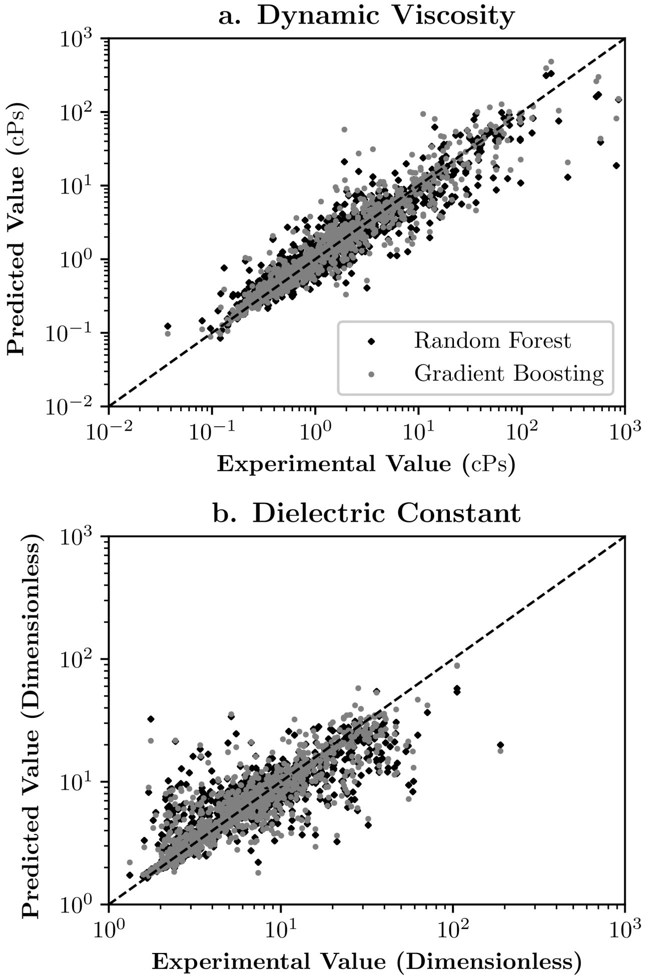


Figure 4.1: Scatter plots for (a) the dynamic viscosity and (b) the dielectric constant, respectively. ML predictions are obtained using 5-fold cross-validation tasks over 1,375 data points, which are taken from the DIPPR 801 database.

Chapter 5

Concluding Remarks

In the present study, we introduced a new approach for the solvation energy prediction, which has a great potential to provide physicochemical insights on the solvation process. The novelty of our neural network model is that the model does not involve the perceptron networks for readout of encoded features and estimation of the target property. Alternatively, we designed the model such that it is possible to calculate pairwise atomic interactions from inner products of atomistic feature vectors[50]. As a result, the model produces the solvation free energy from the group-contribution based prediction.

In Chapter 2, we reviewed our previous ML solvation model, Delfos. The extensive calculations on 2495 experimental values[62] demonstrate that Delfos exhibits excellent prediction accuracy, which is comparable with several well-known QM solvation models[1, 2] when the neural network is trained with sufficiently varied chemical structures. Decline in performances about 0.5 to $0.7\ \mathrm{kcal/mol}$ at the cluster CV tasks represents the accuracy for a structurally new compound, suggesting the importance of preparation of the ML databases even though Delfos still demonstrates comparable predictions with some theoretical approaches such as MD with AMOEBA force field[44] or DFT with pure COSMO[2]. The score matrix taken from the attention mechanism gives us an interaction map between atoms and substructure; our model does provide not only a simple estimation of target property but offers important pieces of information about which substructures play a dominant role in solvation processes.

In Chapter 3, we introduced a new model for the solvation energy estimation and quantified the proposed model's prediction performances for 6,493 experimental data points of solvation energies, which were taken from the FreeSolv[14] and Solv@TUM database[37, 79]. We found a significant geometrical correlation between molecular feature vectors and predicted properties, which implies that the proposed model is actually working as we designed. The estimated prediction MUEs from K-fold CV are $0.19\ \mathrm{kcal/mol}$ for the BiLM encoder and $0.23\ \mathrm{kcal/mol}$ for the GCN model, respectively.

The K-fold CV results from the scaffold-based split[43] showed the prediction accuracy decreases by a factor of three in extreme extrapolation situations, but they still exhibit moderate performances, which were $0.60\ \mathrm{kcal/mol}$. Moreover, we found that the solvent-non-specific structure of the proposed model is appropriate for enlarging dataset size, that is to say, experimental data points for a particular solvent is transferable to other solvents; we conclude that this transferability is the reason for our model's outstanding predictivity[22].

Finally, we examined pairwise atomic interactions that are obtained from the interaction map $\mathbf{I}$ and found a clear tendency between hydrophilic groups and their contributions to the hydration free energy. However, the BiLM model with the recurrent network has some faulty aspects in symmetric or cyclic compounds, albeit it showed better predictions in terms of the total solvation energy. This fact implies the sequential nature of the recurrent network is inappropriate for constructing a group-contribution model, and an explicit usage of the chemical bonding information is inevitable. Although our results need an extra investigation from a quantitative point of view[10], we believe that our model can provide detailed information on the solvation mechanism, not only the predicted value of the target property.

Appendix A

Analyzing Kinetic Trapping as a First-Order Dynamical Phase Transition in the Ensemble of Stochastic Trajectories

A.1 Introduction

Self-assembly is the spontaneous process of disordered components to form ordered patterns or structures. It is one of the most extensively studied research area for complex systems[88–95]. Physical interactions between components play a major role in the self-assembly process. Strength and specificity of the interactions induce the assembling process and determine their assembled structure in the equilibrium condition. However, an obstacle due to an energetic and/or entropic barrier makes it difficult for the system to relax via the reversible dynamics, which hinders the formation of desired assembly structure. The irreversible behavior in bond making and breaking will hinder misbounded components to adjust their bonds easily[91, 96]. Occasionally the system will get trapped in the meta-stable glassy state instead of its equilibrium structure. This behavior is usually called *kinetic trapping*. There have been numerous works in computer simulation studies[97–104] in order to avoid kinetic trapping and achieve effective assembled structure.

A molecular dynamics study of viral capsid growth reported the importance of reversibility and interaction strength in self-assembly at submicroscopic scale[105]. In the work, the authors inspected the time evolution of the cluster size distributions and argued excessive early growth makes monomers trapped in the imperfect shell, resulting in a shortage of free monomer. Analyzing the fluctuation-dissipation ratio (FDR) is another useful strategy for analyzing reversibility[102]. The correlation-response relation showed that the system is in short-time quasi-equilibrium states and reversible in that time scale when the system shows a good assembly kinetics. A notable advance is demonstrained from the direct measurements of bond making and breaking events[99, 103]. In Refs. 99 and 103, the authors defined the *flux* and the *traffic*, which represents the net rate of bond making and total events time scale, respectively. These two quantities give us knowledge of the microscopic reversible behavior of bond-making and breaking progress.

Since the self-assembly is an out-of-equilibrium process, studying its behavior through equilibrium statistical mechanics is usually not valid. For that reason, as we have mentioned earlier, a majority of preceding studies have been based on manners of non-equilibrium statistical mechanics. Meanwhile, recent progresses in the non-equilibrium statistical mechanics framework introduced a useful method to handle out-of-equilibrium processes by biasing trajectories[106–115]. The essential idea of the theory is to implement the large deviation principle in trajectory space as the traditional framework of statistical mechanics has done in phase space. The theory successfully proved that there exists dynamical symmetry breaking in several models of glass formers by both analytical and numerical scheme[107, 108, 114]. Besides, this approach suggested there is practicability of to manage thermodynamic properties like configuration, local structure or energy via a purely dynamical method[116–118].

The self-assembly process has its analogy with the glass forming system in that both systems usually prepared up via temperature quenching from the disordered structure to ordered equilibrium or metastable structure. Focused on this point, we make an attempt to implement the above-mentioned non-equilibrium ensemble of trajectories in the self-assembly system, which has never been tried before, to analyze and quantify the dynamics of the process. Our goal is to understand the obstacle due to the restricted dynamics in the self-assembly process as a dynamical symmetry breaking in trajectory space. We expect our work will give an entirely new perspective to understand the kinetic trapping and the reversible dynamics in self-assembly processes.

A.2 Theory

In this study, we use the activity of a given trajectory as a measurable observable, which is projecting the reversibility of the self-assembling system. Consider a stochastic trajectory $\mathbf{X}$ of classical and discrete Markov process; we can regard the trajectory as a set of time-evolving configurations $(\mathbf{x}, t)$: $\mathbf{X} = \{(\mathbf{x}_K, t_K), \ \cdots, \ (\mathbf{x}_0, t_0)\}$. The probability of finding a single trajectory when observing a given system is described as successive products of transition probability $p(\mathbf{x}_{i+1}, t_{i+1}|\mathbf{x}_i, t_i)$ from the current configuration $(\mathbf{x}_i, t_i)$ to next one $(\mathbf{x}_{i+1}, t_{i+1})$ and the population of its starting configuration $p(\mathbf{x}_0, t_0)$[119, 120]:

$$\begin{aligned} P[\mathbf{X}] =& p(\mathbf{x}_K, t_K|\mathbf{x}_{K-1}, t_{K-1}) \\ & \cdots p(\mathbf{x}_1, t_1|\mathbf{x}_0, t_0) p(\mathbf{x}_0, t_0). \end{aligned} \tag{A.1}$$

We assume that the dynamics of the system is governed by the master equation $\partial_t |\mathbf{p}(t)\rangle = \mathbb{W} |\mathbf{p}(t)\rangle$ and since the model is a discrete process, the master operator is defined as a matrix form:

$$\mathbb{W} = \sum_{\mathbf{x}' \neq \mathbf{x}} w(\mathbf{x}'|\mathbf{x}) |\mathbf{x}'\rangle \langle \mathbf{x}| - \sum_{\mathbf{x}} r(\mathbf{x}) |\mathbf{x}\rangle \langle \mathbf{x}| . \tag{A.2}$$

Here, $w(\mathbf{x}'|\mathbf{x})$ in the off-diagonal elements corresponds to the transition rate from configuration $\mathbf{x}$ to $\mathbf{x}'$, and the diagonal term, $r(\mathbf{x})$ denotes the rate of escape from current configuration $\mathbf{x}$, respectively. With transition rates defined at the master equation, the transition probability of each step will be $w(\mathbf{x}_i|\mathbf{x}_{i-1})e^{-(t_i-t_{i-1})r(\mathbf{x}_{i-1})}$. Therefore, the probability distribution functional of trajectory $P[\mathbf{X}]$ is given as follows[112]:

$$\begin{aligned} P[\mathbf{X}] =& e^{-(\tau-t_K)r(\mathbf{x}_K)}p(\mathbf{x}_0) \\ & \times \prod_{i=1}^{K} w(\mathbf{x}_i|\mathbf{x}_{i-1})e^{-(t_i-t_{i-1})r(\mathbf{x}_{i-1})}. \end{aligned} \tag{A.3}$$

There are two ways in measuring the length of given trajectory: the total trajectory time (or observation time) τ and the number of configuration changes during the trajectory, generally we call this *activity*, K. In a more general approach, one can consider a time-extensive physical observable O over the trajectory and its increment o. Then O will be incremented each configuration change[107, 112]:

$$O[\mathbf{X}] = \sum_{i=1}^{K} o(\mathbf{x}_{i-1}, \mathbf{x}_i). \tag{A.4}$$

The observable O surely becomes activity K when the incremental value is $o = 1-\delta_{\mathbf{x}_{i-1},\mathbf{x}}$, that is 1, when the configuration changes, otherwise 0. If the system had made its final Kth configuration jump at time t_K and the final configuration $\mathbf{x}_K$ survives until the observation time τ, the first exponential term remains. Or we can simply stop measuring the time evolution of the system when the final configuration jump happened. In this case, the first exponential term will be not be needed.

There exist similar relations between extensive properties in the thermodynamic ensemble: the particle number N and the volume V[110, 112]. In the typical experimental scenario, we measure some physical observables in fixed trajectory time τ. However, occasionally, it is much more convenient to fix the activity of trajectory K when simulate systems exhibit very slow dynamics[121].

A.3 Lattice Gas Model

We use an Ising lattice-gas in the two-dimensional square lattice as a model of self-assembly process. More than two particles cannot occupy the same lattice position, and a particle only interacts with the other particles in its nearest neighbor lattice sites. The interaction energy of the system is defined as follows:

$$H = \frac{\epsilon}{2} \sum_{p} n_p. \tag{A.5}$$

Here, ϵ denotes the strength of bonds between the particles, p is the index of the nearest neighbor, and n_p is the occupancy (0 or 1) of the site p, respectively. The model consists of $N = 2048$ particles on the two-dimensional square lattice of $V = 144 \times 144$, and the number density is $\rho \sim 0.10$, accordingly. From the theoretical perspective, the system exhibits liquid-gas phase coexistence when $\sinh^4(\epsilon/2T_\mathrm{c}) > [1-(2\rho-1)^8]^{-1}$. In the equilibrium condition below the critical temperature, the assembly yield should increase monotonically, and particles also ought to form a single large cluster. But kinetic trapping due to the lack of reversibility in bond-making and breaking processes makes it hard for the system to relax into equilibrium configurations. As a result, below a specific temperature point, the system is trapped in metastable states, which are composed of relatively small clusters, and the assembly yield starts to decrease drastically. This phase separating behavior of the Ising lattice gas is in analogy with general self-assembly processes[99].

We perform an extensive numerical simulation to obtain assemble trajectories via a stochastic Monte Carlo scheme. To achieve this, we use the classical *kinetic Monte Carlo* (kMC) method[121]. Given the current phase-space position $\mathbf{x}$ of the system, the time interval to the next jump Δt can be calculated along the probability $p_\mathbf{x}(\Delta t) \propto \exp[-r(\mathbf{x})\Delta t]$, and a transition $\mathbf{x} \to \mathbf{x}'$ is selected from all possible moves with transition rate $w(\mathbf{x}'|\mathbf{x})$. The algorithm is appropriate for sampling trajectories with fixed activity since kMC is a rejection-free process, and each Monte Carlo step corresponds to a single jump between configurations[112].

We calculate the temperature dependence of the assembly yield n_4, which denotes the fraction of particles that have exactly four occupied nearest neighbors, and the intensive trajectory time, τ/K. Since our simulation model is a typical model of the Ising lattice gas, the results shown in Fig. A.1(a) reveal archetypal non-monotonic behavior as expected from the other studies[101–103, 121]. Even if at thermodynamic equilibrium the structure in the very low temperature range should form a single, large cluster, kinetic trapping disrupts the assembling process and the system breaks up into many, relatively small clusters. Consequentially, the system shows the maximum assembly efficiency near the $T \sim 0.3$, and it drops towards to decreasing temperature. The intensive trajectory time in Fig. A.1 shows a comparable temperature dependency with assembly yield.

For more detailed examination, the time evolution of the assembly yield and the intensive trajectory time are plotted in Fig. A.1(c). The relation between two properties gives a more clear idea of trapping phenomena at local minima. Both the assembly yield (Fig. A.1(a)) and the trajectory time (Fig. A.1(b)) exhibit the local minima followed by a long plateau behavior. After enough time has passed, eventually the plateau in the assembly yield ends first; the trajectory time follows. This mechanism makes a kink behavior in n_4 as shown in the Fig. A.1(c). At the lower temperature regime exhibits kinetic trapping, the system trapped in the point near $\tau/K \sim 15$ and $n_4 \sim 8 \times 10^{-3}$, and the graph sharply shoots up when the plateau in the assembly yield disappears. This tendency gradually vanishes as the temperature increases, and the system just bypasses that trapping region and directly into assembling in the temperature range where good assemble is

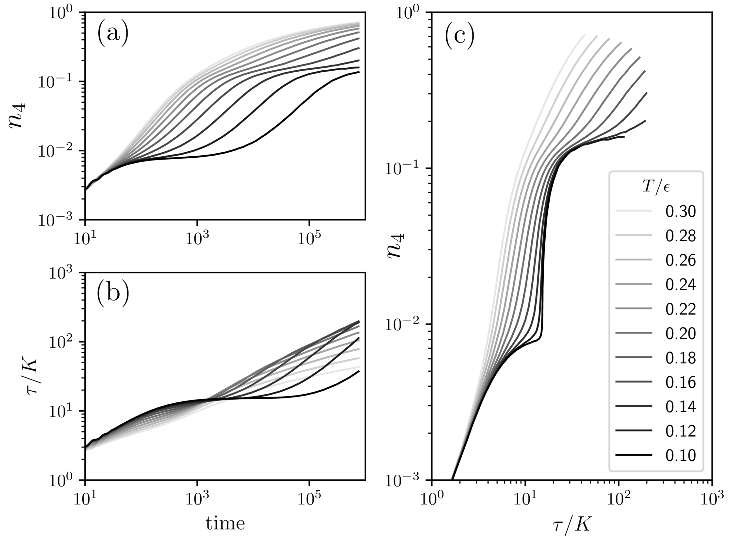


Figure A.1: Time evolution plots of (a) the assembly yield, (b) intensive trajectory length and (c) their relations (c) in the Ising lattice gas. Colors of lines represent the temperature of the system (from $T = 0.10$ to 0.30). In the temperature regime $T < 0.15$, where the kinetic trapping is strongly happens, Plateaux in the structure and the dynamics cause a kink nearby $\tau K / \sim 15$ and $n_4 \sim 8 \times 10^{-3}$.

taking place in the end.

A.4 Mathematical Model

To get more advanced insight, we propose a minimal model that exhibits kinetic trapping behavior as like as the lattice gas model. Grant and Whitelam already presented the prototype of our model to illustrate the non-monotonical growth in self-assembly processes[96, 99]. Essentially the system has three different energy levels. The *unbound* state represents nonbonded free particles and has the highest energy ($E = 0$), the *misbound* states of intermediate energy value ($E = -\epsilon_{\text{mis}}$) and the optimally *bound* state ($E = -\epsilon_{\text{opt}}$) on the ground level; it is obvious that $\epsilon_{\text{opt}} > \epsilon_{\text{mis}}$. Passing through the unbound state is necessary if the system intends to transit from metastable misbound states to the stable bound state. Additionally, there is degeneracy Ω_{mis} in the misbound state to achieve an entropic barrier.

The transition rate matrix (master operator) of the original model is described as 3×3 matrix and the degeneracy is simply multiplied by transition and escape rates of unbound and misbound states[99]. We modify the original model to accomplish the 'rattling' dynamics between degenerated misbound states. For example, the master operator of the $\Omega_{\text{mis}} = 2$ case is expressed as 4×4 matrix[112, 120]:

$$\mathbb{W} = \begin{bmatrix} -1-\gamma & 1 & 1 & 0 \\ 1 & -1-\gamma & 1 & 0 \\ \gamma & \gamma & -3 & \nu \\ 0 & 0 & 1 & -\nu \end{bmatrix}. \tag{A.6}$$

Each state can be described as a vector: misbound states($|1\rangle$, $\cdots$, $|\Omega_{\text{mis}}\rangle$), unbound state ($|\Omega_{\text{mis}} + 1\rangle$) and bound state ($|\Omega_{\text{mis}} + 2\rangle$), respectively. Based on the detailed balance condition, transition rates from misbound to unbound is $\gamma = \exp(-\epsilon_{\text{mis}}/T)$ and bound to unbound is $\nu = \exp(-\epsilon_{\text{opt}}/T)$, respectively. Rates toward to opposite directions are simply 1 by traditional Metropolis acceptance criteria. Notwithstanding our modified model has complicated dynamics more than the original one, it is obvious that the probability of the bound state, $P_{\text{bound}} = \langle \Omega_{\text{mis}} + 2|\mathbf{p}(t)\rangle$ will have exactly the same equilibrium value $\nu/(1 + \nu + \Omega_{\text{mis}}\gamma)$ when $t \to \infty$.

We perform numerical calculations for our minimal model using matrix algebra to confirm that whether or not the model successfully reproduces results from the Ising lattice gas. The time-evolution of a system can be described as $|\mathbf{p}(t)\rangle = \exp(t\mathbb{W})\,|\mathbf{p}(0)\rangle$ and mean value of certain observable O at the time t can be calculated from $\langle O(t)\rangle = \langle \mathbf{e}|\,\mathbb{O}\,|\mathbf{p}(t)\rangle$ where $|\mathbf{e}\rangle = \sum_{\mathbf{x}} |\mathbf{x}\rangle$ is the *projection* state[108, 120]. We let binding energies of misbound and bound states are $\epsilon_{\text{mis}} = \epsilon$ and $\epsilon_{\text{opt}} = 2\epsilon$, respectively. Results from numerical matrix calculations are shown in Fig. A.3. Outcomes well correspond with the results obtained from the Ising lattice gas, especially assembly yield versus intensive trajectory time graph demonstrates the same kink in the kinetic trapping regime.

A.5 Dynamical Phase Transitions

In previous sections, we demonstrated there are kink behaviors between structure (assembly yield, n_4 or P_{Bound}) and dynamics (step time, τ/K) in both numerical models during the kinetic trapping occur. Focused on this

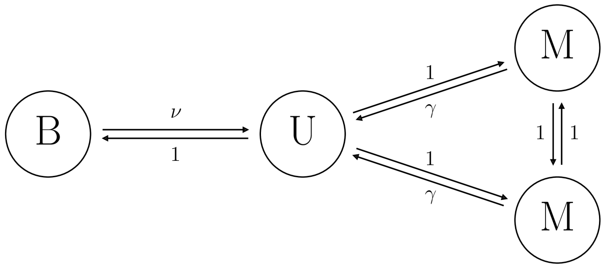


Figure A.2: A minimal three-state model of self-assembly. There are two misbound states (M), which have the same intermediate energy, can transit without any energy barrier. The transition rate from bound state (B, has the lowest energy) to unbound state (U, has the highest energy) is ν, from misbound states to unbound state is γ and rates to reverse directions are 1 due to the Metropolis criteria; jumping between misbound and bound states are impossible.

fact, we suggest the possible existence of a crossover between two different dynamical phases between in self-assembly processes. Recent advances in the dynamic ensemble theory give us a crucial insight by introducing a virtual field that biases trajectory length, which as an conjugate variable of the ensemble of trajectories[107–114].

From the definition of observation probability of a given trajectory as expressed in eqn (A.3), we can calculate the PDF of the τ in K-fixed trajectories

$$P(\tau|K) = \int D\mathbf{X}_K \, \delta(\tau - \hat{\tau}[\mathbf{X}_K])P[\mathbf{X}_K], \tag{A.7}$$

and its corresponding partition function with a conjugate field x of trajectory

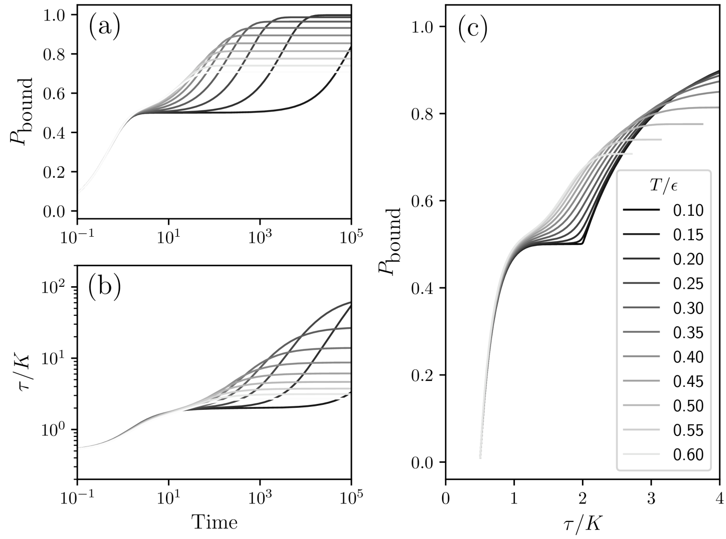


Figure A.3: Time evolution of (a) the assembly yield, (b) total trajectory time per activity (b) and their relations (c) of the three-state minimal model. The structural plateau and the dynamical plateau create a kink in the kinetic trapping regime. These results are consistent with the more realistic model.

time τ[107, 112]:

$$Z(x,K) = \int d\tau e^{-x\tau} P(\tau|K). \tag{A.8}$$

We call these ensembles as (τ, K) and (x, K) ensemble, named after their fixed variables, respectively. Non-equilibrium free energies of two cases are defined as: $\Psi(\tau, K) = \ln P(\tau|K)$ and $\Phi(x, K) = \ln Z(x, K)$. If both quantities have the large deviation limit $\Psi(\tau, K) \sim K\psi(\tau)$ and $\Phi(x, K) \sim K\phi(x)$, ψ and ϕ are convex conjugate to each other by Legendre-Fenchel transform[115]. Finally we can describe the physical meaning of x from Legendre duality:

$$\frac{\partial \Psi}{\partial \tau} \equiv x(\tau, K). \tag{A.9}$$

One can explain x as an external field that biasing trajectory time, like what the chemical potential μ and the pressure P does in traditional thermodynamic ensemble. For Markov processes, we can get the partition sum of trajectories using matrix product

$$Z(x,K) = \langle \mathbf{e} | \mathbb{T}^K(x) | \mathbf{p}(0) \rangle, \tag{A.10}$$

with off-diagonal transfer operator obtained from Laplace transform of the probability matrix of the system[109, 112]:

$$\mathbb{T}(x) = \sum_{\mathbf{x}'=\mathbf{x}} \frac{w(\mathbf{x}'|\mathbf{x})}{x + r(\mathbf{x})} \left|\mathbf{x}'\right\rangle \left\langle\mathbf{x}\right| . \tag{A.11}$$

If the system is in the thermodynamic limit, when K is large enough in other words, we can directly obtain $\phi(x)$ from the largest eigenvalue of the operator $\mathbb{T}(x)$[112, 115]. Many works analytically or numerically demonstrated that the nonequilibrium ensemble exhibits dynamical first-order phase transitions in several abstract or realistic (atomistic) systems which describing glassy dynamics[109, 111, 114]. For example, the kinetically constrained model shows criticality at $T = 0$; therefore, there is always a phase coexistence between low- (inactive) and high-activity (active) phases at any finite temperature[107, 108].

The trajectory time per kMC step plays a relevant role in the assembling process as we discussed in previous sections. Now our purpose is to control assemble dynamics of Ising lattice gas via biasing step time using the (x, K) ensemble. We use the transition path sampling (TPS) scheme[122] for sample ensembles of assembling trajectories in various T and x ranges. The dynamical free energy, $\Phi(x, K)$ is calculated from the multistate Bennet acceptance ratio (MBAR)[123, 124]. As shown in the Fig. A.4 (b), as in other model systems, our results clearly exhibit an active-inactive dynamical phase transition when the field x is applied for total lengths (or time) of trajectories.

We also calculate the same quantity for the minimal model of matrix products in Fig. A.4 (a). The results for the infinite-activity limit is obtained from numerically diagonalized eigenvalue of $\mathbb{T}(x)$:

$$\mathbb{T}(x) = \begin{bmatrix} 0 & \frac{1}{x+\gamma+1} & \frac{1}{x+3} & 0 \\ \frac{1}{x+\gamma+1} & 0 & \frac{1}{x+3} & 0 \\ \frac{\gamma}{x+\gamma+1} & \frac{\gamma}{x+\gamma+1} & 0 & \frac{\nu}{x+\nu} \\ 0 & 0 & \frac{1}{x+3} & 0 \end{bmatrix}. \tag{A.12}$$

A noteworthy feature is that first-order dynamical phase coexistences become apparent as the temperature decreases in both two models. Namely, it seems there is a finite critical temperature $T_{\mathrm{c}} > 0$ exists, and when compared with previous results, the criticality is located in the kinetic trapping regime. This phenomenon is observed both in the Ising lattice gas and the minimal model and is the distinguishable feature when compared to results from the other models: the KCM or the TLG model[107, 108, 125]. Thus, we argue that there are dynamical first-order phase transitions in self-assembly systems, and one can understand the kinetic trapping behavior as a consequence of the phase separation in the ensemble of trajectories.

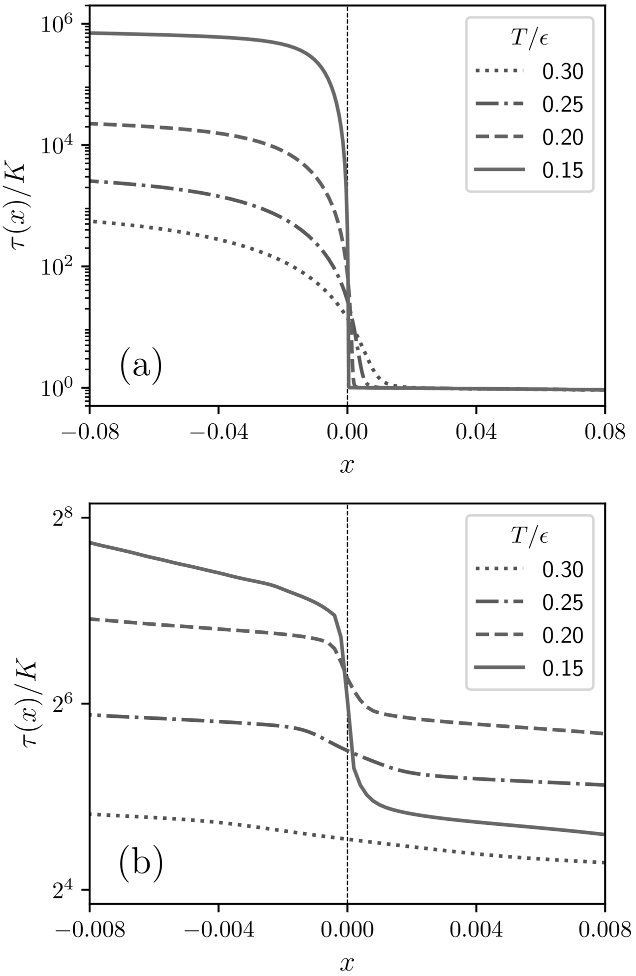


Figure A.4: (a) Plot of the intensive trajectory time τ/K of the minimal model from numerically diagonalized transfer matrix, $\mathbb{T}(x)$. The temperature range is from $T/\epsilon = 0.15$ (blue line) to 0.30 (red line). (b) The same quantity in the Ising lattice gas. Shooting TPS algorithm is applied for sampling ensemble of trajectories. Singularity at low-temperature demonstrates there is active-inactive coexistence near the $x = 0$.

A.6 Conclusion

Adopting the activity concept as a projection of the reversibility of the self-assembly process, we can easily understand the relation between structural relaxations and dynamical properties due to kinetic trapping in a self-assemble system. Using Monte Carlo simulation and numerical calculation, we discovered there are two dominant factors in trapping behavior in the local minimum. When the temperature is low enough to exhibit kinetic trapping, both structure and activity display plateau behaviors at a similar time scale during assembly progress. Then the plateau due to structural trap disappears first; escaping from the dynamical trap then follows. The minimal model that we proposed successfully reproduces the results taken from both the thermodynamic and the dynamic behavior of the relatively realistic lattice gas model.

With the dynamic ensemble of trajectories approach using large deviation formalism[109, 112], it seems that there is a a finite critical temperature that exhibits a dynamical active-inactive first-order phase transition below the temperature. In contrast, for the KCM of glass formers[107, 108], such phase transitions always appear for $T > 0$. If the dynamic critical temperature indeed exists, the kinetic trapping behavior might be described as an active-inactive crossover in assemble trajectories.

As a perspective of the self-assembly process from disordered structure to ordered equilibrium structure can be regarded as a feature of the quenched disorder, we anticipate our mathematical model would be helpful for understanding dynamical and structural properties of many other models handling quenched system; glass forming fluids for example[109, 111, 114]. Certainly, it also might be a useful topic when applying for more realistic models of self-assembly processes.

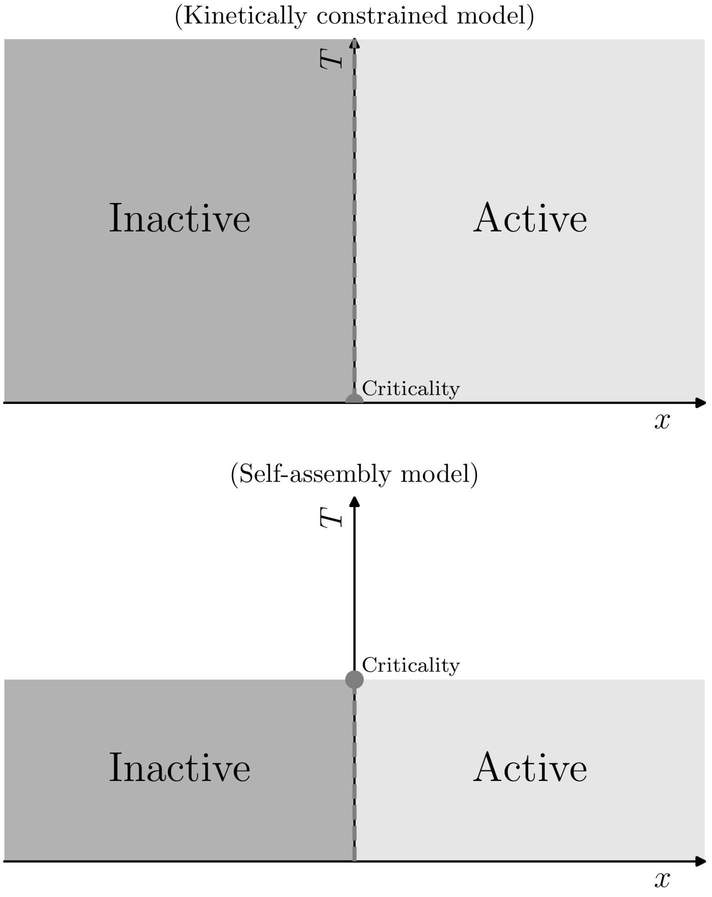


Figure A.5: Estimated dynamical phase diagram of (left) the kinetically constrained model and (right) our model of the self-assembly processes. A distinguishable feature of the our model is in comparison with the KCMs is there is a finite critical temperature $T_c > 0$ which exhibits a dynamic phase coexistence below the T_c.

Appendix B

Reaction-Path Thermodynamics of the Michaelis-Menten Kinetics

B.1 Introduction

Michaelis-Menten kinetics[126, 127] is one of the most fundamental mechanism for describing catalytic or enzymatic reactions and it presents crucial insights into the understanding of many biochemical or physical processes in living systems[128]: enzyme reactions in the living cell, DNA hybridization[68], gene regulation[129, 130], or molecular motors[131, 132]. Over a hundred years since its birth, there have been numerous theoretical and experimental advances for studying the enzymatic mechanism in various systems and methods, especially spectroscopic quantifications at the single-molecule level[133, 134]. Such a series of experimental successes in the microscopic scale promoted studies in theoretical manners[130, 135–

141]. A major topic in theoretical approaches is the timescale of enzymatic turnover[142, 143], which means time duration until a single reaction ends. Many theoretical approaches have been developed to calculate turnover time and to quantify its fluctuation behavior: from the solution of the linear differential equation[134, 142, 143] in the ideal scenario to reaction time distribution (RTD) methods in disordered systems with non-Poissonian kinetics[135, 137, 138].

$$\mathrm{E} + \mathrm{S} \underset{k_u}{\overset{k_b}{\rightleftharpoons}} \mathrm{ES} \xrightarrow{k_c} \mathrm{E} + \mathrm{P} \tag{B.1}$$

The principal idea of the Michaelis-Menten mechanism is there are two stages in the enzymatic reaction process[126, 127]: (i) the reversible binding-unbinding reactions between the substrate (S) and the enzyme (E) molecule, $\mathrm{E} + \mathrm{S} \rightleftharpoons \mathrm{ES}$ and (ii) the irreversible catalytic reaction from the bound enzyme-substrate complex (ES) to the product (P), $\mathrm{ES} \longrightarrow \mathrm{E} + \mathrm{P}$. We need to pay attention to the unbinding (disassociation) reaction at the stage (i) because the unbinding makes the process return to its initial state. Thus, essentially, the Michaelis-Menten mechanism can be interpreted as a renewal process[135, 140], and 'events' of unbinding play an essential role for the entire process. For example, chemical intuition tells that the increase of unbinding rate k_u has to result in the decrease of turnover rate, which is true at least in ideal models which exhibit Poisson kinetics. However, paradoxically, in some cases where the waiting time distribution of catalysis is not a single exponential form, slower disassociation may cause the faster turnover[139, 140]. Such nonmonotonic dependencies between unbinding and turnover suggest that we can classify enzymatic processes into two different dynamical phases, the *inhibitory* and *excitatory* unbinding.

The importance of the unbinding as we mentioned before signifies the necessity of quantifying unbinding events in enzymatic reaction processes. In the present work, we study several kinetic aspects of the Michaelis-Menten mechanism in the single molecule level in the framework of the the nonequilibrium statistical mechanics and quantify the statistical feature of unbinding events. Recent statistical mechanical studies present a notable perspective for handling systems in out-of-equilibrium. The core concept is a stochastic trajectory (or path) can be thought as a microstate in the statistical ensemble theory[113]. This idea and a mathematical formulation named the large deviation principle[115] leads to *nonequilibrium ensemble* theory. The main purpose of the theory is to draw out-of-equilibrium or dynamical properties of the system from theoretical or computer simulation methods. Furthermore, the nonequilibrium ensemble also successfully described the heterogeneous dynamical behavior in many systems, e.g., glass forming liquids[114, 117], kinetic networks[111, 144], active matters[145–147], or protein folding pathways[148] as an order-disorder symmetry breaking phenomenon between metastable states when one uses 'dynamical events' as an order parameter. Based on preceding studies, we believe the nonequilibrium ensemble theory will be a powerful tool for quantifying enzyme kinetics since most chemical reactions, including enzymatic processes, are also out-of-equilibrium processes.

This chapter is outlined as follows: In the second section, we suggest a concept of a reaction-path entropy, construct the statistical thermodynamics of enzymatic reaction paths, and calculate several major reaction timescales of the single-enzyme and single-substrate model via the large deviations principle and the nonequilibrium ensemble theory. In the third section, we quantify the number of unbinding events K when we observe the system at fixed timescale and evaluate the heterogeneous kinetics of the same model as a dynamic order-disorder in unbinding rates. In the last section, we summarize and conclude our results.

B.2 Reaction Path Thermodynamics

We use the single-molecule variant of the chemical master equation (CME) of the Michaelis-Menten equation. The stochastic equation considers finite numbers of molecules in a discrete manner, instead of their concentrations in a continuous manner and each combination of quantities corresponds to a different state of the system. Due to the law of conservation of mass, we can assume that the system contains $N = n_{\mathrm{E}} + n_{\mathrm{ES}}$ of enzyme-type molecules and $M = n_{\mathrm{S}} + n_{\mathrm{E}} + n_{\mathrm{P}}$ of ligand-type molecules[136, 142]. The master equation of the system is as follows:

$$\begin{aligned}
\dot{p}(n_{\mathrm{S}}, n_{\mathrm{ES}}, t) = &- [w_b n_{\mathrm{S}}(N - n_{\mathrm{ES}}) + w_u n_{\mathrm{ES}} + w_c n_{\mathrm{ES}}] p(n_{\mathrm{S}}, n_{\mathrm{ES}}, t) \\
&+ w_b(n_{\mathrm{S}} + 1)(N - n_{\mathrm{ES}} + 1) p(n_{\mathrm{S}} + 1, n_{\mathrm{ES}} - 1, t) \\
&+ w_u(n_{\mathrm{ES}} + 1) p(n_{\mathrm{S}} - 1, n_{\mathrm{ES}} + 1, t) \\
&+ w_c(n_{\mathrm{ES}} + 1) p(n_{\mathrm{S}}, n_{\mathrm{ES}} + 1, t).
\end{aligned} \tag{B.2}$$

Here, $w_b = k_b/V_u^2$, $w_u = k_u/V_u$, and $w_c = k_c/V_u$ are the reaction rate constants per unit volume V_u and subscripts b, u, and c denote the *binding*, the *unbinding*, and the *catalysis* event, respectively. Since the model considers a discrete number of components, we use probabilities of states $p(n_{\mathrm{S}}, n_{\mathrm{ES}}, t)$, instead of continuous concentrations. If the system contains only one enzyme and substrate molecules, $N = 1$ and $M = 1$ in other words, the equation B.2 can be reduced to the following form:

$$\dot{p}_{\mathrm{S}}(t) = w_u p_{\mathrm{ES}}(t) - w_b p_{\mathrm{S}}(t), \tag{B.3a}$$

$$\dot{p}_{\mathrm{ES}}(t) = w_b p_{\mathrm{S}}(t) - (w_u + w_c) p_{\mathrm{ES}}(t), \tag{B.3b}$$

$$\dot{p}_{\mathrm{P}}(t) = w_c p_{\mathrm{ES}}(t). \tag{B.3c}$$

We omit the time evolution of the probability of enzyme E since it has the relation with ES, $p_{\mathrm{E}}(t) = 1 - p_{\mathrm{ES}}(t)$. If one considers a single reaction path

$\mathrm{E+S} \to \cdots \to \mathrm{E+P}$ of the equation B.3 which has K unbinding events, then one can find the given path with the probability $\rho[\{\mathrm{path}\}]$[113, 119]:

$$\rho[\{\mathrm{path}\}] = w_b e^{-w_b \Delta t_0} \left(\prod_{i=1}^{K} w_u e^{-(w_u + w_c)\Delta t'_i} w_b e^{-w_b \Delta t_i} \right) \times w_c e^{-(w_u+w_c)\Delta t'_0}. \tag{B.4}$$

Here, time intervals Δt_i and $\Delta t'_i$ denote lifetimes of S and ES at individual reaction stage, respectively. If we define the 'total' lifetime of each component as the sum of individual lifetimes, $\sum_{i=0}^{K} \Delta t_i = t_{\mathrm{S}}$ and $\sum_{i=0}^{K} \Delta t'_i = t_{\mathrm{ES}}$, then we can simplify the equation B.4 to

$$\rho[\{\mathrm{path}\}] = w_{\mathrm{b}} w_{\mathrm{c}} (w_{\mathrm{u}} w_{\mathrm{b}})^K e^{-w_{\mathrm{b}} t_{\mathrm{S}}} e^{-(w_{\mathrm{b}}+w_{\mathrm{u}}) t_{ES}}, \tag{B.5}$$

which only depends on three nonequilibrium observables: the number of unbinding events (K), the total lifetime of the substrate molecule (t_{S}) and the enzyme-substrate complex (t_{ES}), respectively. That is to say; we can find a single reaction path with identical probability if three observables K, t_{S}, and t_{ES} are conserved. Hence, similar to N, V, and E in the canonical equilibrium ensemble case, the principle of equal a *priori* probabilities is valid, and it leads to the definition of the *nonequilibrium* microcanonical ensemble, described by the following path-dependent reaction entropy.

$$\mathcal{S} \equiv - \sum_{\{\text{path}\}} \rho[\{\text{path}\}] \ln \rho[\{\text{path}\}] = - \ln \rho[\{\text{path}\}] \tag{B.6}$$

The microscopic number of all possible reaction paths (similar to *microstates* in equilibrium statistical mechanics) $\Omega = 1/\rho$ depends on combinations of Δt_i and $\Delta t_i'$[149]:

$$\begin{aligned} \Omega(K, t_{\text{ES}}, t_{\text{S}}) &= \int_{\sum \Delta t_i = t_{\text{S}}} d\Delta t^{K+1} \int_{\sum \Delta t_i' = t_{\text{ES}}} d\Delta t'^{K+1} \\ &= \frac{K+1}{K!K!} t_{\text{ES}}^K t_{\text{S}}^K . \end{aligned} \tag{B.7}$$

In the equation B.7, each integral denotes the area of the $(K+1)$-dimension hyper-sphere. Accordingly, we can evaluate the entropy of reaction paths in the $(K, t_{\text{ES}}, t_{\text{S}})$-fixed ensemble, $\mathcal{S}(K, t_{\text{ES}}, t_{\text{S}}) = \ln \Omega(K, t_{\text{ES}}, t_{\text{S}})$. Now quantifying the MM kinetics with the language of statistical thermodynamics is feasible by cause of the definition of the reaction path entropy and the large deviations principle[115]. The Gärtner-Ellis theorem presents partition functions of the following nonequilibrium canonical (K, t_{ES}, μ) and grand canonical (K, ν, μ) ensembles

$$\mathcal{Z}(K, t_{\text{ES}}, \mu) = \int_0^\infty dt_{\text{S}} e^{-\mu t_{\text{S}}} \Omega(K, t_{\text{ES}}, t_{\text{S}}), \tag{B.8a}$$

$$\mathcal{Q}(K, \nu, \mu) = \int_0^\infty dt_{\text{ES}} e^{-\nu t_{\text{ES}}} \mathcal{Z}(K, t_{\text{ES}}, \mu), \tag{B.8b}$$

and their free energies spontaneously with certain conjugate fields μ and ν, which biases t_{S} and t_{ES}, respectively,

$$\mathcal{F}(K, t_{\mathrm{ES}}, \mu) = K \ln \mu - K \ln t_{\mathrm{ES}} + K \ln K - K, \tag{B.9a}$$

$$\mathcal{G}(K, \nu, \mu) = K \ln \nu + K \ln \mu. \tag{B.9b}$$

Equations B.5 and B.8 suggest that $\mu = w_b$ and $\nu = w_u + w_c$, which in fact means that escaping rates and lifetimes are mutually conjugate variables. Therefore, the fundamental relations of the nonequilibrium thermodynamics, $\mathcal{F} = \mu t_{\mathrm{S}} - \mathcal{S}$ and $\mathcal{G} = \nu t_{\mathrm{ES}} - \mathcal{F}$ are valid. From equations B.7 and B.8, conditional probability distributions of t_{S} and t_{ES} in the K-fixed ensemble are Poissonian as follows:

$$\rho(t_{\mathrm{S}}|K) = \frac{\mu^{K+1} t_{\mathrm{S}}^K}{K!} e^{-\mu t_{\mathrm{S}}}, \tag{B.10a}$$

$$\rho(t_{\mathrm{ES}}|K) = \frac{\nu^{K+1} t_{\mathrm{ES}}^K}{K!} e^{-\nu t_{\mathrm{ES}}}. \tag{B.10b}$$

Note that the two lifetimes t_{S} and t_{ES} are mutually independent. Since the enzymatic turnover time, t_{t}, is the sum of t_{S} and t_{ES}, its conditional probability distribution $\rho(t_{\mathrm{t}}|K)$ takes a convolution form of $\rho(t_{\mathrm{S}}|K)$ and $\rho(t_{\mathrm{ES}}|K)$. The convolution is quite complicated for calculation due to t^K

term, but it can be easily obtained in the Laplace domain:

$$\rho(x_{\mathrm{t}}|K) = \left[\frac{\mu\nu}{(\mu + x_{\mathrm{t}})(\nu + x_{\mathrm{t}})}\right]^{K+1}. \tag{B.11}$$

With Bayes' theorem and considerations of the marginal probability of unbinding events is products of transition probabilities $\rho(K) = (w_c w_u^K)/(w_u + w_c)^{K+1}$ by its definition[113, 119], Eqns. B.10 and B.11 finally give marginal probability distributions of liftimes of S, ES, and turnover time

$$\rho(t_{\mathrm{S}}) = (w_b w_c/(w_u + w_c)) \exp(-w_b w_c t_{\mathrm{S}}/(w_u + w_c)), \tag{B.12a}$$

$$\rho(t_{\mathrm{ES}}) = w_c \exp(-w_c t_{\mathrm{ES}}), \tag{B.12b}$$

$$\rho(t_{\mathrm{t}}) = \alpha\beta(e^{-\alpha t_{\mathrm{t}}} - e^{-\beta t_{\mathrm{t}}})/(\beta - \alpha), \tag{B.12c}$$

where two constants α and β in the turnover time distribution are:

$$\alpha = \frac{\lambda + \sqrt{\lambda^2 - 4w_b w_c}}{2}, \tag{B.13a}$$

$$\beta = \frac{\lambda - \sqrt{\lambda^2 - 4w_b w_c}}{2}. \tag{B.13b}$$

In the avobe equation, $\lambda = w_b + w_u + w_c$. The probability distribution of turnover time we obtained in the equation B.12 is identical with results from the solution of linear differential equations[134, 142, 143]. We finally obtain nonequilibrium ensemble average of the total lifetimes of S, ES, and the turnover time:

$$\langle t_{\mathrm{S}} \rangle = \frac{w_u + w_c}{w_b w_c}, \tag{B.14a}$$

$$\langle t_{\mathrm{ES}} \rangle = \frac{1}{w_c}, \tag{B.14b}$$

$$\langle t_{\mathrm{t}} \rangle = \frac{w_b + w_u + w_c}{w_b w_c}. \tag{B.14c}$$

B.3 Fixed Observation Time

In a certain theoretical or experimental scenario, it might be more convenient to sample reaction paths with arbitrary *observation time* τ[150, 151], instead of the fixed number of enzyme-substrate unbinding events K. Since the kinetics of the system is governed by the master equation B.3, the time evolution of the system can be described as $|p(\tau)\rangle = \mathbb{U}(\tau)\,|p(0)\rangle$ with the propagator $\mathbb{U}(\tau) = \exp(\tau\mathbb{W})$. As we fix the observation time, we have to consider not only '*completed*' reaction paths but also sample '*incompleted*' reaction paths which remain in $|\mathrm{S}\rangle$ or $|\mathrm{ES}\rangle$ at the observation time τ. Because the propagator can be decomposed into the operators of conditional probabilities of unbinding events K as $\mathbb{U}(\tau) = \sum_K \mathbb{P}(K|\tau)$, the conditional probability of K at τ is $P(K|\tau) = \langle \mathrm{e}|\,\mathbb{P}(K|\tau)\,|\mathrm{S}\rangle$ where $|\mathrm{e}\rangle = |\mathrm{S}\rangle + |\mathrm{ES}\rangle + |\mathrm{P}\rangle$ is the *projection state*. For *completed* reaction paths ($\mathrm{E} + \mathrm{S} \to \cdots \to \mathrm{E} + \mathrm{P}$) where the final state is $|\mathrm{P}\rangle$, the conditional probability of K

at fixed τ is

$$\langle \mathrm{P}| \, \mathbb{P}(K|\tau) \, |\mathrm{S}\rangle = \int_0^\tau dt_\mathrm{t} \rho(t_\mathrm{t}, K), \tag{B.15}$$

where $\rho(t_\mathrm{t}, K) = \rho(t_\mathrm{t}|K)\rho(K)$ is the joint probability distribution of t_t and K because $\langle \mathrm{P}| \, \mathbb{P}(K|\tau) \, |\mathrm{S}\rangle$ contains all the possible reaction paths that have K unbinding events and turnover times smaller than τ.

For *incompleted* reaction paths where observation states are $|\mathrm{E}\rangle$ or $|\mathrm{ES}\rangle$, we must consider the value of K at time τ, not t_t due to the reaction is not terminated yet at the observation time. It means $\tau = t_\mathrm{S} + t_\mathrm{ES} < t_\mathrm{t}$ and we have to consider the reaction path entropies of both cases, $\mathrm{E} + \mathrm{S} \to \cdots \to \mathrm{E} + \mathrm{S}$ and $\mathrm{E} + \mathrm{S} \to \cdots \to \mathrm{ES}$. First, we calculate Ω_S, which describes the microscopic number of paths which end at $|\mathrm{S}\rangle$ and $(K, t_\mathrm{S}, t_\mathrm{ES})$:

$$\Omega_\mathrm{S}(K, t_\mathrm{S}, t_\mathrm{ES}) = \frac{\sqrt{K}}{(K-1)!} \frac{\sqrt{K+1}}{K!} t_\mathrm{S}^{K-1} t_\mathrm{ES}^{K}. \tag{B.16}$$

We also have to consider reaction paths which end at $|\mathrm{ES}\rangle$:

$$\Omega_\mathrm{ES}(K, t_\mathrm{S}, t_\mathrm{ES}) = \frac{K+1}{K!K!} t_\mathrm{S}^{K} t_\mathrm{ES}^{K}. \tag{B.17}$$

Since Ω_ES is identical to Ω and Ω_S also has a similar form with Ω, we suppose that the Bayesian probability of (τ, K) for incompleted paths and (t_t, K) for completed paths have a nearly same analytical shape when K is large enough. Therefore, we can approximate $|\mathrm{S}\rangle$- and $|\mathrm{ES}\rangle$-contributions of the $P(K|\tau)$:

$$\langle \mathrm{S}|\, \mathbb{P}(K|\tau)\, |\mathrm{S}\rangle + \langle \mathrm{ES}|\, \mathbb{P}(K|\tau)\, |\mathrm{S}\rangle \simeq \frac{\rho(t_{\mathrm{t}} = \tau, K)}{\rho(t_{\mathrm{t}} = \tau)} \int_{\tau}^{\infty} dt_{\mathrm{t}} \rho(t_{\mathrm{t}}). \quad \text{(B.18)}$$

Here, the overall shape of the probability distribution comes from $\rho(t_{\mathrm{t}} = \tau, K)$ and $\rho(t_{\mathrm{t}} > \tau)/\rho(t_{\mathrm{t}} = \tau)$ is a normalization factor. Equations B.15 and B.18 present an approximate form of the conditional probability of the number of unbinding events at fixed observation time:

$$P(K|\tau) \simeq \rho(t_{\mathrm{t}} < \tau, K) + \rho(t_{\mathrm{t}} = \tau, K) \frac{\rho(t_{\mathrm{t}} < \tau)}{\rho(t_{\mathrm{t}} = \tau)}. \quad \text{(B.19)}$$

We plot equation B.15, B.18, and B.19 for $w_b = 0.5$, $w_u = 1.0$, and $w_c = 0.025$ case in the Figure B.1-(a). The equation B.15 has the maximum value at $K = 0$ and shows almost the same decay behavior with $\rho(K)$ in the early stage; it drastically decreases where K is near the peak of Eqn. B.18. This tendency results in a bimodal shape in their sum. The bimodal behavior of $P(K|\tau)$ signifies that we can divide the probability distribution into two different paths[115]: the unbinding-rich one and the unbinding-poor one. Recent studies showed that there exist more than two dynamical phases in systems which exhibit heterogeneous or glassy dynamics[106–108, 111, 112, 114, 117, 137, 144–148]. In the same way, the Michaelis-Menten mechanism shows heterogeneous kinetics in its unbinding events and results in the inactive-phase of 'reaction-completed' paths and active-phase of 'reaction-incompleted' paths.

Again, we use the formalism of the large deviation principle to evaluate the moment-generating function of K with corresponding virtual, conjugate variable s[106, 112, 115]:

$$Z(s,\tau) = \sum_{K=0}^{\infty} e^{-sK} P(K|\tau). \tag{B.20}$$

The n-th derivative of $Z(s,\tau)$ gives the n-th moment of unbinding events at fixed observation time τ, $\langle K^n \rangle_\tau = (-1)^n Z^{-1} \partial_s Z(s,\tau)$. One can also calculate the cumulants from the cumulant generating function (or intensive free energy), $\phi(s,\tau) = \ln Z(s,\tau)/\tau$. The dynamic susceptibility, $\chi_k(s,\tau)$ is the second derivative of $\phi(s,\tau)$ and denotes the amount of fluctuations of unbinding rates per observation time, $k = K/\tau$. In the Fig. B.2-(a), we plot the observation time dependence of $\chi_k(s,\tau)$. The dynamic susceptibility has its maximum value at the point $s = s^*$, which separates the reaction paths into two different dynamical phases, the active ($s < s^*$) one and the inactive ($s > s^*$) one. We must note that the conjugate variable s is virtual and it is barely known about its real physical meaning. The only thing we know for sure is that we have to regard as s is zero for when one samples the system's reaction paths in ordinary conditions. Therefore, now what we have to do is finding the phase-coexistence timescale τ^* where the $s^*(\tau)$

becomes zero.

We need to know the general analytical behavior of $s^*(\tau)$ before obtaining τ^*. As shown in Fig. B.2-(b), $s^*(\tau)$ shows a power-law-like decay over observation time and in the large deviation limit $\tau \gg 1$, the value of s^* converges to a particular value, s_{c}. We take a different mathematical approach in order to evaluate s_{c}; one can obtain identical results with equation B.19 from algebraic calculations[106, 107, 109, 112]. First, we start from the definition of the master operator $\mathbb{W}$:

$$\mathbb{W} = \begin{bmatrix} -w_b & w_u & 0 \\ w_b & -(w_u + w_c) & 0 \\ 0 & w_c & 0 \end{bmatrix}. \tag{B.21}$$

What we have to do is to decompose the master operator into two matrices, $\mathbb{W} = \mathbb{W}_{\mathrm{m}} + \mathbb{W}_{\mathrm{r}}$. Here, $\mathbb{W}_{\mathrm{m}}$ is the operator of *monitored* reactions and the other operator, $\mathbb{W}_{\mathrm{r}}$ denotes the rest of transitions. Since we count the number of unbinding reactions, we let $\mathbb{W}_{\mathrm{m}} \equiv w_u \left|1\right\rangle \left\langle 2\right|$. The propagator $\mathbb{U}(\tau) = \exp(\tau\mathbb{W})$ is an exponential form of the master operator so we can decompose it as

$$\mathbb{P}(K|\tau) = \sum_{n=0}^{\infty} \frac{\tau^{K+n}}{(K+n)!}\mathbb{O}(K,n), \tag{B.22}$$

where $\mathbb{O}(K,n)$ is K-th order term of $\mathbb{W}_{\mathrm{m}}$ from polynomial $(\mathbb{W}_{\mathrm{m}}+\mathbb{W}_{\mathrm{r}})^{K+n}$

and can be calculated from the recurrence formula, $\mathbb{O}(K, n) = \mathbb{W}_{\mathrm{m}}\mathbb{O}(K-1, n) + \mathbb{W}_{\mathrm{r}}\mathbb{O}(K, n-1)$. We plot Eqns. B.19 and B.22 for cutoff $n_{\max} = 4096$ in Fig. B.1-(b) in order to compare their precision. As we approximate $\Omega_{\mathrm{S}} \simeq \Omega$, we believe Eqn. B.22 shows more accurate results; the $|\mathrm{S}\rangle$-contribution in Eqn. B.19 causes a minor error in the active phase due to approximated Ω_{S}.

The moment generating function $Z(s, \tau)$ and cumulant generating function $\phi(s, \tau)$ can be calculated from matrix product states:

$$Z(s, \tau) = \langle \mathrm{e} | \exp(\tau e^{-s}\mathbb{W}_{\mathrm{m}} + \tau \mathbb{W}_{\mathrm{r}}) | \mathrm{S} \rangle \, . \tag{B.23}$$

In the 'thermodynamic' limit where τ is long enough, the largest eigenvalue of the matrix $\mathbb{W}_s = e^{-s}\mathbb{W}_{\mathrm{m}} + \mathbb{W}_{\mathrm{r}}$ gives the large deviation function of $P(K|\tau)$, $\phi(s) = \lim_{\tau \to \infty} \phi(s, \tau)$. As the system has two different dynamical phases, $\phi(s)$ shows a singularity at s_{c}

$$\phi(s) = \begin{cases} 0 & s > s_{\mathrm{c}} \\ \frac{-\lambda + \sqrt{\lambda^2 - 4\gamma(s)}}{2} & s \leq s_{\mathrm{c}} \end{cases} \tag{B.24}$$

where $\gamma(s) = w_b w_u + w_b w_c - w_b w_u e^{-s}$. The second part of equation B.24 is smaller than zero when s is greater than s_{c}, which makes s_{c} to the boundary between active and inactive phases. The value of $s^*(\tau)$, as we treated before, always converges to the negative value $s_{\mathrm{c}} = -\ln(1 + w_c/w_u)$ from $\gamma(s = s_{\mathrm{c}}) = 0$.

$\phi(s, \tau)$ and $s^*(\tau)$ for finite τ are much more complicated. In fact, Eqn. B.23 can be evaluated from an analytical manner, however, the resulting expression is extremely abstruse for handling. Instead, we perform numerical calculations, and also we consider both τ^* and t_{t} are functions of three rate constants: w_b, w_u, and w_c. Then, the chain rule gives a relation between τ^* and $\langle t_{\mathrm{t}} \rangle$:

$$\begin{aligned} \frac{d\tau^*}{d\left\langle t_{\mathrm{t}}\right\rangle} &= \frac{\partial\tau^*}{\partial w_b}\frac{\partial w_b}{\partial\left\langle t_{\mathrm{t}}\right\rangle} + \frac{\partial\tau^*}{\partial w_u}\frac{\partial w_u}{\partial\left\langle t_{\mathrm{t}}\right\rangle} + \frac{\partial\tau^*}{\partial w_c}\frac{\partial w_c}{\partial\left\langle t_{\mathrm{t}}\right\rangle} \\ &= -w_b w_c\left(\frac{w_b}{w_u + w_c}\frac{\partial\tau^*}{\partial w_b} - \frac{\partial\tau^*}{\partial w_u} + \frac{w_c}{w_b + w_u}\frac{\partial\tau^*}{\partial w_c}\right) \end{aligned} \qquad \text{(B.25)}$$

We plot mean values of turnover times, numerically calculate transition times at various binding, unbinding and catalysis rates in the Fig. B.3. We find that there is strong linear correlations between τ^* and $\langle t_{\mathrm{t}} \rangle$. Each data set represents the case where two of the three rate constants are fixed, and the remainder one varies; the linear relation, $d\tau^*/d\left\langle t_{\mathrm{t}}\right\rangle \sim 1.3$ becomes apparent when $w_u \gg w_c$. Since we let the catalysis stage is irreversible, once a single reaction is over, the number of unbinding events of the given path does not increase any more. It results in the population of the inactive phase is continually increasing as observation time increases and for the active phase, *vice versa*. In the thermodynamic limit, when the time is passed long enough in other words, only inactive paths are survived and $P(K|\tau)$ converges to $\rho(K)$, which is we presented in the previous section. So we can also calculate the value of s_{c} in the large deviation limit from the convergence of Eqn. B.20, $\sum_{K=0}^{\infty} e^{-sK}\rho(K)$. Such preference for the inactive phase in a long observation time scale of the system would causes active-inactive phase transition at τ^* if the reaction process had started from the active phase at short observation time scale. Understandably, the logic can be different depending on the relative rate constants; the phase transition will not be happening if the rate of catalysis, w_c is sufficiently greater than the rate of unbinding, w_u. In that scenario, $s^*(\tau)$ always has negative value even at the very short observation time τ, and the system stays in the inactive phase from beginning till the end of reactions. This principle provides a lower boundary in Fig. (reffig:timescale.

B.4 Conclusions

In the present study, we demonstrate that a series of mathematical formalisms of the statistical thermodynamics in equilibrium systems are also suitable for treating systems in out-of-equilibrium, especially single-molecule enzymatic reactions under the Poissonian Michaelis-Menten mechanism. Three physical observables in nonequilibrium manner -the number of unbinding events, total lifetimes of substrate and enzyme-substrate complex- lead us

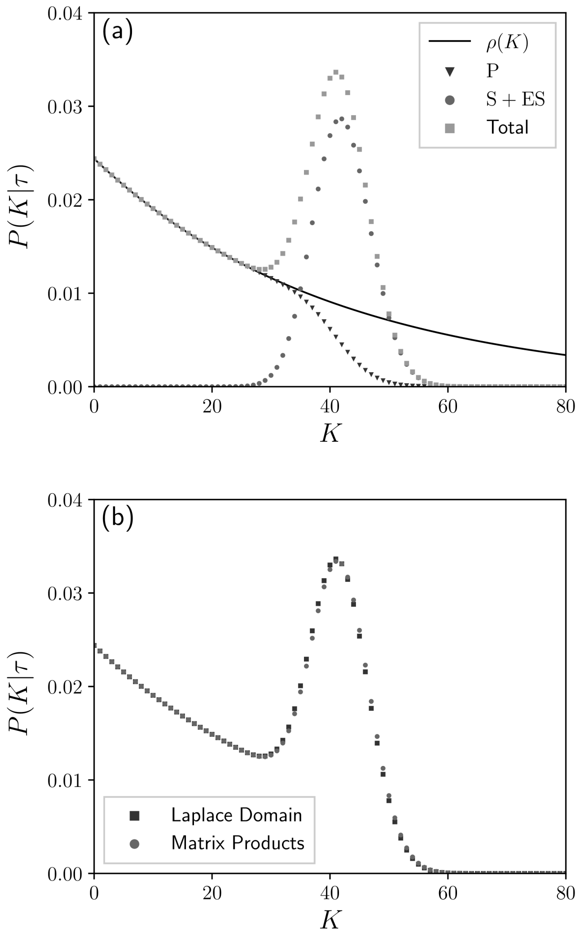


Figure B.1: (a) Conditional probability distribution $P(K|\tau)$, calculated using equation B.19 and inverse Laplace transform. The data obtained under the condition $w_b = 0.5$, $w_u = 1.0$, $w_c = 0.025$, and $\tau = 128$. Red triangles of *completed* paths are maldistributed in inactive state at maximum $K = 0$, while blue squares of *incompleted* paths make active state at maximum $K \simeq 40$. (b) Comparision plot of the equation B.19 (square) and B.22 (circle). The approximation applied for evaluating Ω_{S} makes subtle deviation in active phase.

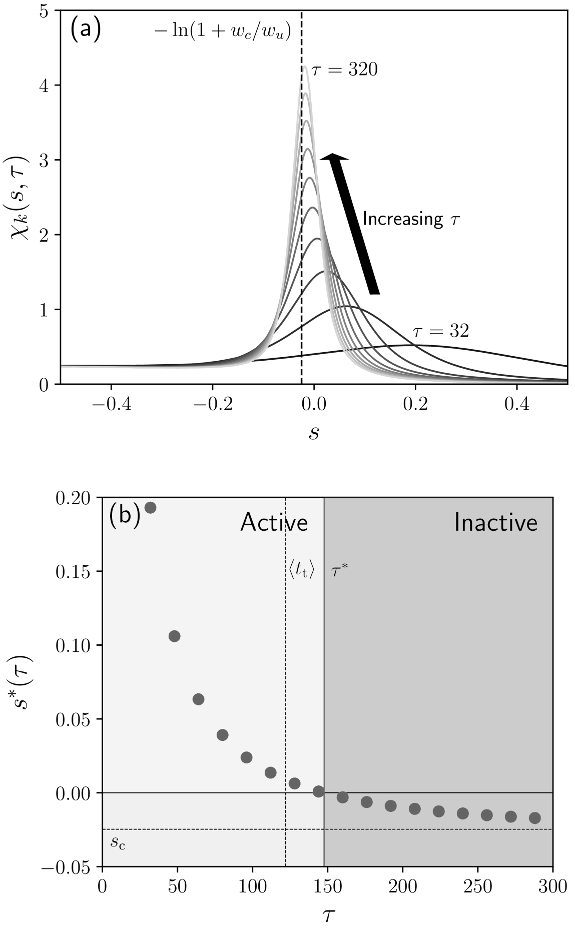


Figure B.2: (a) Susceptibilities of the intensive number of unbinding events, $k = K/\tau$ in various observation time scale and (b) their maximum position $s^*(\tau)$ in variation of the observation time. The dataset is from the condition $w_b = 0.5$, $w_u = 1.0$, and $w_c = 0.025$ s^* converges to negative value, $s_\mathrm{c} = -\ln(1 + w_c/w_u)$ in the thermodynamic limit, while it becomes zero at $\tau \simeq 147$ which exhibits coexistence active paths and inactive paths.

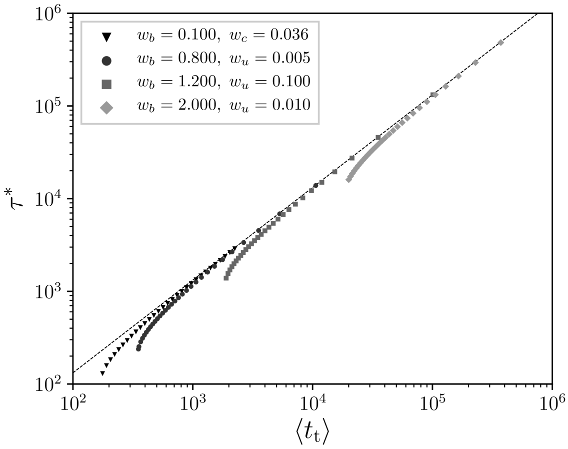


Figure B.3: Relation between mean-turnover times, $\langle t_{\mathrm{t}} \rangle$ and active-inactive phase transition times,τ^*. Two of three reaction constants are fixed while the remainder one is variating. The black dashed line clarifies that all datasets represent linearly correlated tendency, approximately $d\tau^*/d\,\langle t_{\mathrm{t}} \rangle \simeq 1.32$ in the large turnover time scale.

to the principle of a *priori* probabilities and the definition of the reaction path entropy. Based on this idea, we successfully evaluated three statistical ensembles of the out-of-equilibrium process: microcanonical $(K, t_{\mathrm{ES}}, t_{\mathrm{S}})$, canonical $(K, t_{\mathrm{ES}}, \mu)$ and grand canonical (K, ν, μ) ensemble. Conjugate intensive variables in these ensembles, μ and ν bias statistical weights of trajectories, with the lifetimes of components t_{S} and t_{ES}, respectively, and one can uncover from the definition of a single reaction path that ν and μ are just escaping ratios of the Markov process. Thermodynamic relations between nonequilibrium ensembles give us probability distributions of several important reaction time scales. Results obtained from the reaction path thermodynamics reproduces previous results based on mean-field theory.

Furthermore, for the considerations of the various theoretical or experimental scenarios, we extended our results for fixed observation time, τ. We evaluate Bayesian statistics and perform numerical calculations in order to demonstrate that the enzymatic reaction has two different dynamical phases, in fact, if one uses the number of unbinding events per the observation time, $k = K/\tau$ as an order parameter. We name these two phases as the inactive (unbinding-poor) phase and the active (unbinding-rich) phase, respectively. Because the system always takes inactive phases when observation time is long enough (in the thermodynamic limit), a first-order phase transition from the active to the inactive phase may appear during the reaction process, depending on the combination of reaction rate constants. The transition time τ^*, which is the timescale that such phase transition appears, show an approximately linear relation with the average value of enzymatic turnover time, $\langle t_\mathrm{t} \rangle$.

Since there are various evidences that the unbinding of enzyme-substrate doing a crucial role in the kinetics of complex enzymatic processes, we believe our work proposes a potential way for quantifying dynamical behaviors of systems under the MM mechanism. We will extend our study to general models, especially non-Poisson (or heterogeneous) enzymatic reaction process of the enzymatic reaction process. Also, we expect that our work on the nonequilibrium ensemble theory can be applied to various systems in out-of-equilibrium.

Bibliography

[1] Marenich, A. V.; Cramer, C. J.; Truhlar, D. G. Generalized Born Solvation Model SM12. *Journal of Chemical Theory and Computation* **2013**, *9*, 609–620.

[2] Klamt, A.; Diedenhofen, M. Calculation of Solvation Free Energies with DCOSMO-RS. *The Journal of Physical Chemistry A* **2015**, *119*, 5439–5445.

[3] Klamt, A.; Schüürmann, G. COSMO: a new approach to dielectric screening in solvents with explicit expressions for the screening energy and its gradient. *J. Chem. Soc., Perkin Trans. 2* **1993**, 799–805.

[4] Delaney, J. S. ESOL: Estimating Aqueous Solubility Directly from Molecular Structure. *Journal of Chemical Information and Computer Sciences* **2004**, *44*, 1000–1005.

[5] Tomasi, J.; Mennucci, B.; Cammi, R. Quantum Mechanical Continuum Solvation Models. *Chemical Reviews* **2005**, *105*, 2999–3094.

[6] Cramer, C. J.; Truhlar, D. G. A Universal Approach to Solvation Modeling. *Accounts of Chemical Research* **2008**, *41*, 760–768.

[7] Marenich, A. V.; Cramer, C. J.; Truhlar, D. G. Universal Solvation Model Based on Solute Electron Density and on a Continuum Model of the Solvent Defined by the Bulk Dielectric Constant and Atomic Surface Tensions. *The Journal of Physical Chemistry B* **2009**, *113*, 6378–6396.

[8] Klamt, A.; Eckert, F.; Arlt, W. COSMO-RS: An Alternative to Simulation for Calculating Thermodynamic Properties of Liquid Mixtures. *Annual Review of Chemical and Biomolecular Engineering* **2010**, *1*, 101–122.

[9] Shivakumar, D.; Williams, J.; Wu, Y.; Damm, W.; Shelley, J.; Sherman, W. Prediction of Absolute Solvation Free Energies using Molecular Dynamics Free Energy Perturbation and the OPLS Force Field. *Journal of Chemical Theory and Computation* **2010**, *6*, 1509–1519.

[10] Chong, S.-H.; Ham, S. Atomic decomposition of the protein solvation free energy and its application to amyloid-beta protein in water. *The Journal of Chemical Physics* **2011**, *135*, 034506.

[11] Mennucci, B. Polarizable continuum model: Polarizable continuum

model. *Wiley Interdisciplinary Reviews: Computational Molecular Science* **2012**, *2*, 386–404.

[12] Sato, H. A modern solvation theory: quantum chemistry and statistical chemistry. *Physical Chemistry Chemical Physics* **2013**, *15*, 7450.

[13] König, G.; Pickard, F. C.; Mei, Y.; Brooks, B. R. Predicting hydration free energies with a hybrid QM/MM approach: an evaluation of implicit and explicit solvation models in SAMPL4. *Journal of Computer-Aided Molecular Design* **2014**, *28*, 245–257.

[14] Mobley, D. L.; Guthrie, J. P. FreeSolv: a database of experimental and calculated hydration free energies, with input files. *Journal of Computer-Aided Molecular Design* **2014**, *28*, 711–720.

[15] Skyner, R. E.; McDonagh, J. L.; Groom, C. R.; van Mourik, T.; Mitchell, J. B. O. A review of methods for the calculation of solution free energies and the modelling of systems in solution. *Physical Chemistry Chemical Physics* **2015**, *17*, 6174–6191.

[16] Zhang, J.; Tuguldur, B.; van der Spoel, D. Force Field Benchmark of Organic Liquids. 2. Gibbs Energy of Solvation. *Journal of Chemical Information and Modeling* **2015**, *55*, 1192–1201.

[17] Harder, E. et al. OPLS3: A Force Field Providing Broad Coverage of

Drug-like Small Molecules and Proteins. *Journal of Chemical Theory and Computation* **2016**, *12*, 281–296.

[18] Coley, C. W.; Barzilay, R.; Green, W. H.; Jaakkola, T. S.; Jensen, K. F. Convolutional Embedding of Attributed Molecular Graphs for Physical Property Prediction. *Journal of Chemical Information and Modeling* **2017**, *57*, 1757–1772.

[19] Duarte Ramos Matos, G.; Kyu, D. Y.; Loeffler, H. H.; Chodera, J. D.; Shirts, M. R.; Mobley, D. L. Approaches for Calculating Solvation Free Energies and Enthalpies Demonstrated with an Update of the FreeSolv Database. *Journal of Chemical & Engineering Data* **2017**, *62*, 1559–1569.

[20] Wu, Z.; Ramsundar, B.; Feinberg, E.; Gomes, J.; Geniesse, C.; Pappu, A. S.; Leswing, K.; Pande, V. MoleculeNet: a benchmark for molecular machine learning. *Chemical Science* **2018**, *9*, 513–530.

[21] Borhani, T. N.; García-Muñoz, S.; Vanesa Luciani, C.; Galindo, A.; Adjiman, C. S. Hybrid QSPR models for the prediction of the free energy of solvation of organic solute/solvent pairs. *Physical Chemistry Chemical Physics* **2019**, *21*, 13706–13720.

[22] Lim, H.; Jung, Y. Delfos: deep learning model for prediction of sol-

vation free energies in generic organic solvents. *Chemical Science* **2019**, *10*, 8306–8315.

[23] Popova, M.; Isayev, O.; Tropsha, A. Deep reinforcement learning for de novo drug design. *Science Advances* **2018**, *4*, eaap7885.

[24] Reichardt, C.; Welton, T. *Solvents and Solvent Effects in Organic Chemistry: REICHARDT:SOLV.EFF. 4ED O-BK*; Wiley-VCH Verlag GmbH & Co. KGaA: Weinheim, Germany, 2010.

[25] Takeda, T.; Taniki, R.; Masuda, A.; Honma, I.; Akutagawa, T. Electron-deficient anthraquinone derivatives as cathodic material for lithium ion batteries. *Journal of Power Sources* **2016**, *328*, 228–234.

[26] Park, H.; Lim, H.-D.; Lim, H.-K.; Seong, W. M.; Moon, S.; Ko, Y.; Lee, B.; Bae, Y.; Kim, H.; Kang, K. High-efficiency and high-power rechargeable lithium–sulfur dioxide batteries exploiting conventional carbonate-based electrolytes. *Nature Communications* **2017**, *8*, 14989.

[27] Allam, O.; Cho, B. W.; Kim, K. C.; Jang, S. S. Application of DFT-based machine learning for developing molecular electrode materials in Li-ion batteries. *RSC Advances* **2018**, *8*, 39414–39420.

[28] Kim, J.; Ko, S.; Noh, C.; Kim, H.; Lee, S.; Kim, D.; Park, H.; Kwon, G.; Son, G.; Ko, J. W.; Jung, Y.; Lee, D.; Park, C. B.; Kang, K.

Biological Nicotinamide Cofactor as a Redox-Active Motif for Reversible Electrochemical Energy Storage. *Angewandte Chemie International Edition* **2019**, *58*, 16764–16769.

[29] Jia, X.; Wang, M.; Shao, Y.; König, G.; Brooks, B. R.; Zhang, J. Z. H.; Mei, Y. Calculations of Solvation Free Energy through Energy Reweighting from Molecular Mechanics to Quantum Mechanics. *Journal of Chemical Theory and Computation* **2016**, *12*, 499–511.

[30] Tropsha, A. Best Practices for QSAR Model Development, Validation, and Exploitation. *Molecular Informatics* **2010**, *29*, 476–488.

[31] Cherkasov, A. et al. QSAR Modeling: Where Have You Been? Where Are You Going To? *Journal of Medicinal Chemistry* **2014**, *57*, 4977–5010.

[32] Schmidhuber, J. Deep learning in neural networks: An overview. *Neural Networks* **2015**, *61*, 85–117.

[33] Butler, K. T.; Davies, D. W.; Cartwright, H.; Isayev, O.; Walsh, A. Machine learning for molecular and materials science. *Nature* **2018**, *559*, 547–555.

[34] Kearnes, S.; McCloskey, K.; Berndl, M.; Pande, V.; Riley, P. Molec-

ular graph convolutions: moving beyond fingerprints. *Journal of Computer-Aided Molecular Design* **2016**, *30*, 595–608.

[35] Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; Dahl, G. E. Neural Message Passing for Quantum Chemistry. *arXiv:1704.01212 [cs]* **2017**, arXiv: 1704.01212.

[36] Schütt, K. T.; Arbabzadah, F.; Chmiela, S.; Müller, K. R.; Tkatchenko, A. Quantum-chemical insights from deep tensor neural networks. *Nature Communications* **2017**, *8*, 13890.

[37] Smith, J. S.; Isayev, O.; Roitberg, A. E. ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost. *Chemical Science* **2017**, *8*, 3192–3203.

[38] Ryu, S.; Lim, J.; Hong, S. H.; Kim, W. Y. Deeply learning molecular structure-property relationships using attention- and gate-augmented graph convolutional network. *arXiv:1805.10988 [cs, stat]* **2018**, arXiv: 1805.10988.

[39] Schütt, K. T.; Sauceda, H. E.; Kindermans, P.-J.; Tkatchenko, A.; Müller, K.-R. SchNet – A deep learning architecture for molecules and materials. *The Journal of Chemical Physics* **2018**, *148*, 241722.

[40] Sifain, A. E.; Lubbers, N.; Nebgen, B. T.; Smith, J. S.; Lokhov, A. Y.; Isayev, O.; Roitberg, A. E.; Barros, K.; Tretiak, S. Discovering a

Transferable Charge Assignment Model Using Machine Learning. *The Journal of Physical Chemistry Letters* **2018**, *9*, 4495–4501.

[41] Ryu, S.; Kwon, Y.; Kim, W. Y. A Bayesian graph convolutional network for reliable prediction of molecular properties with uncertainty quantification. *Chemical Science* **2019**, *10*, 8438–8446.

[42] Schütt, K. T.; Kessel, P.; Gastegger, M.; Nicoli, K. A.; Tkatchenko, A.; Müller, K.-R. SchNetPack: A Deep Learning Toolbox For Atomistic Systems. *Journal of Chemical Theory and Computation* **2019**, *15*, 448–455.

[43] Winter, R.; Montanari, F.; Noé, F.; Clevert, D.-A. Learning continuous and data-driven molecular descriptors by translating equivalent chemical representations. *Chemical Science* **2019**, *10*, 1692–1701.

[44] Mohamed, N. A.; Bradshaw, R. T.; Essex, J. W. Evaluation of solvation free energies for small molecules with the AMOEBA polarizable force field. *Journal of Computational Chemistry* **2016**, *37*, 2749–2758.

[45] Kromann, J. C.; Steinmann, C.; Jensen, J. H. Improving solvation energy predictions using the SMD solvation method and semiempirical electronic structure methods. *The Journal of Chemical Physics* **2018**, *149*, 104102.

[46] Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed Representations of Words and Phrases and their Compositionality. *arXiv:1310.4546 [cs, stat]* **2013**, arXiv: 1310.4546.

[47] Pennington, J.; Socher, R.; Manning, C. Glove: Global Vectors for Word Representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Doha, Qatar, 2014; pp 1532–1543.

[48] Bahdanau, D.; Cho, K.; Bengio, Y. Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv:1409.0473 [cs, stat]* **2016**, arXiv: 1409.0473.

[49] Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhutdinov, R.; Zemel, R.; Bengio, Y. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. *arXiv:1502.03044 [cs]* **2016**, arXiv: 1502.03044.

[50] Luong, M.-T.; Pham, H.; Manning, C. D. Effective Approaches to Attention-based Neural Machine Translation. *arXiv:1508.04025 [cs]* **2015**, arXiv: 1508.04025.

[51] Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient Estimation of Word Representations in Vector Space. *arXiv:1301.3781 [cs]* **2013**, arXiv: 1301.3781.

[52] Asgari, E.; Mofrad, M. R. K. Continuous Distributed Representation of Biological Sequences for Deep Proteomics and Genomics. *PLOS ONE* **2015**, *10*, e0141287.

[53] Goh, G. B.; Hodas, N. O.; Siegel, C.; Vishnu, A. SMILES2Vec: An Interpretable General-Purpose Deep Neural Network for Predicting Chemical Properties. *arXiv:1712.02034 [cs, stat]* **2018**, arXiv: 1712.02034.

[54] Jaeger, S.; Fulle, S.; Turk, S. Mol2vec: Unsupervised Machine Learning Approach with Chemical Intuition. *Journal of Chemical Information and Modeling* **2018**, *58*, 27–35.

[55] Morgan, H. L. The Generation of a Unique Machine Description for Chemical Structures-A Technique Developed at Chemical Abstracts Service. *Journal of Chemical Documentation* **1965**, *5*, 107–113.

[56] Rogers, D.; Hahn, M. Extended-Connectivity Fingerprints. *Journal of Chemical Information and Modeling* **2010**, *50*, 742–754.

[57] Cereto-Massagué, A.; Ojeda, M. J.; Valls, C.; Mulero, M.; Garcia-Vallvé, S.; Pujadas, G. Molecular fingerprint similarity search in virtual screening. *Methods* **2015**, *71*, 58–63.

[58] Schuster, M.; Paliwal, K. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing* **1997**, *45*, 2673–2681.

[59] Bengio, Y.; Simard, P.; Frasconi, P. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks* **1994**, *5*, 157–166.

[60] Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Computation* **1997**, *9*, 1735–1780.

[61] Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *arXiv:1412.3555 [cs]* **2014**, arXiv: 1412.3555.

[62] Marenich, A. V.; Kelly, C. P.; Thompson, J. D.; Hawkins, G. D.; Chambers, C. C.; Giesen, D. J.; Winget, P.; Cramer, C. J.; Truhlar, D. G. *Minnesota Solvation Database – version 2012*; 2012; Published: University of Minnesota, Minneapolis.

[63] Swain, M.; Kurniawan, E.; Powers, Z.; Yi, H.; Lazzaro, L.; Dahlgren, B.; Sjorgen, R. *PubChemPy*; 2014.

[64] Chollet, F.; others, *Keras*; 2015.

[65] Martín Abadi, et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*; 2015.

[66] Pedregosa, F. et al. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* **2011**, *12*, 2825–2830.

[67] Lusci, A.; Pollastri, G.; Baldi, P. Deep Architectures and Deep Learning in Chemoinformatics: The Prediction of Aqueous Solubility for Drug-Like Molecules. *Journal of Chemical Information and Modeling* **2013**, *53*, 1563–1575.

[68] Zheng, X.; Liu, Q.; Jing, C.; Li, Y.; Li, D.; Luo, W.; Wen, Y.; He, Y.; Huang, Q.; Long, Y.-T.; Fan, C. Catalytic Gold Nanoparticles for Nanoplasmonic Detection of DNA Hybridization. *Angewandte Chemie International Edition* **2011**, *50*, 11994–11998.

[69] Genheden, S. Solvation free energies and partition coefficients with the coarse-grained and hybrid all-atom/coarse-grained MARTINI models. *Journal of Computer-Aided Molecular Design* **2017**, *31*, 867–876.

[70] Dupont, C.; Andreussi, O.; Marzari, N. Self-consistent continuum solvation (SCCS): The case of charged systems. *The Journal of Chemical Physics* **2013**, *139*, 214110.

[71] Sundararaman, R.; Goddard, W. A. The charge-asymmetric nonlocally determined local-electric (CANDLE) solvation model. *The Journal of Chemical Physics* **2015**, *142*, 064107.

[72] Klamt, A. The COSMO and COSMO-RS solvation models: COSMO

and COSMO-RS. *Wiley Interdisciplinary Reviews: Computational Molecular Science* **2018**, *8*, e1338.

[73] Klamt, A. Conductor-like Screening Model for Real Solvents: A New Approach to the Quantitative Calculation of Solvation Phenomena. *The Journal of Physical Chemistry* **1995**, *99*, 2224–2235.

[74] Mayr, A.; Klambauer, G.; Unterthiner, T.; Steijaert, M.; Wegner, J. K.; Ceulemans, H.; Clevert, D.-A.; Hochreiter, S. Large-scale comparison of machine learning methods for drug target prediction on ChEMBL. *Chemical Science* **2018**, *9*, 5441–5451.

[75] Marenich, A. V.; Olson, R. M.; Kelly, C. P.; Cramer, C. J.; Truhlar, D. G. Self-Consistent Reaction Field Model for Aqueous and Nonaqueous Solutions Based on Accurate Polarized Partial Charges. *Journal of Chemical Theory and Computation* **2007**, *3*, 2011–2033.

[76] Zheng, S.; Yan, X.; Yang, Y.; Xu, J. Identifying Structure–Property Relationships through SMILES Syntax Analysis with Self-Attention Mechanism. *Journal of Chemical Information and Modeling* **2019**, *59*, 914–923.

[77] Mikolov, T.; Kopecky, J.; Burget, L.; Glembek, O.; Cernocky, J. Neural network based language models for highly inflective languages.

2009 IEEE International Conference on Acoustics, Speech and Signal Processing. Taipei, Taiwan, 2009; pp 4725–4728.

[78] Stolov, M. A.; Zaitseva, K. V.; Varfolomeev, M. A.; Acree, W. E. Enthalpies of solution and enthalpies of solvation of organic solutes in ethylene glycol at 298.15 K: Prediction and analysis of intermolecular interaction contributions. *Thermochimica Acta* **2017**, *648*, 91–99.

[79] Sedov, I. A.; Salikov, T. M.; Wadawadigi, A.; Zha, O.; Qian, E.; Acree, W. E.; Abraham, M. H. Abraham model correlations for describing the thermodynamic properties of solute transfer into pentyl acetate based on headspace chromatographic and solubility measurements. *The Journal of Chemical Thermodynamics* **2018**, *124*, 133–140.

[80] Peters, M. E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; Zettlemoyer, L. Deep contextualized word representations. *arXiv:1802.05365 [cs]* **2018**, arXiv: 1802.05365.

[81] Kipf, T. N.; Welling, M. Semi-Supervised Classification with Graph Convolutional Networks. *arXiv:1609.02907 [cs, stat]* **2017**, arXiv: 1609.02907.

[82] Sterling, T.; Irwin, J. J. ZINC 15 – Ligand Discovery for Every-

one. *Journal of Chemical Information and Modeling* **2015**, *55*, 2324–2337.

[83] Searles, D. J.; Evans, D. J. The fluctuation theorem and Green–Kubo relations. *The Journal of Chemical Physics* **2000**, *112*, 9727–9735.

[84] Kotsiantis, S. B. Decision trees: a recent overview. *Artificial Intelligence Review* **2013**, *39*, 261–283.

[85] Breiman, L. Random Forests. *Machine Learning* **2001**, *45*, 5–32.

[86] Friedman, J. H. Greedy Function Approximation: A Gradient Boosting Machine. *The Annals of Statistics* **2001**, *29*, 1189–1232.

[87] Thomson, G. H. The DIPPR databases. *International Journal of Thermophysics* **1996**, *17*, 223–232.

[88] Whitesides, G.; Mathias, J.; Seto, C. Molecular self-assembly and nanochemistry: a chemical strategy for the synthesis of nanostructures. *Science* **1991**, *254*, 1312–1319.

[89] Rothemund, P. W. K. Folding DNA to create nanoscale shapes and patterns. *Nature* **2006**, *440*, 297–302.

[90] Prybytak, P.; Frith, W. J.; Cleaver, D. J. Hierarchical self-assembly of chiral fibres from achiral particles. *Interface Focus* **2012**, *2*, 651–657.

[91] Whitesides, G. M.; Boncheva, M. Beyond molecules: Self-assembly of mesoscopic and macroscopic components. *Proceedings of the National Academy of Sciences* **2002**, *99*, 4769–4774.

[92] Stupp, S. I. Self-Assembly and Biomaterials. *Nano Letters* **2010**, *10*, 4783–4786.

[93] Glotzer, S. C.; Solomon, M. J. Anisotropy of building blocks and their assembly into complex structures. *Nature Materials* **2007**, *6*, 557–562.

[94] Langer, R.; Tirrell, D. A. Designing materials for biology and medicine. *Nature* **2004**, *428*, 487–492.

[95] Klotsa, D.; Jack, R. L. Predicting the self-assembly of a model colloidal crystal. *Soft Matter* **2011**, *7*, 6294.

[96] Whitelam, S.; Hedges, L. O.; Schmit, J. D. Self-Assembly at a Nonequilibrium Critical Point. *Physical Review Letters* **2014**, *112*, 155504.

[97] Whitelam, S.; Geissler, P. L. Avoiding unphysical kinetic traps in Monte Carlo simulations of strongly attractive particles. *The Journal of Chemical Physics* **2007**, *127*, 154101.

[98] Whitelam, S.; Jack, R. L. The Statistical Mechanics of Dynamic Path-

ways to Self-Assembly. *Annual Review of Physical Chemistry* **2015**, *66*, 143–163.

[99] Grant, J.; Jack, R. L.; Whitelam, S. Analyzing mechanisms and microscopic reversibility of self-assembly. *The Journal of Chemical Physics* **2011**, *135*, 214505.

[100] Klotsa, D.; Jack, R. L. Controlling crystal self-assembly using a real-time feedback scheme. *The Journal of Chemical Physics* **2013**, *138*, 094502.

[101] Whitelam, S.; Feng, E. H.; Hagan, M. F.; Geissler, P. L. The role of collective motion in examples of coarsening and self-assembly. *Soft Matter* **2009**, *5*, 1251–1262.

[102] Jack, R. L.; Hagan, M. F.; Chandler, D. Fluctuation-dissipation ratios in the dynamics of self-assembly. *Physical Review E* **2007**, *76*, 021119.

[103] Grant, J.; Jack, R. L. Quantifying reversibility in a phase-separating lattice gas: An analogy with self-assembly. *Physical Review E* **2012**, *85*, 021112.

[104] Perkett, M. R.; Hagan, M. F. Using Markov state models to study self-assembly. *The Journal of Chemical Physics* **2014**, *140*, 214101.

[105] Rapaport, D. C. Role of Reversibility in Viral Capsid Growth: A Paradigm for Self-Assembly. *Physical Review Letters* **2008**, *101*, 186101.

[106] Garrahan, J. P.; Jack, R. L.; Lecomte, V.; Pitard, E.; van Duijvendijk, K.; van Wijland, F. Dynamical First-Order Phase Transition in Kinetically Constrained Models of Glasses. *Physical Review Letters* **2007**, *98*, 195702.

[107] Garrahan, J. P.; Jack, R. L.; Lecomte, V.; Pitard, E.; van Duijvendijk, K.; van Wijland, F. First-order dynamical phase transition in models of glasses: an approach based on ensembles of histories. *Journal of Physics A: Mathematical and Theoretical* **2009**, *42*, 075007.

[108] Garrahan, J. P. Classical stochastic dynamics and continuous matrix product states: gauge transformations, conditioned and driven processes, and equivalence of trajectory ensembles. *Journal of Statistical Mechanics: Theory and Experiment* **2016**, *2016*, 073208.

[109] Jack, R. L.; Sollich, P. Large Deviations and Ensembles of Trajectories in Stochastic Models. *Progress of Theoretical Physics Supplement* **2010**, *184*, 304–317.

[110] Chetrite, R.; Touchette, H. Nonequilibrium Microcanonical and

Canonical Ensembles and Their Equivalence. *Physical Review Letters* **2013**, *111*, 120601.

[111] Vaikuntanathan, S.; Gingrich, T. R.; Geissler, P. L. Dynamic phase transitions in simple driven kinetic networks. *Physical Review E* **2014**, *89*, 062108.

[112] Budini, A. A.; Turner, R. M.; Garrahan, J. P. Fluctuating observation time ensembles in the thermodynamics of trajectories. *Journal of Statistical Mechanics: Theory and Experiment* **2014**, *2014*, P03012.

[113] Lecomte, V.; Appert-Rolland, C.; van Wijland, F. Thermodynamic Formalism for Systems with Markov Dynamics. *Journal of Statistical Physics* **2007**, *127*, 51–106.

[114] Hedges, L. O.; Jack, R. L.; Garrahan, J. P.; Chandler, D. Dynamic Order-Disorder in Atomistic Models of Structural Glass Formers. *Science* **2009**, *323*, 1309–1313.

[115] Touchette, H. The large deviation approach to statistical mechanics. *Physics Reports* **2009**, *478*, 1–69.

[116] Speck, T.; Chandler, D. Constrained dynamics of localized excitations causes a non-equilibrium phase transition in an atomistic model of glass formers. *The Journal of Chemical Physics* **2012**, *136*, 184509.

[117] Speck, T.; Malins, A.; Royall, C. P. First-Order Phase Transition in a Model Glass Former: Coupling of Local Structure and Dynamics. *Physical Review Letters* **2012**, *109*, 195703.

[118] Jack, R. L.; Hedges, L. O.; Garrahan, J. P.; Chandler, D. Preparation and Relaxation of Very Stable Glassy States of a Simulated Liquid. *Physical Review Letters* **2011**, *107*, 275702.

[119] Gaspard, P. Time-Reversed Dynamical Entropy and Irreversibility in Markovian Random Processes. *Journal of Statistical Physics* **2004**, *117*, 599–615.

[120] Gardiner, C. W. *Handbook of Stochastic Methods: For Physics, Chemistry and Natural Sciences*; Springer, 1985.

[121] Bortz, A.; Kalos, M.; Lebowitz, J. A new algorithm for Monte Carlo simulation of Ising spin systems. *Journal of Computational Physics* **1975**, *17*, 10–18.

[122] Bolhuis, P. G.; Chandler, D.; Dellago, C.; Geissler, P. L. Transition path sampling: throwing ropes over rough mountain passes, in the dark. *Annual Review of Physical Chemistry* **2002**, *53*, 291–318.

[123] Bennett, C. H. Efficient estimation of free energy differences from Monte Carlo data. *Journal of Computational Physics* **1976**, *22*, 245–268.

[124] Shirts, M. R.; Chodera, J. D. Statistically optimal analysis of samples from multiple equilibrium states. *The Journal of Chemical Physics* **2008**, *129*, 124105.

[125] Elmatad, Y. S.; Jack, R. L.; Chandler, D.; Garrahan, J. P. Finite-temperature critical point of a glass transition. *Proceedings of the National Academy of Sciences* **2010**, *107*, 12793–12798.

[126] Michaelis, L.; Menten, M. L. Die Kinetik der Invertinwirkung. *Biochem. Z.* **1913**, *49*, 333.

[127] Johnson, K. A.; Goody, R. S. The Original Michaelis Constant: Translation of the 1913 Michaelis–Menten Paper. *Biochemistry* **2011**, *50*, 8264–8269.

[128] Cornish-Bowden, A. One hundred years of Michaelis–Menten kinetics. *Perspectives in Science* **2015**, *4*, 3–9.

[129] Ronen, M.; Rosenberg, R.; Shraiman, B. I.; Alon, U. Assigning numbers to the arrows: Parameterizing a gene regulation network by using accurate expression kinetics. *Proceedings of the National Academy of Sciences* **2002**, *99*, 10555–10560.

[130] Pulkkinen, O.; Metzler, R. Variance-corrected Michaelis-Menten equation predicts transient rates of single-enzyme reactions and re-

sponse times in bacterial gene-regulation. *Scientific Reports* **2015**, *5*, 17820.

[131] Schnitzer, M. J.; Visscher, K.; Block, S. M. Force production by single kinesin motors. *Nature Cell Biology* **2000**, *2*, 718–723.

[132] Asbury, C. L. Kinesin Moves by an Asymmetric Hand-Over-Hand Mechanism. *Science* **2003**, *302*, 2130–2134.

[133] van Oijen, A. M. Single-Molecule Kinetics of Exonuclease Reveal Base Dependence and Dynamic Disorder. *Science* **2003**, *301*, 1235–1238.

[134] English, B. P.; Min, W.; van Oijen, A. M.; Lee, K. T.; Luo, G.; Sun, H.; Cherayil, B. J.; Kou, S. C.; Xie, X. S. Ever-fluctuating single enzyme molecules: Michaelis-Menten equation revisited. *Nature Chemical Biology* **2006**, *2*, 87–94.

[135] Cao, J.; Silbey, R. J. Generic Schemes for Single-Molecule Kinetics. 1: Self-Consistent Pathway Solutions for Renewal Processes. *The Journal of Physical Chemistry B* **2008**, *112*, 12867–12880.

[136] Qian, H.; Bishop, L. M. The Chemical Master Equation Approach to Nonequilibrium Steady-State of Open Biochemical Systems: Linear Single-Molecule Enzyme Kinetics and Nonlinear Biochemical Re-

action Networks. *International Journal of Molecular Sciences* **2010**, *11*, 3472–3500.

[137] Jung, W.; Yang, S.; Sung, J. Novel Chemical Kinetics for a Single Enzyme Reaction: Relationship between Substrate Concentration and the Second Moment of Enzyme Reaction Time. *The Journal of Physical Chemistry B* **2010**, *114*, 9840–9847.

[138] Yang, S.; Cao, J.; Silbey, R.; Sung, J. Quantitative Interpretation of the Randomness in Single Enzyme Turnover Times. *Biophysical Journal* **2011**, *101*, 519–524.

[139] Reuveni, S.; Urbakh, M.; Klafter, J. Role of substrate unbinding in Michaelis-Menten enzymatic reactions. *Proceedings of the National Academy of Sciences* **2014**, *111*, 4391–4396.

[140] Rotbart, T.; Reuveni, S.; Urbakh, M. Michaelis-Menten reaction scheme as a unified approach towards the optimal restart problem. *Physical Review E* **2015**, *92*, 060101.

[141] Park, S. J.; Song, S.; Jeong, I.-C.; Koh, H. R.; Kim, J.-H.; Sung, J. Nonclassical Kinetics of Clonal yet Heterogeneous Enzymes. *The Journal of Physical Chemistry Letters* **2017**, *8*, 3152–3158.

[142] Qian, H.; L. Elson, E. Single-molecule enzymology: stochastic

Michaelis–Menten kinetics. *Biophysical Chemistry* **2002**, *101-102*, 565–576.

[143] Kou, S. C.; Cherayil, B. J.; Min, W.; English, B. P.; Xie, X. S. Single-Molecule MichaelisMenten Equations. *The Journal of Physical Chemistry B* **2005**, *109*, 19068–19081.

[144] Murugan, A.; Vaikuntanathan, S. Biological Implications of Dynamical Phases in Non-equilibrium Networks. *Journal of Statistical Physics* **2016**, *162*, 1183–1202.

[145] Klymko, K.; Garrahan, J. P.; Whitelam, S. Similarity of ensembles of trajectories of reversible and irreversible growth processes. *Physical Review E* **2017**, *96*, 042126.

[146] Whitelam, S. Large deviations in the presence of cooperativity and slow dynamics. *Physical Review E* **2018**, *97*, 062109.

[147] Klymko, K.; Geissler, P. L.; Garrahan, J. P.; Whitelam, S. Rare behavior of growth processes via umbrella sampling of trajectories. *Physical Review E* **2018**, *97*, 032123.

[148] Weber, J. K.; Jack, R. L.; Pande, V. S. Emergence of Glass-like Behavior in Markov State Models of Protein Folding Dynamics. *Journal of the American Chemical Society* **2013**, *135*, 5501–5504.

[149] Tuckerman, M. E. *Statistical mechanics: theory and molecular simulation*; Oxford University Press, 2010.

[150] Min, W.; Jiang, L.; Yu, J.; Kou, S. C.; Qian, H.; Xie, X. S. Nonequilibrium Steady State of a Nanometric Biochemical System: Determining the Thermodynamic Driving Force from Single Enzyme Turnover Time Traces. *Nano Letters* **2005**, *5*, 2373–2378.

[151] Sinitsyn, N. A.; Nemenman, I. The Berry phase and the pump flux in stochastic chemical kinetics. *Europhysics Letters (EPL)* **2007**, *77*, 58001.

국문초록

최근 기계학습 기술의 급격한 발전과 이의 화학 분야에 대한 적용은 다양한 화학적 성질에 대한 구조-성질 정량 관계를 기반으로 한 예측 모형의 개발을 가속하고 있다. 용매화 자유 에너지는 그러한 기계학습의 적용 예 중 하나이며 다양한 용매 내의 화학반응에서 중요한 역할을 하는 근본적 성질 중 하나이다. 본 연구에서 우리는 목표로 하는 용매화 자유 에너지를 원자간의 상호작용으로부터 구할 수 있는 새로운 심층학습 기반 용매화 모형을 소개한다. 제안된 심층학습 모형의 계산 과정은 용매와 용질 분자에 대한 부호화 함수가 각 원자와 분자들의 구조적 성질에 대한 벡터 표현을 추출하며, 이를 토대로 원자간 상호작용을 복잡한 퍼셉트론 신경망 대신 벡터간의 간단한 내적으로 구할 수 있다. 952가지의 유기용질과 147가지의 유기용매를 포함하는 6,493가지의 실험치를 토대로 기계학습 모형의 교차 검증 시험을 실시한 결과, 평균 절대 오차 기준 0.2 kcal/mol 수준으로 매우 높은 정확도를 가진다. 스캐폴드-기반 교차 검증의 결과 역시 0.6 kcal/mol 수준으로, 외삽으로 분류할 수 있는 비교적 새로운 분자 구조에 대한 예측에 대해서도 우수한 정확도를 보인다. 또한, 제안된

특정 기계학습 모형은 그 구조 상 특정 용매에 특화되지 않았기 때문에 높은 양도성을 가지며 학습에 이용할 데이터의 수를 늘이는 데 용이하다. 원자간 상호작용에 대한 분석을 통해 제안된 심층학습 모형 용매화 자유 에너지에 대한 그룹-기여도를 잘 재현할 수 있음을 알 수 있으며, 기계학습을 통해 단순히 목표로 하는 성질만을 예측하는 것을 넘어 더욱 상세한 물리화학적 이해를 하는 것이 가능할 것이라 기대할 수 있다.

주요어: 심층학습, 구조-성질 정량 관계, 용매화 자유 에너지, 용해도, 액체 성질, 액체계

학번: 2010-23098

이학박사 학위논문

Deep Learning Approaches to Predictions of Liquid Properties

심층학습을 이용한 액체계의 성질 예측

2020년 2월

서울대학교 대학원

화학부 물리화학 전공

임 현 태

Deep Learning Approaches to Predictions of Liquid Properties

by

Hyuntae Lim

Supervised by

Professor **YounJoon Jung**

A Dissertation
Submitted to the Faculty of
Seoul National University
in Partial Fulfillment of
the Requirements for the Degree of
Doctor of Philosophy

February 2020

Department of Chemistry
Graduate School
Seoul National University

Abstract

Recent advances in machine learning technologies and their chemical applications lead to the developments of diverse structure-property relationship based prediction models for various chemical properties; the free energy of solvation is one of them and plays a dominant role as a fundamental measure of solvation chemistry. Here, we introduce a novel machine learning-based solvation model, which calculates the target solvation free energy from pairwise atomistic interactions. The novelty of our proposed solvation model involves rather simple architecture: two encoding function extracts vector representations of the atomic and the molecular features from the given chemical structure, while the inner product between two atomistic features calculates their interactions, instead of black-boxed perceptron networks. The cross-validation result on 6,493 experimental measurements for 952 organic solutes and 147 organic solvents achieves an outstanding performance, which is 0.2 kcal/mol in MUE. The scaffold-based split method exhibits 0.6 kcal/mol, which shows that the proposed model guarantees reasonable accuracy even for extrapolated cases. Moreover, the proposed model shows an excellent transferability for enlarging training data due to its solvent-non-specific nature. Analysis of the atomistic interaction map shows there is a great potential that our proposed model reproduces group contributions on the solvation energy, which makes us believe that the proposed model not only provides the predicted target property, but also gives us more detailed physicochemical insights.

Keywords: Deep learning, Structure-property relationship, Solvation free energy, Solubility, Liquid property, Liquid system

Student Number: 2010-23098

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The importance of solvation or hydration mechanism and its involved free energy change has made various *in silico* calculation methods for the solvation energy a major topic in computational chemistry.[1–22] The solvation free energy directly influences to many chemical properties in solution and plays a dominant role in various chemical reactions: drug delivery[4, 15, 17, 23], organic synthesis[24], electrochemical redox reactions[25–28], *et cetera.*

The realistic computer simulation approaches for the solvent and the solute molecules directly offer the microscopic structure of the solvation shell, which surrounds the solutes molecule.[9, 10, 13, 16, 17, 29] The solvation shell structure could provide us detailed physicochemical information like microscopic mechanisms on solvation or the interplay between the solvent and the solute molecules when we use an appropriate force field model and parameters. However, those *explicit solvation* methods we stated above need an extensive amount of numerical calculations since we have to simulate each individual molecule in the solvated system. Moreover, the free energy calculation procedure with an explicitly implemented solvent model necessarily involves rare-event sampling methods, which make the task even more computationally expensive. The realistic problems on the explicit solvation model restrict its applications to classical molecular mechanics simulations,[9, 10, 16] or a limited QM/MM approaches.[13, 29]

For classical mechanics approaches for macromolecules or calculations for small compounds at quantum-mechanical level, the idea of *implicit solvation* enables us to calculate solvation energy with feasible time and computational costs when one considers a given solvent as a continuous and isotropic medium in the Poisson-Boltzmann equation.[1–3, 5–8, 11] Many theoretical advances have introduced to construct the PB-based equation, which involves parameterized solvent properties: the polarizable continuum model (PCM),[11] the conductor-like screening model (COSMO),[3] generalized Born approximations like solvation model based on density (SMD)[7] or solvation model 6, 8, 12, ... (SMx).[1, 6] The conductor-like screening model for realistic solvents (COSMO-RS) is a noteworthy solvation model since it is believed to be the state-of-the-art method.[2] This is realized by statistical thermodynamics treatment on the polarization charge densities, which helps COSMO-RS with making successful predictions even in polar solvents where the fundamental idea of the dielectric continuum solvation collapses.[8]

The quantitative structure activity relationship (QSAR) or the quantitative structure property relationship (QSPR) is a rather new approach, which predicts the solvation free energy with a completely different point of view when compared to computer simulation approaches with precisely defined theoretical backgrounds[30, 31]. The underlying architecture of QSPR consists of two elementary mathematical functions[30]: one is the *encoding function*, which encodes the structural or chemical features of a given compound into a *molecular descriptor*. The other, the *mapping function*, predicts the target property (or activity) that we intend to find out using the descriptor from the encoding function. Although we cannot expect detailed chemical or physical insights other than the target property since the QSAR/QSPR is a regression analysis in its intrinsic nature, It has shown its advantages in terms of transferability and outstanding computational efficiency[20, 30, 31].

Recent successes in the machine learning (ML) technique[32] and their implementations in computational chemistry[20, 33] are promoting broad applications of QSAR/QSPR in numerous chemical studies[4, 18, 21, 23, 27, 34–43]. Those studies proved that ML guarantees faster calculations than computer simulations and more precise estimations than traditional

QSPR estimations; a decent number of models showed accuracies comparable to *ab initio* solvation models in the aqueous system[20].

In this thesis, we introduce a novel artificial neural-network-based ML model called *Delfos* that predicts free energies of solvation for generic organic solvents in the previous work[22]. The model not only has a great potential of showing an accuracy comparable to the state-of-the-art computational chemistry methods[1, 2] but offers information about which substructures play a dominant role in the solvation process. As a further development, we propose an improved ML model for the solvation energy estimation, which is based on the group-contribution method. The key idea of the proposed model is the calculation of pairwise atomic interactions by inner products of atomic feature vectors, while each encoder network for the solvent and the solute extracts such atomic features.

The outline of the rest of the present thesis is as follows: in Chapter 2, we mainly discuss the performance of Delfos, with both MD and *ab initio* simulation strategies[1, 2, 44, 45] and analyze database sensitivity using cluster cross-validation method. We also visualize important substructures in solvation via attention mechanism. In Chapter 3, we introduce a new ML model for the solvation energy prediction, which is based on pairwise atom-by-atom interactions. The chapter quantifies the proposed model's performance with 6,594 data points, mainly focused on group contributions and pairwise atomistic interactions. In the last chapter of the thesis, we summarize and conclude our work.

Chapter 2

Delfos: Deep Learning Model for Prediction of Solvation Free Energies in Generic Organic Solvents

2.1 Methods

2.1.1 Embedding of Chemical Contexts

Natural language processing (NLP) is one of the most cutting-edge subfields of computer science in varied applications of machine learning and neural networks[46–50]. To process human languages using computers, we need to encode words and sentences and extract their linguistic properties. The process is commonly implemented via *word embedding* method[46, 47]. To perform the task, unsupervised learning schemes such as skip-gram and continuous bag of words (CBOW) algorithms generate a vector representation of the given word in an arbitrary vector space[47, 51]. If the necessary vector space is well-defined, one can conjecture the semantic or syntactic features of the given word from the position of the embedded vector, and the inner product of two vectors corresponding to two different words provides information about their semantic similarity.

It is worthwhile to note that we can employ the embedding technique for chemical or biophysical processes if we consider an atom or a substructure as a word and a compound as a sentence[52–54]. In that case, positions of molecular substructures in the embedded vector space represent their chemical and physical properties, instead of linguistic information. Several models have already been developed along the line of this idea. For example, bio-vector models[52] that have been developed to encode sequences of proteins or DNAs, and atomic-vector embedding models have been introduced recently to encode structural features of chemical compounds[53, 54]. Mol2Vec is one of such embedding techniques, and it generates vector representations of a given molecule from the *molecular sentence*[54]. To make molecular sentences, Mol2Vec uses the Morgan algorithm[55] that assorts identical atoms in the molecule. The algorithm is commonly used to generate ECFP fingerprints[56], which are the *de facto* standard in cheminformatics[57], and they make identifiers of the given atom from the chemical environment where the atom is positioned. An atom may have multiple identifiers depending on the pre-set maximum value of *radius* $r_{\max}$, which denotes the maximum topological distance between the given atom and its neighboring atoms. The atom itself is identified

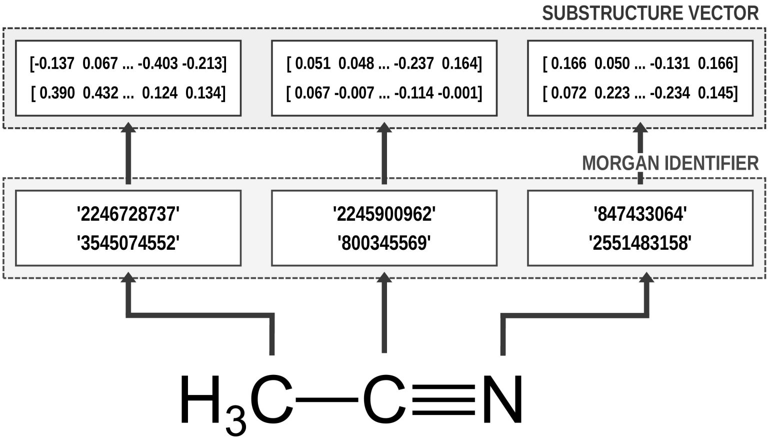


Figure 2.1: Schematic illustration of the molecular embedding process for acetonitrile (SMILES: CC#N) and $r_{\max} = 1$. The Morgan algorithm discriminates identifiers between two substructures: one is for itself ($r = 0$) and the other considers its nearest neighbor atoms ($r = 1$). Then the embedding layer calculates the vector representation from the given identifier.

by $r = 0$, and additional substructure identifiers for adjacent atoms are denoted by $r = 1$ (nearest neighbor), $r = 2$ (next nearest neighbor), and so on. Since Mol2Vec has demonstrated promising performances in several applications of QSAR/QSPR[54], Delfos uses Mol2Vec as the primary encoding means. We schematically illustrated embedding procedure for acetonitrile in Fig. 2.1.

2.1.2 Encoder-Predictor Network

As shown in Fig. 2.2, the fundamental architecture of Delfos involves three sub-neural networks: the solvent and the solute encoders extract dominant structural features of the given compound from SMILES strings, while the predictor calculates the solvation energy of the given solvent-solute pair from their encoded features.

The primary architecture of the encoder is based on two bidirectional recurrent neural networks (BiRNNs)[58]. The network is designed for handling sequential data and we consider the molecular sentence $[\mathbf{x}_1, \cdots, \mathbf{x}_N]$ as a sequence of embedded substructures, $\mathbf{x}_i$. RNNs may have a failure when input sequences are lengthy; gradients of the loss function can be diluted or amplified because of accumulated precision error from the back-propagation process[59]. The excessive or restrained gradient may cause a decline in learning performance, and we call these two problems as vanishing or exploding gradient. To overcome these limits which stem from lengthy input sequences, one may consider using both forward-directional RNN ($\overrightarrow{\text{RNN}}$) and backward-directional RNN ($\overleftarrow{\text{RNN}}$) within a single layer:

$$\overrightarrow{\text{RNN}}([\mathbf{x}_1, \cdots, \mathbf{x}_N]) = [\overrightarrow{\mathbf{h}_1}, \cdots, \overrightarrow{\mathbf{h}_N}], \tag{2.1a}$$

$$\overleftarrow{\text{RNN}}([\mathbf{x}_1, \cdots, \mathbf{x}_N]) = [\overleftarrow{\mathbf{h}_1}, \cdots, \overleftarrow{\mathbf{h}_N}], \tag{2.1b}$$

$$\overleftrightarrow{\text{RNN}}([\mathbf{x}_1, \cdots, \mathbf{x}_N]) = [\mathbf{h}_1, \cdots, \mathbf{h}_N]. \tag{2.1c}$$

In Eqn. 2.1, $\mathbf{x}_i$ is the embedded atomic vector of a given molecule, $\overrightarrow{\mathbf{h}_i}$ and $\overleftarrow{\mathbf{h}_i}$ are hidden state outputs of each recurrent unit, and $\mathbf{h}_i = \overrightarrow{\mathbf{h}_i}; \overleftarrow{\mathbf{h}_i}$ means concatenation of two hidden states, respectively. The long-short term memory[60] (LSTM) and gated recurrent unit[61] (GRU) networks, which are modifications of RNN, are invented to handle lengthy input sequences. They introduce *gates* in each RNN cell state to memorize important information of the previous cell state and minimize vanishing and exploding gradient problem.

After RNN layers, the molecular sentences of both the solvent $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_N]$ and the solute $\mathbf{Y} = [\mathbf{y}_1, \cdots, \mathbf{y}_M]$ are converted to hidden states, $\mathbf{H} = [\mathbf{h}_1, \cdots \mathbf{h}_N]$ and $\mathbf{G} = [\mathbf{g}_1, \cdots, \mathbf{g}_M]$, respectively. Each hidden state is then put into the shared *attention* layer and weighted. The attention mechanism, which was originally proposed to enhance performances of machine translator[48], is an essential technique in diverse NLP applications nowadays[49, 50]. Principles of the attention start from the definition of the score function of hidden states and its normalization with the softmax function:

$$\alpha_{ij} = \frac{\exp(\text{score}(\mathbf{h}_i, \mathbf{g}_j))}{\sum_k \exp(\text{score}(\mathbf{h}_i, \mathbf{g}_k))}, \tag{2.2a}$$

$$\mathbf{p}_i = \sum_j^M \alpha_{ij} \mathbf{g}_j, \tag{2.2b}$$

$$\text{score}(\mathbf{h}_i, \mathbf{g}_j) = \mathbf{h}_i \cdot \mathbf{g}_j. \tag{2.2c}$$

There are various score functions that have been introduced to achieve efficient predictions[48–50], and among them we use Luong's dot-product attention[50] in Eqn. 2.2c as a score function since it is computationally efficient. The solvent context, $\mathbf{P} = \alpha\mathbf{G}$ denotes an *emphasized* hidden state $\mathbf{H}$ with the attention alignment, α. We also get the solute context $\mathbf{Q}$ using the same procedure. The context weighted from the attention layer is an $L \times 2D$ matrix, where L is the sequence length and D is the dimension of two RNN hidden layers since we use bidirectional RNN (BiRNN). Two max-pooling layers, which is the last part of each encoder reduces contexts $\mathbf{H}$, $\mathbf{G}$, $\mathbf{P}$, and $\mathbf{Q}$ to $2D$-dimensional feature vectors $\mathbf{u}$ and $\mathbf{v}$[50]:

$$\mathbf{u} = \mathrm{MaxPooling}([\mathbf{h}_1; \mathbf{p}_1, \cdots, \mathbf{h}_N; \mathbf{p}_N]), \tag{2.3a}$$

$$\mathbf{v} = \mathrm{MaxPooling}([\mathbf{g}_1; \mathbf{q}_1, \cdots, \mathbf{g}_M; \mathbf{q}_M]). \tag{2.3b}$$

The predictor has a single fully-connected perceptron layer with rectifier unit (ReLU) and an output layer. It uses the concatenated feature of the solvent and solute $[\mathbf{u}; \mathbf{v}]$ as an input. The overall architecture of our model is shown in Figure 2.2. We also consider encoders without RNN and attention layers in order to quantify the impact of these layers on prediction performances of the network; each encoding network contains only the embedding layer and directly connected to the MLP layer. The solvent and solute features are simple summations of atomic vectors, $\mathbf{u} = \sum_i^N \mathbf{x}_i$ and $\mathbf{v} = \sum_i^M \mathbf{y}_i$, respectively. This model was initially used for gradient boost-

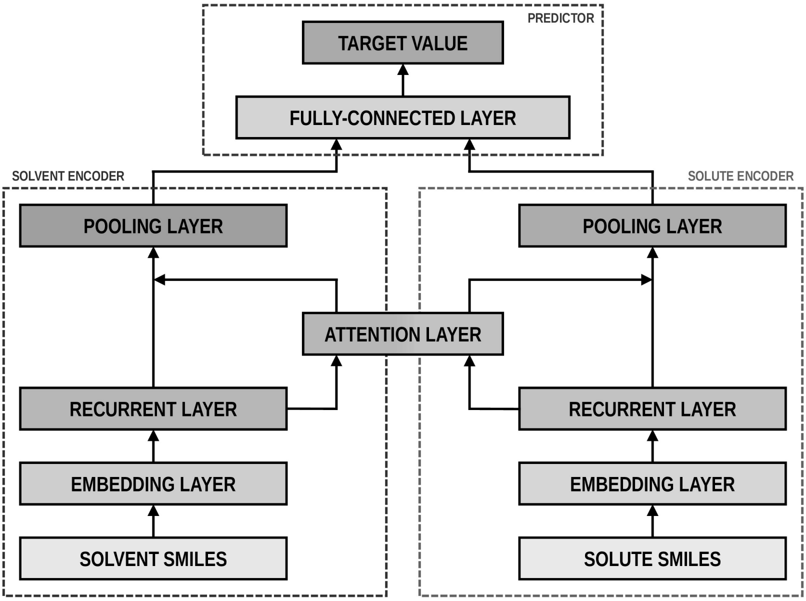


Figure 2.2: The fundamental architecture of Delfos. Each encoder network has one embedding and one recurrent layer, while the predictor has a fully-connected MLP layer. Two encoders share an attention layer, which weights outputs from recurrent layers. Black arrows indicate flow of input data.

ing (GBM) regression analysis for aqueous solubilities and toxicities[54].

2.2 Results and Discussions

2.2.1 Computational Setup and Results

We use the Minnesota solvation database[62] (MNSOL) as the dataset over which we train and test, and it provides 3,037 experimental measures of free energies of solvation and transfer energies for 790 unique solutes in 92 solvents. Because the MNSOL only contains common names of compounds, we perform an automated searching process using PubChemPy[63] script and receive SMILES strings of compounds from PubChem database. There are 363 results for charged solutes and 144 results for transfer free energies in the MNSOL which are excluded from machine learning dataset, and 35 results of solvent-solute combinations are not valid in PubChem. We finally prepare SMILES specifications of 2,495 solutions for 418 solutes and 91 solvents for the machine learning input.

For the implementation of the proposed neural networks, we use Keras 2.2.4 framework[64] with TensorFlow 1.12 backend[65]. At the very first stage, Morgan algorithm for $r = 0$ and $r = 1$ generates molecular sentences of the solvent and solute from their SMILES strings. Then the given molecular sentence is embedded to a sequence of 300-dimensional substructure vectors by the skip-gram pretrained Word2Vec model available at https://github.com/samoturk/mol2vec, which contains information of $\sim 20,000,000$ compounds and $\sim 20,000$ substructures from the ZINC15 database[54]. We consider BiLSTM and BiGRU layers in both solvent and solute encoders to compare their performances. Since our model is a regression problem, we use mean squared error (MSE) as the loss function.

We employ 10-fold cross-validation (CV) for secure representativeness of the test data because the dataset we use has a limited number of experimental measures; the total dataset is uniformly and randomly split into 10 subsets, and we iteratively choose one of the subsets as a test set and the training run uses the remaining 9 subsets. Consequentially, a 10-fold CV task performs 10 independent training and test runs, and relative sizes of the training and test sets are 9 to 1. We use Scikit-Learn library[66] to implement the CV task and perform an extensive grid search for tuning hyperparameters: learning algorithms, learning rates, and dimensions of hidden layers. We select the stochastic gradient descent (SGD) algorithm with Nesterov momentum, whose learning rate is 0.0002 and momentum is 0.9. Optimized hidden dimensions are 150 for recurrent layers and 2000 for the fully connected layer. To minimize the variance of the test run, we take averages for all results over 9 independent random CV, split from different random states.

Solvation free energies that we calculated from the MNSOL using attentive BiRNN encoders are exhibited in Fig. 2.3 and 2.4. Prediction errors for the BiLSTM model are $\pm 0.57\ \mathrm{kcal/mol}$ in RMSE, $\pm 0.30\ \mathrm{kcal/mol}$ in MAE, and the Pearson correlation coefficient is $R^2 = 0.96$ while results from the BiGRU model indicate there is no meaningful difference between the two recurrent models. The encoder without BiRNN and attention layers produces much less accurate results, whose error metrics are $\pm 0.77\ \mathrm{kcal/mol}$ in RMSE, $\pm 0.43\ \mathrm{kcal/mol}$ in MAE, and 0.92 in R^2 value, respectively.

We cannot directly compare our results with other ML models because Delfos is the first ML-based study using the MNSOL database. Nonetheless, several studies on aqueous system have previously calculated solubilities or hydration free energies using various ML techniques and molecular descriptors[4, 20, 53, 54, 67, 68]. For comparison, we have tested our neural network model for hydration free energy. A benchmark study of Wu et al. [20] provides hydration energies of 642 small molecules in a group of QSPR/ML models. Their RMSEs were up to $1.15\ \mathrm{kcal/mol}$ while our prediction from the BiLSTM encoder attains $1.19\ \mathrm{kcal/mol}$ for the same dataset and split method. This result suggests our neural network model guarantees considerably good performances even in a specific solvent of water.

Meanwhile, for studies which are not ML-based, there are several results from both classical and quantum-mechanical simulation studies that use the MNSOL as the reference data[1, 2, 44, 45, 69–71]. In Table 2.1, we choose two DFT studies which employ several widely-used QM solvation models[1, 2] for comparison with our proposed ML model: solvation model 8/12 (SM8/SM12), solvation model based on density (SMD), and full/direct conductor-like screening model for realistic solvation (COSMO-RS/D-COSMO-RS). Albeit all of those QM methods exhibited excellent performances given chemical accuracy $1.0\ \mathrm{kcal/mol}$, among the rest, full COSMO-RS is a noteworthy solvation model since it is believed to be the state-of-the-art method which shows the best accuracy[72]. This is realized by statistical thermodynamics treatment on the polarization charge densities, which helps COSMO-RS with making successful predictions even in polar solvents where the key idea of the dielectric continuum solvation collapses[8, 72, 73]. As a result, COSMO-RS calculations with BP86 functional and TZVP basis set achieved $0.52\ \mathrm{kcal/mol}$ for 274 aqueous, $0.41\ \mathrm{kcal/mol}$ for 2,072 organic solvents, and $0.43\ \mathrm{kcal/mol}$ for the full dataset in mean absolute error[2].

For the proposed ML models, Delfos with BiLSTM shows a comparable accuracy in water solvent, which MAE is $0.64\ \mathrm{kcal/mol}$. Delfos makes much better predictions in non-aqueous organic solvents; machine learning for 2121 non-aqueous systems result in 0.24 kcal/mol, which is 44% of SM12CM5 and 59% of COSMO-RS. However, one may argue that K-fold CV from random split does not produce the real prediction accuracy of the model. That is, the random-CV results only indicate the accuracy for *trained* or *practiced* chemical structures. Accordingly, one may ask the following questions. For example, will the ML model ensure the comparable prediction accuracy in "structurally" new compounds? What happens if the ML model couldn't learn sufficiently varied chemical structures? We will discuss these questions in the next section.

2.2.2 Transferability of the Model for New Compounds

Since our study uses techniques of machine learning with empirical data from experimental measures, there is a likelihood that Delfos would not

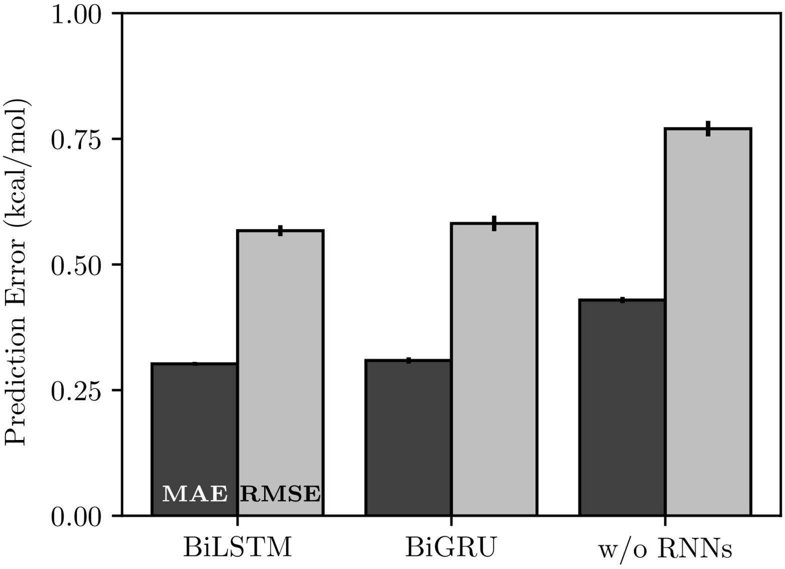


Figure 2.3: Benchmark chart for three kinds of encoder networks, for two metrics (MAE and RMSE). The BiLSTM and the BiGRU models show no significant differences, while it makes relatively inaccurate predictions without recurrent networks. All results are averaged over 9 independent test runs and black lines on tops of boxes denote variances.

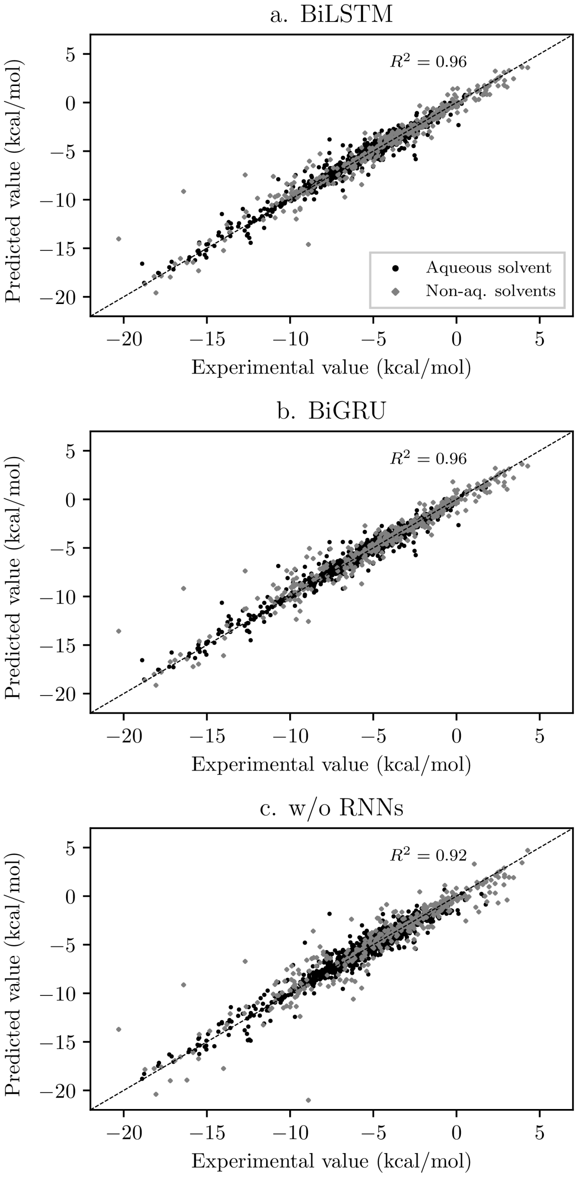


Figure 2.4: Scatter plot for true (x-axis) and ML predicted (y-axis) values of solvation energies in three different models: (a) BiLSTM, (b) BiGRU, and (c) without recurrent layers. All results are averaged over 9 independent 10-fold CV runs.

Solvent	Method	N_{data}	MAE	Ref
Aqueous	SM12CM5/B3LYP/MG3S	374	0.77	[1]
	SM8/M06-2X/6-31G(d)	366	0.89	[1]
	SMD/M05-2X/6-31G(d)	366	0.88	[1]
	COSMO-RS/BP86/TZVP	274	0.52	[2]
	D-COSMO-RS/BP86/TZVP	274	0.94	[2]
	Delfos/BiLSTM	**374**	**0.64**	
	Delfos/BiGRU	**374**	**0.68**	
	Delfos w/o RNNs	**374**	**0.90**	
Non-aqueous	SM12CM5/B3LYP/MG3S	2129	0.54	[1]
	SM8/M06-2X/6-31G(d)	2129	0.61	[1]
	SMD/M05-2X/6-31G(d)	2129	0.67	[1]
	COSMO-RS/BP86/TZVP	2072	0.41	[2]
	D-COSMO-RS/BP86/TZVP	2072	0.62	[2]
	Delfos/BiLSTM	**2121**	**0.24**	
	Delfos/BiGRU	**2121**	**0.24**	
	Delfos w/o RNNs	**2121**	**0.36**	

Table 2.1: Comparisons between encoder-predictor networks and various quantum-mechanical solvation models for aqueous and non-aqueous solutions. The error metric is MAE and $\mathrm{kcal/mol}$. Data in bold texts are our results, while QM results are taken from the work of Marenich et al. [1] and Klamt and Diedenhofen [2].

guarantee prediction accuracy for structurally new solvents or solutes which are not present in the dataset, although the MNSOL contains a considerable number commonly-used solvents and solutes.[62]. In order to investigate this potential issue, we perform another train and test runs with the *cluster cross-validation*[43, 74], instead of using the random-split CV. As a start, we individually obtain 10 clusters for solvents and solutes using the K-mean clustering algorithm and the molecular vector. The molecular vector is a simple summation of substructure vectors as we used for the simple MLP model without RNN encoders[54]: $\mathbf{u} = \sum_i^N \mathbf{x}_i$ for solvents and $\mathbf{v} = \sum_i^M \mathbf{y}_i$ for solutes, respectively. Then, we iteratively perform cross-validation process over each cluster. The size of each cluster is (422, 482, 186, 231, 443, 243, 143, 251, 15, 79) for solvents and (401, 672, 514, 75, 64, 6, 512, 54, 42, 155) for solutes, respectively.

Results from the solvent and the solute cluster CV tasks shown in Table 2.2 exhibit generalized expectation error ranges for new solvents or solutes which are not in the dataset. Winter et al. [43] reported that the split method based on the clustering brings an apparent degradation of prediction performances in various properties; we find that our proposed model exhibits a similar tendency as well. For the BiLSTM encoder model, increments of MAE are $0.52\ \mathrm{kcal/mol}$ for the solvent clustering and $0.69\ \mathrm{kcal/mol}$ for the solute clustering. The reason why the random K-fold CV exhibits superior performances is obvious; if we have a pair ($\mathcal{A}$, $\mathcal{B}$) of solvent $\mathcal{A}$ and solute $\mathcal{B}$ in the test set and the training set have $(\mathcal{A}, \mathcal{C})$ and $(\mathcal{D}, \mathcal{B})$ pairs, then both $(\mathcal{A}, \mathcal{C})$ and $(\mathcal{D}, \mathcal{B})$ could enhance prediction accuracy of $(\mathcal{A}, \mathcal{B})$. However, the clustering limits the location of a specific compound, and pairs of specific solvent or solute should be either in the test set or the train set.

For an additional comparison, Table 2.2 also contains results taken from SMD calculations with semi-empirical methods[45], COSMO, COSMO-RS[2], and classical molecular dynamics[44] for four small organic solvents: toluene ($C_6H_5CH_3$), chloroform ($CHCl_3$), acetonitrile (CH_3CN), and dimethyl sulfoxide ($(CH_3)_2SO$), respectively. Albeit MD is based on classical dynamics, the results of generalized amber force field (GAFF) tells us that an explicit solvation model with a suitable force field could make considerably good predictions. The bottom line of cluster CV is if the dataset for train contains at least one side of the solvent-solvent pair of which we want to estimate the solvation free energy, the expectation error of Delfos lies within chemical accuracy $1.0\ \mathrm{kcal/mol}$, which is the general error of computer simulation scheme. Also, results for four organic solvents demonstrate that predictions from the cluster CV have the accuracy that is comparable with MD simulations using AMOEBA polarizable force field[44].

Results from the cluster CV highlight the necessity for discussion on the importance of database preparation. As described earlier, the cluster CV causes a considerable increase in prediction error, and we suspect that the degradation mainly comes from the decline in the diversity of the training set. Namely, the number of substructures that the neural network learns in training process is not so many as the random CV if we use the cluster CV. To prove this speculation, we define *unique* substructures, which are substructures that only exists in the test cluster. As shown in Figure 2.5, in the solute cluster CV, MAE for 1,226 pairs which do not have any unique substructures in solutes is $0.54\ \mathrm{kcal/mol}$, while the prediction error for the rest 1,269 solutions is $1.64\ \mathrm{kcal/mol}$. The solvent cluster CV shows even more extreme results: the MAE for 374 aqueous solvents is $2.48\ \mathrm{kcal/mol}$, while non-aqueous solvents exhibit $0.52\ \mathrm{kcal/mol}$ in contrast. We believe that the outlying behavior of water is due to its distinctive nature. Water has only one, unique substructure since the oxygen atom does not have any neighbors. So the solvent clustering makes the network unable to learn the structure of water in indirect ways, results in a prediction failure. This logic tells us that the most critical thing in an ML prediction task is securement of the training dataset which contains as many as possible kinds of solvents and solutes. We believe that computational approaches would be as helpful as experimental measures for enriching structural diversity of the training data, given recent advances on QM solvation models[1, 2, 75] such as COSMO-RS. Furthermore, since there are 418 solutes and 91 solvents in the dataset we use[62], which make up 38,038 possible pairs, we expect Delfos and MNSOL would guarantee similar precision levels with the random CV for numerous systems.

Solvent	Method	N_{data}	MAE	RMSE	Ref
All	COSMO/TZVP	2346	2.15	2.57	[2]
	COSMO-RS/TZVP	2346	0.42	0.75	[2]
	SMD/PM6	2500	-	3.6	[45]
	Random CV	**2495**	**0.30**	**0.57**	
	Solvent Clustering	**2495**	**0.82**	**1.45**	
	Solute Clustering	**2495**	**0.99**	**1.61**	
Toluene	MD/GAFF	21	0.48	0.63	[44]
	MD/AMOEBA	21	0.92	1.18	[44]
	COSMO/TZVP	21	2.17	2.71	[2]
	COSMO-RS/TZVP	21	0.27	0.34	[2]
	Solvent Clustering	**21**	**0.66**	**1.10**	
	Solute Clustering	**21**	**0.93**	**1.46**	
Chloroform	MD/GAFF	21	0.92	1.11	[44]
	MD/AMOEBA	21	1.68	1.97	[44]
	COSMO/TZVP	21	1.76	2.12	[2]
	COSMO-RS/TZVP	21	0.50	0.66	[2]
	Solvent Clustering	**21**	**0.78**	**0.87**	
	Solute Clustering	**21**	**1.14**	**1.62**	
Acetonitrile	MD/GAFF	6	0.43	0.52	[44]
	MD/AMOEBA	6	0.73	0.77	[44]
	COSMO/TZVP	6	1.42	1.58	[2]
	COSMO-RS/TZVP	6	0.33	0.38	[2]
	Solvent Clustering	**6**	**0.74**	**0.82**	
	Solute Clustering	**6**	**0.80**	**0.94**	
DMSO	MD/GAFF	6	0.61	0.75	[44]
	MD/AMOEBA	6	1.12	1.21	[44]
	COSMO/TZVP	6	1.31	1.42	[2]
	COSMO-RS/TZVP	6	0.56	0.73	[2]
	Solvent Clustering	**6**	**0.93**	**1.19**	
	Solute Clustering	**6**	**0.91**	**1.11**	

Table 2.2: Prediction accuracy of the random-split CV, the solvent and solute cluster CVs using K-mean algorithm, and several theoretical solvation models for four different organic solvents: toluene ($C_6H_5CH_3$), chloroform ($CHCl_3$), acetonitrile (CH_3CN), and dimethyl sulfoxide ($(CH_3)_2SO$), respectively. Units of MAE and RMSE are $\mathrm{kcal/mol}$.

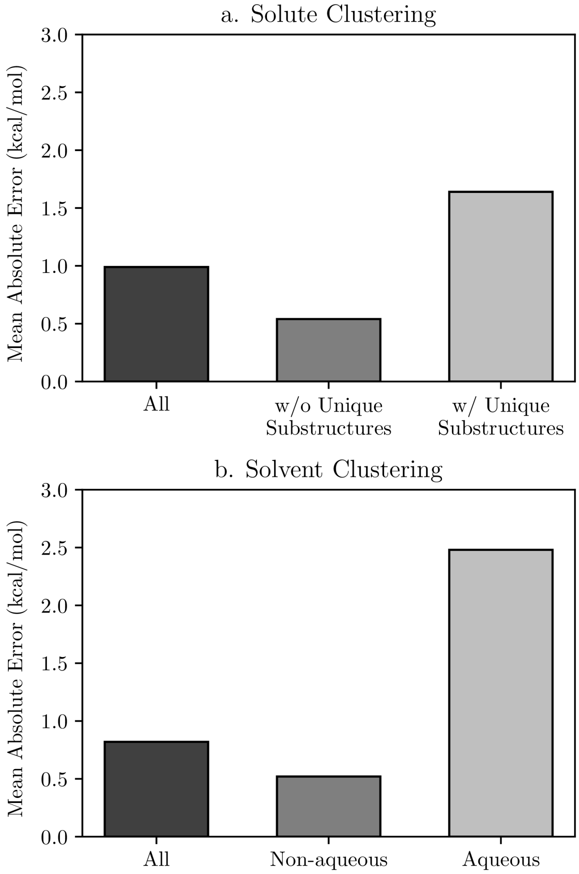


Figure 2.5: Results of cross-validation tasks using K-mean clustering algorithm for (a) solutes and (b) solvents. We conclude that unique substructures in the given compounds are the main cause of the decline in prediction accuracy. Each encoder network includes a BiLSTM layer and we use the same hyperparameters which are optimized in the random CV task.

2.2.3 Visualization of Attention Mechanism

A useful aspect of attention mechanism is that the model provides not only the prediction value of solvation energy of a given input but also a clue to why the neural network makes such a prediction based on the correlations between recurrent hidden states[49, 53, 76]. In this section, we visualize how the attention layer operates, and verify how well such correlations correspond to chemical intuitions for inter-molecular interactions. The matrix of attention alignments, α from Eqn. 2.2a indicates which substructures in the given solvent and solute are strongly correlated with each other so they play dominant roles in determining their solvation energy. In Figure 2.6, we demonstrate attention alignments of nitromethane (CH_3NO_2) solute in four different solvents: 1-octanol ($C_8H_{17}OH$, 3.51 kcal/mol), 1-butanol (C_4H_9OH, 3.93 kcal/mol), ethanol (C_2H_5OH, 4.34 kcal/mol), and acetonitrile (CH_3CN, 5.62 kcal/mol). The scheme for visualizing attention alignments is as follows: (i) first, we calculate the average alignment $\langle\alpha\rangle_j$ of each substructure j of the solute over the entire solvent structure $\{i\}$, $\langle\alpha\rangle_j = \sum_i^N \alpha_{ij}/N$. (ii) Then, we get relative amounts of averaged attention alignments $[\tilde{\alpha_1}, \cdots, \tilde{\alpha_M}]$ from dividing $\langle\alpha\rangle_j$ by the maximum value, $\tilde{\alpha}_j = \langle\alpha\rangle_j / \max(\langle\alpha\rangle_1, \cdots, \langle\alpha\rangle_M)$. (iii) Also, since the embedding algorithm which we use generates two substructure vectors per an atom, we individually visualize two alignments maps, $[\tilde{\alpha}_1, \tilde{\alpha}_3, \cdots, \tilde{\alpha}_{M-1}]$ (for $r = 0$)

and $[\tilde{\alpha}_2, \tilde{\alpha}_4, \cdots, \tilde{\alpha}_M]$ (for $r = 1$) for simpler and more intuitive illustration. (iv) Finally, the color representation of each atom in Fig. 2.6 denotes the amount of $\tilde{\alpha}_j$; the neural network judges that red-colored substructures (higher $\tilde{\alpha}_j$) in the solute are more "similar" to the solvent and the model puts more weights on them during the prediction task. In contrast, green-colored substructures have lower $\tilde{\alpha}_j$, which means they do not have similarity with the solvent molecule so much as red-colored one.

Overall results in Fig. 2.6 imply that the *chemical similarity* taken from the attention layer has a significant connection to fundamental knowledge of chemistry like polarity or hydrophilicity. Each alcoholic solvent has one hydrophilic $-$OH group, and it results in increasing contributions of the nitro group in the solute as hydrocarbon chains of alcohols shorten. For the acetonitrile-nitromethane solution, the attention mechanism reflects the highest contributions of $-NO_2$ groups due to strong polarity and aprotic nature of the solvent. Although the attention mechanism seems to reproduce molecular interactions in a faithful way, however, we find there is a defective prediction which does not agree with chemical knowledge. Two oxygen atoms =O and $-O^-$ in the nitro group are indistinguishable due to the resonance structure, thus they must have equivalent contributions in any solvents, but we find they show different attention scores in our model. We believe those problems happen because the SMILES string of nitromethane (C[N+](=O)[O-]) does not encode the resonance effect in the

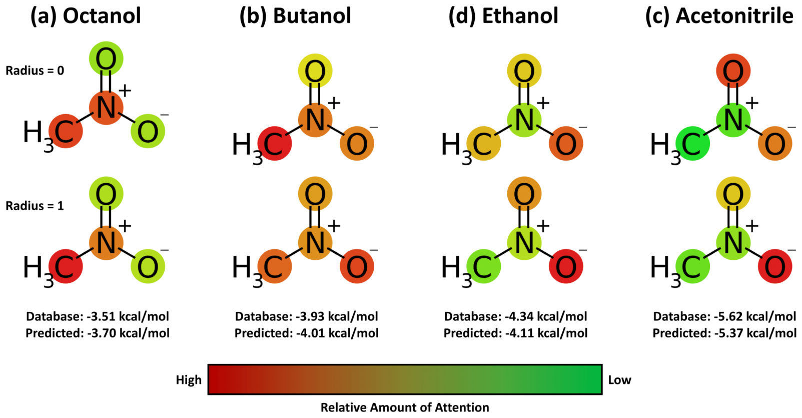


Figure 2.6: Relative and mean attention alignments map for nitromethane in four different solvents: (a) octanol, (b) butanol, (c) ethanol, (d) and acetonitrile, respectively. Color representations denote that the neural network invests more weights on red, while green substructures have relatively low contributions for the solvation energy.

nitro group. Indeed, the Morgan algorithm generates different identifiers for two oxygen atoms in the nitro group, [864942730, 2378779377] for $=\mathrm{O}$ and [864942795, 2378775366] for $-\mathrm{O}^{-}$. The absence of resonance might be a problem worthwhile considering when one intends to use word embedding models with SMILES strings[43, 53, 54], although estimated solvation energies for nitromethane from the BiLSTM model are within a moderate error range as shown in Fig. 2.6.

Chapter 3

Group Contribution Method for the Solvation Energy Estimation with Vector Representations of Atom

3.1 Model Description

3.1.1 Word Embedding

In the proposed work, the primary strategy for the encoding of the input compound's structure is the *word embedding*, mainly inspired by Google's word2vec model[46, 51]. The first attempt of continuous vector representations of human vocabularies in arbitrary space introduced in the mid-1980s[51], however, the remarkable breakthrough has been made by developments of neural network language model (NNLM) and recurrent neural network language model[77] (RNNLM).

The general procedure of word embedding starts from the construction of a one-hot encoded vector $\mathbf{x}(I) = [x_1(I), \cdots, x_V(I)]$ of a given, tokenized input word I, where V is the vocabulary size[46]. By the nature of one-hot encoding, we know the vector $\mathbf{x}$ has only one non-zero element at the corresponding dimension to the given word, $x_I(I) = 1$ and the other elements are 0, in short, $x_i(I) = \delta_{i,I}$. Fig. 3.1 illustrates the embedding procedure when the input context has only one word.

$$\mathbf{h}(I) = \mathbf{x}(I)\mathbf{W}, \tag{3.1a}$$

$$\mathbf{y}(I) = \text{Softmax}(\mathbf{h}(I)\mathbf{W}^{'}). \tag{3.1b}$$

In Eqn. 3.1 and Fig. 3.1, the first fully-connected layer $\mathbf{W}$ forms a $V \times N$ matrix, and the second, $\mathbf{W}^{'}$ is $N \times V$. So the hidden layer (or the *projection layer*) $\mathbf{h}(I)$ has a shape of N-dimensional vector and is identical to the I-th row of $\mathbf{W}$, $\mathbf{w}_I$. The second FC layer calculates the output $\mathbf{y}(I)$, following the equations shown below:

$$\mathbf{h}(I)\mathbf{W}^{'} = \left[\mathbf{w}_1^{'} \cdot \mathbf{w}_I, \cdots, \mathbf{w}_V^{'} \cdot \mathbf{w}_I\right], \tag{3.2a}$$

$$y_i(I) = \frac{\exp(\mathbf{w}_i^{'} \cdot \mathbf{w}_I)}{\sum_{j=1}^{V} \exp(\mathbf{w}_j^{'} \cdot \mathbf{w}_I)}. \tag{3.2b}$$

Each projecting element for the second FC layer in Eqn. 3.2, $\mathbf{w}_j^{'}$ is the j-th column of $\mathbf{W}^{'}$. Both $\mathbf{w}$ and $\mathbf{w}^{'}$ have the same shape, and one can either use them as the N-dimensional embedded vector representation of the input word. Since we train the embedding model as classification tasks with a specific target word T, the conditional probability of finding T given an input I is:

$$P(T|I) = y_J(I). \tag{3.3}$$

The general optimization scheme for the classification model is logistic regression that is maximizing $P(T|I)$ and minimizing the binary cross-entropy loss function.

$$\begin{aligned} L &= -\mathbf{x}(T) \cdot \log \mathbf{y}(I) && \text{(3.4a)} \\ &= -\mathbf{w}_T^{'} \cdot \mathbf{w}_I + \log \sum_{j=1}^{V} \exp(\mathbf{w}_j^{'} \cdot \mathbf{w}_I). && \text{(3.4b)} \end{aligned}$$

Another essential feature of the word embedding is that both the input word and the target word are taken from a single context. That is to say, an embedding model calculates predictivity or co-occurrence between the target word and the input word in a single sentence. This strategy makes the embedding model as an unsupervised machine learning problem, so one can easily enlarge the size of the pre-training dataset. There are two models in Word2Vec: the continuous bag of words (CBOW) model and the skip-gram model. As shown in Fig. 3.2, the CBOW model predicts the central word from its neighboring words; the skip-gram model uses the central word as the input to predict its neighbors. The model complexity of a CBOW model,

Q is dependent on the embedding dimension D, the window length N and the vocabulary size V.

$$Q = D(N + \log_2 V), \tag{3.5}$$

and for a skip-gram model, Q is as follows:

$$Q = ND(1 + \log_2 V). \tag{3.6}$$

The logarithmic dependence on the vocabulary size $\log_2 V$ is originated from the *hierarchical softmax* activation function, which makes it unncessary for the model to update all weights in $\mathbf{W}$ and $\mathbf{W}'$[51].

A number of studies showed that the the unsupervised context learning in the word embedding scheme can also be a powerful tool for encoding structural features of chemical compounds[18, 23, 43, 54]. The idea is realized by the consideration of a given molecular structure as *chemical contexts* of atoms of substructure; positions of projected atomic feature vectors in the embedded vector space now represent their chemical or physical properties, instead of linguistic information. In the present study, we use Mol2Vec embedding model as the primary encoding means[54], which uses the Morgan algorithm to assort atoms in an identical chemical environment and generate the chemical context of a given compound[56].

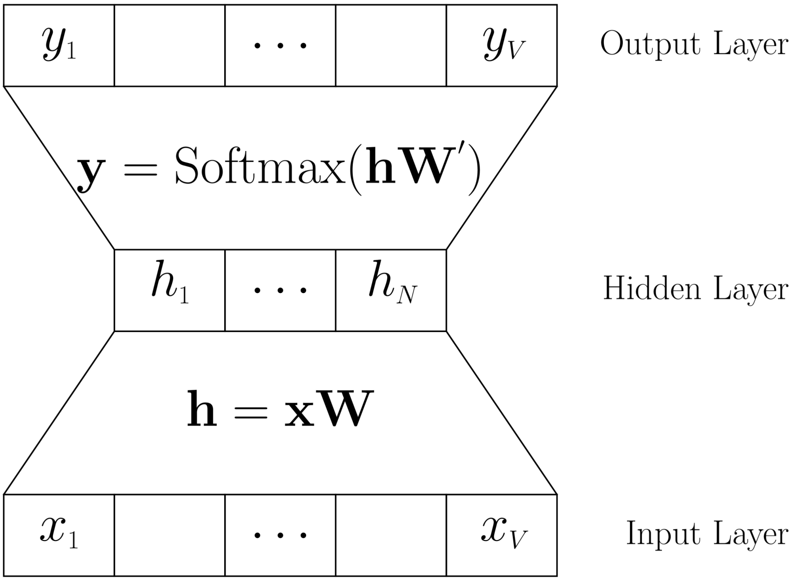


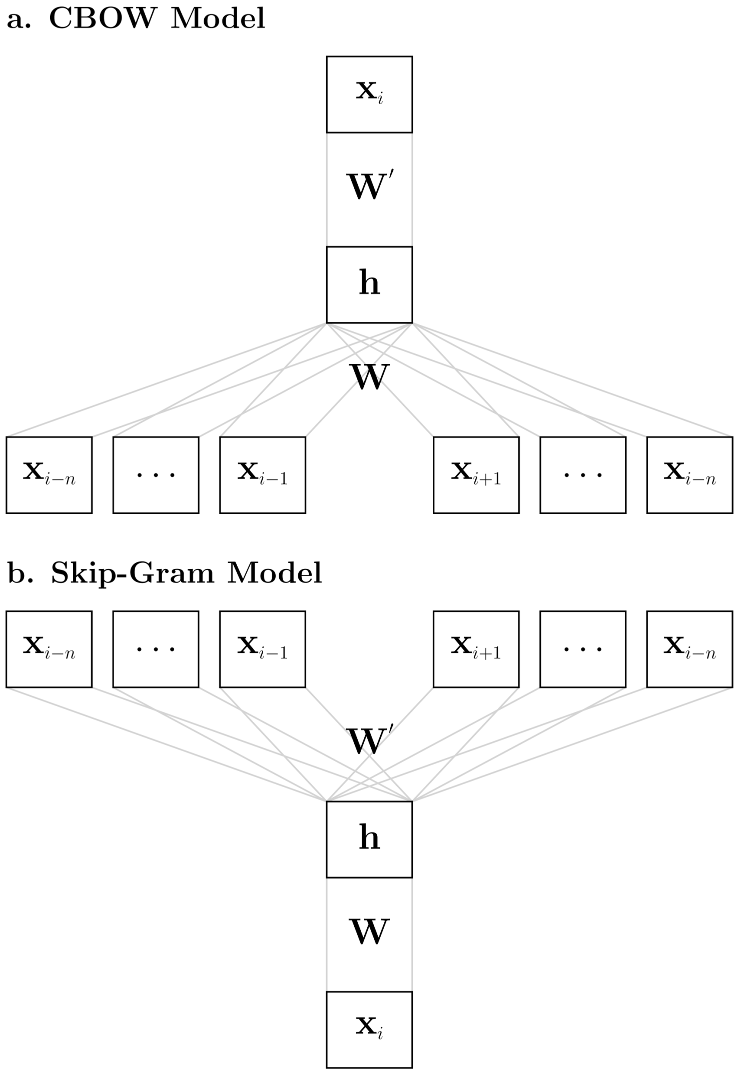
Figure 3.1: Embedding procedure for simple one-word context.

3.1.2 Network Architecture

In the proposed model, the linear regression task between the given chemical structures of the solvent and solute molecules and their solvation free energy starts with embedded vector representations of the given solvent $\mathbf{x}_\alpha$ and solute $\mathbf{y}_\gamma$, where α and γ are atom indices. The entire molecular structure is now can be expressed as a sequence of vectors or a matrix:

$$\mathbf{X} = \{\mathbf{x}_\alpha\}\,, \tag{3.7a}$$

$$\mathbf{Y} = \{\mathbf{y}_\gamma\}\,, \tag{3.7b}$$



a. CBOW Model

b. Skip-Gram Model

Figure 3.2: Model architecture diagrams for (a) the CBOW model and (b) the skip-gram model. The CBOW model predicts the current word based on neighboring words, while the skip-gram words predicts surrounding words from the current word.

so $\mathbf{x}_\alpha$ and $\mathbf{y}_\gamma$ are α-th row of $\mathbf{X}$ and γ-th row of $\mathbf{Y}$, respectively. Then the encoder function learns their chemical structures and extracts feature matrices for the solvent $\mathbf{P}$ and the solute $\mathbf{Q}$.

$$\mathbf{P} = \mathrm{Encoder}(\mathbf{X}), \tag{3.8a}$$

$$\mathbf{Q} = \mathrm{Encoder}(\mathbf{Y}). \tag{3.8b}$$

Columns of $\mathbf{P}$ and $\mathbf{Q}$, p_α and q_γ involve atomistic chemical features of atoms α and γ, which are directly related to the target property, the solvation free energy. We now calculate the un-normalized attention (or *chemical similarity*) between α and γ with on Luong's dot-product attention score function[50]:

$$I_{\alpha\gamma} = -\mathbf{p}_\alpha \cdot \mathbf{q}_\gamma. \tag{3.9}$$

Since our target quantity is the free energies of solvation, we expect such chemical similarity $I_{\alpha\gamma}$ to well correspond to atomistic interactions between α and γ, which involves both the energetic and the entropic contributions. Eventually, the free energy of solvation of the given pair, which is the final regression target, is given as a simple summation of atomistic interactions:

$$\Delta G^\circ_{sol} = \sum_{\alpha\gamma} I_{\alpha\gamma}. \tag{3.10}$$

Certainly, one can also calculate the free energies of solvation from two molecular feature vectors, those are representing the solvent properties $\mathbf{u}$ and the solute properties $\mathbf{v}$, respectively:

$$\Delta G^{\circ}_{sol} = \mathbf{u} \cdot \mathbf{v} = \left(\sum_{\alpha} \mathbf{p}_{\alpha} \right) \cdot \left(\sum_{\alpha} \mathbf{q}_{\alpha} \right). \tag{3.11}$$

The inner-product relation between molecular feature vectors $\mathbf{u}$ and $\mathbf{v}$ has a formal analogy with the solvent-gas partition coefficient calculation method via the solvation descriptor approach, which is founded by Abraham and Acree[78, 79]:

$$\log K = c + eE + sS + aA + bB + lL. \tag{3.12}$$

In Eqn. 3.12, the solute descriptor $(1, E, S, A, B, L)$ is determined from a series of experimental measures, and the solvent descriptor (c, e, s, a, b, l) is a fitted value. In our proposed model, both $\mathbf{u}$ and $\mathbf{v}$ are purely fitted quantities from the scratch, with the skip-gram pre-training and the linear regression analysis.

We choose and compare two different neural network models in order to encode the input molecular structure and extract important structural or chemical features which are strongly related to solvation behavior: one is bidirectional language model (BiLM)[80] based on the recurrent neural network (RNN), the other is the graph convolutional neural network (GCN)[81] which explicitly handles the connectivity (bonding) between atoms with the adjacency matrix.

The detailed mathematical expressions of the bidirectional language model are given below[80]:

$$\overrightarrow{\mathbf{H}}^{(i+1)} = \overrightarrow{\mathrm{RNN}}(\overrightarrow{\mathbf{H}}^{(i)}), \tag{3.13a}$$

$$\overleftarrow{\mathbf{H}}^{(i+1)} = \overleftarrow{\mathrm{RNN}}(\overleftarrow{\mathbf{H}}^{(i)}). \tag{3.13b}$$

In Eqn. 3.13, the right-headed arrow in $\overrightarrow{\mathrm{RNN}}$ denotes a forward-directed recurrent unit which propagates from the leftmost of the sequence to the rightmost one. The BiLM also involves the backward-directed recurrent neural network ($\overleftarrow{\mathrm{RNN}}$) and it propagates from the rightmost to the leftmost. The superscript (i) in hidden layers $\mathbf{H}^{(i)}$ denotes the position at the stacked configuration: at the first stack, both forward-directed and backward-directed RNN share the pre-trained sequence $\mathbf{X}$ as an input, $\overrightarrow{\mathbf{H}}^{(0)} = \overleftarrow{\mathbf{H}}^{(0)} = \mathbf{X}$. In addition, use of more improved versions of RNNs, e.g. the gated recurrent unit (GRU)[61] or the long-short term memory (LSTM)[60], are more suitable when one considers cumulated numerical errors due to the deep-structured nature of RNNs[59],

$$\mathbf{H}^{(i)} = \overrightarrow{\mathbf{H}}^{(i)} + \overleftarrow{\mathbf{H}}^{(i)}. \tag{3.14}$$

Hidden layers from the forward and backward RNNs are then merged into a single sequence, as described in Eq. 3.14. Finally, we obtain the sequence of chemical feature vectors of the α-th atom in the given solvent with weighted summation of rnn stacks,

$$\mathbf{P} = \sum_i c_i \mathbf{H}^{(i)}. \tag{3.15}$$

The encoder function for solutes has an identical neural network architecture, which converts the pre-trained solute sequence $\mathbf{Y}$ into the feature sequence $\mathbf{Q}$.

To sum up, the BiLM encoder considers a given molecule as just a simple sequence of atomic vector representations. The idea is quite clear and rather straightfoward for implementation of the neural network. However, this idea may causes "problems" in more complex compounds due to the lack of intramolecular bonding information between atoms. We also consider the graph convolutional neural network (GCN), which is one of the most well-known algorithms in chemical applications of neural networks[34, 81]. The GCN model represents the input molecule as a mathematical graph, instead of a simple sequence: each node corresponds to the atom, and each edge in the adjacency matrix $\mathbf{A}$ involves connectivity (or existence of bonding) between atoms:

$$\mathbf{H}^{(i+1)} = \mathrm{GCN}(\mathbf{H}^{(i)}, \mathbf{A}). \tag{3.16}$$

The role of adjacency matrix in the GCN constrains convolution filters to the node and its nearest neighbors. Eqn. 3.17 describes a more detailed mathematical expression of the skip-connected GCN[81]

$$\mathrm{GCN}(\mathbf{H}, \tilde{\mathbf{A}}) = \sigma(\tilde{\mathbf{A}}\mathbf{H}\mathbf{W}_1 + \mathbf{H}\mathbf{W}_2 + \mathbf{b}), \tag{3.17}$$

where $\mathbf{W}_1$ and $\mathbf{W}_2$ are convolution filters, $\mathbf{b}$ is the bias vector, and σ denotes the activation function - we choose the hyperbolic tangent in the proposed model. The GCN encoder also invloves stacked structure, and we can obtain the feature sequence for each molecule with the same manner as described in Eqn. 3.15.

3.2 Results and Discussions

3.2.1 Computational Details

For the training and test tasks of the proposed neural network, we prepare 6,594 experimental measures of free energies of solvation for 952 organic solvents and 147 organic solutes, including some inert gases. 642 experimental measures for free energies of hydration are taken from the FreeSolv

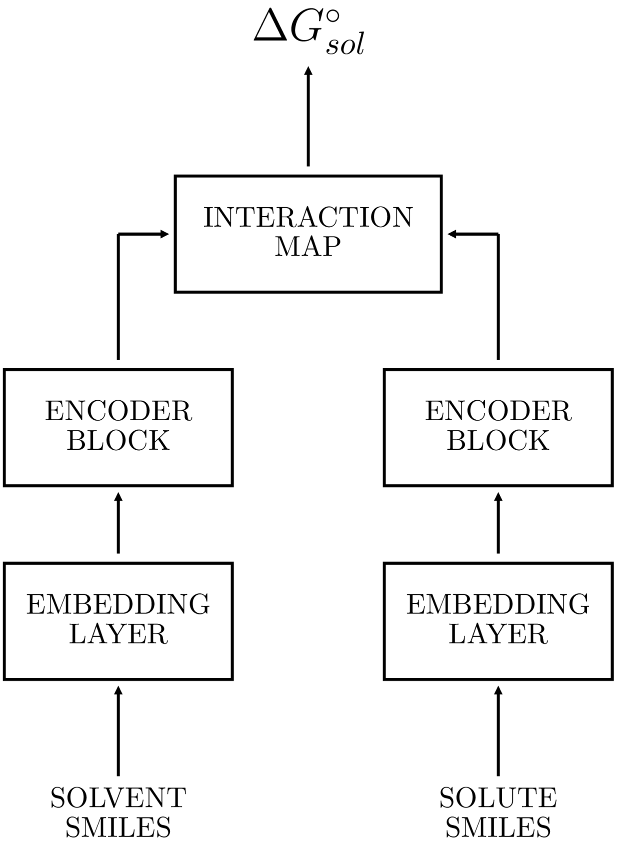


Figure 3.3: Architecture of the proposed model. Each encoder network extracts atomistic feature vectors given pre-trained vector representations, and the interaction map calculates pairwise atomistic interactions.

database[14],and 5,952 data points for non-aqueous solvents are collected with the Solv@TUM database version 1.0[78, 79], which is available at https://github.com/hille721/solvatum. Compounds in the dataset involves 10 kinds of atoms, which are commonly used in organic chemistry: hydrogen (H), carbon (C), oxygen (O), sulfur (S), nitrogen (N), phosphorus (P), fluorine (F), chlorine (Cl), bromine (Br), and iodine (I). The maximum heavy-atom count is 28 for solutes and 18 for solvents.

For the very first stage, we perform the skip-gram pre-training process for 10,229,472 organic compounds, which are collected from the ZINC15 database[82], using Gensim 3.8.1 and Mol2Vec skip-gram model to construct the 128-dimensional embedding lookup table[54]. For the implementation of the neural network model, we mainly use the Tensorflow 2.0 and Keras 2.3.1 frameworks[65]. To construct the BiLM encoder, we both consider CuDNN implementations[65] for the LSTM and the GRU, which are basic layers in the Tensorflow. For GCN encoder, we use codes taken from Spektral library version 0.1.1, which implements the skip-connected graph convolutional network. Each model has L2 regularization to prevent excessive changes on weights and minimize the variance and uses the RMSprop algorithm with 10^{-3} of learning rate and $\rho = 0.9$ for optimizing its loss function, the mean squared error (MSE).

We employ 5-fold cross-validation to evaluate the prediction accuracy of the chosen model; the entire dataset is randomly split into five uniform-sized subsets, and we iteratively choose one of the subsets as a test set, and the training run uses the remainder 4 subsets. Consequentially, a 5-fold CV task performs 5 independent training and test runs, and relative sizes of the training and test sets are 8 to 2. To minimize the variation of results from CV tasks, we take averages for all results over 9 independent random CV, split from different random states. The procedure for CV is implemented with the Scikit-Learn library version 0.2.2[66].

3.2.2 Prediction Accuracy

The selection of the optimized model for the target property is realized by an extensive grid-search task for tuning model hyperparameters. First, we choose 32 as the batch size, and RMSprop as an optimization algorithm with learning rate is 10^{-3}. It is generally known that the smaller batch size generates a better result; however, a too small batch size is computationally inefficient, so we take the value of 32 as the point of compromise between the prediction performance and the computational efficiency. Table 3.2.2 shows additional searching information for the optimized stack size of the encoder networks and maximum epochs are 50 for the BiLM model and 100 for the GCN model, respectively. Fig. 3.4 shows epoch-evolution of training and validation loss for both the BiLM/LSTM encoder and the GCN encoder, where optimized stack size is 3. BiLM encoder shows a much faster convergence behavior untill $\sim$ 50 epochs and overfitting appears, while the

GCN encoder exhibits minimum validation loss around ~ 100 epochs.

The results for test run using 5-fold CV tasks for the optimized models with grid search tasks are shown at Fig. 3.5. We found that the BiLM encoder with the LSTM layer performs slightly better than the GCN encoder, although their differences are not pronounced: the mean unsigned prediction error (MUE) for the BiLM/LSTM encoder model is 0.19 $\mathrm{kcal/mol}$, while the GCN model results in 0.23 $\mathrm{kcal/mol}$. Both MUE values show that the our proposed mechanism is actually working and guarentees excellent prediction accuracies for well-trained chemical structures. Moreover, since we use a simple version of the graph-based neural network as the encoder, we might expect the GCN-based model to perform better than a simple graph-based embedding model or more progressed version of graph neural networks to perform even better for chemical structures: such as the messege-passing neural network (MPNN)[35], the deep tensor neural network (DTNN)[36], and so on.

As the last of this section, we confirm whether or not the proposed neural network architecture is working as we designed. Fig. 3.6 presents t-SNE visualizations for pre-trained solute vectors $\mathbf{y}$ and encoded molecular feature $\mathbf{v}$[38]. Color codes denote predicted hydration free energies for 15,432 points, whose structures are randomly taken from the ZINC15[82]; red dots correpond to the compounds with low hydration free energies while the blue dots correspond to them with high hydration free energies. The correlation

Encoder	Stack	Training	Validation	Test
		RMSE	RMSE	RMSE
	1	0.29 ± 0.00	0.59 ± 0.04	
	2	0.24 ± 0.01	0.44 ± 0.04	
BiLM	3	$\mathbf{0.24 \pm 0.01}$	$\mathbf{0.43 \pm 0.02}$	$\mathbf{0.41 \pm 0.01}$
	4	0.23 ± 0.00	0.49 ± 0.03	
	5	0.20 ± 0.02	0.52 ± 0.02	
	1	0.34 ± 0.00	0.73 ± 0.04	
	2	0.26 ± 0.00	0.70 ± 0.07	
GCN	3	0.25 ± 0.00	0.51 ± 0.08	
	4	$\mathbf{0.26 \pm 0.01}$	$\mathbf{0.46 \pm 0.05}$	$\mathbf{0.44 \pm 0.01}$
	5	0.27 ± 0.01	0.77 ± 0.16	

Table 3.1: Error metrics for training, validation, and test runs with respects to the number of stacked encoder layers. The units of all errors are $\mathrm{kcal/mol}$.

between molecular features and predicted free energies is a clear clue that the model architecture can extract geometrical correlations and calculate free energy. Meanwhile, the pre-trained solute vectors from the skip-gram embedding model exhibit only weak correlations.

3.2.3 Model Transferability

Since our proposed neural network model is a solvent-non-specific one that considers both the solvent structure and the solute structure as seperate inputs, it has a distinct character when compared to the other solvent-specific ML models. The model can train with the structure of a single solute repeatedly when the solute has multiple solvation energy data for different kinds of solvents[22]; this logic is also valid for a single solvent. Therefore, one of the most useful advantages of our model is that we can easily enlarge the

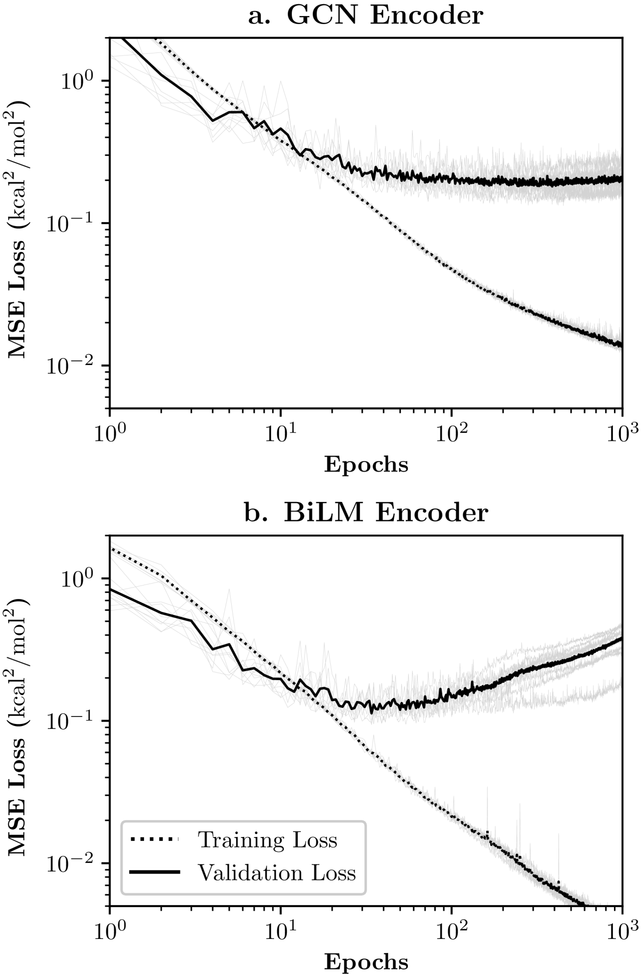


Figure 3.4: Epoch-evolution of mean squared loss functions (RMSE) for (a) the GCN encoder model and (b) the BiLM encoder model. Solid lines denote evolution of training losses while dotted lines denote validation losses. All results are averaged over 8 independent cross-validation runs.

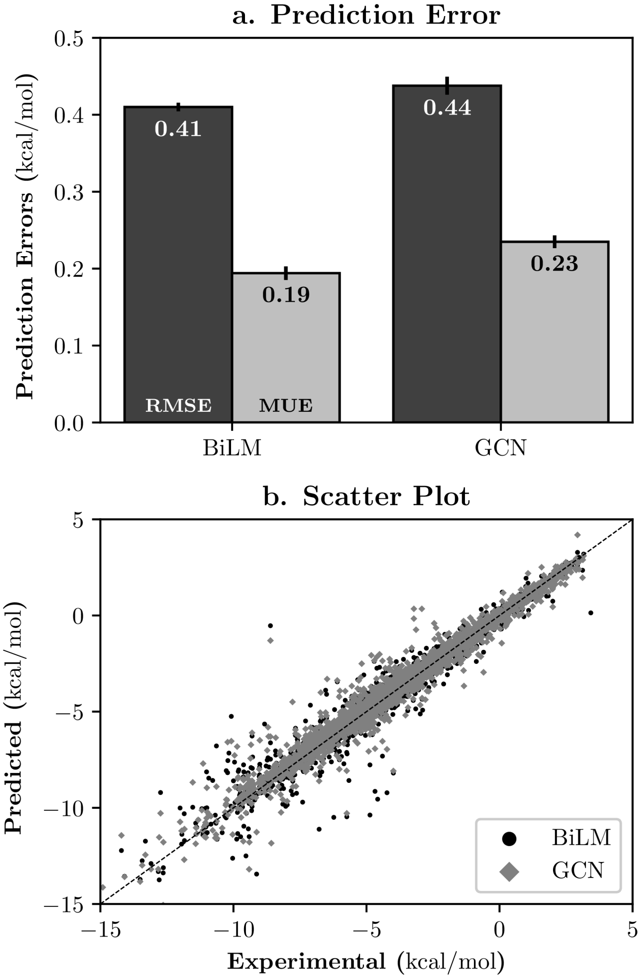


Figure 3.5: (a) Prediction erros for two models in kcal/mol, taken from 5-fold cross validation results. (b) Scatter plot between the experimental value and ML the ML predicted value. Black circles denote the BiLM model while the GCN results are shown in gray diamonds.

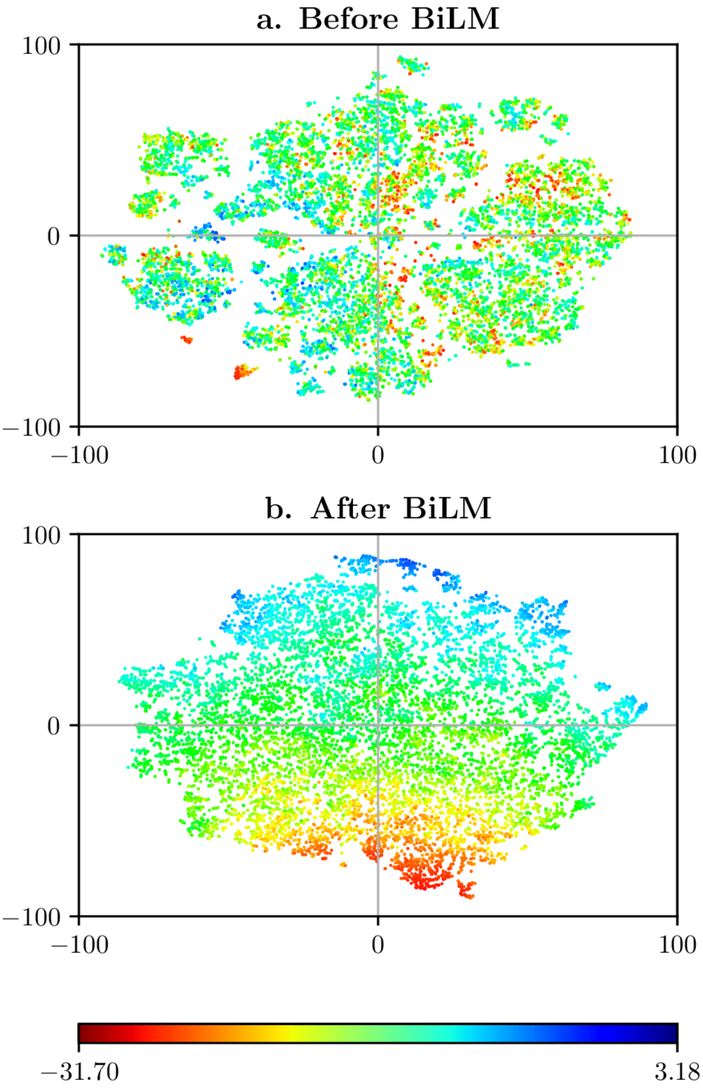


Figure 3.6: 2-dimensional visualizations on (a) the pre-trained vector $\sum_\gamma \mathbf{y}_\gamma$ and (b) the molecular feature vector $\mathbf{v}$ for 15,432 solutes. We reduce the dimension of each vector with the t-SNE algorithm. The color representation denotes the hydration energy of each point.

dataset for training, even in the scenario that we want to predict solvation free energies for a specific solvent. Fig. 3.7 shows 5-fold cv results for 642 hydration free energies (FreeSolv) from both the BiLM and the GCN models, in two different situations. One uses only the FreeSolv[14] database for train and test tasks, and the other additionally uses the Solv@TUM[78, 79]. Although the Solv@TUM database only involves non-aqueous data points, it enhances each model's accuracy by about 20% (BiLM) to 30% (GCN) in terms of mean unsigned errors. Those results imply that there are possible applications of the transfer learning to other solvation-related properties, like aqueous solubilities[4] or octanol-water partition coefficients.

However, in some other situations, the advantage we discussed above might be a downside: the repetitive training for a single compound may make the model tends to overfit, and they could weaken predictivity for the structurally new compound, which is considered as an extrapolation. We investigate the model's predictivity for extrapolation situations with the *scaffold-based* split[22, 35, 43]. Instead of the ordinary K-fold CV task with the random and uniform split method, the K-means clustering algorithm builds each fold with the MACCS substructural fingerprint. One can simulate an extreme extrapolation situation through CV tasks over the clustered fold. As shown in Fig. 3.8, albeit the scaffold-based split degrades MUEs by a factor of three, they are still within an acceptable error range ~ 0.6 kcal/mol, given chemical accuracy 1.0 kcal/mol. Furthermore, we

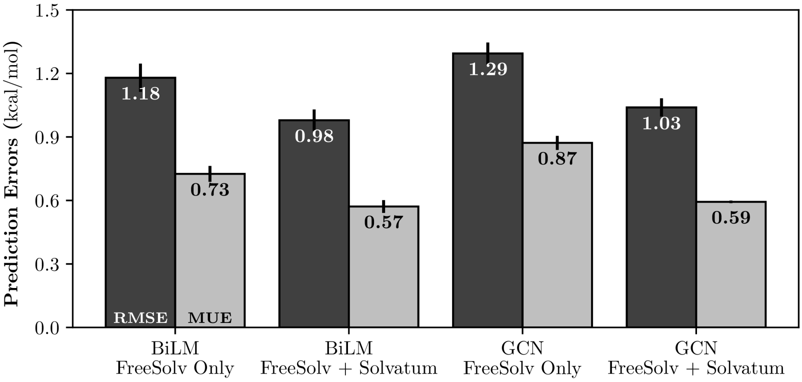


Figure 3.7: CV-results for FreeSolv hydration energies with two different training dataset selection. Deep-colored boxes denote CV results with the augmented dataset with the Solv@TUM database.

do not see any clear evidence that our model tends to overfit more than other solvent-specific models[35, 43].

3.2.4 Group Contributions of Solvation Energy

Although we showed that the proposed NN model guarantees an excellent predictivity for solvation energies of various solute and solvent pairs, the main objective of the present study is obtaining the solvation free energy as the sum of decomposed inter-atomic interactions, as we described at Eq. 3.9 and 3.10. In order to verify whether or not the the model's solvation energy estimation has correspondence to group-contribution based calculation, we define the sum of atomic interactions $\mathbf{I}_{\alpha\gamma}$ over the solvent indices γ as the

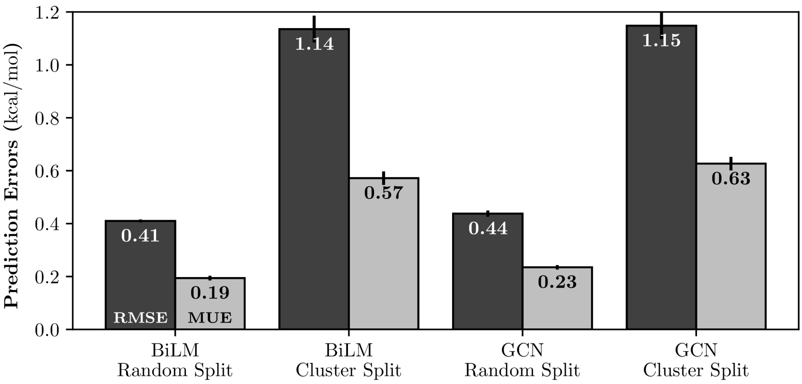


Figure 3.8: Comparison between CV results with the random-split and the scaffold-based split (or cluster split).

group contributions of the α-th solute atom:

$$\mathbf{I}_{\alpha} = \sum_{\gamma} \mathbf{I}_{\alpha\gamma}. \tag{3.18}$$

Figure 3.9 shows hydration free energy contributions for four linear and small organic solutes which have six heavy atoms: n-hexane (CCCCCC), 1-chloropentane (CCCCCCl), pentaldehyde (CCCCC=O), and 1-aminopentane (CCCCCN). As shown in Fig. 3.9, both the BiLM and the GCN model exhibit a resembling tendency in group contributions; the model estimates that atomic interactions between the solute atoms and water increases near the hydrophilic groups. Although the results show that we can find a significant correspondence to intuitive chemical knowledge, it might need further quantified analysis of computer simulation approaches. For example, molecular dynamics simulations with an appropriate explicit solvation model. The Kirkwood charging formula can give atomic free energy contributions with pairwise interactions $u(\mathbf{r}, \lambda)$ and the solvation shell structure $g(\mathbf{r}, \lambda)$[10]:

$$\mu = \rho \int_0^1 d\lambda \int d\mathbf{r} g(\mathbf{r}, \lambda) \frac{\partial u(\mathbf{r}, \lambda)}{\partial \lambda}. \tag{3.19}$$

However, there is an aspect that we can easily verify without quantitative computer simulations. It is obvious that each atom in cyclohexane and benzene must have identical contributions to the free energy, but the results in Fig. 3.10 clearly shows that the BiLM model makes faulty predictions while the GCN model works well as expected. We believe that this malfunctioning of the BiLM model originates from the sequential nature of the recurrent neural network. Since the RNN considers the input molecule is just a simple sequence of atomic vectors and there are no explicit statements that involve bonding information, the model could not be aware of the cyclic shape of the input compound[23, 34]. We conclude that it is inevitable to use explicitly bond (or connectivity) information when one constructs a group-contribution based ML model, although the RNN-based model well predicts in terms of their sum.

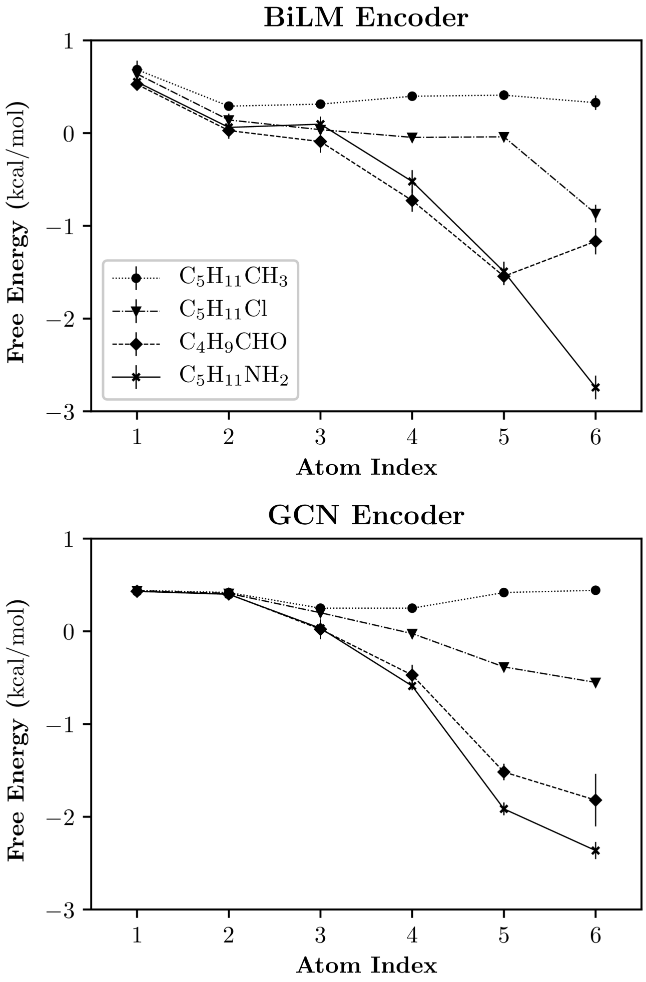


Figure 3.9: ML-calculated atomistic group contributions for four small, linear organic molecules which have six heavy atoms. The atom index starts from the leftmost of the given molecule and only counts heavy atoms.

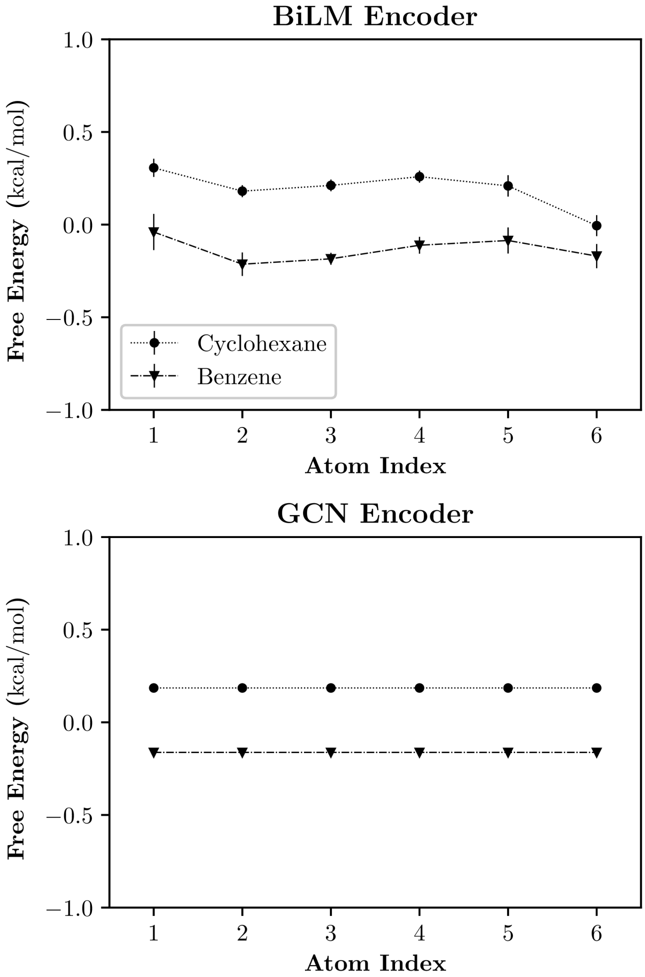


Figure 3.10: Group contributions for two simple cyclic compounds: cyclohexane and benzene.

Chapter 4

Empirical Structure-Property Relationship Model for Liquid Transport Properties

In this chapter, we present a simple structure-property relationship estimation procedure for two major transport properties of the liquid state: the dynamic viscosity (η) and the dielectric constant (ϵ).

Computer simulation approaches for the calculation of transport properties are not easily feasible since they are non-equilibrium measures which are depending on the external field: shear stress (viscosity) and electric field (dielectric constant). Generally, the calculation of transport property via equilibrium simulation needs to generate multiple molecular dynamics trajectories to evaluate the Green-Kubo relation, which is the exact mathematical expression for transport coefficients in the linear response regime[83]:

$$\gamma = \int_0^\infty d\tau \left\langle A(0)A(\tau) \right\rangle . \tag{4.1}$$

Eqn. 4.1 calculates the given transport coefficient γ with the time integration of a specific time correlation function. At high-viscous liquids, it is difficult to sample trajectories and calculate the Green-Kubo relation due to extremely slow relaxation of the liquid system.

In previous chapters, we showed that the structure-property relationship could be a powerful tool for the prediction of the free energy of solvation. Here, we seek another application of SPR estimation of non-equilibrium transport properties, which might be applicable in many systems - even in viscous liquids. The basis of the present SPR model is the decision-tree regression model; the model generates tree-like graphs of decision rules and learns the training database[84]. Also, we employ two *ensemble methods*, the *random forest*[85] (RF) and the *gradient boosting*[86] (GBM) algorithms to minimize bias and variance of the tree-based machine learning model.

The mathematical expression of the ensemble method starts with the mathematical function F of a regression model an input descriptor $\mathbf{x}$ to its label y[86]:

$$\hat{y}_i = F(\mathbf{x}_i; \mathbf{P}), \tag{4.2}$$

where $\mathbf{P}$ is the collection of trainable parameters of the function F and $\hat{y}$ is the predicted value of the model, given input descriptor $\mathbf{x}$. The linear regression task loss function $L(y_i, F(\mathbf{x}_i)) = (y_i - \hat{y}_i)^2$ by the least-square method.

$$\mathbf{P}^* = \arg\min_{\mathbf{P}} \sum_i L(y_i, F(\mathbf{x}_i; \mathbf{P})). \tag{4.3}$$

A random forest regression model involves a set of independent, randomly generated decision-tree *subpredictors* $\{F_1(\mathbf{x}; \mathbf{P}_1), \cdots, F_K(\mathbf{x}; \mathbf{P}_K)\}$, and one can get the optimized model from the ensemble average over K “weakly-optimized” subpredictors[85].

$$\mathbb{F}(\mathbf{x}_i) = \sum_{k=1}^{K} F_k(\mathbf{x}_i; \mathbf{P}_k^*). \tag{4.4}$$

If the model is a classification problem, each subpredictor casts a unit vote for the selection of the most popular class.

The gradient boosting algorithm takes a different approach to the RF model. It has an analogy with the RF that the model consists a set of subpredictors, however, instead of the ensemble average over subpredictors, a GBM model updates its prediction model F_k via the sequential iteration task and chooses the last model F_K^* as the optimized model[86]:

$$F_{k+1}^*(\mathbf{x}) = F_k^*(\mathbf{x}) + h_k(\mathbf{x}). \tag{4.5}$$

Here, we fit the *base learner* h_k with *pseudo-residuals* $\{r_{ik}\}$:

$$r_{ik} = -\left[\frac{\partial L(y_i, F(\mathbf{x}_i))}{\partial F(\mathbf{x}_i)}\right]_{F(\mathbf{x})=F_{k-1}^*(\mathbf{x})}. \tag{4.6}$$

At the very first stage, the initial model F_1^* is equivalent with Eqn. 4.3.

We perform an extensive searching task over tens of elementary structural properties and choose the collection of 19 values, which are shown in Table. 4.1, as the optimized molecular descriptor for liquid transport properties. All properties are available in RDkit 2019.09 python module, and their evaluation process does not require additional simulations or theoretical calculations. For the train and validation tasks, we collect 1,375 experimental data for the liquid dynamic viscosity and the relative permittivity (the dielectric constant) from the web version of DIPPR 801 database[87]. The two decision-tree based ensemble models are implemented using Scikit-learn 0.22[66] and XGBoost 0.90 libraries.

We optimize the hyperparameters and evaluate the predictivity of two models for two transport properties using the 5-fold cross-validation task. The optimized RF model's maximum tree depth is 8, while the GBM model has 6 maximum nodes; both models have the same number of estimators, 100. Fig. 4.1 shows scatter plots between experimental values (x-axis) and predicted values (y-axis). We also specify the Pearson correlation coefficient in order to indicate the prediction accuracy of each model. The GBM model shows better accuracy: R^2 values are 0.91 for the dynamic viscosity and 0.81 for the dielectric constant in the logarithmic scale, respectively. while the RF model shows $R^2 = 0.89$ for the viscosity and 0.78 for the dielectric constant, respectively.

No.	Property	Unit
1	Molecular weight	A. U.
2	Heavy atom weight	A. U.
3	Maximum partial charge	e
4	Minimum partial charge	e
5	Fraction of sp^3 carbons	-
6	Labute accessible surface area	Å^2
7	Topological polar surface area	Å^2
8	Number of aliphatic carbocycles	-
9	Number of aliphatic heterocycles	-
10	Number of aromatic carbocycles	-
11	Number of aromatic heterocycles	-
12	Number of saturated carbocycles	-
13	Number of saturated heterocycles	-
14	Number of stereo centers	-
15	Number of hydrogen bond acceptor	-
16	Number of hydrogen bond donor	-
17	Number of Lipinski hydrogen bond acceptor	-
18	Number of Lipinski hydrogen bond donor	-
19	Number of heteroatoms	-

Table 4.1: Collection of 19 elementary structural properties for the description of a given organic molecule. All properties are available in RDKit python module.

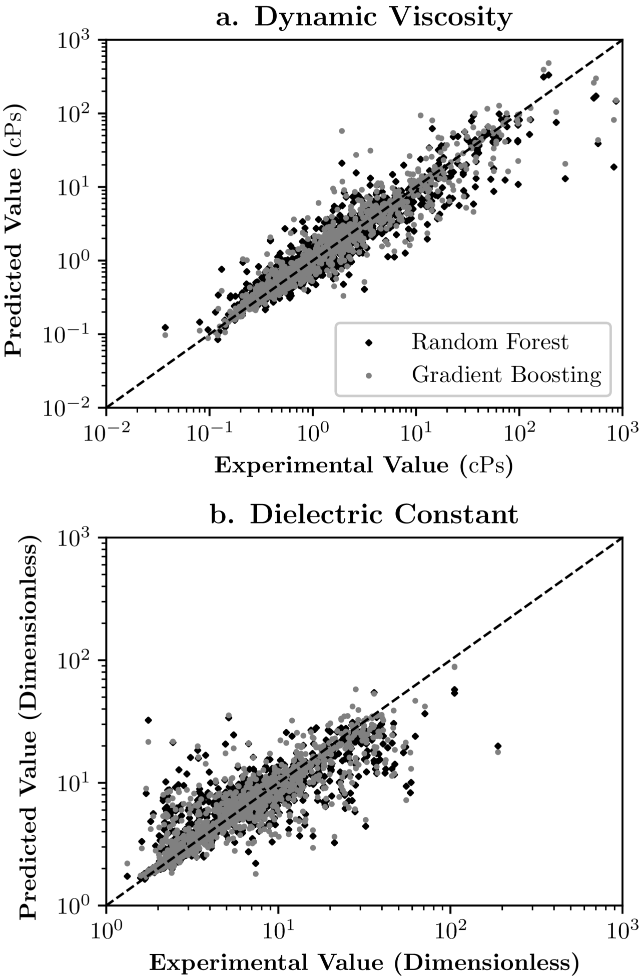


Figure 4.1: Scatter plots for (a) the dynamic viscosity and (b) the dielectric constant, respectively. ML predictions are obtained using 5-fold cross-validation tasks over 1,375 data points, which are taken from the DIPPR 801 database.

Chapter 5

Concluding Remarks

In the present study, we introduced a new approach for the solvation energy prediction, which has a great potential to provide physicochemical insights on the solvation process. The novelty of our neural network model is that the model does not involve the perceptron networks for readout of encoded features and estimation of the target property. Alternatively, we designed the model such that it is possible to calculate pairwise atomic interactions from inner products of atomistic feature vectors[50]. As a result, the model produces the solvation free energy from the group-contribution based prediction.

In Chapter 2, we reviewed our previous ML solvation model, Delfos. The extensive calculations on 2495 experimental values[62] demonstrate that Delfos exhibits excellent prediction accuracy, which is comparable with several well-known QM solvation models[1, 2] when the neural network is trained with sufficiently varied chemical structures. Decline in performances about 0.5 to $0.7\ \mathrm{kcal/mol}$ at the cluster CV tasks represents the accuracy for a structurally new compound, suggesting the importance of preparation of the ML databases even though Delfos still demonstrates comparable predictions with some theoretical approaches such as MD with AMOEBA force field[44] or DFT with pure COSMO[2]. The score matrix taken from the attention mechanism gives us an interaction map between atoms and substructure; our model does provide not only a simple estimation of target property but offers important pieces of information about which substructures play a dominant role in solvation processes.

In Chapter 3, we introduced a new model for the solvation energy estimation and quantified the proposed model's prediction performances for 6,493 experimental data points of solvation energies, which were taken from the FreeSolv[14] and Solv@TUM database[37, 79]. We found a significant geometrical correlation between molecular feature vectors and predicted properties, which implies that the proposed model is actually working as we designed. The estimated prediction MUEs from K-fold CV are $0.19\ \mathrm{kcal/mol}$ for the BiLM encoder and $0.23\ \mathrm{kcal/mol}$ for the GCN model, respectively.

The K-fold CV results from the scaffold-based split[43] showed the prediction accuracy decreases by a factor of three in extreme extrapolation situations, but they still exhibit moderate performances, which were $0.60\ \mathrm{kcal/mol}$. Moreover, we found that the solvent-non-specific structure of the proposed model is appropriate for enlarging dataset size, that is to say, experimental data points for a particular solvent is transferable to other solvents; we conclude that this transferability is the reason for our model's outstanding predictivity[22].

Finally, we examined pairwise atomic interactions that are obtained from the interaction map $\mathbf{I}$ and found a clear tendency between hydrophilic groups and their contributions to the hydration free energy. However, the BiLM model with the recurrent network has some faulty aspects in symmetric or cyclic compounds, albeit it showed better predictions in terms of the total solvation energy. This fact implies the sequential nature of the recurrent network is inappropriate for constructing a group-contribution model, and an explicit usage of the chemical bonding information is inevitable. Although our results need an extra investigation from a quantitative point of view[10], we believe that our model can provide detailed information on the solvation mechanism, not only the predicted value of the target property.

Appendix A

Analyzing Kinetic Trapping as a First-Order Dynamical Phase Transition in the Ensemble of Stochastic Trajectories

A.1 Introduction

Self-assembly is the spontaneous process of disordered components to form ordered patterns or structures. It is one of the most extensively studied research area for complex systems[88–95]. Physical interactions between components play a major role in the self-assembly process. Strength and specificity of the interactions induce the assembling process and determine their assembled structure in the equilibrium condition. However, an obstacle due to an energetic and/or entropic barrier makes it difficult for the system to relax via the reversible dynamics, which hinders the formation of desired assembly structure. The irreversible behavior in bond making and breaking will hinder misbounded components to adjust their bonds easily[91, 96]. Occasionally the system will get trapped in the meta-stable glassy state instead of its equilibrium structure. This behavior is usually called *kinetic trapping*. There have been numerous works in computer simulation studies[97–104] in order to avoid kinetic trapping and achieve effective assembled structure.

A molecular dynamics study of viral capsid growth reported the importance of reversibility and interaction strength in self-assembly at submicroscopic scale[105]. In the work, the authors inspected the time evolution of the cluster size distributions and argued excessive early growth makes monomers trapped in the imperfect shell, resulting in a shortage of free monomer. Analyzing the fluctuation-dissipation ratio (FDR) is another useful strategy for analyzing reversibility[102]. The correlation-response relation showed that the system is in short-time quasi-equilibrium states and reversible in that time scale when the system shows a good assembly kinetics. A notable advance is demonstrained from the direct measurements of bond making and breaking events[99, 103]. In Refs. 99 and 103, the authors defined the *flux* and the *traffic*, which represents the net rate of bond making and total events time scale, respectively. These two quantities give us knowledge of the microscopic reversible behavior of bond-making and breaking progress.

Since the self-assembly is an out-of-equilibrium process, studying its behavior through equilibrium statistical mechanics is usually not valid. For that reason, as we have mentioned earlier, a majority of preceding studies have been based on manners of non-equilibrium statistical mechanics. Meanwhile, recent progresses in the non-equilibrium statistical mechanics framework introduced a useful method to handle out-of-equilibrium processes by biasing trajectories[106–115]. The essential idea of the theory is to implement the large deviation principle in trajectory space as the traditional framework of statistical mechanics has done in phase space. The theory successfully proved that there exists dynamical symmetry breaking in several models of glass formers by both analytical and numerical scheme[107, 108, 114]. Besides, this approach suggested there is practicability of to manage thermodynamic properties like configuration, local structure or energy via a purely dynamical method[116–118].

The self-assembly process has its analogy with the glass forming system in that both systems usually prepared up via temperature quenching from the disordered structure to ordered equilibrium or metastable structure. Focused on this point, we make an attempt to implement the above-mentioned non-equilibrium ensemble of trajectories in the self-assembly system, which has never been tried before, to analyze and quantify the dynamics of the process. Our goal is to understand the obstacle due to the restricted dynamics in the self-assembly process as a dynamical symmetry breaking in trajectory space. We expect our work will give an entirely new perspective to understand the kinetic trapping and the reversible dynamics in self-assembly processes.

A.2 Theory

In this study, we use the activity of a given trajectory as a measurable observable, which is projecting the reversibility of the self-assembling system. Consider a stochastic trajectory $\mathbf{X}$ of classical and discrete Markov process; we can regard the trajectory as a set of time-evolving configurations $(\mathbf{x}, t)$: $\mathbf{X} = \{(\mathbf{x}_K, t_K), \ \cdots, \ (\mathbf{x}_0, t_0)\}$. The probability of finding a single trajectory when observing a given system is described as successive products of transition probability $p(\mathbf{x}_{i+1}, t_{i+1}|\mathbf{x}_i, t_i)$ from the current configuration $(\mathbf{x}_i, t_i)$ to next one $(\mathbf{x}_{i+1}, t_{i+1})$ and the population of its starting configuration $p(\mathbf{x}_0, t_0)$[119, 120]:

$$\begin{aligned} P[\mathbf{X}] =& p(\mathbf{x}_K, t_K|\mathbf{x}_{K-1}, t_{K-1}) \\ & \cdots p(\mathbf{x}_1, t_1|\mathbf{x}_0, t_0) p(\mathbf{x}_0, t_0). \end{aligned} \tag{A.1}$$

We assume that the dynamics of the system is governed by the master equation $\partial_t |\mathbf{p}(t)\rangle = \mathbb{W} |\mathbf{p}(t)\rangle$ and since the model is a discrete process, the master operator is defined as a matrix form:

$$\mathbb{W} = \sum_{\mathbf{x}' \neq \mathbf{x}} w(\mathbf{x}'|\mathbf{x}) |\mathbf{x}'\rangle \langle \mathbf{x}| - \sum_{\mathbf{x}} r(\mathbf{x}) |\mathbf{x}\rangle \langle \mathbf{x}| . \tag{A.2}$$

Here, $w(\mathbf{x}'|\mathbf{x})$ in the off-diagonal elements corresponds to the transition rate from configuration $\mathbf{x}$ to $\mathbf{x}'$, and the diagonal term, $r(\mathbf{x})$ denotes the rate of escape from current configuration $\mathbf{x}$, respectively. With transition rates defined at the master equation, the transition probability of each step will be $w(\mathbf{x}_i|\mathbf{x}_{i-1})e^{-(t_i-t_{i-1})r(\mathbf{x}_{i-1})}$. Therefore, the probability distribution functional of trajectory $P[\mathbf{X}]$ is given as follows[112]:

$$\begin{aligned} P[\mathbf{X}] =& e^{-(\tau-t_K)r(\mathbf{x}_K)}p(\mathbf{x}_0) \\ & \times \prod_{i=1}^{K} w(\mathbf{x}_i|\mathbf{x}_{i-1})e^{-(t_i-t_{i-1})r(\mathbf{x}_{i-1})}. \end{aligned} \tag{A.3}$$

There are two ways in measuring the length of given trajectory: the total trajectory time (or observation time) τ and the number of configuration changes during the trajectory, generally we call this *activity*, K. In a more general approach, one can consider a time-extensive physical observable O over the trajectory and its increment o. Then O will be incremented each configuration change[107, 112]:

$$O[\mathbf{X}] = \sum_{i=1}^{K} o(\mathbf{x}_{i-1}, \mathbf{x}_i). \tag{A.4}$$

The observable O surely becomes activity K when the incremental value is $o = 1 - \delta_{\mathbf{x}_{i-1},\mathbf{x}}$, that is 1, when the configuration changes, otherwise 0. If the system had made its final Kth configuration jump at time t_K and the final configuration $\mathbf{x}_K$ survives until the observation time τ, the first exponential term remains. Or we can simply stop measuring the time evolution of the system when the final configuration jump happened. In this case, the first exponential term will be not be needed.

There exist similar relations between extensive properties in the thermodynamic ensemble: the particle number N and the volume V[110, 112]. In the typical experimental scenario, we measure some physical observables in fixed trajectory time τ. However, occasionally, it is much more convenient to fix the activity of trajectory K when simulate systems exhibit very slow dynamics[121].

A.3 Lattice Gas Model

We use an Ising lattice-gas in the two-dimensional square lattice as a model of self-assembly process. More than two particles cannot occupy the same lattice position, and a particle only interacts with the other particles in its nearest neighbor lattice sites. The interaction energy of the system is defined as follows:

$$H = \frac{\epsilon}{2} \sum_{p} n_p. \tag{A.5}$$

Here, ϵ denotes the strength of bonds between the particles, p is the index of the nearest neighbor, and n_p is the occupancy (0 or 1) of the site p, respectively. The model consists of $N = 2048$ particles on the two-dimensional square lattice of $V = 144 \times 144$, and the number density is $\rho \sim 0.10$, accordingly. From the theoretical perspective, the system exhibits liquid-gas phase coexistence when $\sinh^4(\epsilon/2T_\mathrm{c}) > [1-(2\rho-1)^8]^{-1}$. In the equilibrium condition below the critical temperature, the assembly yield should increase monotonically, and particles also ought to form a single large cluster. But kinetic trapping due to the lack of reversibility in bond-making and breaking processes makes it hard for the system to relax into equilibrium configurations. As a result, below a specific temperature point, the system is trapped in metastable states, which are composed of relatively small clusters, and the assembly yield starts to decrease drastically. This phase separating behavior of the Ising lattice gas is in analogy with general self-assembly processes[99].

We perform an extensive numerical simulation to obtain assemble trajectories via a stochastic Monte Carlo scheme. To achieve this, we use the classical *kinetic Monte Carlo* (kMC) method[121]. Given the current phase-space position $\mathbf{x}$ of the system, the time interval to the next jump Δt can be calculated along the probability $p_{\mathbf{x}}(\Delta t) \propto \exp[-r(\mathbf{x})\Delta t]$, and a transition $\mathbf{x} \to \mathbf{x}'$ is selected from all possible moves with transition rate $w(\mathbf{x}'|\mathbf{x})$. The algorithm is appropriate for sampling trajectories with fixed activity since kMC is a rejection-free process, and each Monte Carlo step corresponds to a single jump between configurations[112].

We calculate the temperature dependence of the assembly yield n_4, which denotes the fraction of particles that have exactly four occupied nearest neighbors, and the intensive trajectory time, τ/K. Since our simulation model is a typical model of the Ising lattice gas, the results shown in Fig. A.1(a) reveal archetypal non-monotonic behavior as expected from the other studies[101–103, 121]. Even if at thermodynamic equilibrium the structure in the very low temperature range should form a single, large cluster, kinetic trapping disrupts the assembling process and the system breaks up into many, relatively small clusters. Consequentially, the system shows the maximum assembly efficiency near the $T \sim 0.3$, and it drops towards to decreasing temperature. The intensive trajectory time in Fig. A.1 shows a comparable temperature dependency with assembly yield.

For more detailed examination, the time evolution of the assembly yield and the intensive trajectory time are plotted in Fig. A.1(c). The relation between two properties gives a more clear idea of trapping phenomena at local minima. Both the assembly yield (Fig. A.1(a)) and the trajectory time (Fig. A.1(b)) exhibit the local minima followed by a long plateau behavior. After enough time has passed, eventually the plateau in the assembly yield ends first; the trajectory time follows. This mechanism makes a kink behavior in n_4 as shown in the Fig. A.1(c). At the lower temperature regime exhibits kinetic trapping, the system trapped in the point near $\tau/K \sim 15$ and $n_4 \sim 8 \times 10^{-3}$, and the graph sharply shoots up when the plateau in the assembly yield disappears. This tendency gradually vanishes as the temperature increases, and the system just bypasses that trapping region and directly into assembling in the temperature range where good assemble is

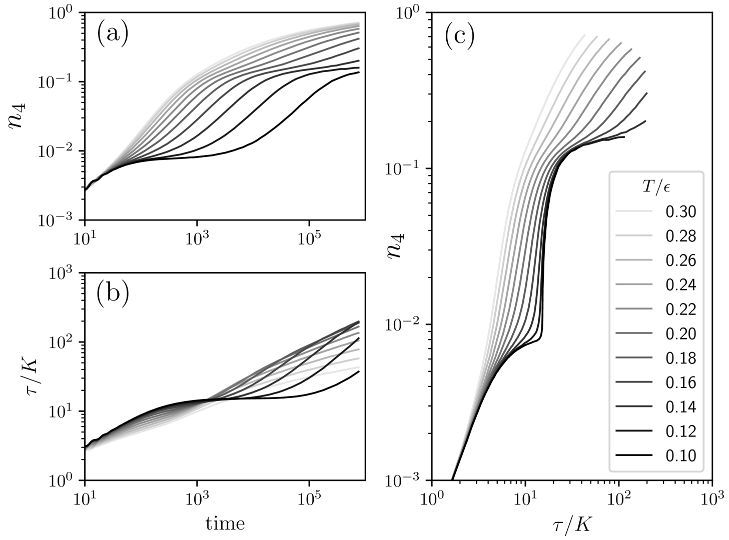


Figure A.1: Time evolution plots of (a) the assembly yield, (b) intensive trajectory length and (c) their relations (c) in the Ising lattice gas. Colors of lines represent the temperature of the system (from $T = 0.10$ to 0.30). In the temperature regime $T < 0.15$, where the kinetic trapping is strongly happens, Plateaux in the structure and the dynamics cause a kink nearby $\tau K / \sim 15$ and $n_4 \sim 8 \times 10^{-3}$.

taking place in the end.

A.4 Mathematical Model

To get more advanced insight, we propose a minimal model that exhibits kinetic trapping behavior as like as the lattice gas model. Grant and Whitelam already presented the prototype of our model to illustrate the non-monotonical growth in self-assembly processes[96, 99]. Essentially the system has three different energy levels. The *unbound* state represents nonbonded free particles and has the highest energy ($E = 0$), the *misbound* states of intermediate energy value ($E = -\epsilon_{\text{mis}}$) and the optimally *bound* state ($E = -\epsilon_{\text{opt}}$) on the ground level; it is obvious that $\epsilon_{\text{opt}} > \epsilon_{\text{mis}}$. Passing through the unbound state is necessary if the system intends to transit from metastable misbound states to the stable bound state. Additionally, there is degeneracy Ω_{mis} in the misbound state to achieve an entropic barrier.

The transition rate matrix (master operator) of the original model is described as 3×3 matrix and the degeneracy is simply multiplied by transition and escape rates of unbound and misbound states[99]. We modify the original model to accomplish the 'rattling' dynamics between degenerated misbound states. For example, the master operator of the $\Omega_{\text{mis}} = 2$ case is expressed as 4×4 matrix[112, 120]:

$$\mathbb{W} = \begin{bmatrix} -1-\gamma & 1 & 1 & 0 \\ 1 & -1-\gamma & 1 & 0 \\ \gamma & \gamma & -3 & \nu \\ 0 & 0 & 1 & -\nu \end{bmatrix}. \tag{A.6}$$

Each state can be described as a vector: misbound states($|1\rangle\,,\ \cdots\,,\ |\Omega_{\text{mis}}\rangle$), unbound state ($|\Omega_{\text{mis}}+1\rangle$) and bound state ($|\Omega_{\text{mis}}+2\rangle$), respectively. Based on the detailed balance condition, transition rates from misbound to unbound is $\gamma = \exp(-\epsilon_{\text{mis}}/T)$ and bound to unbound is $\nu = \exp(-\epsilon_{\text{opt}}/T)$, respectively. Rates toward to opposite directions are simply 1 by traditional Metropolis acceptance criteria. Notwithstanding our modified model has complicated dynamics more than the original one, it is obvious that the probability of the bound state, $P_{\text{bound}} = \langle \Omega_{\text{mis}} + 2|\mathbf{p}(t)\rangle$ will have exactly the same equilibrium value $\nu/(1+\nu+\Omega_{\text{mis}}\gamma)$ when $t \to \infty$.

We perform numerical calculations for our minimal model using matrix algebra to confirm that whether or not the model successfully reproduces results from the Ising lattice gas. The time-evolution of a system can be described as $|\mathbf{p}(t)\rangle = \exp(t\mathbb{W})\,|\mathbf{p}(0)\rangle$ and mean value of certain observable O at the time t can be calculated from $\langle O(t)\rangle = \langle \mathbf{e}|\,\mathbb{O}\,|\mathbf{p}(t)\rangle$ where $|\mathbf{e}\rangle = \sum_{\mathbf{x}} |\mathbf{x}\rangle$ is the *projection* state[108, 120]. We let binding energies of misbound and bound states are $\epsilon_{\text{mis}} = \epsilon$ and $\epsilon_{\text{opt}} = 2\epsilon$, respectively. Results from numerical matrix calculations are shown in Fig. A.3. Outcomes well correspond with the results obtained from the Ising lattice gas, especially assembly yield versus intensive trajectory time graph demonstrates the same kink in the kinetic trapping regime.

A.5 Dynamical Phase Transitions

In previous sections, we demonstrated there are kink behaviors between structure (assembly yield, n_4 or P_{Bound}) and dynamics (step time, τ/K) in both numerical models during the kinetic trapping occur. Focused on this

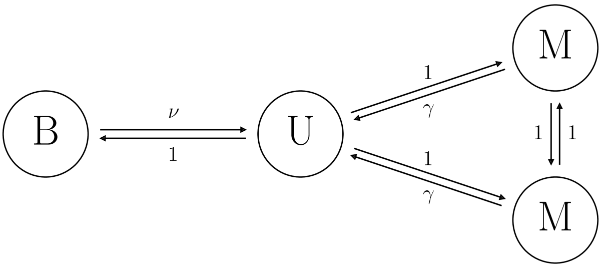


Figure A.2: A minimal three-state model of self-assembly. There are two misbound states (M), which have the same intermediate energy, can transit without any energy barrier. The transition rate from bound state (B, has the lowest energy) to unbound state (U, has the highest energy) is ν, from misbound states to unbound state is γ and rates to reverse directions are 1 due to the Metropolis criteria; jumping between misbound and bound states are impossible.

fact, we suggest the possible existence of a crossover between two different dynamical phases between in self-assembly processes. Recent advances in the dynamic ensemble theory give us a crucial insight by introducing a virtual field that biases trajectory length, which as an conjugate variable of the ensemble of trajectories[107–114].

From the definition of observation probability of a given trajectory as expressed in eqn (A.3), we can calculate the PDF of the τ in K-fixed trajectories

$$P(\tau|K) = \int D\mathbf{X}_K \, \delta(\tau - \hat{\tau}[\mathbf{X}_K])P[\mathbf{X}_K], \tag{A.7}$$

and its corresponding partition function with a conjugate field x of trajectory

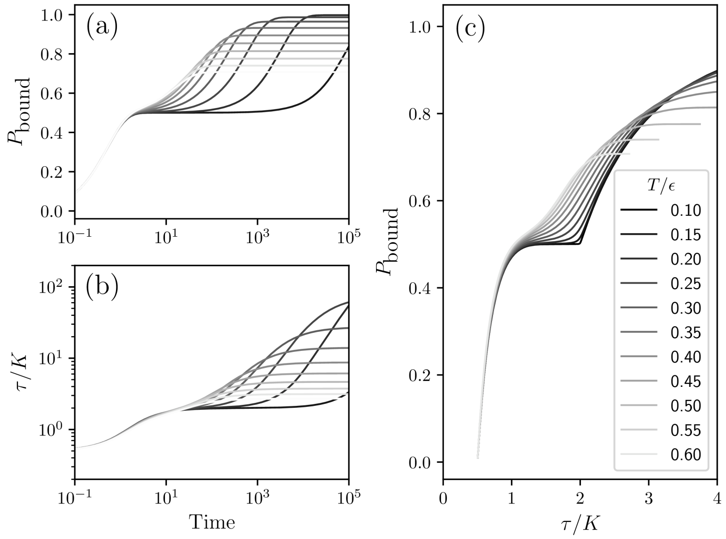


Figure A.3: Time evolution of (a) the assembly yield, (b) total trajectory time per activity (b) and their relations (c) of the three-state minimal model. The structural plateau and the dynamical plateau create a kink in the kinetic trapping regime. These results are consistent with the more realistic model.

time τ[107, 112]:

$$Z(x, K) = \int d\tau e^{-x\tau} P(\tau|K). \tag{A.8}$$

We call these ensembles as (τ, K) and (x, K) ensemble, named after their fixed variables, respectively. Non-equilibrium free energies of two cases are defined as: $\Psi(\tau, K) = \ln P(\tau|K)$ and $\Phi(x, K) = \ln Z(x, K)$. If both quantities have the large deviation limit $\Psi(\tau, K) \sim K\psi(\tau)$ and $\Phi(x, K) \sim K\phi(x)$, ψ and ϕ are convex conjugate to each other by Legendre-Fenchel transform[115]. Finally we can describe the physical meaning of x from Legendre duality:

$$\frac{\partial \Psi}{\partial \tau} \equiv x(\tau, K). \tag{A.9}$$

One can explain x as an external field that biasing trajectory time, like what the chemical potential μ and the pressure P does in traditional thermodynamic ensemble. For Markov processes, we can get the partition sum of trajectories using matrix product

$$Z(x, K) = \langle \mathbf{e} | \, \mathbb{T}^K(x) \, | \mathbf{p}(0) \rangle \,, \tag{A.10}$$

with off-diagonal transfer operator obtained from Laplace transform of the probability matrix of the system[109, 112]:

$$\mathbb{T}(x) = \sum_{\mathbf{x}'=\mathbf{x}} \frac{w(\mathbf{x}'|\mathbf{x})}{x + r(\mathbf{x})} \left|\mathbf{x}'\right\rangle \left\langle \mathbf{x}\right| . \tag{A.11}$$

If the system is in the thermodynamic limit, when K is large enough in other words, we can directly obtain $\phi(x)$ from the largest eigenvalue of the operator $\mathbb{T}(x)$[112, 115]. Many works analytically or numerically demonstrated that the nonequilibrium ensemble exhibits dynamical first-order phase transitions in several abstract or realistic (atomistic) systems which describing glassy dynamics[109, 111, 114]. For example, the kinetically constrained model shows criticality at $T = 0$; therefore, there is always a phase coexistence between low- (inactive) and high-activity (active) phases at any finite temperature[107, 108].

The trajectory time per kMC step plays a relevant role in the assembling process as we discussed in previous sections. Now our purpose is to control assemble dynamics of Ising lattice gas via biasing step time using the (x, K) ensemble. We use the transition path sampling (TPS) scheme[122] for sample ensembles of assembling trajectories in various T and x ranges. The dynamical free energy, $\Phi(x, K)$ is calculated from the multistate Bennet acceptance ratio (MBAR)[123, 124]. As shown in the Fig. A.4 (b), as in other model systems, our results clearly exhibit an active-inactive dynamical phase transition when the field x is applied for total lengths (or time) of trajectories.

We also calculate the same quantity for the minimal model of matrix products in Fig. A.4 (a). The results for the infinite-activity limit is obtained from numerically diagonalized eigenvalue of $\mathbb{T}(x)$:

$$\mathbb{T}(x) = \begin{bmatrix} 0 & \frac{1}{x+\gamma+1} & \frac{1}{x+3} & 0 \\ \frac{1}{x+\gamma+1} & 0 & \frac{1}{x+3} & 0 \\ \frac{\gamma}{x+\gamma+1} & \frac{\gamma}{x+\gamma+1} & 0 & \frac{\nu}{x+\nu} \\ 0 & 0 & \frac{1}{x+3} & 0 \end{bmatrix}. \tag{A.12}$$

A noteworthy feature is that first-order dynamical phase coexistences become apparent as the temperature decreases in both two models. Namely, it seems there is a finite critical temperature $T_{\mathrm{c}} > 0$ exists, and when compared with previous results, the criticality is located in the kinetic trapping regime. This phenomenon is observed both in the Ising lattice gas and the minimal model and is the distinguishable feature when compared to results from the other models: the KCM or the TLG model[107, 108, 125]. Thus, we argue that there are dynamical first-order phase transitions in self-assembly systems, and one can understand the kinetic trapping behavior as a consequence of the phase separation in the ensemble of trajectories.

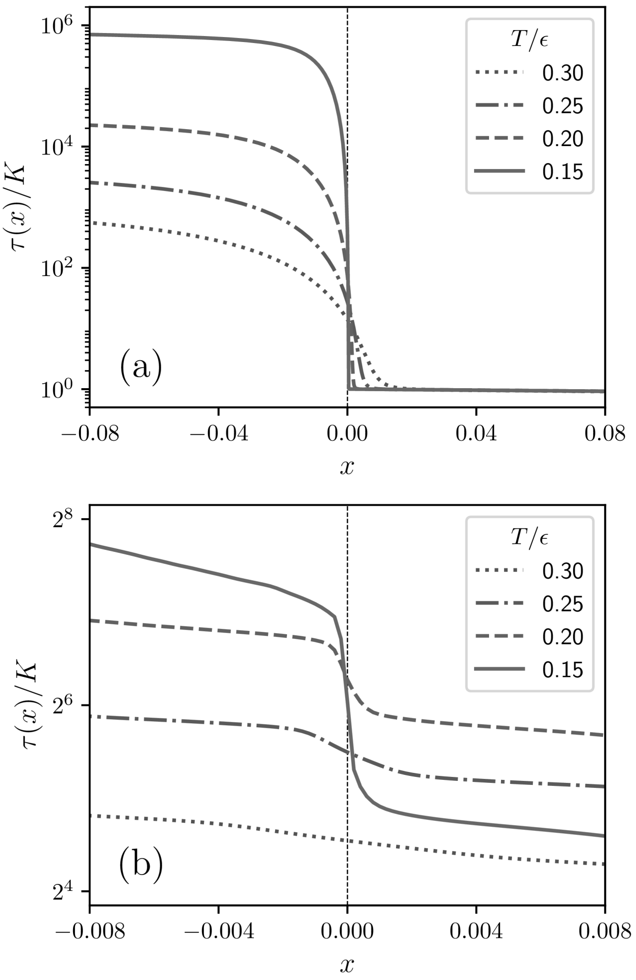


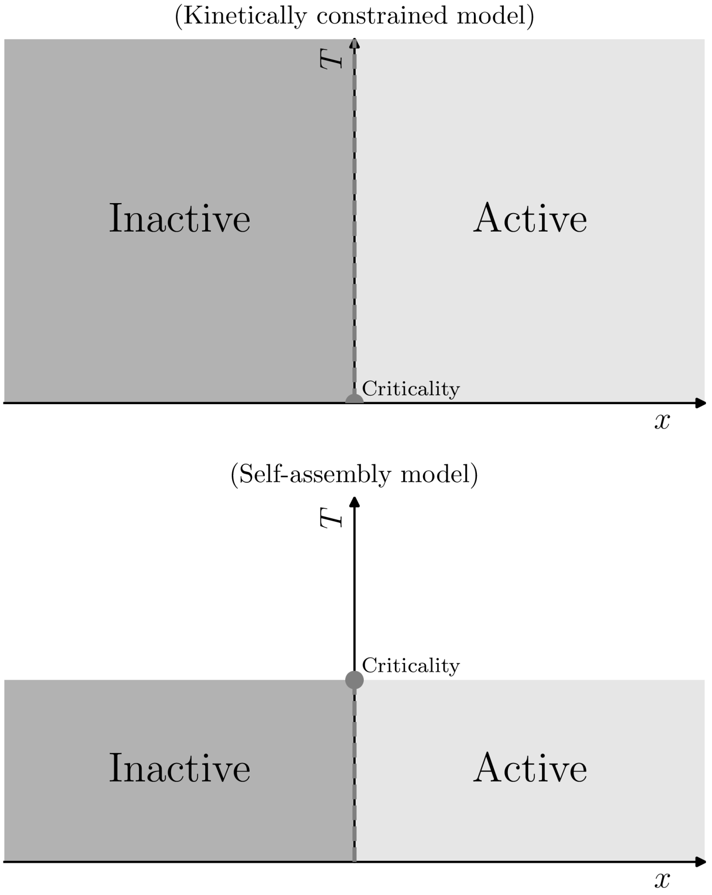
Figure A.4: (a) Plot of the intensive trajectory time τ/K of the minimal model from numerically diagonalized transfer matrix, $\mathbb{T}(x)$. The temperature range is from $T/\epsilon = 0.15$ (blue line) to 0.30 (red line). (b) The same quantity in the Ising lattice gas. Shooting TPS algorithm is applied for sampling ensemble of trajectories. Singularity at low-temperature demonstrates there is active-inactive coexistence near the $x = 0$.

A.6 Conclusion

Adopting the activity concept as a projection of the reversibility of the self-assembly process, we can easily understand the relation between structural relaxations and dynamical properties due to kinetic trapping in a self-assemble system. Using Monte Carlo simulation and numerical calculation, we discovered there are two dominant factors in trapping behavior in the local minimum. When the temperature is low enough to exhibit kinetic trapping, both structure and activity display plateau behaviors at a similar time scale during assembly progress. Then the plateau due to structural trap disappears first; escaping from the dynamical trap then follows. The minimal model that we proposed successfully reproduces the results taken from both the thermodynamic and the dynamic behavior of the relatively realistic lattice gas model.

With the dynamic ensemble of trajectories approach using large deviation formalism[109, 112], it seems that there is a a finite critical temperature that exhibits a dynamical active-inactive first-order phase transition below the temperature. In contrast, for the KCM of glass formers[107, 108], such phase transitions always appear for $T > 0$. If the dynamic critical temperature indeed exists, the kinetic trapping behavior might be described as an active-inactive crossover in assemble trajectories.

As a perspective of the self-assembly process from disordered structure to ordered equilibrium structure can be regarded as a feature of the quenched disorder, we anticipate our mathematical model would be helpful for understanding dynamical and structural properties of many other models handling quenched system; glass forming fluids for example[109, 111, 114]. Certainly, it also might be a useful topic when applying for more realistic models of self-assembly processes.


(Kinetically constrained model)

(Self-assembly model)

Figure A.5: Estimated dynamical phase diagram of (left) the kinetically constrained model and (right) our model of the self-assembly processes. A distinguishable feature of the our model is in comparison with the KCMs is there is a finite critical temperature $T_c > 0$ which exhibits a dynamic phase coexistence below the T_c.

Appendix B

Reaction-Path Thermodynamics of the Michaelis-Menten Kinetics

B.1 Introduction

Michaelis-Menten kinetics[126, 127] is one of the most fundamental mechanism for describing catalytic or enzymatic reactions and it presents crucial insights into the understanding of many biochemical or physical processes in living systems[128]: enzyme reactions in the living cell, DNA hybridization[68], gene regulation[129, 130], or molecular motors[131, 132]. Over a hundred years since its birth, there have been numerous theoretical and experimental advances for studying the enzymatic mechanism in various systems and methods, especially spectroscopic quantifications at the single-molecule level[133, 134]. Such a series of experimental successes in the microscopic scale promoted studies in theoretical manners[130, 135–

141]. A major topic in theoretical approaches is the timescale of enzymatic turnover[142, 143], which means time duration until a single reaction ends. Many theoretical approaches have been developed to calculate turnover time and to quantify its fluctuation behavior: from the solution of the linear differential equation[134, 142, 143] in the ideal scenario to reaction time distribution (RTD) methods in disordered systems with non-Poissonian kinetics[135, 137, 138].

$$\mathrm{E} + \mathrm{S} \underset{k_u}{\overset{k_b}{\rightleftharpoons}} \mathrm{ES} \xrightarrow{k_c} \mathrm{E} + \mathrm{P} \tag{B.1}$$

The principal idea of the Michaelis-Menten mechanism is there are two stages in the enzymatic reaction process[126, 127]: (i) the reversible binding-unbinding reactions between the substrate (S) and the enzyme (E) molecule, $\mathrm{E} + \mathrm{S} \rightleftharpoons \mathrm{ES}$ and (ii) the irreversible catalytic reaction from the bound enzyme-substrate complex (ES) to the product (P), $\mathrm{ES} \longrightarrow \mathrm{E} + \mathrm{P}$. We need to pay attention to the unbinding (disassociation) reaction at the stage (i) because the unbinding makes the process return to its initial state. Thus, essentially, the Michaelis-Menten mechanism can be interpreted as a renewal process[135, 140], and ‘events’ of unbinding play an essential role for the entire process. For example, chemical intuition tells that the increase of unbinding rate k_u has to result in the decrease of turnover rate, which is true at least in ideal models which exhibit Poisson kinetics. However, paradoxically, in some cases where the waiting time distribution of catalysis is not a single exponential form, slower disassociation may cause the faster turnover[139, 140]. Such nonmonotonic dependencies between unbinding and turnover suggest that we can classify enzymatic processes into two different dynamical phases, the *inhibitory* and *excitatory* unbinding.

The importance of the unbinding as we mentioned before signifies the necessity of quantifying unbinding events in enzymatic reaction processes. In the present work, we study several kinetic aspects of the Michaelis-Menten mechanism in the single molecule level in the framework of the the nonequilibrium statistical mechanics and quantify the statistical feature of unbinding events. Recent statistical mechanical studies present a notable perspective for handling systems in out-of-equilibrium. The core concept is a stochastic trajectory (or path) can be thought as a microstate in the statistical ensemble theory[113]. This idea and a mathematical formulation named the large deviation principle[115] leads to *nonequilibrium ensemble* theory. The main purpose of the theory is to draw out-of-equilibrium or dynamical properties of the system from theoretical or computer simulation methods. Furthermore, the nonequilibrium ensemble also successfully described the heterogeneous dynamical behavior in many systems, e.g., glass forming liquids[114, 117], kinetic networks[111, 144], active matters[145–147], or protein folding pathways[148] as an order-disorder symmetry breaking phenomenon between metastable states when one uses ‘dynamical events’ as an order parameter. Based on preceding studies, we believe the nonequilibrium ensemble theory will be a powerful tool for quantifying enzyme kinetics since most chemical reactions, including enzymatic processes, are also out-of-equilibrium processes.

This chapter is outlined as follows: In the second section, we suggest a concept of a reaction-path entropy, construct the statistical thermodynamics of enzymatic reaction paths, and calculate several major reaction timescales of the single-enzyme and single-substrate model via the large deviations principle and the nonequilibrium ensemble theory. In the third section, we quantify the number of unbinding events K when we observe the system at fixed timescale and evaluate the heterogeneous kinetics of the same model as a dynamic order-disorder in unbinding rates. In the last section, we summarize and conclude our results.

B.2 Reaction Path Thermodynamics

We use the single-molecule variant of the chemical master equation (CME) of the Michaelis-Menten equation. The stochastic equation considers finite numbers of molecules in a discrete manner, instead of their concentrations in a continuous manner and each combination of quantities corresponds to a different state of the system. Due to the law of conservation of mass, we can assume that the system contains $N = n_{\mathrm{E}} + n_{\mathrm{ES}}$ of enzyme-type molecules and $M = n_\mathrm{S} + n_\mathrm{E} + n_\mathrm{P}$ of ligand-type molecules[136, 142]. The master equation of the system is as follows:

$$\begin{aligned}\dot{p}(n_\mathrm{S}, n_\mathrm{ES}, t) = &- [w_b n_\mathrm{S}(N - n_\mathrm{ES}) + w_u n_\mathrm{ES} + w_c n_\mathrm{ES}]p(n_\mathrm{S}, n_\mathrm{ES}, t) \\ &+ w_b(n_\mathrm{S} + 1)(N - n_\mathrm{ES} + 1)p(n_\mathrm{S} + 1, n_\mathrm{ES} - 1, t) \\ &+ w_u(n_\mathrm{ES} + 1)p(n_\mathrm{S} - 1, n_\mathrm{ES} + 1, t) \\ &+ w_c(n_\mathrm{ES} + 1)p(n_\mathrm{S}, n_\mathrm{ES} + 1, t).\end{aligned} \tag{B.2}$$

Here, $w_b = k_b/V_u^2$, $w_u = k_u/V_u$, and $w_c = k_c/V_u$ are the reaction rate constants per unit volume V_u and subscripts b, u, and c denote the *binding*, the *unbinding*, and the *catalysis* event, respectively. Since the model considers a discrete number of components, we use probabilities of states $p(n_\mathrm{S}, n_\mathrm{ES}, t)$, instead of continuous concentrations. If the system contains only one enzyme and substrate molecules, $N = 1$ and $M = 1$ in other words, the equation B.2 can be reduced to the following form:

$$\dot{p}_\mathrm{S}(t) = w_u p_\mathrm{ES}(t) - w_b p_\mathrm{S}(t), \tag{B.3a}$$

$$\dot{p}_\mathrm{ES}(t) = w_b p_\mathrm{S}(t) - (w_u + w_c)p_\mathrm{ES}(t), \tag{B.3b}$$

$$\dot{p}_\mathrm{P}(t) = w_c p_\mathrm{ES}(t). \tag{B.3c}$$

We omit the time evolution of the probability of enzyme E since it has the relation with ES, $p_\mathrm{E}(t) = 1 - p_\mathrm{ES}(t)$. If one considers a single reaction path

$\mathrm{E} + \mathrm{S} \to \cdots \to \mathrm{E} + \mathrm{P}$ of the equation B.3 which has K unbinding events, then one can find the given path with the probability $\rho[\{\text{path}\}]$[113, 119]:

$$\begin{aligned}\rho[\{\text{path}\}] = w_b e^{-w_b \Delta t_0} &\left(\prod_{i=1}^{K} w_u e^{-(w_u + w_c)\Delta t'_i} w_b e^{-w_b \Delta t_i} \right) \\ &\times w_c e^{-(w_u + w_c)\Delta t'_0}.\end{aligned} \tag{B.4}$$

Here, time intervals Δt_i and $\Delta t'_i$ denote lifetimes of S and ES at individual reaction stage, respectively. If we define the 'total' lifetime of each component as the sum of individual lifetimes, $\sum_{i=0}^{K} \Delta t_i = t_{\mathrm{S}}$ and $\sum_{i=0}^{K} \Delta t'_i = t_{\mathrm{ES}}$, then we can simplify the equation B.4 to

$$\rho[\{\text{path}\}] = w_{\mathrm{b}} w_{\mathrm{c}} (w_{\mathrm{u}} w_{\mathrm{b}})^K e^{-w_{\mathrm{b}} t_{\mathrm{S}}} e^{-(w_{\mathrm{b}} + w_{\mathrm{u}}) t_{ES}}, \tag{B.5}$$

which only depends on three nonequilibrium observables: the number of unbinding events (K), the total lifetime of the substrate molecule (t_{S}) and the enzyme-substrate complex (t_{ES}), respectively. That is to say; we can find a single reaction path with identical probability if three observables K, t_{S}, and t_{ES} are conserved. Hence, similar to N, V, and E in the canonical equilibrium ensemble case, the principle of equal a *priori* probabilities is valid, and it leads to the definition of the *nonequilibrium* microcanonical ensemble, described by the following path-dependent reaction entropy.

$$\mathcal{S} \equiv -\sum_{\{\text{path}\}} \rho[\{\text{path}\}] \ln \rho[\{\text{path}\}] = -\ln \rho[\{\text{path}\}] \tag{B.6}$$

The microscopic number of all possible reaction paths (similar to *microstates* in equilibrium statistical mechanics) $\Omega = 1/\rho$ depends on combinations of Δt_i and $\Delta t'_i$[149]:

$$\begin{aligned} \Omega(K, t_{\text{ES}}, t_{\text{S}}) &= \int_{\sum \Delta t_i = t_{\text{S}}} d\Delta t^{K+1} \int_{\sum \Delta t'_i = t_{\text{ES}}} d\Delta t'^{K+1} \\ &= \frac{K+1}{K!K!} t_{\text{ES}}^{K} t_{\text{S}}^{K}. \end{aligned} \tag{B.7}$$

In the equation B.7, each integral denotes the area of the $(K+1)$-dimension hyper-sphere. Accordingly, we can evaluate the entropy of reaction paths in the $(K, t_{\text{ES}}, t_{\text{S}})$-fixed ensemble, $\mathcal{S}(K, t_{\text{ES}}, t_{\text{S}}) = \ln \Omega(K, t_{\text{ES}}, t_{\text{S}})$. Now quantifying the MM kinetics with the language of statistical thermodynamics is feasible by cause of the definition of the reaction path entropy and the large deviations principle[115]. The Gärtner-Ellis theorem presents partition functions of the following nonequilibrium canonical (K, t_{ES}, μ) and grand canonical (K, ν, μ) ensembles

$$\mathcal{Z}(K, t_{\text{ES}}, \mu) = \int_0^\infty dt_{\text{S}} e^{-\mu t_{\text{S}}} \Omega(K, t_{\text{ES}}, t_{\text{S}}), \tag{B.8a}$$

$$\mathcal{Q}(K, \nu, \mu) = \int_0^\infty dt_{\text{ES}} e^{-\nu t_{\text{ES}}} \mathcal{Z}(K, t_{\text{ES}}, \mu), \tag{B.8b}$$

and their free energies spontaneously with certain conjugate fields μ and ν, which biases t_{S} and t_{ES}, respectively,

$$\mathcal{F}(K, t_{\mathrm{ES}}, \mu) = K \ln \mu - K \ln t_{\mathrm{ES}} + K \ln K - K, \qquad \text{(B.9a)}$$

$$\mathcal{G}(K, \nu, \mu) = K \ln \nu + K \ln \mu. \qquad \text{(B.9b)}$$

Equations B.5 and B.8 suggest that $\mu = w_b$ and $\nu = w_u + w_c$, which in fact means that escaping rates and lifetimes are mutually conjugate variables. Therefore, the fundamental relations of the nonequilibrium thermodynamics, $\mathcal{F} = \mu t_{\mathrm{S}} - \mathcal{S}$ and $\mathcal{G} = \nu t_{\mathrm{ES}} - \mathcal{F}$ are valid. From equations B.7 and B.8, conditional probability distributions of t_{S} and t_{ES} in the K-fixed ensemble are Poissonian as follows:

$$\rho(t_{\mathrm{S}}|K) = \frac{\mu^{K+1} t_{\mathrm{S}}^K}{K!} e^{-\mu t_{\mathrm{S}}}, \qquad \text{(B.10a)}$$

$$\rho(t_{\mathrm{ES}}|K) = \frac{\nu^{K+1} t_{\mathrm{ES}}^K}{K!} e^{-\nu t_{\mathrm{ES}}}. \qquad \text{(B.10b)}$$

Note that the two lifetimes t_{S} and t_{ES} are mutually independent. Since the enzymatic turnover time, t_{t}, is the sum of t_{S} and t_{ES}, its conditional probability distribution $\rho(t_{\mathrm{t}}|K)$ takes a convolution form of $\rho(t_{\mathrm{S}}|K)$ and $\rho(t_{\mathrm{ES}}|K)$. The convolution is quite complicated for calculation due to t^K

term, but it can be easily obtained in the Laplace domain:

$$\rho(x_{\mathrm{t}}|K) = \left[\frac{\mu\nu}{(\mu + x_{\mathrm{t}})(\nu + x_{\mathrm{t}})}\right]^{K+1}. \tag{B.11}$$

With Bayes' theorem and considerations of the marginal probability of unbinding events is products of transition probabilities $\rho(K) = (w_c w_u^K)/(w_u + w_c)^{K+1}$ by its definition[113, 119], Eqns. B.10 and B.11 finally give marginal probability distributions of liftimes of S, ES, and turnover time

$$\rho(t_{\mathrm{S}}) = (w_b w_c/(w_u + w_c))\exp(-w_b w_c t_{\mathrm{S}}/(w_u + w_c)), \tag{B.12a}$$

$$\rho(t_{\mathrm{ES}}) = w_c \exp(-w_c t_{\mathrm{ES}}), \tag{B.12b}$$

$$\rho(t_{\mathrm{t}}) = \alpha\beta(e^{-\alpha t_{\mathrm{t}}} - e^{-\beta t_{\mathrm{t}}})/(\beta - \alpha), \tag{B.12c}$$

where two constants α and β in the turnover time distribution are:

$$\alpha = \frac{\lambda + \sqrt{\lambda^2 - 4w_b w_c}}{2}, \tag{B.13a}$$

$$\beta = \frac{\lambda - \sqrt{\lambda^2 - 4w_b w_c}}{2}. \tag{B.13b}$$

In the avobe equation, $\lambda = w_b + w_u + w_c$. The probability distribution of turnover time we obtained in the equation B.12 is identical with results from the solution of linear differential equations[134, 142, 143]. We finally obtain nonequilibrium ensemble average of the total lifetimes of S, ES, and the turnover time:

$$\langle t_{\mathrm{S}} \rangle = \frac{w_u + w_c}{w_b w_c}, \tag{B.14a}$$

$$\langle t_{\mathrm{ES}} \rangle = \frac{1}{w_c}, \tag{B.14b}$$

$$\langle t_{\mathrm{t}} \rangle = \frac{w_b + w_u + w_c}{w_b w_c}. \tag{B.14c}$$

B.3 Fixed Observation Time

In a certain theoretical or experimental scenario, it might be more convenient to sample reaction paths with arbitrary *observation time* τ[150, 151], instead of the fixed number of enzyme-substrate unbinding events K. Since the kinetics of the system is governed by the master equation B.3, the time evolution of the system can be described as $|p(\tau)\rangle = \mathbb{U}(\tau)\,|p(0)\rangle$ with the propagator $\mathbb{U}(\tau) = \exp(\tau\mathbb{W})$. As we fix the observation time, we have to consider not only '*completed*' reaction paths but also sample '*incompleted*' reaction paths which remain in $|\mathrm{S}\rangle$ or $|\mathrm{ES}\rangle$ at the observation time τ. Because the propagator can be decomposed into the operators of conditional probabilities of unbinding events K as $\mathbb{U}(\tau) = \sum_K \mathbb{P}(K|\tau)$, the conditional probability of K at τ is $P(K|\tau) = \langle \mathrm{e}|\, \mathbb{P}(K|\tau)\,|\mathrm{S}\rangle$ where $|\mathrm{e}\rangle = |\mathrm{S}\rangle + |\mathrm{ES}\rangle + |\mathrm{P}\rangle$ is the *projection state*. For *completed* reaction paths ($\mathrm{E} + \mathrm{S} \to \cdots \to \mathrm{E} + \mathrm{P}$) where the final state is $|\mathrm{P}\rangle$, the conditional probability of K

at fixed τ is

$$\langle \mathrm{P}| \, \mathbb{P}(K|\tau) \, |\mathrm{S}\rangle = \int_0^\tau dt_\mathrm{t} \rho(t_\mathrm{t}, K), \tag{B.15}$$

where $\rho(t_\mathrm{t}, K) = \rho(t_\mathrm{t}|K)\rho(K)$ is the joint probability distribution of t_t and K because $\langle \mathrm{P}| \, \mathbb{P}(K|\tau) \, |\mathrm{S}\rangle$ contains all the possible reaction paths that have K unbinding events and turnover times smaller than τ.

For *incompleted* reaction paths where observation states are $|\mathrm{E}\rangle$ or $|\mathrm{ES}\rangle$, we must consider the value of K at time τ, not t_t due to the reaction is not terminated yet at the observation time. It means $\tau = t_\mathrm{S} + t_\mathrm{ES} < t_\mathrm{t}$ and we have to consider the reaction path entropies of both cases, $\mathrm{E} + \mathrm{S} \to \cdots \to \mathrm{E} + \mathrm{S}$ and $\mathrm{E} + \mathrm{S} \to \cdots \to \mathrm{ES}$. First, we calculate Ω_S, which describes the microscopic number of paths which end at $|\mathrm{S}\rangle$ and $(K, t_\mathrm{S}, t_\mathrm{ES})$:

$$\Omega_\mathrm{S}(K, t_\mathrm{S}, t_\mathrm{ES}) = \frac{\sqrt{K}}{(K-1)!} \frac{\sqrt{K+1}}{K!} t_\mathrm{S}^{K-1} t_\mathrm{ES}^{K}. \tag{B.16}$$

We also have to consider reaction paths which end at $|\mathrm{ES}\rangle$:

$$\Omega_\mathrm{ES}(K, t_\mathrm{S}, t_\mathrm{ES}) = \frac{K+1}{K!K!} t_\mathrm{S}^{K} t_\mathrm{ES}^{K}. \tag{B.17}$$

Since Ω_ES is identical to Ω and Ω_S also has a similar form with Ω, we suppose that the Bayesian probability of (τ, K) for incompleted paths and (t_t, K) for completed paths have a nearly same analytical shape when K is large enough. Therefore, we can approximate $|\mathrm{S}\rangle$- and $|\mathrm{ES}\rangle$-contributions of the $P(K|\tau)$:

$$\langle \mathrm{S}| \, \mathbb{P}(K|\tau) \, |\mathrm{S}\rangle + \langle \mathrm{ES}| \, \mathbb{P}(K|\tau) \, |\mathrm{S}\rangle \simeq \frac{\rho(t_{\mathrm{t}} = \tau, K)}{\rho(t_{\mathrm{t}} = \tau)} \int_{\tau}^{\infty} dt_{\mathrm{t}} \rho(t_{\mathrm{t}}). \quad \text{(B.18)}$$

Here, the overall shape of the probability distribution comes from $\rho(t_{\mathrm{t}} = \tau, K)$ and $\rho(t_{\mathrm{t}} > \tau)/\rho(t_{\mathrm{t}} = \tau)$ is a normalization factor. Equations B.15 and B.18 present an approximate form of the conditional probability of the number of unbinding events at fixed observation time:

$$P(K|\tau) \simeq \rho(t_{\mathrm{t}} < \tau, K) + \rho(t_{\mathrm{t}} = \tau, K) \frac{\rho(t_{\mathrm{t}} < \tau)}{\rho(t_{\mathrm{t}} = \tau)}. \quad \text{(B.19)}$$

We plot equation B.15, B.18, and B.19 for $w_b = 0.5$, $w_u = 1.0$, and $w_c = 0.025$ case in the Figure B.1-(a). The equation B.15 has the maximum value at $K = 0$ and shows almost the same decay behavior with $\rho(K)$ in the early stage; it drastically decreases where K is near the peak of Eqn. B.18. This tendency results in a bimodal shape in their sum. The bimodal behavior of $P(K|\tau)$ signifies that we can divide the probability distribution into two different paths[115]: the unbinding-rich one and the unbinding-poor one. Recent studies showed that there exist more than two dynamical phases in systems which exhibit heterogeneous or glassy dynamics[106–108, 111, 112, 114, 117, 137, 144–148]. In the same way, the Michaelis-Menten mechanism shows heterogeneous kinetics in its unbinding events and results in the inactive-phase of 'reaction-completed' paths and active-phase of 'reaction-incompleted' paths.

Again, we use the formalism of the large deviation principle to evaluate the moment-generating function of K with corresponding virtual, conjugate variable s[106, 112, 115]:

$$Z(s,\tau)=\sum_{K=0}^{\infty}e^{-sK}P(K|\tau). \tag{B.20}$$

The n-th derivative of $Z(s,\tau)$ gives the n-th moment of unbinding events at fixed observation time τ, $\langle K^n\rangle_\tau=(-1)^nZ^{-1}\partial_sZ(s,\tau)$. One can also calculate the cumulants from the cumulant generating function (or intensive free energy), $\phi(s,\tau)=\ln Z(s,\tau)/\tau$. The dynamic susceptibility, $\chi_k(s,\tau)$ is the second derivative of $\phi(s,\tau)$ and denotes the amount of fluctuations of unbinding rates per observation time, $k=K/\tau$. In the Fig. B.2-(a), we plot the observation time dependence of $\chi_k(s,\tau)$. The dynamic susceptibility has its maximum value at the point $s=s^*$, which separates the reaction paths into two different dynamical phases, the active ($s<s^*$) one and the inactive ($s>s^*$) one. We must note that the conjugate variable s is virtual and it is barely known about its real physical meaning. The only thing we know for sure is that we have to regard as s is zero for when one samples the system's reaction paths in ordinary conditions. Therefore, now what we have to do is finding the phase-coexistence timescale τ^* where the $s^*(\tau)$

becomes zero.

We need to know the general analytical behavior of $s^*(\tau)$ before obtaining τ^*. As shown in Fig. B.2-(b), $s^*(\tau)$ shows a power-law-like decay over observation time and in the large deviation limit $\tau \gg 1$, the value of s^* converges to a particular value, s_c. We take a different mathematical approach in order to evaluate s_c; one can obtain identical results with equation B.19 from algebraic calculations[106, 107, 109, 112]. First, we start from the definition of the master operator $\mathbb{W}$:

$$\mathbb{W} = \begin{bmatrix} -w_b & w_u & 0 \\ w_b & -(w_u + w_c) & 0 \\ 0 & w_c & 0 \end{bmatrix}. \tag{B.21}$$

What we have to do is to decompose the master operator into two matrices, $\mathbb{W} = \mathbb{W}_\mathrm{m} + \mathbb{W}_\mathrm{r}$. Here, $\mathbb{W}_\mathrm{m}$ is the operator of *monitored* reactions and the other operator, $\mathbb{W}_\mathrm{r}$ denotes the rest of transitions. Since we count the number of unbinding reactions, we let $\mathbb{W}_\mathrm{m} \equiv w_u \left|1\right\rangle \left\langle 2\right|$. The propagator $\mathbb{U}(\tau) = \exp(\tau\mathbb{W})$ is an exponential form of the master operator so we can decompose it as

$$\mathbb{P}(K|\tau) = \sum_{n=0}^{\infty} \frac{\tau^{K+n}}{(K+n)!} \mathbb{O}(K,n), \tag{B.22}$$

where $\mathbb{O}(K,n)$ is K-th order term of $\mathbb{W}_\mathrm{m}$ from polynomial $(\mathbb{W}_\mathrm{m}+\mathbb{W}_\mathrm{r})^{K+n}$

and can be calculated from the recurrence formula, $\mathbb{O}(K,n) = \mathbb{W}_{\rm m}\mathbb{O}(K-1,n)+\mathbb{W}_{\rm r}\mathbb{O}(K,n-1)$. We plot Eqns. B.19 and B.22 for cutoff $n_{\max} = 4096$ in Fig. B.1-(b) in order to compare their precision. As we approximate $\Omega_{\rm S} \simeq \Omega$, we believe Eqn. B.22 shows more accurate results; the $|{\rm S}\rangle$-contribution in Eqn. B.19 causes a minor error in the active phase due to approximated $\Omega_{\rm S}$.

The moment generating function $Z(s,\tau)$ and cumulant generating function $\phi(s,\tau)$ can be calculated from matrix product states:

$$Z(s,\tau) = \langle {\rm e}| \exp(\tau e^{-s}\mathbb{W}_{\rm m} + \tau\mathbb{W}_{\rm r}) |{\rm S}\rangle \,. \tag{B.23}$$

In the 'thermodynamic' limit where τ is long enough, the largest eigenvalue of the matrix $\mathbb{W}_s = e^{-s}\mathbb{W}_{\rm m} + \mathbb{W}_{\rm r}$ gives the large deviation function of $P(K|\tau)$, $\phi(s) = \lim_{\tau\to\infty}\phi(s,\tau)$. As the system has two different dynamical phases, $\phi(s)$ shows a singularity at $s_{\rm c}$

$$\phi(s) = \begin{cases} 0 & s > s_{\rm c} \\ \frac{-\lambda+\sqrt{\lambda^2-4\gamma(s)}}{2} & s \le s_{\rm c} \end{cases} \tag{B.24}$$

where $\gamma(s) = w_b w_u + w_b w_c - w_b w_u e^{-s}$. The second part of equation B.24 is smaller than zero when s is greater than $s_{\rm c}$, which makes $s_{\rm c}$ to the boundary between active and inactive phases. The value of $s^*(\tau)$, as we treated before, always converges to the negative value $s_{\rm c} = -\ln(1 + w_c/w_u)$ from $\gamma(s = s_{\rm c}) = 0$.

$\phi(s,\tau)$ and $s^*(\tau)$ for finite τ are much more complicated. In fact, Eqn. B.23 can be evaluated from an analytical manner, however, the resulting expression is extremely abstruse for handling. Instead, we perform numerical calculations, and also we consider both τ^* and $t_{\rm t}$ are functions of three rate constants: w_b, w_u, and w_c. Then, the chain rule gives a relation between τ^* and $\langle t_{\rm t}\rangle$:

$$\begin{aligned}\frac{d\tau^*}{d\langle t_{\rm t}\rangle} &= \frac{\partial\tau^*}{\partial w_b}\frac{\partial w_b}{\partial\langle t_{\rm t}\rangle} + \frac{\partial\tau^*}{\partial w_u}\frac{\partial w_u}{\partial\langle t_{\rm t}\rangle} + \frac{\partial\tau^*}{\partial w_c}\frac{\partial w_c}{\partial\langle t_{\rm t}\rangle} \\ &= -w_b w_c\left(\frac{w_b}{w_u + w_c}\frac{\partial\tau^*}{\partial w_b} - \frac{\partial\tau^*}{\partial w_u} + \frac{w_c}{w_b + w_u}\frac{\partial\tau^*}{\partial w_c}\right)\end{aligned} \tag{B.25}$$

We plot mean values of turnover times, numerically calculate transition times at various binding, unbinding and catalysis rates in the Fig. B.3. We find that there is strong linear correlations between τ^* and $\langle t_{\rm t}\rangle$. Each data set represents the case where two of the three rate constants are fixed, and the remainder one varies; the linear relation, $d\tau^*/d\langle t_{\rm t}\rangle \sim 1.3$ becomes apparent when $w_u \gg w_c$. Since we let the catalysis stage is irreversible, once a single reaction is over, the number of unbinding events of the given path does not increase any more. It results in the population of the inactive phase is continually increasing as observation time increases and for the active phase, *vice versa*. In the thermodynamic limit, when the time is passed long enough in other words, only inactive paths are survived and $P(K|\tau)$ converges to $\rho(K)$, which is we presented in the previous section. So we can also calculate the value of s_{c} in the large deviation limit from the convergence of Eqn. B.20, $\sum_{K=0}^{\infty} e^{-sK}\rho(K)$. Such preference for the inactive phase in a long observation time scale of the system would causes active-inactive phase transition at τ^* if the reaction process had started from the active phase at short observation time scale. Understandably, the logic can be different depending on the relative rate constants; the phase transition will not be happening if the rate of catalysis, w_c is sufficiently greater than the rate of unbinding, w_u. In that scenario, $s^*(\tau)$ always has negative value even at the very short observation time τ, and the system stays in the inactive phase from beginning till the end of reactions. This principle provides a lower boundary in Fig. (reffig:timescale.

B.4 Conclusions

In the present study, we demonstrate that a series of mathematical formalisms of the statistical thermodynamics in equilibrium systems are also suitable for treating systems in out-of-equilibrium, especially single-molecule enzymatic reactions under the Poissonian Michaelis-Menten mechanism. Three physical observables in nonequilibrium manner -the number of unbinding events, total lifetimes of substrate and enzyme-substrate complex- lead us

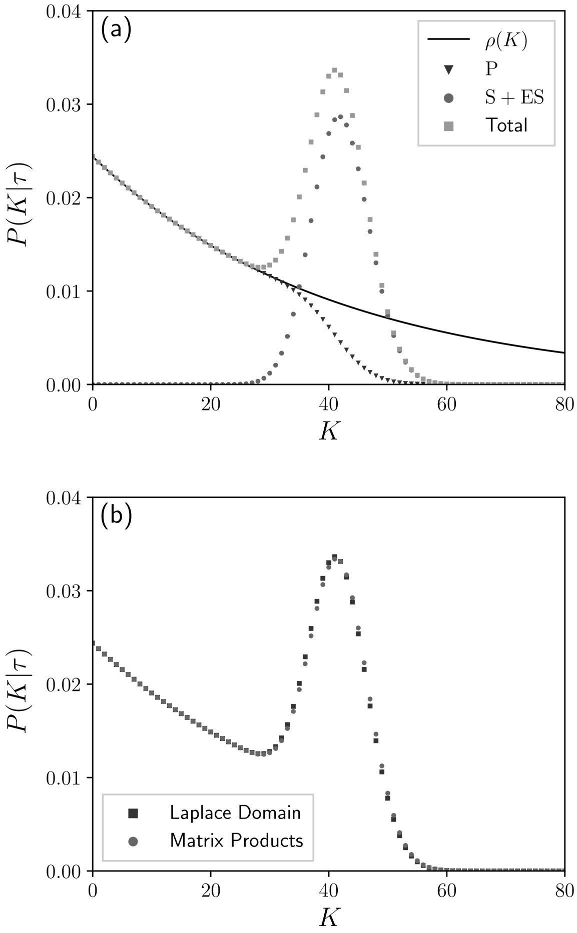


Figure B.1: (a) Conditional probability distribution $P(K|\tau)$, calculated using equation B.19 and inverse Laplace transform. The data obtained under the condition $w_b = 0.5$, $w_u = 1.0$, $w_c = 0.025$, and $\tau = 128$. Red triangles of *completed* paths are maldistributed in inactive state at maximum $K = 0$, while blue squares of *incompleted* paths make active state at maximum $K \simeq 40$. (b) Comparision plot of the equation B.19 (square) and B.22 (circle). The approximation applied for evaluating Ω_S makes subtle deviation in active phase.

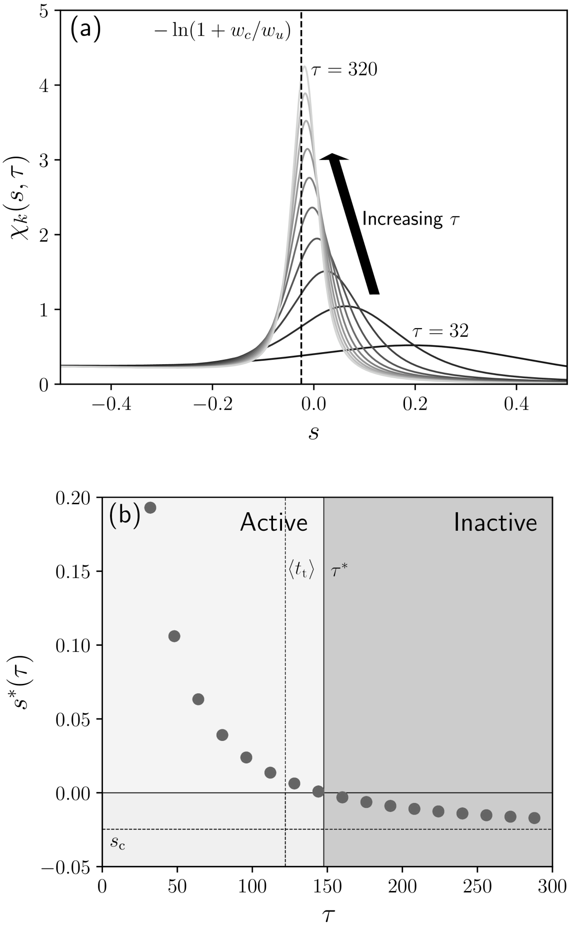


Figure B.2: (a) Susceptibilities of the intensive number of unbinding events, $k = K/\tau$ in various observation time scale and (b) their maximum position $s^*(\tau)$ in variation of the observation time. The dataset is from the condition $w_b = 0.5$, $w_u = 1.0$, and $w_c = 0.025$ s^* converges to negative value, $s_{\mathrm{c}} = -\ln(1 + w_c/w_u)$ in the thermodynamic limit, while it becomes zero at $\tau \simeq 147$ which exhibits coexistence active paths and inactive paths.

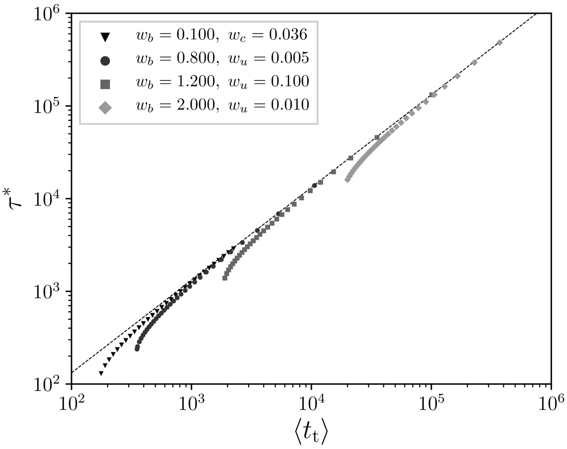


Figure B.3: Relation between mean-turnover times, $\langle t_\mathrm{t} \rangle$ and active-inactive phase transition times,τ^*. Two of three reaction constants are fixed while the remainder one is variating. The black dashed line clarifies that all datasets represent linearly correlated tendency, approximately $d\tau^*/d\,\langle t_\mathrm{t} \rangle \simeq 1.32$ in the large turnover time scale.

to the principle of a *priori* probabilities and the definition of the reaction path entropy. Based on this idea, we successfully evaluated three statistical ensembles of the out-of-equilibrium process: microcanonical $(K, t_\mathrm{ES}, t_\mathrm{S})$, canonical (K, t_ES, μ) and grand canonical (K, ν, μ) ensemble. Conjugate intensive variables in these ensembles, μ and ν bias statistical weights of trajectories, with the lifetimes of components t_S and t_ES, respectively, and one can uncover from the definition of a single reaction path that ν and μ are just escaping ratios of the Markov process. Thermodynamic relations between nonequilibrium ensembles give us probability distributions of several important reaction time scales. Results obtained from the reaction path thermodynamics reproduces previous results based on mean-field theory.

Furthermore, for the considerations of the various theoretical or experimental scenarios, we extended our results for fixed observation time, τ. We evaluate Bayesian statistics and perform numerical calculations in order to demonstrate that the enzymatic reaction has two different dynamical phases, in fact, if one uses the number of unbinding events per the observation time, $k = K/\tau$ as an order parameter. We name these two phases as the inactive (unbinding-poor) phase and the active (unbinding-rich) phase, respectively. Because the system always takes inactive phases when observation time is long enough (in the thermodynamic limit), a first-order phase transition from the active to the inactive phase may appear during the reaction process, depending on the combination of reaction rate constants. The transition time τ^*, which is the timescale that such phase transition appears, show an approximately linear relation with the average value of enzymatic turnover time, $\langle t_{\mathrm{t}} \rangle$.

Since there are various evidences that the unbinding of enzyme-substrate doing a crucial role in the kinetics of complex enzymatic processes, we believe our work proposes a potential way for quantifying dynamical behaviors of systems under the MM mechanism. We will extend our study to general models, especially non-Poisson (or heterogeneous) enzymatic reaction process of the enzymatic reaction process. Also, we expect that our work on the nonequilibrium ensemble theory can be applied to various systems in out-of-equilibrium.

Bibliography

[1] Marenich, A. V.; Cramer, C. J.; Truhlar, D. G. Generalized Born Solvation Model SM12. *Journal of Chemical Theory and Computation* **2013**, *9*, 609–620.

[2] Klamt, A.; Diedenhofen, M. Calculation of Solvation Free Energies with DCOSMO-RS. *The Journal of Physical Chemistry A* **2015**, *119*, 5439–5445.

[3] Klamt, A.; Schüürmann, G. COSMO: a new approach to dielectric screening in solvents with explicit expressions for the screening energy and its gradient. *J. Chem. Soc., Perkin Trans. 2* **1993**, 799–805.

[4] Delaney, J. S. ESOL: Estimating Aqueous Solubility Directly from Molecular Structure. *Journal of Chemical Information and Computer Sciences* **2004**, *44*, 1000–1005.

[5] Tomasi, J.; Mennucci, B.; Cammi, R. Quantum Mechanical Continuum Solvation Models. *Chemical Reviews* **2005**, *105*, 2999–3094.

[6] Cramer, C. J.; Truhlar, D. G. A Universal Approach to Solvation Modeling. *Accounts of Chemical Research* **2008**, *41*, 760–768.

[7] Marenich, A. V.; Cramer, C. J.; Truhlar, D. G. Universal Solvation Model Based on Solute Electron Density and on a Continuum Model of the Solvent Defined by the Bulk Dielectric Constant and Atomic Surface Tensions. *The Journal of Physical Chemistry B* **2009**, *113*, 6378–6396.

[8] Klamt, A.; Eckert, F.; Arlt, W. COSMO-RS: An Alternative to Simulation for Calculating Thermodynamic Properties of Liquid Mixtures. *Annual Review of Chemical and Biomolecular Engineering* **2010**, *1*, 101–122.

[9] Shivakumar, D.; Williams, J.; Wu, Y.; Damm, W.; Shelley, J.; Sherman, W. Prediction of Absolute Solvation Free Energies using Molecular Dynamics Free Energy Perturbation and the OPLS Force Field. *Journal of Chemical Theory and Computation* **2010**, *6*, 1509–1519.

[10] Chong, S.-H.; Ham, S. Atomic decomposition of the protein solvation free energy and its application to amyloid-beta protein in water. *The Journal of Chemical Physics* **2011**, *135*, 034506.

[11] Mennucci, B. Polarizable continuum model: Polarizable continuum

model. *Wiley Interdisciplinary Reviews: Computational Molecular Science* **2012**, *2*, 386–404.

[12] Sato, H. A modern solvation theory: quantum chemistry and statistical chemistry. *Physical Chemistry Chemical Physics* **2013**, *15*, 7450.

[13] König, G.; Pickard, F. C.; Mei, Y.; Brooks, B. R. Predicting hydration free energies with a hybrid QM/MM approach: an evaluation of implicit and explicit solvation models in SAMPL4. *Journal of Computer-Aided Molecular Design* **2014**, *28*, 245–257.

[14] Mobley, D. L.; Guthrie, J. P. FreeSolv: a database of experimental and calculated hydration free energies, with input files. *Journal of Computer-Aided Molecular Design* **2014**, *28*, 711–720.

[15] Skyner, R. E.; McDonagh, J. L.; Groom, C. R.; van Mourik, T.; Mitchell, J. B. O. A review of methods for the calculation of solution free energies and the modelling of systems in solution. *Physical Chemistry Chemical Physics* **2015**, *17*, 6174–6191.

[16] Zhang, J.; Tuguldur, B.; van der Spoel, D. Force Field Benchmark of Organic Liquids. 2. Gibbs Energy of Solvation. *Journal of Chemical Information and Modeling* **2015**, *55*, 1192–1201.

[17] Harder, E. et al. OPLS3: A Force Field Providing Broad Coverage of

Drug-like Small Molecules and Proteins. *Journal of Chemical Theory and Computation* **2016**, *12*, 281–296.

[18] Coley, C. W.; Barzilay, R.; Green, W. H.; Jaakkola, T. S.; Jensen, K. F. Convolutional Embedding of Attributed Molecular Graphs for Physical Property Prediction. *Journal of Chemical Information and Modeling* **2017**, *57*, 1757–1772.

[19] Duarte Ramos Matos, G.; Kyu, D. Y.; Loeffler, H. H.; Chodera, J. D.; Shirts, M. R.; Mobley, D. L. Approaches for Calculating Solvation Free Energies and Enthalpies Demonstrated with an Update of the FreeSolv Database. *Journal of Chemical & Engineering Data* **2017**, *62*, 1559–1569.

[20] Wu, Z.; Ramsundar, B.; Feinberg, E.; Gomes, J.; Geniesse, C.; Pappu, A. S.; Leswing, K.; Pande, V. MoleculeNet: a benchmark for molecular machine learning. *Chemical Science* **2018**, *9*, 513–530.

[21] Borhani, T. N.; García-Muñoz, S.; Vanesa Luciani, C.; Galindo, A.; Adjiman, C. S. Hybrid QSPR models for the prediction of the free energy of solvation of organic solute/solvent pairs. *Physical Chemistry Chemical Physics* **2019**, *21*, 13706–13720.

[22] Lim, H.; Jung, Y. Delfos: deep learning model for prediction of sol-

vation free energies in generic organic solvents. *Chemical Science* **2019**, *10*, 8306–8315.

[23] Popova, M.; Isayev, O.; Tropsha, A. Deep reinforcement learning for de novo drug design. *Science Advances* **2018**, *4*, eaap7885.

[24] Reichardt, C.; Welton, T. *Solvents and Solvent Effects in Organic Chemistry: REICHARDT:SOLV.EFF. 4ED O-BK*; Wiley-VCH Verlag GmbH & Co. KGaA: Weinheim, Germany, 2010.

[25] Takeda, T.; Taniki, R.; Masuda, A.; Honma, I.; Akutagawa, T. Electron-deficient anthraquinone derivatives as cathodic material for lithium ion batteries. *Journal of Power Sources* **2016**, *328*, 228–234.

[26] Park, H.; Lim, H.-D.; Lim, H.-K.; Seong, W. M.; Moon, S.; Ko, Y.; Lee, B.; Bae, Y.; Kim, H.; Kang, K. High-efficiency and high-power rechargeable lithium–sulfur dioxide batteries exploiting conventional carbonate-based electrolytes. *Nature Communications* **2017**, *8*, 14989.

[27] Allam, O.; Cho, B. W.; Kim, K. C.; Jang, S. S. Application of DFT-based machine learning for developing molecular electrode materials in Li-ion batteries. *RSC Advances* **2018**, *8*, 39414–39420.

[28] Kim, J.; Ko, S.; Noh, C.; Kim, H.; Lee, S.; Kim, D.; Park, H.; Kwon, G.; Son, G.; Ko, J. W.; Jung, Y.; Lee, D.; Park, C. B.; Kang, K.

Biological Nicotinamide Cofactor as a Redox-Active Motif for Reversible Electrochemical Energy Storage. *Angewandte Chemie International Edition* **2019**, *58*, 16764–16769.

[29] Jia, X.; Wang, M.; Shao, Y.; König, G.; Brooks, B. R.; Zhang, J. Z. H.; Mei, Y. Calculations of Solvation Free Energy through Energy Reweighting from Molecular Mechanics to Quantum Mechanics. *Journal of Chemical Theory and Computation* **2016**, *12*, 499–511.

[30] Tropsha, A. Best Practices for QSAR Model Development, Validation, and Exploitation. *Molecular Informatics* **2010**, *29*, 476–488.

[31] Cherkasov, A. et al. QSAR Modeling: Where Have You Been? Where Are You Going To? *Journal of Medicinal Chemistry* **2014**, *57*, 4977–5010.

[32] Schmidhuber, J. Deep learning in neural networks: An overview. *Neural Networks* **2015**, *61*, 85–117.

[33] Butler, K. T.; Davies, D. W.; Cartwright, H.; Isayev, O.; Walsh, A. Machine learning for molecular and materials science. *Nature* **2018**, *559*, 547–555.

[34] Kearnes, S.; McCloskey, K.; Berndl, M.; Pande, V.; Riley, P. Molec-

ular graph convolutions: moving beyond fingerprints. *Journal of Computer-Aided Molecular Design* **2016**, *30*, 595–608.

[35] Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; Dahl, G. E. Neural Message Passing for Quantum Chemistry. *arXiv:1704.01212 [cs]* **2017**, arXiv: 1704.01212.

[36] Schütt, K. T.; Arbabzadah, F.; Chmiela, S.; Müller, K. R.; Tkatchenko, A. Quantum-chemical insights from deep tensor neural networks. *Nature Communications* **2017**, *8*, 13890.

[37] Smith, J. S.; Isayev, O.; Roitberg, A. E. ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost. *Chemical Science* **2017**, *8*, 3192–3203.

[38] Ryu, S.; Lim, J.; Hong, S. H.; Kim, W. Y. Deeply learning molecular structure-property relationships using attention- and gate-augmented graph convolutional network. *arXiv:1805.10988 [cs, stat]* **2018**, arXiv: 1805.10988.

[39] Schütt, K. T.; Sauceda, H. E.; Kindermans, P.-J.; Tkatchenko, A.; Müller, K.-R. SchNet – A deep learning architecture for molecules and materials. *The Journal of Chemical Physics* **2018**, *148*, 241722.

[40] Sifain, A. E.; Lubbers, N.; Nebgen, B. T.; Smith, J. S.; Lokhov, A. Y.; Isayev, O.; Roitberg, A. E.; Barros, K.; Tretiak, S. Discovering a

Transferable Charge Assignment Model Using Machine Learning. *The Journal of Physical Chemistry Letters* **2018**, *9*, 4495–4501.

[41] Ryu, S.; Kwon, Y.; Kim, W. Y. A Bayesian graph convolutional network for reliable prediction of molecular properties with uncertainty quantification. *Chemical Science* **2019**, *10*, 8438–8446.

[42] Schütt, K. T.; Kessel, P.; Gastegger, M.; Nicoli, K. A.; Tkatchenko, A.; Müller, K.-R. SchNetPack: A Deep Learning Toolbox For Atomistic Systems. *Journal of Chemical Theory and Computation* **2019**, *15*, 448–455.

[43] Winter, R.; Montanari, F.; Noé, F.; Clevert, D.-A. Learning continuous and data-driven molecular descriptors by translating equivalent chemical representations. *Chemical Science* **2019**, *10*, 1692–1701.

[44] Mohamed, N. A.; Bradshaw, R. T.; Essex, J. W. Evaluation of solvation free energies for small molecules with the AMOEBA polarizable force field. *Journal of Computational Chemistry* **2016**, *37*, 2749–2758.

[45] Kromann, J. C.; Steinmann, C.; Jensen, J. H. Improving solvation energy predictions using the SMD solvation method and semiempirical electronic structure methods. *The Journal of Chemical Physics* **2018**, *149*, 104102.

[46] Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed Representations of Words and Phrases and their Compositionality. *arXiv:1310.4546 [cs, stat]* **2013**, arXiv: 1310.4546.

[47] Pennington, J.; Socher, R.; Manning, C. Glove: Global Vectors for Word Representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Doha, Qatar, 2014; pp 1532–1543.

[48] Bahdanau, D.; Cho, K.; Bengio, Y. Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv:1409.0473 [cs, stat]* **2016**, arXiv: 1409.0473.

[49] Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhutdinov, R.; Zemel, R.; Bengio, Y. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. *arXiv:1502.03044 [cs]* **2016**, arXiv: 1502.03044.

[50] Luong, M.-T.; Pham, H.; Manning, C. D. Effective Approaches to Attention-based Neural Machine Translation. *arXiv:1508.04025 [cs]* **2015**, arXiv: 1508.04025.

[51] Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient Estimation of Word Representations in Vector Space. *arXiv:1301.3781 [cs]* **2013**, arXiv: 1301.3781.

[52] Asgari, E.; Mofrad, M. R. K. Continuous Distributed Representation of Biological Sequences for Deep Proteomics and Genomics. *PLOS ONE* **2015**, *10*, e0141287.

[53] Goh, G. B.; Hodas, N. O.; Siegel, C.; Vishnu, A. SMILES2Vec: An Interpretable General-Purpose Deep Neural Network for Predicting Chemical Properties. *arXiv:1712.02034 [cs, stat]* **2018**, arXiv: 1712.02034.

[54] Jaeger, S.; Fulle, S.; Turk, S. Mol2vec: Unsupervised Machine Learning Approach with Chemical Intuition. *Journal of Chemical Information and Modeling* **2018**, *58*, 27–35.

[55] Morgan, H. L. The Generation of a Unique Machine Description for Chemical Structures-A Technique Developed at Chemical Abstracts Service. *Journal of Chemical Documentation* **1965**, *5*, 107–113.

[56] Rogers, D.; Hahn, M. Extended-Connectivity Fingerprints. *Journal of Chemical Information and Modeling* **2010**, *50*, 742–754.

[57] Cereto-Massagué, A.; Ojeda, M. J.; Valls, C.; Mulero, M.; Garcia-Vallvé, S.; Pujadas, G. Molecular fingerprint similarity search in virtual screening. *Methods* **2015**, *71*, 58–63.

[58] Schuster, M.; Paliwal, K. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing* **1997**, *45*, 2673–2681.

[59] Bengio, Y.; Simard, P.; Frasconi, P. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks* **1994**, *5*, 157–166.

[60] Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Computation* **1997**, *9*, 1735–1780.

[61] Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *arXiv:1412.3555 [cs]* **2014**, arXiv: 1412.3555.

[62] Marenich, A. V.; Kelly, C. P.; Thompson, J. D.; Hawkins, G. D.; Chambers, C. C.; Giesen, D. J.; Winget, P.; Cramer, C. J.; Truhlar, D. G. *Minnesota Solvation Database – version 2012*; 2012; Published: University of Minnesota, Minneapolis.

[63] Swain, M.; Kurniawan, E.; Powers, Z.; Yi, H.; Lazzaro, L.; Dahlgren, B.; Sjorgen, R. *PubChemPy*; 2014.

[64] Chollet, F.; others, *Keras*; 2015.

[65] Martín Abadi, et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*; 2015.

[66] Pedregosa, F. et al. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* **2011**, *12*, 2825–2830.

[67] Lusci, A.; Pollastri, G.; Baldi, P. Deep Architectures and Deep Learning in Chemoinformatics: The Prediction of Aqueous Solubility for Drug-Like Molecules. *Journal of Chemical Information and Modeling* **2013**, *53*, 1563–1575.

[68] Zheng, X.; Liu, Q.; Jing, C.; Li, Y.; Li, D.; Luo, W.; Wen, Y.; He, Y.; Huang, Q.; Long, Y.-T.; Fan, C. Catalytic Gold Nanoparticles for Nanoplasmonic Detection of DNA Hybridization. *Angewandte Chemie International Edition* **2011**, *50*, 11994–11998.

[69] Genheden, S. Solvation free energies and partition coefficients with the coarse-grained and hybrid all-atom/coarse-grained MARTINI models. *Journal of Computer-Aided Molecular Design* **2017**, *31*, 867–876.

[70] Dupont, C.; Andreussi, O.; Marzari, N. Self-consistent continuum solvation (SCCS): The case of charged systems. *The Journal of Chemical Physics* **2013**, *139*, 214110.

[71] Sundararaman, R.; Goddard, W. A. The charge-asymmetric nonlocally determined local-electric (CANDLE) solvation model. *The Journal of Chemical Physics* **2015**, *142*, 064107.

[72] Klamt, A. The COSMO and COSMO-RS solvation models: COSMO

and COSMO-RS. *Wiley Interdisciplinary Reviews: Computational Molecular Science* **2018**, *8*, e1338.

[73] Klamt, A. Conductor-like Screening Model for Real Solvents: A New Approach to the Quantitative Calculation of Solvation Phenomena. *The Journal of Physical Chemistry* **1995**, *99*, 2224–2235.

[74] Mayr, A.; Klambauer, G.; Unterthiner, T.; Steijaert, M.; Wegner, J. K.; Ceulemans, H.; Clevert, D.-A.; Hochreiter, S. Large-scale comparison of machine learning methods for drug target prediction on ChEMBL. *Chemical Science* **2018**, *9*, 5441–5451.

[75] Marenich, A. V.; Olson, R. M.; Kelly, C. P.; Cramer, C. J.; Truhlar, D. G. Self-Consistent Reaction Field Model for Aqueous and Nonaqueous Solutions Based on Accurate Polarized Partial Charges. *Journal of Chemical Theory and Computation* **2007**, *3*, 2011–2033.

[76] Zheng, S.; Yan, X.; Yang, Y.; Xu, J. Identifying Structure–Property Relationships through SMILES Syntax Analysis with Self-Attention Mechanism. *Journal of Chemical Information and Modeling* **2019**, *59*, 914–923.

[77] Mikolov, T.; Kopecky, J.; Burget, L.; Glembek, O.; Cernocky, J. Neural network based language models for highly inflective languages.

2009 IEEE International Conference on Acoustics, Speech and Signal Processing. Taipei, Taiwan, 2009; pp 4725–4728.

[78] Stolov, M. A.; Zaitseva, K. V.; Varfolomeev, M. A.; Acree, W. E. Enthalpies of solution and enthalpies of solvation of organic solutes in ethylene glycol at 298.15 K: Prediction and analysis of intermolecular interaction contributions. *Thermochimica Acta* **2017**, *648*, 91–99.

[79] Sedov, I. A.; Salikov, T. M.; Wadawadigi, A.; Zha, O.; Qian, E.; Acree, W. E.; Abraham, M. H. Abraham model correlations for describing the thermodynamic properties of solute transfer into pentyl acetate based on headspace chromatographic and solubility measurements. *The Journal of Chemical Thermodynamics* **2018**, *124*, 133–140.

[80] Peters, M. E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; Zettlemoyer, L. Deep contextualized word representations. *arXiv:1802.05365 [cs]* **2018**, arXiv: 1802.05365.

[81] Kipf, T. N.; Welling, M. Semi-Supervised Classification with Graph Convolutional Networks. *arXiv:1609.02907 [cs, stat]* **2017**, arXiv: 1609.02907.

[82] Sterling, T.; Irwin, J. J. ZINC 15 – Ligand Discovery for Every-

one. *Journal of Chemical Information and Modeling* **2015**, *55*, 2324–2337.

[83] Searles, D. J.; Evans, D. J. The fluctuation theorem and Green–Kubo relations. *The Journal of Chemical Physics* **2000**, *112*, 9727–9735.

[84] Kotsiantis, S. B. Decision trees: a recent overview. *Artificial Intelligence Review* **2013**, *39*, 261–283.

[85] Breiman, L. Random Forests. *Machine Learning* **2001**, *45*, 5–32.

[86] Friedman, J. H. Greedy Function Approximation: A Gradient Boosting Machine. *The Annals of Statistics* **2001**, *29*, 1189–1232.

[87] Thomson, G. H. The DIPPR databases. *International Journal of Thermophysics* **1996**, *17*, 223–232.

[88] Whitesides, G.; Mathias, J.; Seto, C. Molecular self-assembly and nanochemistry: a chemical strategy for the synthesis of nanostructures. *Science* **1991**, *254*, 1312–1319.

[89] Rothemund, P. W. K. Folding DNA to create nanoscale shapes and patterns. *Nature* **2006**, *440*, 297–302.

[90] Prybytak, P.; Frith, W. J.; Cleaver, D. J. Hierarchical self-assembly of chiral fibres from achiral particles. *Interface Focus* **2012**, *2*, 651–657.

[91] Whitesides, G. M.; Boncheva, M. Beyond molecules: Self-assembly of mesoscopic and macroscopic components. *Proceedings of the National Academy of Sciences* **2002**, *99*, 4769–4774.

[92] Stupp, S. I. Self-Assembly and Biomaterials. *Nano Letters* **2010**, *10*, 4783–4786.

[93] Glotzer, S. C.; Solomon, M. J. Anisotropy of building blocks and their assembly into complex structures. *Nature Materials* **2007**, *6*, 557–562.

[94] Langer, R.; Tirrell, D. A. Designing materials for biology and medicine. *Nature* **2004**, *428*, 487–492.

[95] Klotsa, D.; Jack, R. L. Predicting the self-assembly of a model colloidal crystal. *Soft Matter* **2011**, *7*, 6294.

[96] Whitelam, S.; Hedges, L. O.; Schmit, J. D. Self-Assembly at a Nonequilibrium Critical Point. *Physical Review Letters* **2014**, *112*, 155504.

[97] Whitelam, S.; Geissler, P. L. Avoiding unphysical kinetic traps in Monte Carlo simulations of strongly attractive particles. *The Journal of Chemical Physics* **2007**, *127*, 154101.

[98] Whitelam, S.; Jack, R. L. The Statistical Mechanics of Dynamic Path-

ways to Self-Assembly. *Annual Review of Physical Chemistry* **2015**, *66*, 143–163.

[99] Grant, J.; Jack, R. L.; Whitelam, S. Analyzing mechanisms and microscopic reversibility of self-assembly. *The Journal of Chemical Physics* **2011**, *135*, 214505.

[100] Klotsa, D.; Jack, R. L. Controlling crystal self-assembly using a real-time feedback scheme. *The Journal of Chemical Physics* **2013**, *138*, 094502.

[101] Whitelam, S.; Feng, E. H.; Hagan, M. F.; Geissler, P. L. The role of collective motion in examples of coarsening and self-assembly. *Soft Matter* **2009**, *5*, 1251–1262.

[102] Jack, R. L.; Hagan, M. F.; Chandler, D. Fluctuation-dissipation ratios in the dynamics of self-assembly. *Physical Review E* **2007**, *76*, 021119.

[103] Grant, J.; Jack, R. L. Quantifying reversibility in a phase-separating lattice gas: An analogy with self-assembly. *Physical Review E* **2012**, *85*, 021112.

[104] Perkett, M. R.; Hagan, M. F. Using Markov state models to study self-assembly. *The Journal of Chemical Physics* **2014**, *140*, 214101.

[105] Rapaport, D. C. Role of Reversibility in Viral Capsid Growth: A Paradigm for Self-Assembly. *Physical Review Letters* **2008**, *101*, 186101.

[106] Garrahan, J. P.; Jack, R. L.; Lecomte, V.; Pitard, E.; van Duijvendijk, K.; van Wijland, F. Dynamical First-Order Phase Transition in Kinetically Constrained Models of Glasses. *Physical Review Letters* **2007**, *98*, 195702.

[107] Garrahan, J. P.; Jack, R. L.; Lecomte, V.; Pitard, E.; van Duijvendijk, K.; van Wijland, F. First-order dynamical phase transition in models of glasses: an approach based on ensembles of histories. *Journal of Physics A: Mathematical and Theoretical* **2009**, *42*, 075007.

[108] Garrahan, J. P. Classical stochastic dynamics and continuous matrix product states: gauge transformations, conditioned and driven processes, and equivalence of trajectory ensembles. *Journal of Statistical Mechanics: Theory and Experiment* **2016**, *2016*, 073208.

[109] Jack, R. L.; Sollich, P. Large Deviations and Ensembles of Trajectories in Stochastic Models. *Progress of Theoretical Physics Supplement* **2010**, *184*, 304–317.

[110] Chetrite, R.; Touchette, H. Nonequilibrium Microcanonical and

Canonical Ensembles and Their Equivalence. *Physical Review Letters* **2013**, *111*, 120601.

[111] Vaikuntanathan, S.; Gingrich, T. R.; Geissler, P. L. Dynamic phase transitions in simple driven kinetic networks. *Physical Review E* **2014**, *89*, 062108.

[112] Budini, A. A.; Turner, R. M.; Garrahan, J. P. Fluctuating observation time ensembles in the thermodynamics of trajectories. *Journal of Statistical Mechanics: Theory and Experiment* **2014**, *2014*, P03012.

[113] Lecomte, V.; Appert-Rolland, C.; van Wijland, F. Thermodynamic Formalism for Systems with Markov Dynamics. *Journal of Statistical Physics* **2007**, *127*, 51–106.

[114] Hedges, L. O.; Jack, R. L.; Garrahan, J. P.; Chandler, D. Dynamic Order-Disorder in Atomistic Models of Structural Glass Formers. *Science* **2009**, *323*, 1309–1313.

[115] Touchette, H. The large deviation approach to statistical mechanics. *Physics Reports* **2009**, *478*, 1–69.

[116] Speck, T.; Chandler, D. Constrained dynamics of localized excitations causes a non-equilibrium phase transition in an atomistic model of glass formers. *The Journal of Chemical Physics* **2012**, *136*, 184509.

[117] Speck, T.; Malins, A.; Royall, C. P. First-Order Phase Transition in a Model Glass Former: Coupling of Local Structure and Dynamics. *Physical Review Letters* **2012**, *109*, 195703.

[118] Jack, R. L.; Hedges, L. O.; Garrahan, J. P.; Chandler, D. Preparation and Relaxation of Very Stable Glassy States of a Simulated Liquid. *Physical Review Letters* **2011**, *107*, 275702.

[119] Gaspard, P. Time-Reversed Dynamical Entropy and Irreversibility in Markovian Random Processes. *Journal of Statistical Physics* **2004**, *117*, 599–615.

[120] Gardiner, C. W. *Handbook of Stochastic Methods: For Physics, Chemistry and Natural Sciences*; Springer, 1985.

[121] Bortz, A.; Kalos, M.; Lebowitz, J. A new algorithm for Monte Carlo simulation of Ising spin systems. *Journal of Computational Physics* **1975**, *17*, 10–18.

[122] Bolhuis, P. G.; Chandler, D.; Dellago, C.; Geissler, P. L. Transition path sampling: throwing ropes over rough mountain passes, in the dark. *Annual Review of Physical Chemistry* **2002**, *53*, 291–318.

[123] Bennett, C. H. Efficient estimation of free energy differences from Monte Carlo data. *Journal of Computational Physics* **1976**, *22*, 245–268.

[124] Shirts, M. R.; Chodera, J. D. Statistically optimal analysis of samples from multiple equilibrium states. *The Journal of Chemical Physics* **2008**, *129*, 124105.

[125] Elmatad, Y. S.; Jack, R. L.; Chandler, D.; Garrahan, J. P. Finite-temperature critical point of a glass transition. *Proceedings of the National Academy of Sciences* **2010**, *107*, 12793–12798.

[126] Michaelis, L.; Menten, M. L. Die Kinetik der Invertinwirkung. *Biochem. Z.* **1913**, *49*, 333.

[127] Johnson, K. A.; Goody, R. S. The Original Michaelis Constant: Translation of the 1913 Michaelis–Menten Paper. *Biochemistry* **2011**, *50*, 8264–8269.

[128] Cornish-Bowden, A. One hundred years of Michaelis–Menten kinetics. *Perspectives in Science* **2015**, *4*, 3–9.

[129] Ronen, M.; Rosenberg, R.; Shraiman, B. I.; Alon, U. Assigning numbers to the arrows: Parameterizing a gene regulation network by using accurate expression kinetics. *Proceedings of the National Academy of Sciences* **2002**, *99*, 10555–10560.

[130] Pulkkinen, O.; Metzler, R. Variance-corrected Michaelis-Menten equation predicts transient rates of single-enzyme reactions and re-

sponse times in bacterial gene-regulation. *Scientific Reports* **2015**, *5*, 17820.

[131] Schnitzer, M. J.; Visscher, K.; Block, S. M. Force production by single kinesin motors. *Nature Cell Biology* **2000**, *2*, 718–723.

[132] Asbury, C. L. Kinesin Moves by an Asymmetric Hand-Over-Hand Mechanism. *Science* **2003**, *302*, 2130–2134.

[133] van Oijen, A. M. Single-Molecule Kinetics of Exonuclease Reveal Base Dependence and Dynamic Disorder. *Science* **2003**, *301*, 1235–1238.

[134] English, B. P.; Min, W.; van Oijen, A. M.; Lee, K. T.; Luo, G.; Sun, H.; Cherayil, B. J.; Kou, S. C.; Xie, X. S. Ever-fluctuating single enzyme molecules: Michaelis-Menten equation revisited. *Nature Chemical Biology* **2006**, *2*, 87–94.

[135] Cao, J.; Silbey, R. J. Generic Schemes for Single-Molecule Kinetics. 1: Self-Consistent Pathway Solutions for Renewal Processes. *The Journal of Physical Chemistry B* **2008**, *112*, 12867–12880.

[136] Qian, H.; Bishop, L. M. The Chemical Master Equation Approach to Nonequilibrium Steady-State of Open Biochemical Systems: Linear Single-Molecule Enzyme Kinetics and Nonlinear Biochemical Re-

action Networks. *International Journal of Molecular Sciences* **2010**, *11*, 3472–3500.

[137] Jung, W.; Yang, S.; Sung, J. Novel Chemical Kinetics for a Single Enzyme Reaction: Relationship between Substrate Concentration and the Second Moment of Enzyme Reaction Time. *The Journal of Physical Chemistry B* **2010**, *114*, 9840–9847.

[138] Yang, S.; Cao, J.; Silbey, R.; Sung, J. Quantitative Interpretation of the Randomness in Single Enzyme Turnover Times. *Biophysical Journal* **2011**, *101*, 519–524.

[139] Reuveni, S.; Urbakh, M.; Klafter, J. Role of substrate unbinding in Michaelis-Menten enzymatic reactions. *Proceedings of the National Academy of Sciences* **2014**, *111*, 4391–4396.

[140] Rotbart, T.; Reuveni, S.; Urbakh, M. Michaelis-Menten reaction scheme as a unified approach towards the optimal restart problem. *Physical Review E* **2015**, *92*, 060101.

[141] Park, S. J.; Song, S.; Jeong, I.-C.; Koh, H. R.; Kim, J.-H.; Sung, J. Nonclassical Kinetics of Clonal yet Heterogeneous Enzymes. *The Journal of Physical Chemistry Letters* **2017**, *8*, 3152–3158.

[142] Qian, H.; L. Elson, E. Single-molecule enzymology: stochastic

Michaelis–Menten kinetics. *Biophysical Chemistry* **2002**, *101-102*, 565–576.

[143] Kou, S. C.; Cherayil, B. J.; Min, W.; English, B. P.; Xie, X. S. Single-Molecule MichaelisMenten Equations. *The Journal of Physical Chemistry B* **2005**, *109*, 19068–19081.

[144] Murugan, A.; Vaikuntanathan, S. Biological Implications of Dynamical Phases in Non-equilibrium Networks. *Journal of Statistical Physics* **2016**, *162*, 1183–1202.

[145] Klymko, K.; Garrahan, J. P.; Whitelam, S. Similarity of ensembles of trajectories of reversible and irreversible growth processes. *Physical Review E* **2017**, *96*, 042126.

[146] Whitelam, S. Large deviations in the presence of cooperativity and slow dynamics. *Physical Review E* **2018**, *97*, 062109.

[147] Klymko, K.; Geissler, P. L.; Garrahan, J. P.; Whitelam, S. Rare behavior of growth processes via umbrella sampling of trajectories. *Physical Review E* **2018**, *97*, 032123.

[148] Weber, J. K.; Jack, R. L.; Pande, V. S. Emergence of Glass-like Behavior in Markov State Models of Protein Folding Dynamics. *Journal of the American Chemical Society* **2013**, *135*, 5501–5504.

[149] Tuckerman, M. E. *Statistical mechanics: theory and molecular simulation*; Oxford University Press, 2010.

[150] Min, W.; Jiang, L.; Yu, J.; Kou, S. C.; Qian, H.; Xie, X. S. Nonequilibrium Steady State of a Nanometric Biochemical System: Determining the Thermodynamic Driving Force from Single Enzyme Turnover Time Traces. *Nano Letters* **2005**, *5*, 2373–2378.

[151] Sinitsyn, N. A.; Nemenman, I. The Berry phase and the pump flux in stochastic chemical kinetics. *Europhysics Letters (EPL)* **2007**, *77*, 58001.

국문초록

최근 기계학습 기술의 급격한 발전과 이의 화학 분야에 대한 적용은 다양한 화학적 성질에 대한 구조-성질 정량 관계를 기반으로 한 예측 모형의 개발을 가속하고 있다. 용매화 자유 에너지는 그러한 기계학습의 적용 예중 하나이며 다양한 용매 내의 화학반응에서 중요한 역할을 하는 근본적 성질 중 하나이다. 본 연구에서 우리는 목표로 하는 용매화 자유 에너지를 원자간의 상호작용으로부터 구할 수 있는 새로운 심층학습 기반 용매화 모형을 소개한다. 제안된 심층학습 모형의 계산 과정은 용매와 용질 분자에 대한 부호화 함수가 각 원자와 분자들의 구조적 성질에 대한 벡터 표현을 추출하며, 이를 토대로 원자간 상호작용을 복잡한 퍼셉트론 신경망 대신 벡터간의 간단한 내적으로 구할 수 있다. 952가지의 유기용질과 147가지의 유기용매를 포함하는 6,493가지의 실험치를 토대로 기계학습 모형의 교차 검증 시험을 실시한 결과, 평균 절대 오차 기준 0.2 kcal/mol 수준으로 매우 높은 정확도를 가진다. 스캐폴드-기반 교차 검증의 결과 역시 0.6 kcal/mol 수준으로, 외삽으로 분류할 수 있는 비교적 새로운 분자 구조에 대한 예측에 대해서도 우수한 정확도를 보인다. 또한, 제안된

특정 기계학습 모형은 그 구조 상 특정 용매에 특화되지 않았기 때문에 높은 양도성을 가지며 학습에 이용할 데이터의 수를 늘이는 데 용이하다. 원자간 상호작용에 대한 분석을 통해 제안된 심층학습 모형 용매화 자유 에너지에 대한 그룹-기여도를 잘 재현할 수 있음을 알 수 있으며, 기계학습을 통해 단순히 목표로 하는 성질만을 예측하는 것을 넘어 더욱 상세한 물리화학적 이해를 하는 것이 가능할 것이라 기대할 수 있다.

주요어: 심층학습, 구조-성질 정량 관계, 용매화 자유 에너지, 용해도, 액체 성질, 액체계

학번: 2010-23098