Surrogate Molecular Dynamics Simulation Model for Dielectric Constants with Ensemble Neural Networks

*Tong Gao[1], Cameron J. Shock[1], Mark J. Stevens[2], Amalie L. Frischknecht[2], and Issei Nakamura[1]**

[1]Department of Physics, Michigan Technological University, Houghton, Michigan 49931, United States

[2]Center for Integrated Nanotechnologies, Sandia National Laboratories, Albuquerque, New Mexico 87185, United States

*inakamur@mtu.edu

We develop ensemble neural networks (ENN) that serve as computationally fast surrogate models of Stockmayer fluid (SF) molecular dynamics (MD) simulations for determining the dielectric constants of polar solvents and NaCl solutions. The ENNs are trained using 50-times less data than is used to calculate the dielectric constants from MD simulations. The predictions of ENNs trained on this small amount of data and using batch normalization (BN) or bagging are in relatively good agreement with the full MD results. These ENN methods are thus able to extract reliable values from statistically noisy data.

INTRODUCTION

The discovery and invention of novel materials are ever-growing interests in materials science research, and an in-depth understanding of the atomistic or molecular mechanisms of chemical and physical properties is often required when fabricating and controlling electrochemical devices. From this perspective, the computational modeling of materials is an essential methodology for addressing these objectives and bypassing undesired processes before performing experiments. Specifically, the use of neural network (NN) techniques enables the construction of surrogate models that capture highly nonlinear features that are dependent on a number of model parameters, which used to be a challenging task when performed using traditional optimization algorithms.

Of particular importance in materials science is that NNs can considerably reduce the required computational expense and burden when applied to molecular dynamics (MD) simulations, as largely reviewed in Ref. [1]. Recent examples of the application of NNs in MD simulations include NNs trained to calculate the free energy of coarse-grained molecules [2], NNs trained on quantum molecular dynamics configurations to obtain the dielectric constant of water [3], machine-learning (ML) potentials obtained from NNs for the structure and permeability of water [4], NNs used to produce a Hamiltonian for an electronic structure [5], and an NN-assisted MD method to reduce the computations required when conducting open-boundary simulations [6].

Ensembles of NNs are straightforward methods for enhancing the versatility of NNs [7] and for improving the accuracy of the resulting models when combined with the optimal linear combination of trained NNs [8]. Typically, an ensemble of NNs reduces the variance of prediction errors made by constituent models and enhances model capability (generalization) to adapt to previously unseen data. The effectiveness of ensemble NNs (ENNs) in various systems has been explored for a few decades, but the applications involving MD simulations are still limited. These examples include ENNs for determining the potential energy surface of a bimolecular reaction [9], a combination of different ML models for studying the dynamics of water droplets [10], and the analysis of the potential energy surfaces of various Lennard–Jones fluids with two types of ENNs [11]. Nevertheless, the study of ENNs combined with MD simulations remains significantly limited, particularly with respect to the dielectric properties of materials [3].

The parametrization of simulation models and the statistical convergence of noisy simulation data are common challenges in computational material studies and designs. The former issue is generic in most MD simulations, as seen when determining the parameters of interaction potentials. Statistically noisy simulations often require large numbers of samples to achieve good statistical convergence. These two issues together are particularly challenging. Often, long MD simulations must be run iteratively to determine optimal model

parameters. The computational expense of performing many long simulations can make such parameterizations intractable. In particular, MD simulations involving electrostatic interactions tend to be computationally expensive because of the long-range nature of the electrostatic interactions. Accurate simulations require algorithms such as the particle mesh Ewald summation that scale as $O(N \log N)$ with the number of particles N, respectively. This computational expense can be substantially mitigated by reducing the number N of particles by using coarse-grained simulations instead of atomistic simulations. For ion-containing organic solvents, we recently showed that Stockmayer fluid (SF) simulation methods that model polar molecules as dipolar spheres agree with the experimental values of the solvation energies of various ions [12] and thus serve as computationally efficient alternatives to fully atomistic simulations. Nevertheless, the brute-force search process used to determine the model parameters, such as the interaction parameter and particle size, is a daunting task, particularly when we aim to simultaneously determine the molecular parameters for various solvents. Here we are interested in obtaining accurate values of the dielectric constant. Calculation of the dielectric constant from MD simulations is statistically noisy because the thermal fluctuations of solvent dipoles are large.

In this article, we show that ENNs serve as efficient surrogate models for SF MD simulations that provide the dielectric constants of polar solvents and NaCl solutions. This article aims to show how neural networks can help address the problem of calculating dielectric constants, including our general theoretical framework for reducing computational costs arising from the statistical ensemble of MD simulations. ENNs trained with low-sampling MD simulation datasets can provide predictions of the dielectric constants for various model parameters without running additional MD simulations. Advantages of the use of ENNs include the following objectives: (1) ENNs can provide the model parameters for multiple solvents and salt ions. (2) Unlike statistical averages calculated from ensembles of vast numbers of configurations, ENNs require relatively small amounts of simulation data for training. (3) Even if a single large neural network performed equivalent computations, training data might be too large for the memory. With ENNs, however, various batch and ensemble methods can be used to split the data into parts that can be more quickly processed. (4) Typically, the objective (loss) function of NNs has many local minima, and the backpropagation algorithm for the optimization of the objective function often results in a movement towards these minima (the so-called "local minimum problem"). However, ENNs tend to prevent model training from getting stuck in these local minima.

To boost and stabilize the training performance and to tame the large statistical noise, we also examine batch normalization (BN), a recently proposed method for the normalization of layer inputs in NNs [13], and bagging (or bootstrap aggregating) [14], a standard ML ensemble meta-algorithm. Furthermore, unlike the majority of NNs, our approach employs the sine function as an activation function, which is essential for capturing the nonmonotonic features of the training data in the present study. NNs with hidden layers consisting of sinusoidal activation functions have been largely ignored in the literature, partly because they are considered difficult to train, as recently pointed out in Ref. [15]. However, they can learn faster and better on specific tasks [15], and indeed, sinusoidal activation functions considerably enhance the performance of ENNs in the present study.

COMPUTATIONAL METHOD

MD Simulation

We consider seven common polar solvents, water, methanol, ethanol, acetone, 1-propanol, dimethyl sulfoxide (DMSO), and dimethylfuran (DMF) as model systems. Our simulation strategy is to develop a coarse-grained model that treats these charge-neutral polar molecules as spheres with diameters of σ and freely rotating permanent dipoles with dipole moments of $\vec{\mu}$. The hard-core nature of a dipolar sphere is given by the Weeks-Chandler-Anderson (WCA) potential,

$$\begin{cases} U_{WCA} = 4\epsilon_{LJ}\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right] & (r \le 2^{\frac{1}{6}}\sigma) \\ U_{WCA} = 0 & (r > 2^{\frac{1}{6}}\sigma) \end{cases} \quad \text{Eq. 1}$$

where the LJ parameter ϵ_{LJ} and $r = |r_i - r_j|$ designate the well depth of the potential and the distance between particles i and j, respectively. The cutoff of $2^{1/6}\sigma$ ensures the presence of only a repulsive force and no attractive tail in the potential. We write the dipole–dipole interaction between the *i*th and *j*th particles as

$$U_{\mu\mu} = \frac{(\overrightarrow{\mu_i} \cdot \overrightarrow{\mu_j})}{r^3} - \left(\frac{3}{r^5}\right)(\overrightarrow{\mu_i} \cdot \vec{r})(\overrightarrow{\mu_j} \cdot \vec{r}). \quad \text{Eq. 2}$$

For NaCl solutions, we also write the charge-dipole interaction between the ions and the solvent at a distance of r as follows:

$$U_{q\mu} = \frac{q(\vec{\mu} \cdot \vec{r})}{r^3}, \quad \text{Eq. 3}$$

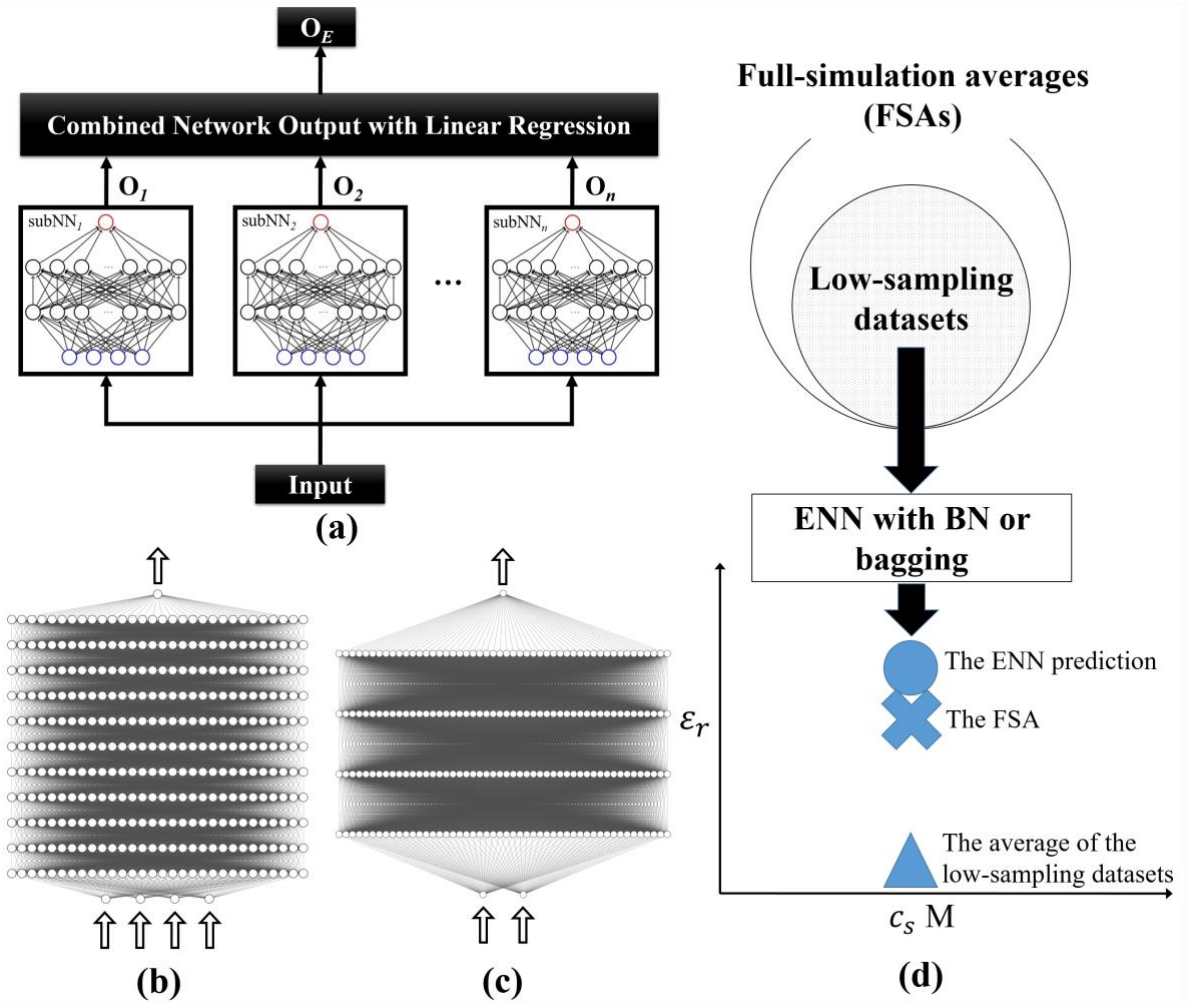

Fig. 1 ENN structure for prediction of the dielectric constant. (a) A hybrid of sub-NNs: each neuron contains an activation function. The sub-NNs are combined via the linear regression method. (b) The sub-NN structure for the salt-free solvents. The inputs are the dipole moment μ, the LJ parameter ϵ_{LJ} for the solvent-solvent interaction, the solvent diameter σ, and the number density ρ. The training data consist of the dielectric constants, calculated from the average of the total dipole moments over 10000 configurations. (c) The sub-NN structure for NaCl-containing water. The inputs are the salt concentration c_s and the LJ parameter ϵ_{LJ} for the ion-ion interaction. The training data consist of the dielectric constants, calculated independently from 40 configurations using Eq. 5. (d) Schematic diagram of the input-output relationship for NaCl-containing water. The ENN prediction derived from the low-sampling dataset is closer to the FSA than the average of the low-sampling dataset.

where q designates the ionic charge. The ion-ion interaction between charges q_i and q_j is given by the standard Coulombic form:

$$U_{q\mu} = \frac{q_i q_j}{r^3}. \qquad \text{Eq. 4}$$

Given that $\epsilon_{LJ}^{(i,j)}$ and $\sigma^{(i,j)}$ designate ϵ_{LJ} and σ in the WCA potential between the ith and jth particles, respectively, the geometric mixing rules for the interaction parameters, $\epsilon_{LJ}^{(i,j)} = \sqrt{\epsilon_{LJ}^{(i,i)} \cdot \epsilon_{LJ}^{(j,j)}}$ and $\sigma^{(i,j)} = \sqrt{\sigma^{(i,i)} \cdot \sigma^{(j,j)}}$, are often assumed. We follow these rules for the solvents, but we independently change $\epsilon_{LJ}^{(i,j)}$ in the case of NaCl solutions to examine the effect of this parameter on the dielectric constant.

We perform MD simulations using LAMMPS [16]. Similarly to the simulation setup in Ref. [12], the present simulations consist of boxes with side lengths of 21 Å for water and 30 Å for the other solvents. For the WCA potential, we use $\epsilon_{LJ}^{(i,j)} = 3.1676$ kJ/mol. The timestep is $\Delta t = 1$ fs. Simulations are equilibrated for 1–20 ns at room temperature (300 K). Production simulations are run to obtain 10000 configurations for the seven salt-free solvents at a given solvent diameter and LJ parameter and 2100 configurations for NaCl-

containing water at a given salt concentration and LJ parameter by performing measurements every 5000 steps. Although the accuracy of the Ewald summation method for electrostatic interactions is expected to be 10^{-5}, in this study, we set the tolerance to 10^{-4} for the salt-doped liquid to speed up the process of obtaining training samples. We calculate the dielectric constants using

$$\langle \varepsilon_r \rangle = 1 + \frac{\left(\langle \vec{M}^2 \rangle - \langle \vec{M} \rangle^2\right)}{3Vk_BT\varepsilon_0}, \qquad \text{Eq. 5}$$

where $\vec{M} = \sum_i \vec{\mu}_i$ is the total dipole moment calculated from the sum of the dipole moments $\vec{\mu}_i$ of all solvent particles. Here, note that Eq. *5* does not account for the contribution from the collective translational dipole moments of ionic charges [17]. However, this contribution is significantly smaller than that from the permanent dipole moments. Thus, we ignore such a small effect in the training data. Further details of our SF simulation methods with the collective dipole moments will also be presented elsewhere.

ENNs

Although individual coarse-grained SF simulations can be substantially faster than atomistic simulations [12], we need to determine the appropriate parameters for the coarse-grained models. Determining the optimal molecular parameters for various liquids over vast parameter spaces (e.g., the number of solvent and ion types or molecular architecture) may easily become impractical, especially for multicomponent liquid and ion mixtures. Thus, developing a more computationally fast surrogate model that captures the predictions of the MD simulations is critical, specifically from the following two perspectives. First, such a surrogate model facilitates the identification of appropriate molecular parameters for SFs in systems involving various types of liquids and, if any, salt ions. Second, the calculation of dielectric constants is often statistically noisy due to the large thermal fluctuations of the electric polarization; thus, these large statistical fluctuations lead to very slow statistical convergence of dielectric constant calculations. The iterative reparameterization of the model by repeatedly checking the statistics of the target dielectric constant after equilibration with a timescale of 1-20 ns is undoubtedly a daunting task. In this regard, a surrogate model should be constructed from a relatively small amount of simulation data. Accordingly, we employ ENNs because they were previously shown to efficiently capture highly nonmonotonic features in stochastic simulations of lithium dendrite growth in ionic liquids that involved similar computational requirements as needed here [18].

An ENN consists of multiple sub-NNs and each sub-NN has its own structure [Fig. 1(a)]. The idea of this method is to combine all predictions obtained from the sub-NNs and provide a better prediction than an individual output [7,8,19-21]. The outputs produced from each sub-NN are combined using linear regression: $O_E = c_0 + c_1O_1 + c_2O_2 + c_3O_3 + \cdots$, where $O_n(n = 1,2,3\ldots)$ is the output from the *n*th sub-NN, O_E is the output of the ENN, and $c_n(n = 0,1,2\ldots)$ is the weight of the corresponding sub-NN. The weights $c_0, c_1, \ldots, c_n$ are determined by minimizing the sum of the squared estimates of the errors between the training data and the predictions O_E from the ENN. The linear regression process yields a low value of $|c_n|$ when a sub-NN is poorly trained, whereas the $|c_n|$ for a well-trained sub-NN is larger. Thus, the predictions of well-trained sub-NNs are weighed more heavily than those of poorly trained sub-NNs.

ENN for Salt-Free Solvents

We use the dipole moment μ, the solvent diameter σ, the LJ parameter ϵ_{LJ}, and the number density ρ as the input descriptors, whereas the output is the dielectric constant ε_r of each solvent. We consider an ensemble of 3 sub-NNs for the prediction of the dielectric constants of salt-free solvents. The network topology of each sub-NN is represented by 4 inputs, 11 hidden layers with 30 neurons per hidden layer, and 1 output (i.e., the layer structure is 4-30-30-30-30-30-30-30-30-30-30-30-1), as illustrated in Fig. 1(b). The neurons between the neighboring layers are fully connected, and all connections have weights between the nodes [22]. The weights of all sub-NNs are randomly initialized and are optimized by minimizing the loss function. That is, we train the model by minimizing the mean squared error (MSE) between the predicted value ($\hat{y}$) and the target value (y) (the loss function) as follows:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \qquad \text{Eq. 6}$$

In addition, the MSE indicates the standard error for the evaluation of NN performance. However, we also provide the mean absolute percentage error (MAPE) $100\% \times (\sum|y - \hat{y}|/|y|)/n$, which serves as an alternative performance analysis measure. For the activation functions, we employ rectified linear unit (ReLU), Gaussian, and sine functions. Specifically, we note that the use of a sandwich structure such as "X-sine-X" or "X-Gaussian-X" (in the caption of Table 1a)

with X=ReLU is critical for enhancing the performance of the sub-NNs and accordingly that of the ENN. To reduce the overfitting of the models, we also stop training early with a patience setting of 30 epochs (the number of passes of the entire training dataset). That is, the training process is terminated when no improvement in the loss function accuracy occurs within 30 epochs.

We produce 76 samples for the set of polar solvent by changing the dipole moment $\vec{\mu}$, the solvent diameters σ, the LJ parameters ϵ_{LJ}, and the number density ρ and calculating the dielectric constant ε_r for each sample from the average of 10000 MD configurations (Supplementary Information). We thus train the ENN with 76 averaged values using the Keras API [23]. To enhance the model accuracy, we duplicate these 76 samples five times to employ 76×5 samples and divide all data into two parts, training and validation datasets. Normally, data samples are classified into training, validation, and test datasets. However, we only split them into training and validation datasets to maximally utilize the limited number of samples that are available for training. After training all sub-NNs, their outputs are combined to yield the output of the ENN via linear regression. For other combination methods with sub-NNs, interested readers can refer to the Supplementary Information.

ENN for NaCl Solutions

We also simulate NaCl ions in water with $\epsilon_{\mathrm{LJ}}^{\mathrm{water}} = 0.758$ kcal/mol and $\epsilon_{\mathrm{LJ}}^{(+,+)} = \epsilon_{\mathrm{LJ}}^{(-,-)} \equiv \epsilon_{\mathrm{LJ}}^{\mathrm{ion}}$. We run MD simulations to obtain 2100 configurations at a given salt concentration and LJ parameter. Here, an MD parameter set consists of a salt concentration and LJ parameter pair. The equilibration process requires a little longer than 1 ns when the salt concentration becomes very high. To examine the robustness and efficacy of the ENN method, we intentionally enhance the statistical noise in the training data via the MD simulations, in which we terminate the equilibration runs for all salt concentrations at 1 ns and begin sampling. For the training data (Supplementary Information), we randomly select 40 configurations from the 2100 configurations for each MD parameter set. Thus, the total numbers of simulation data and training data amount to $2100\times N_p$ and $40\times N_p$, respectively, where N_p designates the number of MD parameter sets. Here, we characterize the latter dataset as a low-sampling dataset. In the present systems, the autocorrelation functions $\langle\varepsilon_r(0)\varepsilon_r(t)\rangle$ and $\langle\vec{M}(0)\vec{M}(t)\rangle$ typically approach plateaus around 20 ps. Accordingly, the randomly selected 40 configurations for each MD parameter set are likely to be uncorrelated.

The dielectric constants calculated from the averages of the total dipole moments $\vec{M}$ over 2100 configurations for each MD parameter set using Eq. 5 serve as full-simulation averages (FSAs) for examining the efficacy of our ensemble networks with BN and bagging, particularly when we substantially reduce the amount of training data. We employ a dipole moment of 1.73 D throughout all these simulations because this dipole moment provides a dielectric constant of 80 for water, the experimental value.

We construct an ENN [Fig. 1 (a) and (c)] for NaCl-containing water, examining the efficacy of our method based on low-sampling datasets [18]. The ENN consists of 5 identical sub-NNs, each with a network topology of 2-50-50-50-50-1 (2 inputs, 4 hidden layers with 50 neurons per hidden layer, and 1 output). The input descriptors are the salt concentration c_s and the LJ parameter ϵ_{LJ}, and the output is the dielectric constant ε_r of NaCl-containing water. The architecture of the sub-NNs is shown in the caption of Table 1b. To increase the diversity of the sub-NNs, we combine them with different patience and batch size parameters in the ranges of 15 to 30 and 20 to 30, respectively. To reduce the probability of getting stuck in the local minima of the loss function while training and accelerate the training speed, we use BN before each ReLU activation function. We also empirically find from the calculation of the MSE that a sub-NN with 50 neurons per hidden layer serves as a good architecture.

Note that unlike in the case involving the salt-free solvents, the training data consist of the values of ε_r obtained before taking the average and thus include large statistical fluctuations. Accordingly, the training and validation errors appear to be large (Supplementary Information), and therefore the error values are not informative. To conduct a more meaningful comparison, we compare our ENN prediction with the FSA to evaluate the model performance. For clarity, we illustrate the input-output relationship in Fig. 1 (d). This noisy-data method was developed to reduce the simulation runs needed to train ENNs on cellular automata simulations of lithium dendrite formation [17]. The statistical fluctuations of the present simulation data are substantially larger. Thus, we examine whether the noisy-data method can still be employed as a general strategy, specifically with BN or bagging. Here, three types of electrostatic interactions (the ion-ion, ion-dipole, and dipole-dipole interactions) in NaCl solutions tend to increase the statistical fluctuations of the dielectric constant calculations. Thus, we directly deal with the simulation data for the test of the noisy-data method.

Stochastic gradient descent, an iterative method that optimizes a loss function via updates of the NNs' weights and biases, is often invoked for training feedforward NNs. However, the gradient is often vanishingly small, and therefore the NNs train slowly, specifically when the input values of the layers are large.

Table 1 MSEs (MAPEs) from the training of salt-free solvents and NaCl-containing water.

Salt-free solvents

(a) Training and validation errors (MSEs) of ENN1 and its three constituent sub-NNs for salt-free solvents. The MAPEs are shown in parentheses. The network topology of each sub-NN is 4-30-30-30-30-30-30-30-30-30-30-1, and the sequence of the activation functions consists of X-Y-X-Z-X-Y-X-Z-X-Y-X-X, where X=ReLU, Y=sine, and Z=Gaussian.

Salt-Free Solvents	Training Error	Validation Error
ENN1	11.7 (6.3%)	7.9 (8.3%)
Sub-NN1	29.1 (17.2%)	25.5 (15.7%)
Sub-NN2	21.1 (20.3%)	18.1 (19.0%)
Sub-NN3	11.8 (7.2%)	8.2 (7.0%)

NaCl-containing water

(b) MSEs (MAPEs in parentheses) of the predictions yielded by the ENNs for NaCl-containing water with and without BN relative to the average of the dielectric constants calculated from the low-sampling datasets and the FSA. These results are calculated from Fig. 3(a-d). The network topology of each sub-NN with BN is (BN-ReLU)-Gaussian-(BN-ReLU)-(BN-ReLU)-ReLU.

(c) MAPEs of the predictions of the ENNs for NaCl-containing water with and without BN under bagging relative to the dielectric constants calculated from the low-sampling datasets and the FSA. These results are calculated from Fig. 3(e-g).

ϵ_{LJ} kcal/mol		Low-sampling datasets	FSA	Low-sampling datasets	FSA
0.01	No BN	78.2 (15.7%)	65.1 (13.0%)	14.7%	10.2%
	BN	99.0 (16.4%)	**25.4 (8.8%)**	**18.1%**	**14.1%**
0.10	No BN	88.7 (15.1%)	130.1 (18.8%)	18.3%	16.4%
	BN	186.1 (18.6%)	**37.0 (9.8%)**	**20.9%**	13.6%
0.50	No BN	28.4 (11.1%)	79.8 (19.9%)	16.1%	6.9%
	BN	155.9 (20.9%)	**54.5 (10.9%)**	**28.7%**	**20.5%**

To circumvent this problem, we employ BN, a normalization scheme recently proposed by Ioffe and Szegedy, to accelerate the training of the NNs [13]. BN tends to reduce the dependence of the gradient on the scales of the parameters or their initial values via a normalization step that fixes the means and variances of each layer's inputs or the output of the activation function from the prior layer. For example, BN in Keras applies a transformation that maintains the mean output close to 0 and the output standard deviation close to 1, mitigating the internal covariate shift problem [23].

RESULTS

Salt-Free Solvents

Table 1a shows the training and validation errors of ENN1, which consists of three sub-NNs. Here, we employ three sub-NNs because the addition of more sub-NNs does not substantially improve the ENN. Sub-NN1 and sub-NN2 have relatively high training and validation errors while sub-NN3 has low errors. ENN1 outperforms all the sub-NNs or remains comparable to sub-NN3. Unlike the sub-NNs, the errors of ENN1 are not substantially changed by the initial model parameters. The robustness of the training performance to the combination of sub-NNs arises mainly from the fact that in linear regression, well-trained sub-NNs are weighted more significantly than sub-NNs with relatively poor performance. In the Supplementary Information, we also show that an ENN (ENN3) consisting of a hybrid of three poorly trained sub-NNs (sub-NN1, sub-NN2, and sub-NN4 in Table S1) can outperform all the sub-NNs and compare favorably with ENN1 (Table S2). Thus, the use of ENNs increases the likelihood of finding a global minimum among the sub-NNs, without having one large NN that can encompass all the data and find the global minimum.

We plot the dielectric constants of seven solvents with different LJ parameters in Fig. 2, in which we compare the predictions of ENN1 with the FSAs from MD simulations. For clarity, the experimental values of dielectric constants of the seven solvents [24] are shown below each solvent's name in Fig. 2. We focus on the solvent diameter and the LJ parameter because the Onsager equation (the standard mean-field theory for the dielectric constant) cannot predict changes in the dielectric constant with respect to the two [25]. Here, the total number of training data is limited to 76 samples (i.e., N_p =76) for all the solvents (Supplementary Information), and Fig. 2 includes 65 samples to illustrate the overall trend of the dielectric constants. Each data point (symbol) designates the dielectric

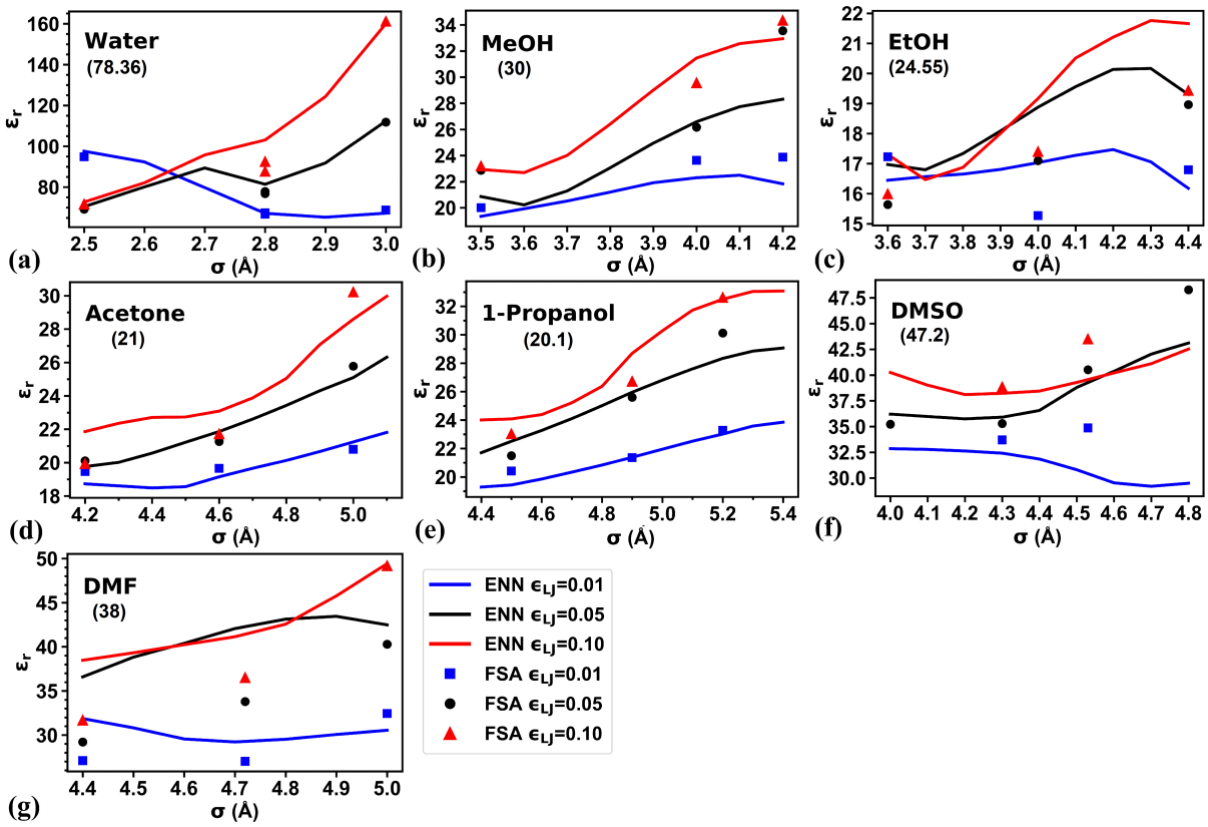

Fig. 2 Dielectric constants vs. solvent diameters predicted by ENN1 (Table 1a) for different solvents. The solid lines indicate the predictions from the ENN, whereas the symbols indicate the full simulation averages (FSAs). The solvents are (a) water, (b) MeOH, (c) EtOH, (d) acetone, (e) 1-propanol, (f) DMSO, and (g) DMF. The values in the brackets indicate the experimental values of their dielectric constants.

constant calculated from the configurations, and we employ the samples to individually train the sub-NNs. Here, we note that conventional NNs consisting of only ReLU and sigmoid functions are unlikely to capture the observed variations in the dielectric constants, such as the nonmonotonicity between the solvent diameter σ and the LJ parameter ϵ_{LJ}. To address this nonmonotonicity, we find that the average of the total dipole moment over 10000 inclusion of a sine function sandwiched by ReLUs considerably improves the training efficiency, and a Gaussian function sandwiched by ReLUs further improves the results. As discussed before, however, the combination of these sub-NNs is the essence of the substantial improvement and robustness of the model accuracy, mitigating the local minimum issue and providing the model robustness against poorly trained sub-NNs. Concomitantly, we test BN in one or more layers and find that BN makes the prediction results worse. The reason for this could be due to the insufficient amount of training data. Nevertheless, in the case of salt-doped solvents, we show that BN averages the results and improves the performance of ENNs when the training data are statistically noisy.

NaCl Solutions

We examine the noisy-data method with an ENN (ENNc1) consisting of five sub-NNs with the following sequence of activation functions: (BN-ReLU)-Gaussian-(BN-ReLU)-(BN-ReLU)-ReLU (see Tables S3 and S4 for the details of training and validation errors). Fig. 3(a) shows the FSAs and the predictions of the ENN with BN, in which the salt concentrations c_s and the LJ parameters ϵ_{LJ} for the ion-ion and ion-water interactions are varied. The number of MD parameter sets is $N_p = 33$. For regularization to force the predictions to fit the dielectric constants of pure water (i.e., $c_s = 0$ M), each of the dielectric constant averages at $c_s = 0$ M is copied 267 times repeatedly in the training data to include about 800 additional samples for the three LJ parameters. Note that the training data at a given salt concentration and LJ parameter (i.e., an MD parameter set) consist of only 40 configurations; however, we used more data to produce the FSAs. Thus, the training data may be disparate from the FSAs, yet the predictions compare favorably with the FSAs.

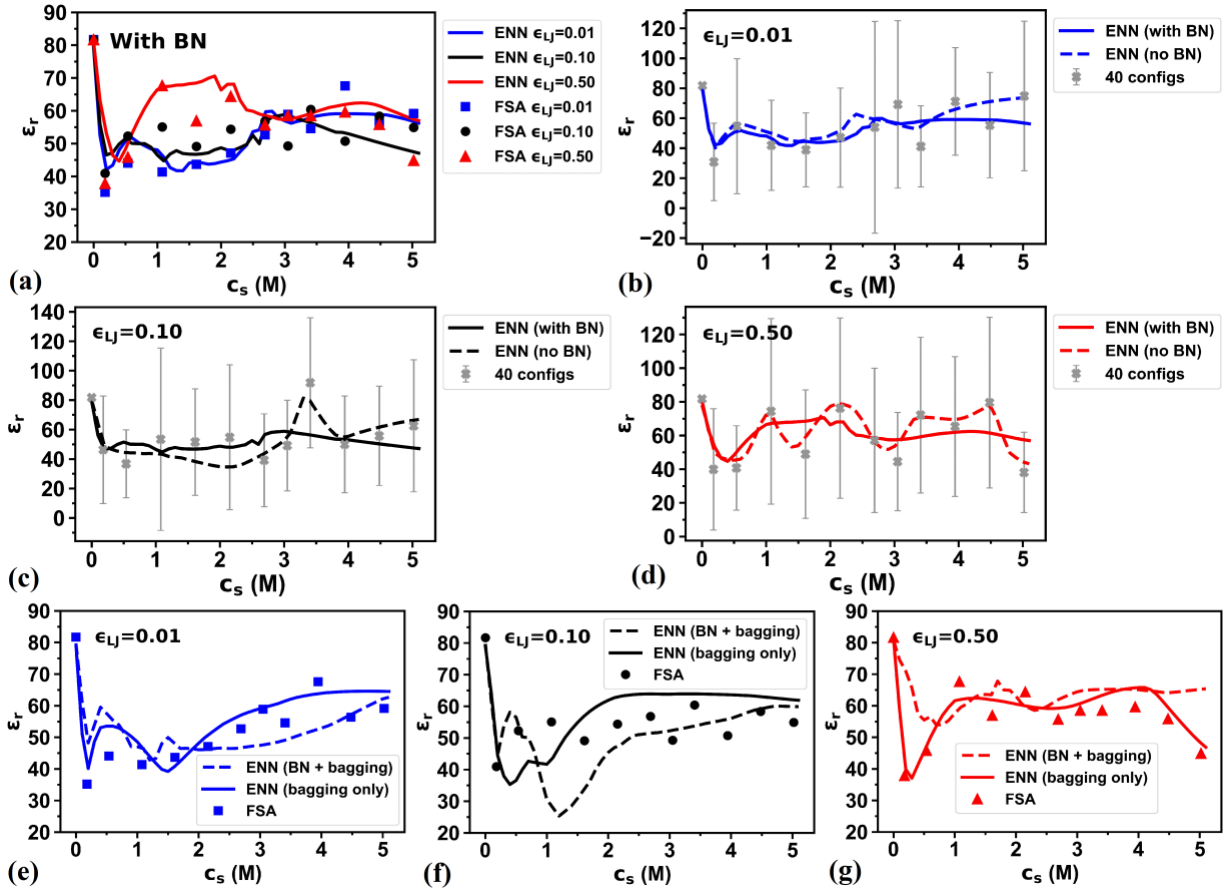

Fig. 3 (a) Comparison between the full-simulation averages (FSAs) and the ENN predictions for the dielectric constants of NaCl solutions. Only BN was used. The solid lines indicate the ENN predictions for different salt concentrations c_s and different LJ parameters between the ions. ϵ_{LJ} designates $\epsilon_{LJ}^{(ion,\,ion)}$, where $\epsilon_{LJ}^{(ion,\,ion)} = \epsilon_{LJ}^{(+,+)} = \epsilon_{LJ}^{(+,-)} = \epsilon_{LJ}^{(-,-)} = \epsilon_{LJ}^{(water,+)} = \epsilon_{LJ}^{(water,-)}$. The symbols indicate the FSAs (the average over 2100 configurations for each MD parameter set). (b)-(d) Efficacy of BN with (b) ϵ_{LJ} = 0.01 kcal/mol, (c) ϵ_{LJ} = 0.10 kcal/mol, and (d) ϵ_{LJ} = 0.50 kcal/mol. The solid and dashed lines indicate the ENN predictions with and without BN, respectively. The grey symbols indicate the average over 40 configurations for each MD parameter set, and the standard deviations are represented by error bars. (e)-(g) BN + bagging (dashed line) and bagging only (solid line) with (e) ϵ_{LJ} = 0.01 kcal/mol, (f) ϵ_{LJ} = 0.10 kcal/mol, and (g) ϵ_{LJ} = 0.50 kcal/mol. The symbols indicate the FSAs.

Table 1b shows the differences between the dielectric constants predicted from the low-sampling datasets and the averages calculated from the low-sampling datasets or the FSAs. Note that the predictions obtained with BN are worse than those without BN when compared with the average calculated from the low-sampling datasets. In contrast, the performance of BN becomes better when compared with the FSAs, as highlighted in bold. This difference indicates that BN drives the predictions derived from the low-sampling datasets to approach the FSAs.

To explain the role of BN, Fig. 3(b)-(d) show the dielectric constants calculated from individual configurations without taking the average in Eq. 5 (i.e., the low-sampling dataset) and the predictions from the ENN. The unaveraged dielectric constants are widely distributed around their mean values. Note that the averages (reduced-set averages) calculated from the low-sampling datasets may not be close to the FSAs; thus, they can be highly inaccurate values. The predictions obtained without BN tend to be close to the reduced-set averages. However, when we employ BN, the predictions derived from the low-sampling datasets do not trace the reduced-set averages and instead tend to approach the FSAs. For example, this is especially clear at the result for $\epsilon_{LJ} = 0.1$ kcal/mol and $c_s = 3.407$ M and at nearly all results for $\epsilon_{LJ} = 0.5$ kcal/mol. Thus, BN-combined ENNs appear to be robust against significant reductions in the number of training data and therefore can considerably speed up the parameter adjustment process and the exploration of new features in noisy data over a vast parameter space. It is probable that this enhanced training performance occurs partly due to the normalization process of BN reducing the

large fluctuation of the training data and mixing information about data distributions at different MD parameter sets.

Finally, we also employ bagging, a treatment for decreasing variance to prevent overfitting. By randomly grouping the training data, we obtain 5 different training datasets and then feed these training data into 5 different sub-NNs. Our results show that bagging can outperform BN, but a hybrid of BN and bagging makes the prediction results less accurate, as highlighted by the bold figures in Table 1c. In Fig. 3(e)-(g), the results obtained from the bagging-only method can more accurately capture the nonmonotonicity of ε_r at low salt concentrations with all LJ parameters and the nonmonotonicity of ε_r at high concentrations in the case where $\epsilon_{LJ} = 0.5$ kcal/mol. The optimization of the bagging size (the number of training data in the bags selected from the original training data to create diverse samples) is crucial for outperforming BN, which is typically encountered in ML when tuning model hyperparameters. However, this optimization may not always be guaranteed or may require further computational effort because the training performance depends on various hyperparameters.

CONCLUSION

We constructed ensemble neural networks for the calculation of the dielectric constants of polar solvents and NaCl solutions. For the training data, we used Stockmayer fluid MD simulation methods that accounted for the dielectric responses of dipolar fluids. We showed that our ensemble neural networks with activation layers containing sine and Gaussian functions captured the trends of the dielectric constants of seven polar solvents. Sinusoidal activation functions often yield large numbers of shallow local minima [15], but our ensemble neural networks appeared to mitigate this issue. For NaCl solutions, we trained the ensemble neural networks with low-sampling datasets that were insufficient to produce the full-simulation averages. Moreover, the training data were widely distributed around the mean values for each MD parameter set. In this case, with highly noisy data, batch normalization or bagging enabled the predictions of ensemble neural networks to compare favorably with the full-simulation averages when employed independently. In other words, batch normalization and bagging can make the predictions of ensemble neural networks trace the optimal values derived from noisy datasets. Thus, we suggest that ensemble neural networks with batch normalization or bagging serve as efficient tools to substantially reduce the number of simulation runs and thus save computational effort. For example, the ensemble neural networks trained with a small number of possible input parameters can be used to choose values of the model parameters that best match target experiments, without having to run more MD simulations. The present methods would also be useful to simultaneously determine a number of model parameters for multicomponent systems.

ACKNOWLEDGMENTS

This material is based upon work supported by the Faculty Early Career Development Program of the National Science Foundation under grant DMR-1944211 and Michigan Tech's doctoral finishing fellowship. This work was performed, in part, at the Center for Integrated Nanotechnologies, an Office of Science User Facility operated for the U.S. Department of Energy (DOE) Office of Science. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. DOE's National Nuclear Security Administration under contract DE-NA-0003525. We are also grateful to the High-Performance Computing Shared Facility, Superior, at MTU for their essential support.

DATA AVAILABILITY

The training data can be found in the Supplementary Information.

CONFLICT OF INTEREST

All authors declare that they have no conflict of interest.

REFERENCES

[1] F. Noé, A. Tkatchenko, K.-R. Müller, and C. Clementi, Annual Review of Physical Chemistry **71**, 361 (2020).

[2] J. Wang, S. Olsson, C. Wehmeyer, A. Pérez, N. E. Charron, G. de Fabritiis, F. Noé, and C. Clementi, Acs Central Sci **5**, 755 (2019).

[3] A. Krishnamoorthy *et al.*, Physical Review Letters **126**, 216403 (2021).

[4] H. Ghorbanfekr, J. Behler, and F. M. Peeters, J Phys Chem Lett **11**, 7363 (2020).

[5] Q. Q. Gu, L. F. Zhang, and J. Feng, Science Bulletin **67**, 29 (2022).
[6] J. E. Floyd and J. R. Lukes, The Journal of Chemical Physics **156**, 184114 (2022).
[7] L. K. Hansen and P. Salamon, IEEE Transactions on Pattern Analysis and Machine Intelligence **12**, 993 (1990).
[8] S. Hashem, B. Schmeiser, and Y. Yih, in *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94)*(1994), pp. 1507.
[9] H. M. Le and L. M. Raff, The Journal of Physical Chemistry A **114**, 45 (2010).
[10] S. K. Singh, K. K. Bejagam, Y. An, and S. A. Deshmukh, The Journal of Physical Chemistry A **123**, 5190 (2019).
[11] D. A. R. S. Latino, F. F. M. Freitas, J. Aires-De-Sousa, and F. M. S. Silva Fernandes, Int J Quantum Chem **107**, 2120 (2007).
[12] C. J. Shock, M. J. Stevens, A. L. Frischknecht, and I. Nakamura, The Journal of Physical Chemistry B **124**, 4598 (2020).
[13] S. Ioffe and C. Szegedy, arXiv:1502.03167 (2015).
[14] L. Breiman, Machine Learning **24**, 123 (1996).
[15] G. Parascandolo, H. Huttunen, and T. Virtanen, *Taming the waves: sine as activation function in deep neural networks* (2016).
[16] S. Plimpton, Journal of Computational Physics **117**, 1 (1995).
[17] J. Grunenberg, *Computational spectroscopy: methods, experiments and applications* (Wiley-VCH, Weinheim, 2010).
[18] T. Gao, Z. Qian, H. Chen, R. Shahbazian-Yassar, and I. Nakamura, Molecular Systems Design & Engineering **7**, 260 (2022).
[19] S. Hashem and B. Schmeiser, IEEE Trans. Neural Networks **6**, 792 (1995).
[20] D. J. Reid, Economica **35**, 431 (1968).
[21] S. Hashem, Neural Networks **10**, 599 (1997).
[22] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*, Adaptive computation and machine learning.
[23] F. Chollet and others, https://keras.io (2015).
[24] W. M. Haynes, D. R. Lide, and T. J. Bruno, *CRC handbook of chemistry and physics* (CRC press, 2016).
[25] G. G. Raju, *Dielectrics in electric fields* (CRC Press, Taylor & Francis Group, 2016), 2nd edn.