Deep Neural Networks for Multicomponent Molecular Systems

Kyohei Hanaoka*

ABSTRACT: Deep neural networks (DNNs) represent promising approaches to molecular machine learning (ML). However, their applicability remains limited to single-component materials and a general DNN model capable of handling various multicomponent molecular systems with composition data is still elusive, while current ML approaches for multicomponent molecular systems are still molecular descriptor-based. Here, a general DNN architecture extending existing molecular DNN models to multicomponent systems called MEIA is proposed. Case studies showed that the MEIA architecture could extend two exiting molecular DNN models to multicomponent systems with the same procedure, and that the obtained models could learn both the molecular structure and composition information with equal or better accuracies compared to a well-used molecular descriptor-based model in the best model for each case study. Furthermore, the case studies also showed that, for ML tasks where the molecular structure information plays a minor role, the performance improvements by DNN models were small; while for ML tasks where the molecular structure information plays a major role, the performance improvements by DNN models were large, and DNN models showed notable predictive accuracies for an extremely sparse dataset, which cannot be modeled without the molecular structure information. The enhanced predictive ability of DNN models for sparse datasets of multicomponent systems will extend the applicability of ML in the multicomponent material design. Furthermore, the general capability of MEIA to extend DNN models to multicomponent systems will provide new opportunities to utilize the progress of actively developed single-component DNNs for the modeling of multicomponent systems.

INTRODUCTION

Multicomponent molecular systems such as polymer alloys, mixtures, and composite materials are used in various applications due to multiple functions and tunable properties. The properties of these classes of materials vary with both chemical structures and component compositions, and composition freedom means that design spaces of multicomponent molecular systems are exponentially larger than those of single-component systems, which makes an optimal recipe for multicomponent materials hard to be found. To solve this problem, high-throughput screening methods are actively developed and used for a wide range of multicomponent systems including active layers of organic solar cells, electrolyte additives for lithium ion batteries, surface coating materials, and industrial products. Nevertheless, due to tremendously large design space of multi-component molecular systems, even high-throughput screening methods are compelled to limit their exploration space to a fixed combination or composition with a moderate number of component materials. However, in most cases, to find better recipes for multicomponent materials, it is needed to explore much larger combination and composition freedoms of multicomponent systems, and therefore, the primary approach for the multicomponent material design is still trial-and-error-based.

To accelerate the exploration of material design spaces, the material science community has recently paid much attention to machine learning-based data-driven approaches to the material design. Although most of molecular machine learning studies have focused on single-component systems, attempts have also been made to model properties of multicomponent molecular systems using datasets generated from massive computer simulations or systematic experiments. However, it is rarely possible to obtain sufficient datasets for multicomponent materials due to the huge chemical space involved, and therefore, handling of a machine learning task on sparse datasets, which are produced on a trial-and-error basis during the process of the material design, is needed to extend the applicability of machine learning-based approaches for the multicomponent materials design. However, such a machine learning task is still challenging for current molecular machine learning models.

Conversely, recent years have seen that the accuracy and applicability of molecular machine learning was significantly improved by using deep neural networks (DNNs) for molecules. One of the major advantage of DNNs is the ability of learning rules to generate fixed-size feature vectors suitable for a given task directly from unstructured data such as...
images and molecular structures. This ability of DNNs is widely called “feature representation learning”. Figure 1a shows a

![Diagram](https://dx.doi.org/10.1021/acsomega.0c02599)

Figure 1. Workflows of molecular machine learning. In these workflows, monomer units in polymers are used as examples of input molecular structures. (a) DNN for single-component molecular systems with feature representation learning ability. (b) Descriptor-based model for single-component molecular systems. (c) DNN for multicomponent molecular systems with feature representation learning ability. (d) Descriptor-based model for multicomponent molecular systems.

Therefore, most of existing approaches for machine learning of multicomponent molecular systems are still descriptor-based⁶,⁷,⁹,¹²–³³ (see Figure 1d for typical cases). Another requirement for molecular DNN models in terms of application on multicomponent molecular systems is permutation invariance of input components, a mathematical property of a model whereby the order of components in its input is essentially independent of its output. For example, for a binary system composed of component materials \(z_1 (r_1\%)\) and \(z_2 (r_2\%)\), a model should satisfy the following equation

\[
\text{model}([z_1, r_1], [z_2, r_2]) = \text{model}([z_2, r_1], [z_1, r_1])
\]  

(1)

Permutation invariance is a natural requirement for a machine learning model in multicomponent molecular systems with randomly associated components, such as mixtures, random copolymers, and polymer blends, because if the model lacks permutation invariance, its prediction changes artificially depending on the order of inputting component material data to the model. Although, there is a DNN model performing feature representation learning from the dataset of two-component systems, binary copolymers, with molecular structure and composition informations,⁸ this model is not applicable to materials with more than two components and does not have permutation invariance. Meanwhile, there are also DNN models that can handle nonmolecular solid materials with an arbitrary number of component species in a permutation invariant manner;³⁸,³⁹ however, these models use only the composition information as an input and cannot perform feature representation learning from unstructured data such as molecular structures. It is also possible to treat molecular simulation models of a multicomponent copolymer as a single long-chain monomer, which can be handled by machine learning models for a single molecule;⁶ however, such an approach is not applicable to randomly associated experimental copolymers with arbitrary compositions. Therefore, to extend the applicability of feature representation learning of molecular DNNs to multicomponent molecular systems, a new class of models is needed; capable of learning feature representation and composition information from a multicomponent molecular system with an arbitrary number of unordered component molecules in a permutation invariant manner.

In this study, a general DNN architecture extending DNN models for single-component systems to the multicomponent while maintaining permutation invariance is presented. To benefit from feature representation learning abilities of existing DNN models for single molecules, the proposed DNN architecture uses embedding networks of the existing models to generate a fixed-size feature vector for each component (Figure 1c). Then, a mixing network merges the obtained feature vectors with composition information in a permutation-invariant manner to obtain a single fixed-size vector (Figure 1c), and this vector is finally used as an input for a prediction network to make a prediction. Such DNN models are attractive because, by substituting their embedding networks, one can easily extend the models to any type of unstructured chemical data such as images, spectrum, and simulation results. However, there is no proposed mixing network that can merge a set of feature vectors with composition information in a permutation invariant manner. Instead, we start from considering a special case of the multicomponent molecular dataset where the compositions are fixed and only the combinations of molecular components determine the materials properties. In this special case, multicomponent molecular system data can be regarded as set
data and properly handled by the recently emerging class of DNN models for the set proposed in the field of machine learning research by substituting their embedding networks with that of molecular DNN models. Note that set data is a variable-sized collection of components without any natural order between them. Although the DNN models for sets cannot handle composition information, most of these models are permutation invariant and composed of embedding, mixing, and prediction networks as with the DNN architecture for multicomponent molecular systems considered here (Figure 1c). Therefore, DNN models for sets provide a good starting point for designing a DNN architecture for multicomponent molecular systems.

In the following, the DNN architecture for multicomponent molecular systems is constructed based on the DNN models for sets, and the performances of obtained models are examined in four types of case studies. These case studies showed that, by exploiting embedding networks of existing molecular DNN models, the proposed DNN architecture enables learning of both feature representation and composition information with equal or better accuracies compared to a well-used molecular descriptor-based model in the best model for each case study, and the performance improvements by DNN models were notable for machine learning tasks where the molecular structure information in multicomponent materials data plays a major role. Furthermore, DNN models showed notable predictive accuracies for an extremely sparse dataset.

## METHODS

In the following part, the permutation-invariant DNN models for sets are decomposed into common building blocks and the building blocks thus obtained are used to reconstruct DNN models for multicomponent molecular systems with composition information in the next part.

### DNN Models for Sets

Figure 2 shows a network architecture shared by most DNN models for sets, which is referred to in this study as the embed-interact-aggregate architecture.

![Figure 2: Schematic of the embed-interact-aggregate architecture for sets.](image)

Here, embed block corresponds to the embedding network in Figure 1c, which generates a set of fixed-size feature vectors, \( Z \), from a set of molecular structures, \( X \). Note that, the same network parameters are used for embedding all input molecular structures. While the pair of interact and aggregate blocks corresponds to the mixing network, which merges the set of feature vectors, \( Z \), into a single vector, \( a \). The interact block is a neural network incorporating interactions among the input set. Because the interact block does not change the size of the set, this block is omitted. The aggregate block can be both a neural network and unlearnable function, such as simple summation, and merges a set of feature vectors, \( M \), into the single vector, \( a \).

More formally, for an input set of molecules, \( X = \{ x_1, x_2, ..., x_N \} \), the embed-interact-aggregate architecture models calculate the output values as follows:

\[
\begin{align*}
    z_i &= \text{embed}(x_i) \\
    Z &= \{ z_1, z_2, ..., z_N \} \\
    m_i &= \text{interact}(z_i, z_j | j \neq i) \\
    M &= \{ m_1, m_2, ..., m_N \} \\
    a &= \text{aggregate}(M) \\
    \text{output} &= \text{predict}(a)
\end{align*}
\]

To conserve overall permutation invariance, interact and aggregate blocks are invariant to permutation of feature vectors in \( z_j (j \neq i) \) and \( M \), respectively. The overall model can be trained using backpropagation. The embed-interact-aggregate architecture-based DNN models have been successfully applied to machine learning tasks on sets such as regression using multiple images. Brief descriptions of these models are provided in the Supporting Information.

### DNN Models for Multicomponent Molecular Systems

Next, a permutation invariant mix-embed-interact-aggregate (MEIA) architecture is proposed for multicomponent molecular systems, using the building blocks of the embed-interact-aggregate architecture and incorporating composition information into them (Figure 3).

In this architecture, mix functions are introduced to incorporate composition information into interact and/or aggregate blocks of the embed-interact-aggregate architecture. A requirement for the mix function is conserving permutation invariance of the overall model, and this is accomplished by individually converting each feature vector in the input set of a building block using composition information of each molecular component. Following is an example of such mix function, concat-mix, which is applied to interact blocks in this study:

\[
\begin{align*}
    \text{concat-mix}(z_i, r_j) &= \text{concatenate}(z_i, r_j) \\
    z_i' &= \text{concat-mix}(z_i, r_i) \\
    Z' &= \{ z_1', z_2', ..., z_N' \} \\
    m_i &= \text{interact}(z_i', z_j' | j \neq i)
\end{align*}
\]

Here, \( R = \{ r_1, r_2, ..., r_N \} \) is composition of each component. Such individual conversion of the feature vectors of molecular components in the set does not obviously break overall permutation invariance. By this way, the MEIA architecture can handle an arbitrary number of molecular structures with composition information in a permutation invariant manner.

Combining the building blocks, embed, interact, aggregate, and mix functions, allows a total of seven types of MEIA-based models to be evaluated in the following case studies. These models are shown in Table 1 alongside the interact, aggregate, and mix functions used, and examples of MEIA-based models are shown in Figure 3 for WS, Concat-MHA, Concat-Self-MHA, and WS-Self-MHA models. Note that, as the embed block, the embedding network of the graph convolution (GC) model was used unless otherwise stated and all MEIA-based models examined here use FCNN with a single-hidden layer as the prediction network. In Table 1, summation represents the element-wise summation of feature vectors, and weight-mix is used as the mixing network of WS model employing these functions is a simple-weighted summation function (Figure 3 WS). The other six models utilize attention...
neural networks to improve the treatment of interactions between components. In the field of molecular DNN research, attention neural networks are used to merge a set of feature vectors of atoms in a molecule into a single-molecular feature vector incorporating interactions between the atoms.\textsuperscript{31} Intuitively, attention neural networks learn a weight for each component in a set of feature vectors and perform weighted summation based on the obtained weights in permutation invariant manners. Therefore, attention neural networks can be used as aggregate blocks in the MEIA architecture. In the case study, two popular models of attention neural networks, RNN-based\textsuperscript{40} and multihead\textsuperscript{47} attentions, were used as aggregate blocks (Concat-MHA and Concat-RNNA in Table 1). As an example of the MEIA-based model with an attention neural network, the neural network architecture of the Concat-MHA model is shown in Figure 3. Note that only difference between Concat-MHA and Concat-RNNA models is the choice of the aggregate block.

Meanwhile, attention neural networks can also be used to update a set of feature vectors without merging them by updating all vectors in the set based on the weighted sum of

Table 1. MEIA-Based Models Used in Case Studies

| MEIA-based model\textsuperscript{a} | interact | aggregate | mix function |
|-------------------------------|-----------|-----------|---------------|
| WS                            | summation | Weight-mix for aggregate |
| Concat-Self-MHA               | multihead self-attention | summation | Concat-mix for interact |
| WS-Self-MHA                   | multihead self-attention | summation | Weight-mix for aggregate |
| Concat-MHA                    | multihead attention | Concat-mix for aggregate |
| Concat-Self-RNNA              | RNN-based self-attention | summation | Concat-mix for interact |
| WS-Self-RNNA                  | RNN-based self-attention | summation | Weight-mix for aggregate |
| Concat-RNNA                   | RNN-based attention | Concat-mix for aggregate |

\textsuperscript{a}“WS” in the names of models stands for the combination of weight-mix and summation. “Concat”, “Self”, “MHA”, and “RNNA” stand for concat-mix, self-attention, multihead attention, and RNN-based attention, respectively.

Figure 3. Schematic of the MEIA architecture for the multicomponent molecular system. Examples of neural network architectures and how they process multiple molecular structures with composition information are shown for WS, Concat-MHA, Concat-Self-MHA, and WS-Self-MHA models. Each colored pillar represents a feature vector. Elements of the feature vectors that have composition information incorporated by mix functions, concat-mix and weight-mix, are highlighted in red.
vectors generated by these neural networks. Such a type of attention is called as self-attention and can be used as interact blocks in the MEIA architecture. As with aggregate blocks, RNN-based and multhead self-attentions were used as interact blocks (Concat-Self-MHA, WS-Self-MHA, Concat-Self-RNNA, and WS-Self-RNNA in Table 1). As examples of the MEIA-based model with self-attentions, the neural network architectures of Concat-Self-MHA and WS-Self-MHA models are shown in Figure 3. Note that the only difference between models using RNN-based and multhead self-attentions is the choice of the interact block.

Detailed architectures of the building blocks using the attention neural networks are shown in Figures S3 and S2; meanwhile, full details of overall architectures of MEIA-based models are shown in Figures S3–S8.

■ COMPUTATIONAL DETAILS

Neural Networks. Embedding networks of DeepChem implementations of GC and message passing neural network (MPNN) models are used as embed blocks with their default featurizers to generate the initial atom and bond features. All models were implemented as tensor graph objects in DeepChem and optimized using the L2 loss and ADAM optimizer. To reduce over fitting, dropout and batch normalization layers were introduced to all models examined here (Figures S3–S9).

Dataset. The chemical compositions in the weight fraction are used as composition information. All datasets in the experiments were split into train and test sets, with the former used in training and hyper parameter tuning and the test sets limited to performance evaluations. In the case study 1, the dataset was randomly split; while in the case study 2, the dataset was split according to the setting of “mixture-out validation” where each multicomponent material data in test sets comprised an unknown combination of component molecules. Subsequently, in the case studies of 3 and 4, datasets were split according to the setting of “compound-out validation” where each multicomponent material data in test sets contains at least one unknown component molecule.

Input Molecular Structure. 2D molecular graphs without hydrogen atoms were used as input molecular structures for neural network models in all case studies. For random copolymers, monomer units (i.e., substructures in polymers) were used as input molecular structures, and the connections between repeating units were not considered. Note that although this simple treatment for monomers may not be generally applicable to all the properties of polymers, this approach is reasonable for the modelling of glass transition temperature (Tg) of random copolymers, which can often be modelled based on the information about component monomers, although other factors, such as molecular weights, are also known to change Tg of copolymers.

Performance Comparison. The performances of models for the test sets were compared by coefficient of determination (R2), mean absolute error (MAE), root-mean-square error (RMSE), and Spearman’s rank correlation coefficient (ρ) for average predictions of 50 and 20 randomly initialized models for GC- and MPNN-based models, respectively. As baselines, the performances of FCNN models with three hidden layers were also evaluated for composition-only and extended-connectivity fingerprint (ECFP) inputs, respectively. The ECFP fingerprints were obtained using an RDKit with a radius of 2 in an unhashed manner, and molar-weighted sums of nonzero elements of the obtained fingerprints were used as inputs for the FCNN model. A detailed architecture of the FCNN model is shown in Figure S9.

Note that all MEIA-based models examined here share the same architecture of prediction networks in the last parts of networks, and the last part of the FCNN model also has the network architecture identical to the prediction networks of the MEIA-based models. Therefore, model performance differences are attributable to the way a fixed size vector used in the last part of the network is generated from molecular structures and compositions. In the MEIA-based models, this is done by embedding and mixing networks, whereas, in the FCNN models with an ECFP input, this is done by ECFP, weighted summation, and initial two fully connected layers of the FCNN model.

Hyper Parameter Optimization. The hyper parameters of the models were obtained by Bayesian hyper parameter optimizations using GPYOpt. In the Bayesian optimizations, the MAE in cross validations was minimized in 5 initial random sampling and 20 subsequent steps of optimization. Meanwhile, for “mixture-out” and “compound-out” validation tasks, cross validations in Bayesian optimizations are also performed according to the setting of “compound-out validation”. To construct a data set for “k-fold compound-out cross validation”, all the component molecules in the training set are first split into k subsets, whereupon training samples containing the component molecules in the subset are used for validation and the remainder for learning for each subset. Details of the hyper parameters used in the Bayesian optimizations are shown in Tables S1 and S2, while the settings of the other hyper parameters are shown in Table S3.

■ RESULTS AND DISCUSSION

In the following, to examine whether the MEIA architecture can properly extend single-component DNN models and whether the MEIA architecture can allow DNN models to learn both feature representation and composition information, performances of obtained models were compared with two shallow learning baselines, composition-only and descriptor-based (Figure 1d) models, in four case studies. For the composition-only model, a FCNN with composition input was used, and for the descriptor-based model, a FCNN with molar-weighted sum of ECFP fingerprints was used. Note that ECFP has often been used in benchmark studies of molecular DNNs because ECFP uses only the information of the molecular graph and does not use further detailed information about the molecules, such as physicochemical properties, generate descriptors, as with most molecular DNN models.

Case Study 1: Interpolation of the Composition Space. The first case study is on a previously examined interpolation task within a composition space of copolymers where the test set contains no unknown components. This task could be handled without structural information, by interpolating the data points of a training set within the composition space. A dominant contribution of the composition information renders this task suitable for examining whether the MEIA models can properly handle the composition information for each component. Furthermore, this task is suitable to assess the effect of permutation invariance introduced in MEIA-based models because this task was previously modelled by a recursive NN model, a DNN models without permutation invariance. Therefore, in this case study, the result of previous study using recursive NN model was also used as a baseline, in addition to composition-only and descriptor-based baselines.


Table 2. Prediction Performances for the Binary Copolymer Tg

| models            | R2     | MAE   | RMSE  | Spearman’s ρ |
|-------------------|--------|-------|-------|--------------|
| recursive NN      | 0.982  | 4.930 | 6.137 | 0.987        |
| Composition-only  | 0.987  | 3.690 | 5.121 | 0.994        |
| Descriptor-based  | 0.994  | 2.619 | 3.534 | 0.994        |
| WS                | 0.990  | 3.163 | 4.511 | 0.994        |
| Concat-Slef-MHA   | 0.988  | 3.295 | 4.880 | 0.988        |
| WS-Self-MHA       | 0.993  | 2.744 | 3.785 | 0.993        |
| Concat-MHA        | 0.994  | 2.454 | 3.605 | 0.996        |
| Concat-Slef-RNNA  | 0.981  | 4.450 | 6.232 | 0.986        |
| WS-Self-RNNA      | 0.993  | 2.872 | 3.725 | 0.994        |
| Concat-RNNA       | 0.992  | 2.813 | 3.940 | 0.994        |
| MEIA (average)    | 0.990(0.986–0.994) | 3.113(3.510–3.716) | 4.383(3.515–5.251) | 0.992(0.989–0.996) |

Scores of the model showing performance superior relative to the Recursive NN baseline in all performance metrics are shown in bold. Scores reported for MEIA (average) were calculated by averaging the scores reported above over all seven MEIA-based models and mean that plus/minus one standard deviations are reported in brackets.

Table 3. Three-Fold Cross-Validation Performances for the Binary Copolymer Tg

| models            | R2     | MAE   | RMSE  | Spearman’s ρ |
|-------------------|--------|-------|-------|--------------|
| Composition-only  | 0.985(0.980–0.993) | 3.572(2.610–4.122) | 5.182(3.498–6.701) | 0.993(0.989–0.997) |
| Descriptor-based  | 0.988(0.984–0.990) | 3.407(3.035–3.934) | 4.833(4.285–5.885) | 0.991(0.989–0.993) |
| WS                | 0.989(0.986–0.992) | 3.234(2.464–4.014) | 4.507(3.993–4.975) | 0.991(0.991–0.996) |
| Concat-Slef-MHA   | 0.980(0.976–0.983) | 4.020(3.873–4.158) | 6.122(5.337–7.274) | 0.986(0.981–0.991) |
| WS-Self-MHA       | 0.977(0.959–0.989) | 4.664(3.353–5.976) | 6.377(4.370–9.534) | 0.992(0.988–0.994) |
| Concat-MHA        | 0.985(0.971–0.991) | 3.817(2.942–5.322) | 5.293(3.890–7.767) | 0.992(0.987–0.995) |
| Concat-Slef-RNNA  | 0.979(0.966–0.986) | 4.478(3.787–5.541) | 6.188(5.013–7.620) | 0.987(0.984–0.991) |
| WS-Self-RNNA      | 0.990(0.988–0.994) | 3.019(2.366–1.695) | 4.244(3.393–5.165) | 0.993(0.990–0.995) |
| Concat-RNNA       | 0.983(0.974–0.990) | 4.046(3.181–5.167) | 5.438(4.712–6.646) | 0.990(0.987–0.992) |
| MEIA (average)    | 0.983(0.980–0.988) | 3.868(3.514–4.192) | 5.453(4.619–6.480) | 0.990(0.988–0.992) |

Mean scores are reported with the minimum and maximum scores in the three runs (in brackets). Scores for MEIA (average) were calculated by averaging the scores of the all seven MEIA-based models in each run. Then, the mean, minimum, and maximum scores in the three runs are reported.

The dataset used for this task was a collection of compositions and Tg of 275 random copolymers and homopolymers comprising 12 monomers. Among them, 57 samples were selected as the test set, with the remaining samples used as the train set.

The evaluated performances are shown in Table 2, with the average scores over all the seven MEIA-based models, called MEIA (average). Six out of seven MEIA-based models evaluated here outperformed the recursive NN baseline in all performance metrics. Furthermore, scores for the recursive NN baseline fell outside one standard deviation from the scores of MEIA (average) in all performance metrics. On the other hand, all scores of composition-only and descriptor-based baselines fell within one standard deviation from the scores of MEIA (average), and relative superiority of these models is unclear from this comparison. The significantly enhanced performance relative to the permutation-dependent recursive NN baseline is considered attributable to the introduction of permutation invariance. Unexpectedly, the performance of the recursive NN baseline is worse than that of the composition-only baseline in all performance metrics. It is considered that in the recursive NN model, the negative effect of an artifact induced by permutation dependence exceeds the positive effect of incorporating structural information. Accordingly, in the following experiments, the recursive NN model was not considered in the performance comparisons.

To make a clear comparison among prediction performances of MEIA-based models and composition-only and descriptor-based baselines, three-fold cross-validation test was performed where union of the original train and test sets was randomly divided into three new test sets. Average scores of the cross-validation test are shown in Table 3 with the minimum and maximum scores in the three runs. As a result, WS and WS-Self-RNNA showed at least comparative performances relative to the composition-only and descriptor-based baselines with slightly better average performances in R2, MAE, and RMSE; however, considering the minimum and maximum ranges in three runs (brackets in Table 3), there is no significant performance improvement by these MEIA-based models relative to the baselines. Furthermore, average scores of MEIA (average) over three runs were slightly worse than the composition-only and descriptor-based baselines for all the performance metrics, although the minimum and maximum range of each score was overlapped between them. Considering the dominant role of composition information, which can be represented as a tabulated data, this result is consistent with the well-known superiority of shallow learning models for the tabulated data relative to DNNs. Although there is no significant performance improvement by MEIA-based models, the result that two MEIA-based models showed at least comparative accuracies relative to the composition-only and descriptor-based baselines in this interpolation task showed that MEIA-based models can properly handle the composition information for each component.

Case Study 2: Regression for Unknown Molecular Combinations. The second case study is on a previously examined regression tasks for multicomponent materials with
unknown combinations of component molecules. In this task, each test data comprised unknown combinations of component molecules. Such a setting of dataset is known as “mixture-out validation” and seen in quantitative structure–property relationship (QSPR) studies for mixtures. Among them, Oprisiu et al. exhaustively studied regression models for datasets comprising thousands of densities of binary mixtures, by changing molecular descriptors, mixing rules, and machine learning models. They generated feature vectors of mixtures from the classical molecular descriptors, such as ChemAxon, Dragon, and Inductive descriptors by applying simple mixing rules such as weighted summation (Figure 1d). Their study is one of the most sophisticated forms of machine learning modeling for multicomponent materials. Although the best model reported by Oprisiu et al. used the support vector regression (SVR) and Chemaxon descriptors, in this study, the FCNN with ECPF is used as the baseline because, in contrast to ECFP and MEIA-based models, Chemaxon uses much information about molecules including 3D molecular structures and physicochemical properties in the descriptor calculation. Also, the actual best model for this task has not been determined in this study.

The “mixture-out” dataset of mixture density is available in the website of Oprisiu et al., OCHEN, and comprises 3857 and 672 samples of the train and test sets, featured 118 and 46 component molecules, respectively. To evaluate the prediction performances of MEIA-based models on multiple “mixture-out” data splits, the train and test sets used by Oprisiu et al. were merged into the single dataset, and 387 combinations of component molecules found in the resulting dataset were randomly divided into three groups. Then, three test sets were reconstructed according to this grouping, and for each test set, samples not contained in the test set were assigned to a train set. Finally, test set mixtures containing a component molecule that is not contained in the train set were reassigned to the train set. The regression objective of this task is deviations of the experimental densities of mixtures from the ideal densities defined as weighted averages of the density of each component.

The evaluated average performances for baselines and MEIA-based models (models with GC in embed column) are shown in Table 4 with the minimum and maximum scores in the three runs. As a result, the WS-Self-RNNA model showed at least comparative performance relative to the descriptor-based baseline with better average scores in all the performance metrics with no overlap of the minimum and maximum ranges of R2 and RMSE. However, the other six MEIA-based models and s MEIA (average) showed worse average performances relative to the descriptor-based baseline in most scores. This is probably because the embedding network of the GC model, which is used as the embed block in these MEIA-based models, was unsuitable for this dataset. Accordingly, the embedding network of MPNN, another popular molecular DNN model, was also examined as the embed block.

The evaluated performances for MPNN-based models are also shown in Table 4 (models with MPNN in embed column). As expected, enhanced average performances relative to corresponding GC-based models were observed for all MPNN-based models in all performance metrics. Moreover, four out of seven MPNN-based models showed better average performances than the descriptor-based baseline for all performance metrics, with almost no overlap of the minimum and maximum ranges of scores. MEIA (average) also showed better average scores than the descriptor-based baselines for all performance metrics, with almost no overlap of the minimum and maximum ranges of scores. These results show that the MEIA architecture can extend both GC and MPNN models to multicomponent systems using the same procedure and that performances of MEIA-based models are dependent on the suitability of the embedding networks for the given task.

The observation of clearly improved performances of MEIA-based models relative to the descriptor-based baseline is contrast to the case study 1 where the effect of the structural information on the modelling was small. To reveal the role of the structural information in this task, the performance of the composition-only baseline was also evaluated. In contrast to the case study 1,
Table 5. Average Prediction Performances for the Mixture Boiling Point in Three Runs

| embed   | models | R2       | MAE     | RMSE   | Spearman’s $\rho$ |
|---------|--------|----------|---------|--------|-------------------|
| GC      | Composition-only | 0.430 (0.305−0.521) | 11.316 (9.076−12.753) | 18.616 (15.496−22.293) | 0.728 (0.712−0.754) |
|         | Descriptor-based  | 0.626 (0.548−0.721) | 9.619 (8.235−11.093) | 15.047 (11.817−16.748) | 0.795 (0.771−0.837) |
|         | WS      | 0.651 (0.570−0.699) | 9.409 (8.426−10.042) | 14.555 (12.280−17.544) | 0.798 (0.757−0.829) |
|         | Concat-Slef-MHA | 0.597 (0.562−0.663) | 10.423 (10.011−10.958) | 15.589 (14.326−17.629) | 0.779 (0.712−0.816) |
|         | WS-Self-MHA   | 0.674 (0.648−0.717) | 8.793 (8.489−9.010) | 14.015 (13.118−15.650) | 0.831 (0.741−0.880) |
|         | Concat-MHA    | 0.673 (0.645−0.690) | 8.994 (8.268−9.878) | 14.091 (12.620−15.930) | 0.820 (0.786−0.865) |
|         | Concat-Slef-RNNA | 0.637 (0.605−0.654) | 9.309 (8.755−10.101) | 14.793 (14.074−15.801) | 0.813 (0.748−0.869) |
|         | WS-Self-RNNA  | 0.607 (0.581−0.652) | 9.842 (9.584−10.038) | 15.393 (14.490−17.128) | 0.793 (0.726−0.830) |
|         | Concat-RNNA   | 0.620 (0.542−0.705) | 9.851 (9.426−10.636) | 15.135 (13.402−18.089) | 0.795 (0.733−0.855) |
|         | MEIA (average)$^b$ | 0.637 (0.603−0.681) | 9.517 (9.123−9.769) | 14.796 (13.638−16.824) | 0.804 (0.743−0.837) |
| MPNN    | WS      | 0.628 (0.336−0.783) | 8.651 (6.793−11.311) | 14.716 (10.434−21.785) | 0.825 (0.749−0.880) |
|         | Concat-Slef-MHA | 0.673 (0.529−0.746) | 8.622 (7.238−10.122) | 14.033 (11.317−18.356) | 0.827 (0.769−0.873) |
|         | WS-Self-MHA   | 0.665 (0.491−0.766) | 8.670 (7.160−10.592) | 14.174 (10.836−19.075) | 0.822 (0.762−0.858) |
|         | Concat-MHA    | 0.541 (0.259−0.728) | 11.386 (8.040−15.232) | 16.535 (11.673−23.013) | 0.753 (0.601−0.833) |
|         | Concat-Slef-RNNA | 0.617 (0.433−0.764) | 8.842 (7.854−10.326) | 15.092 (11.977−20.134) | 0.832 (0.776−0.882) |
|         | WS-Self-RNNA  | 0.655 (0.447−0.778) | 8.410 (7.272−10.312) | 14.306 (11.413−19.880) | 0.831 (0.763−0.890) |
|         | Concat-RNNA   | 0.668 (0.585−0.795) | 8.127 (7.554−9.028) | 13.697 (11.167−17.218) | 0.844 (0.812−0.892) |
|         | MEIA (average)$^b$ | 0.638 (0.440−0.746) | 8.958 (7.451−10.989) | 14.651 (11.649−19.923) | 0.819 (0.747−0.872) |

“Mean scores are reported with the minimum and maximum scores in the three runs (in brackets). $^b$Scores for MEIA (average) were calculated by averaging the scores of the all seven MEIA-based models in each run. Then, the mean, minimum, and maximum scores in the three runs are reported.

The descriptor-based baseline showed clearly improved performances relative to the composition-only baseline, with no overlap of the minimum and maximum ranges of scores in all performance metrics, which shows that structural information plays a major role in this task. This indicates that, for a machine learning task on a multicomponent molecular systems where structural information plays a major role, the feature representation learning by embedding networks largely benefits the modelling, which is consistent with excellent performances of single-component molecular DNN models for machine learning tasks on molecular structures.

Results of the comparison between MEIA-based models and the best model reported by Oprisiu et al. $^{12}$, which employs SVR and Chemaxon descriptors, using the same training and test sets are also provided in Table S4. Although three MPNN-based models showed better performances than the Chemaxon-based model in all the performance metrics, however, further experiments are needed to show reproducibility of these results and determine the best model for this dataset.

**Case Study 3: Regression for Unknown Component Molecules.** The third case study is on a “compound-out validation” for boiling points of binary mixtures, which was also studied by Oprisiu et al. $^{12}$ In this task, each test data contains at least one unknown component molecule and cannot properly be predicted by interpolating the composition space. This setting is more difficult than the other two validation settings.

The mixture boiling point dataset comprising 3239 and 1309 train and test data, featured 67 and 56 component molecules, respectively. To evaluate the prediction performances of MEIA-based models on multiple “compound-out” data splits, the train and test sets used by Oprisiu et al. were merged into the single dataset, and 98 component molecules found in the resulting dataset were randomly divided into three groups. Then, three test sets were reconstructed according to this grouping, by assigning all mixtures containing molecules belonging to each group to the test set. Also, for each test set, samples not contained in the test set were assigned to the train set. Although the best model reported by Oprisiu et al. used the SVR and Dragon descriptors, $^{12}$ in this study, the FCNN with ECFP is used as the baseline because, in contrast to ECFP and MEIA-based models, Dragon uses much information about molecules including 3D molecular structures and physicochemical properties in the descriptor calculation. Then, the actual best model for this task has not been determined in this study.

As with the mixture density modeling in the case study 2, the performances of both GC- and MPNN-based MEIA models were examined. The evaluated average performances are shown in Table 5 with the minimum and maximum scores in the three runs. As a result, 9 out of 14 GC and MPNN-based MEIA models showed at least comparative performances relative to the descriptor-based baseline with better average scores over the three run in all the performance metrics, and MEIA (average) also showed slightly better performances than the descriptor-based baseline in the average over the three runs for all performance metrics. These results again showed that the MEIA-architecture can extend both GC and MPNN models to multicomponent systems in the same manner, with at least comparative performances with the descriptor-based baseline. However, the minimum and maximum ranges for all scores are overlapped between all the MEIA-based models and descriptor-based baseline. Therefore, the performance improvement by the DNN models in this task is not clear. This is due to large deviations of performances of models among the three different data splits.

Noted that in contrast to the case study 2, in this case study, clear superiority of MPNN-based models relative to GC-based models was not observed.

The performances of the composition-only and descriptor-based baselines were also compared, and the descriptor-based baseline showed improved average performances relative to the composition-only model in all performance metrics. However, the minimum and maximum ranges for MAE and RMSE were overlapped between the composition-only and descriptor-based baselines due to large deviations of performances of models among the three different data splits.
Although 4 out of 14 GC- and MPNN-Dragon descriptors, using the same training and test sets are also prepared by collecting compositions and Tg of small sets of experimental data. The dataset for this task was most challenging task, compound-out validation for a collection performances of MEIA-based models were examined for the results and determine the best model for this dataset. Further experiments are needed to show reproducibility of these Dragon-based model in all the performance metrics, however, based MEIA models showed better performances than the best model reported by Oprisiu et al., which employs SVR and composition-only and descriptor-based baselines, and the average results are shown in Table 6. Scatter plots of predicted versus experimental values are also shown in Figure 4.

Results of comparison between the MEIA-based models and best model reported by Oprisiu et al., which employs SVR and Descriptor-based 0.309(0.258−0.359) Composition-only 0.076(−0.009−0.161) 62.615(59.477−65.753) 86.19(81.496−90.883) 0.163(0.006−0.321) Descriptor-based 0.309(0.258−0.359) 56.294(54.505−58.084) 74.567(71.218−77.917) 0.581(0.520−0.642) WS 0.562(0.533−0.590) 45.953(43.811−48.095) 59.352(57.928−60.775) 0.811(0.779−0.843) Concat-Slef-MHA 0.424(0.404−0.444) 53.627(52.683−54.571) 68.085(67.474−68.696) 0.742(0.699−0.786) WS-Self-MHA 0.540(0.511−0.570) 45.458(43.638−47.278) 60.788(59.352−62.224) 0.792(0.769−0.816) Concat-MHA 0.504(0.498−0.510) 48.088(46.671−49.504) 63.174(62.249−64.098) 0.737(0.728−0.746) Concat-Self-RNNA 0.456(0.430−0.481) 52.813(50.613−55.013) 66.154(65.147−67.161) 0.746(0.687−0.805) WS-Self-RNNA 0.552(0.547−0.557) 46.810(45.990−47.629) 60.051(59.196−60.906) 0.810(0.777−0.844) Concat-RNNA 0.485(0.473−0.496) 47.376(46.079−48.673) 64.400(63.160−65.640) 0.758(0.707−0.810) MEIA (average) 0.503(0.492−0.515) 48.589(47.069−50.109) 63.143(62.935−63.352) 0.771(0.735−0.807)

“Mean scores are reported with the minimum and maximum scores in the two runs (in brackets). Scores of models are shown in bold if the worst scores in the two runs are equal or better than the best score for the descriptor-based baseline. Scores for MEIA (average) were calculated by averaging the scores of all seven MEIA-based models in each run. Then, the mean, minimum, and maximum scores in the two runs are reported.

Table 6. Average Prediction Performances for the Ternary Copolymer Tg in Two Runsa,b

| models                  | R2          | MAE          | RMSE         | Spearman’s ρ   |
|-------------------------|-------------|--------------|--------------|----------------|
| Composition-only        | 0.076(−0.009−0.161) | 62.615(59.477−65.753) | 86.19(81.496−90.883) | 0.163(0.006−0.321) |
| Descriptor-based        | 0.309(0.258−0.359) | 56.294(54.505−58.084) | 74.567(71.218−77.917) | 0.581(0.520−0.642) |
| WS                      | 0.562(0.533−0.590) | 45.953(43.811−48.095) | 59.352(57.928−60.775) | 0.811(0.779−0.843) |
| Concat-Slef-MHA         | 0.424(0.404−0.444) | 53.627(52.683−54.571) | 68.085(67.474−68.696) | 0.742(0.699−0.786) |
| WS-Self-MHA             | 0.540(0.511−0.570) | 45.458(43.638−47.278) | 60.788(59.352−62.224) | 0.792(0.769−0.816) |
| Concat-MHA              | 0.504(0.498−0.510) | 48.088(46.671−49.504) | 63.174(62.249−64.098) | 0.737(0.728−0.746) |
| Concat-Self-RNNA        | 0.456(0.430−0.481) | 52.813(50.613−55.013) | 66.154(65.147−67.161) | 0.746(0.687−0.805) |
| WS-Self-RNNA            | 0.552(0.547−0.557) | 46.810(45.990−47.629) | 60.051(59.196−60.906) | 0.810(0.777−0.844) |
| Concat-RNNA             | 0.485(0.473−0.496) | 47.376(46.079−48.673) | 64.400(63.160−65.640) | 0.758(0.707−0.810) |
| MEIA (average)          | 0.503(0.492−0.515) | 48.589(47.069−50.109) | 63.143(62.935−63.352) | 0.771(0.735−0.807) |

Mean scores are reported with the minimum and maximum scores in the two runs (in brackets). Scores of models are shown in bold if the worst scores in the two runs are equal or better than the best score for the descriptor-based baseline. Scores for MEIA (average) were calculated by averaging the scores of all seven MEIA-based models in each run. Then, the mean, minimum, and maximum scores in the two runs are reported.

Case Study 4: Regression on a Sparse Dataset. Finally, performances of MEIA-based models were examined for the most challenging task, compound-out validation for a collection of small sets of experimental data. The dataset for this task was prepared by collecting compositions and Tg of film-shaped ternary linear random copolymers from the open-access PoLyInfo database. Although some records in the PoLyInfo do not contain complete information about their composition, records were collected as far as possible if a summation of molecular weights of component monomers were smaller than 1000. Also, collected data considered to be at least in some extent reflect the natural distribution of multicomponent linear copolymers in literature. The collected dataset contains 83 copolymers comprising 55 component monomers and was produced by 36 literature studies. Figure S10 is a histogram for the number of appearances of each monomer in the dataset and shows that most monomers in the dataset are used only a few times. Note that although DNN models in this study does not consider some factors that affect Tg of polymers, such as molecular weights, this dataset can be used to examine numerical properties of models on the extremely sparse ternary system.

To create dataset for “compound-out validation”, the 41 monomers, which appear in the collected copolymers less than five times, were chosen as “unknown compounds” in “compound-out validation”. To evaluate the prediction performances of MEIA-based models on multiple “compound-out” data splits, these “unknown compounds” were randomly divided into ten groups. Then, ten test sets were reconstructed according to this grouping, by assigning all copolymers containing monomers belonging to each group to the test set. Also, for the each test set, samples not contained in the test set were assigned to the train set. Because the dataset for this task is small, all scores were evaluated by merging all the predicted values for the ten test sets. Then, the reproducibilities of obtained results were confirmed by repeating the same test twice with different data splits and evaluate the mean, minimum, and maximum over two repeated tests.

The performances of models were evaluated and compared with the composition-only and descriptor-based baselines, and the average results are shown in Table 6. Scatter plots of predicted versus experimental values are also shown in Figure 4.

Here, the composition-only baseline showed an average R2 of 0.076 with almost no predictive ability as shown in Figure 4, underlining the importance of the contribution of monomers contained only in the test set. Meanwhile, the performance of the descriptor-based baseline remains terrible with an average R2 of 0.309. Conversely, MEIA-based models showed remarkable predictive abilities for this dataset. Enhanced average performances relative to the descriptor-based baseline were observed for all MEIA-based models, with no overlap of the minimum and maximum ranges of scores in all the performance metrics except the MAE of Concat-Slef-MHA and Concat-Self-RNNA models. Also, the best R2 model, WS, showed a
significantly improved prediction accuracy (Figure 4 WS) with an average R2 of 0.562. This remarkable performance improvement is attributable to the contribution of molecular structure information to modelling of the sparse dataset. Then, these results again indicate that, for a machine learning task on a multicomponent molecular system where structural information plays a major role, the feature representation learning by embedding networks largely benefit the modelling.

**CONCLUSIONS**

In this study, the permutation-invariant DNN architecture, MEIA for multicomponent molecular systems was proposed. Introducing permutation invariance allows for significant reductions in prediction errors relative to the previously proposed DNN model without permutation invariance. Moreover, using the MEIA architecture, both GC and MPNN models were successfully extended to multicomponent systems by the same procedure, and the best one in MEIA-based models consistently showed comparative or higher accuracies relative to the descriptor-based baseline in all four case studies. These results showed that the MEIA architecture can extend single-component molecular DNN models to multicomponent systems and can allow DNN models to learn both feature representation and composition information directly from the information of molecular structures and their compositions.

It is well known that deep learning models are suitable for unstructured data, while shallow learning models are suitable for tabulated data. On the other hand, suitability of machine learning models for multicomponent material data is not obvious. The comparison of the four case studies on multicomponent molecular systems indicates that for machine learning tasks where the molecular structure information plays a minor role, the performance improvements by DNN models are small as with the case study 1; while for machine learning tasks where the molecular structure information plays a major role as with case studies 2 and 4, the performance improvements by DNN models are large. This trend was notable for the sparse dataset in the case study 4 where MEIA-based models showed significantly improved predictive abilities, whereas the conventional descriptor-based model showed poor performance. Currently, the usage of molecular machine learning in the multicomponent material design remains limited due to the scarcity and nonsystematic nature of experimental datasets for this class of materials. Accordingly, enhancing the prediction ability for such sparse datasets by MEIA-based models will extend the applicability of molecular machine learning in modelling of multicomponent materials.

**FUTURE WORK**

Improved predictive capabilities of MEIA-based models for the sparse dataset have the potential to change current screening approaches for multicomponent molecular systems. Currently, even in recent high-throughput screening studies, the exploration space was limited to fixed combinations or compositions with moderate numbers of component materials because a screening effort on full combination and composition freedoms of multicomponent systems by a limited number of experiments produces uninterpretable results with sparse data. However, there is no guarantee that materials with desired properties will be found in the predefined design space for high-throughput screening studies. To find optimal recipes, redefinition of the design space and screening within it often need to be repeated, which hinders the application of high-throughput screening methods to the multicomponent materials design, and therefore, the trial-and-error-based small-scale screening on combination and composition spaces of multicomponent systems is still the primary approach for the development of multicomponent materials. Meanwhile, MEIA-based models will allow screening on full combination and composition freedoms of all candidate raw materials, in combination with self-driving laboratories composed of high-throughput screening equipment and machine learning-based sequential optimization methods such as Bayesian optimization, which will largely expand accessible design spaces of multicomponent systems. Furthermore, MEIA-based models will speed-up the non-high-throughput screening by providing recommendations for better raw materials that is expected to improve material properties from libraries of available materials based on sparse experimental datasets. Another attractive feature of DNNs is the transfer learning capability where feature representations learned by other datasets are used to improve machine learning performance on small datasets. By employing embedding networks of prelearned models, MEIA-based models can also perform the transfer learning, which is useful especially for the non-high-throughput screening studies generating small experimental data.

Although this study focused on the general extension of DNNs for multicomponent systems and special treatments for each input material and output property were not considered, there will be room for improvement of the performances of the MEIA-based models for individual applications. For copolymers, the treatment of input data of monomers is a hopeful direction for the performance improvement, which includes the processing for connection points of monomers and the introduction of the reaktant information of copolymers. Another direction for performance improvement is the variation of the form of the summation function, which allows generating various types of mixing rules, such as the weighted geometric mean and weighted square sum, which are also used in the QSPR studies for mixtures. Furthermore, the introduction of the embedding networks of state-of-the-art molecular machine learning models for each type of dataset as the embed block will be a convenient way to improve MEIA-based models because prediction performances of MEIA-based models largely depend on the architectures of the embedding networks, as seen in the case study 2 where the MPNN-based models showed the improved performances relative to the GC-based models in the binary mixture density prediction.

It is worth to point out that most of DNN models for single-component systems can be divided into embedding and prediction networks, and the MEIA architecture can extend these models to multicomponent systems by exploiting their embedding networks. Therefore, the MEIA architecture that enables feature representation learning from unstructured data incorporating composition information will extend DNN models for a wide range of chemical data to multicomponent systems. A promising extension of current study is DNN models for nonmolecular multicomponent systems including inorganic crystals and metal alloys using unstructured data such as spectrum and simulation results.

**ASSOCIATED CONTENT**

*Supporting Information*

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acsomega.0c02599.
Brief descriptions of embed-interact-aggregate models for sets, neural network architectures, histogram for the number of appearances of each monomer in the ternary copolymer dataset, hyper-parameter settings, and performances of models (PDF)

List of ternary copolymers used in the case study (XLS)

■ AUTHOR INFORMATION

Corresponding Author
Kyohei Hanaoka – Hitachi Chemical Company, Ltd., Tsukuba City 300-4247, Japan; orcid.org/0000-0001-5634-0029; Email: k-hanaoka@hitachi-chem.co.jp

Complete contact information is available at: https://pubs.acs.org/doi/10.1021/acsomega.0c02599

Notes
The author declares no competing financial interest.

■ ACKNOWLEDGMENTS

This work was supported by Hitachi Chemical Company, Ltd. The author gratefully acknowledges all the support from Hitachi Chemical Company, Ltd. and his colleagues. There is no funding received for this study.

■ REFERENCES

(1) Harillo-Baños, A.; Rodríguez-Martínez, X.; Campoy-Quiles, M. Efficient Exploration of the Composition Space in Ternary Organic Solar Cells by Combining High-Throughput Material Libraries and Hyperspectral Imaging. Adv. Energy Mater. 2020, 10, 1902417.

(2) Matsuda, S.; Nishioka, K.; Nakanishi, S. High-Throughput Combinatorial Screening of Multi-Component Electrolyte Additives to Improve the Performance of Li Metal Secondary Batteries. Sci. Rep. 2019, 9, 6211.

(3) Gomez, I. J.; Wu, J.; Roper, J.; Beckham, H.; Meredith, J. C. High Throughput Screening of Mechanical Properties and Scratch Resistance of Tricomponent Polyurethane Coatings. ACS Appl. Polym. Mater. 2019, 1, 3064–3073.

(4) Kuo, T. C.; Malvadkar, N. A.; Drumright, R.; Cesaretti, R.; Bishop, M.; T.; High-Throughput Industrial Coatings Research at the Dow Chemical Company. ACS Comb. Sci. 2016, 18, 507.

(5) Liu, Y.; Zhao, T.; Ju, W.; Shi, S. Materials Discovery and Design Using Machine Learning. J. Mater. 2017, 3, 159–177.

(6) Wilbraham, L.; Sprick, R. S.; Jelfs, K. E.; Zwijsen, M. A. Mapping Binary Copolymer Property Space with Neural Networks. Chem. Sci. 2019, 10, 4973–4984.

(7) Ajmani, S.; Rogers, S. C.; Barley, M. H.; Livingstone, D. J. Application of QSPR to Mixtures. J. Chem. Inf. Model. 2006, 46, 2043–2053.

(8) Pertinetto, C. G.; Duce, C.; Micheli, A.; Solaro, R.; Tiné, M. R. QSPR Analysis of Copolymers by Recursive Neural Networks: Prediction of the Glass Transition Temperature of (Meth)Acrylic Random Copolymers. Mol. Inf. 2010, 29, 635–643.

(9) Katritzky, A. R.; Sotynova-SlavoVA, I. B.; TamM, K.; TamM, T.; Karelson, M. Application of the QSPR Approach to the Boiling Points of Azeotropes. J. Phys. Chem. A 2011, 115, 3475–3479.

(10) Oliiferenko, A. A.; Oliiferenko, P. V.; Torrecilla, J. S.; Katritzky, A. R. Boiling Points of Ternary Azeotropic Mixtures Modeled with the Use of the Universal Solvation Equation and Neural Networks. Ind. Eng. Chem. Res. 2012, 51, 9123–9128.

(11) Oprisiu, I.; Varlamova, E.; Muratov, E.; Artemenko, A.; Marcou, G.; Polischuk, P.; Kuz’Min, V.; Varnek, A. QSPR Approach to Predict Nonadditive Properties of Mixtures. Application to Bubble Point Temperatures of Binary Mixtures of Liquids. Mol. Inf. 2012, 31, 491–502.

(12) Oprisiu, I.; Novotarskyi, S.; Tetko, I. V. Modeling of Non-Additive Mixture Properties Using the Online CHEmical Database and Modeling Environment (OChEM). Aust. J. Chem. 2013, 5, 4.

(13) Saldana, D. A.; Starck, L.; Mougin, P.; Rousseau, B.; Creton, B. Prediction of Flash Points for Fuel Mixtures Using Machine Learning and a Novel Equation. Energy Fuels 2013, 27, 3811–3820.

(14) Gaudin, T.; Rotureau, P.; Fayet, G. Combining Mixing Rules with QSPR Models for Pure Chemicals to Predict the Flash Points of Binary Organic Liquid Mixtures. Fire Saf. J. 2015, 74, 61–70.

(15) Gaudin, T.; Rotureau, P.; Fayet, G. Mixture Descriptors toward the Development of Quantitative Structure-Property Relationship Models for the Flash Points of Organic Mixtures. Ind. Eng. Chem. Res. 2015, 54, 6596–6604.

(16) Jiao, L.; Zhang, X.; Qin, Y.; Wang, X.; Li, H. QSPR Study on the Flash Point of Organic Binary Mixtures by Using Electrotropological State Index. Chemom. Intell. Lab. Syst. 2016, 156, 211–216.

(17) Wang, T.; Tang, L.; Luan, F.; Cordeiro, M. N. D. S. Prediction of the Toxicity of Binary Mixtures by Qsar Approach Using the Hypothetical Descriptors. Int. J. Mol. Sci. 2018, 19, 3423.

(18) Kar, S.; Ghosh, S.; Leszczynski, J. Single or Mixture Halogenated Chemicals? Risk Assessment and Developmental Toxicity Prediction on Zebrafish Embryos Based on Weighted Descriptors Approach. Chemosphere 2018, 210, 588–596.

(19) Qin, L. T.; Chen, Y. H.; Zhang, X.; Mo, L. Y.; Zeng, H. H.; Liang, Y. P. QSAR Prediction of Additive and Non-Additive Mixture Toxicities of Antibiotics and Pesticide. Chemosphere 2018, 198, 122–129.

(20) Wang, B.; Park, H.; Xu, K.; Wang, Q. Prediction of Lower Flammability Limits of Blended Gases Based on Quantitative Structure–Property Relationship. J. Therm. Anal. Calorim. 2018, 132, 1125–1130.

(21) Wang, Y.; Yan, F.; Jia, Q.; Wang, Q. Distributive Structure-Property Relationship for Flash Point of Multiple Components Mixture. Fluid Phase Equilib. 2018, 474, 1–5.

(22) Pan, Y.; Ji, X.; Ding, L.; Jiang, J. Prediction of Lower Flammability Limits for Binary Hydrocarbon Gases by Quantitative Structure—a Property Relationship Approach. Molecules 2019, 24, 748.

(23) Fayet, G.; Rotureau, P. New QSPR Models to Predict the Flammability of Binary Liquid Mixtures. Mol. Inf. 2019, 38, 1800122.

(24) Lusci, A.; Pollastri, G.; Baldi, P. Deep Architectures and Deep Learning in Chemoinformatics: The Prediction of Aqueous Solubility for Drug-like Molecules. J. Chem. Inf. Model. 2013, 53, 1563–1575.

(25) Duvenaud, D.; Maclaurin, D.; Aguilera-Iparraguirre, J.; Gemez-Bombarelli, R.; Hirzel, T.; Aspuru-Guzik, A.; Adams, R. P. Convolutional Networks on Graphs for Learning Molecular Fingerprints. In Advances in Neural Information Processing Systems; Neural Information Processing Systems Foundation, Inc.: 2015; pp 2224–411.

(26) Kearnes, S.; McCloskey, K.; Berndl, M.; Pande, V.; Riley, P. Molecular Graph Convolutions: Moving beyond Fingerprints. J. Comput-Aided Mol. Des. 2016, 30, 595–608.

(27) Schütz, K. T.; Arbabzadah, F.; Chmiela, S.; Müller, K. R.; Tkatchenko, A. Quantum-Chemical Insights from Deep Tensor Neural Networks. Nat. Commun. 2017, 8, 13890.

(28) Smith, J. S.; Isayev, O.; Roitberg, A. E. ANI-1: An Extensible Neural Network Potential with DFT Accuracy at Force Field Computational Cost. Chem. Sci. 2017, 8, 3192–3203.

(29) Yao, K.; Herr, J. E.; Brown, S. N.; Parkhill, J. Intrinsic Bond Energies from a Bonds-in-Molecules Neural Network. J. Phys. Chem. Lett. 2017, 8, 2689–2694.

(30) Schütz, K. T.; Kindermans, P. J.; Sauceda, H. E.; Chmiela, S.; Tkatchenko, A.; Müller, K. R. SchNet: A Continuous-Filter Convolutional Neural Network for Modeling Quantum Interactions. In Advances in Neural Information Processing Systems; Neural Information Processing Systems Foundation, Inc.: 2017; pp 992–1002.

(31) Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; Dahl, G. E. Neural Message Passing for Quantum Chemistry. In Proceedings of the 34th International Conference on Machine Learning-Volume 70; ICML: 2017; pp 1263–1272, DOI: 10.1002/nme.2457.
(32) Tsukinoki, M.; Mizoguchi, T. Fast and Accurate Molecular Property Prediction: Learning Atomic Interactions and Potentials with Neural Networks. J. Phys. Chem. Lett. 2018, 9, 5733–5741.

(33) Matlock, M. K.; Dang, N. L.; Swamidass, S. J. Learning a Local-Variable Model of Aromatic and Conjugated Systems. ACS Cent. Sci. 2018, 4, 52–62.

(34) Krizhevsky, A.; Sutskever, I.; Hinton, G. E. ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of the 25th International Conference on Neural Information Processing Systems; Neural Information Processing Systems Foundation, Inc.: 2012; pp 1097–1105.

(35) Wu, Y.; Schuster, M.; Chen, Z.; Le, Q. V.; Norouzi, M.; Macherey, W.; Krikun, M.; Cao, Y.; Gao, Q.; Macherey, K.; et al. Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. arXiv Prepr. arXiv1609.08144 2016.

(36) Wu, Z.; Ramsundar, B.; Feinberg, E. N.; Gomes, J.; Geniesse, C.; Pappu, A. S.; Leswing, K.; Pande, V. MoleculeNet: A Benchmark for Molecular Machine Learning. Chem. Sci. 2018, 9, 513–530.

(37) Faber, F. A.; Hutchison, L.; Huang, B.; Gilmer, J.; Schoenholz, S.; Dahl, G. E.; Vinyals, O.; Kearnes, S.; Riley, P. F.; Von Lilienfeld, O. A. Prediction Errors of Molecular Machine Learning Models Lower than Hybrid DFT Error. J. Chem. Theory Comput. 2017, 13, 5255–5264.

(38) Jha, D.; Ward, L.; Paul, A.; Liao, W. K.; Choudhary, A.; Wollowton, C.; Agrawal, E. ElemNet: Deep Learning the Chemistry of Materials From Only Elemental Composition. Sci. Rep. 2018, 8, 17593.

(39) Zeng, S.; Zhao, Y.; Li, G.; Wang, R.; Wang, X.; Ni, J. Atom Table Convolutional Neural Networks for an Accurate Prediction of Compounds Properties. npj Comput. Mater. 2019, 5, 84.

(40) Vinyals, O.; Bengio, S.; Kudlur, M. Order Matters: Sequence to Sequence for Sets. arXiv Prepr. arXiv1511.06391 2015.

(41) Zaheer, M.; Kottur, S.; Ravabakhsh, S.; Póczos, B.; Salakhutdinov, R.; Smola, A. J. Deep Sets. In Advances in Neural Information Processing Systems 30; Neural Information Processing Systems Foundation, Inc.: 2017; pp 3391–3401.

(42) Qi, C. R.; Su, H.; Mo, K.; Guibas, L. J. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; IEEE: 2017; pp 652–660.

(43) Qi, C. R.; Yi, L.; Su, H.; Guibas, L. J. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In Advances in neural information processing systems; Neural Information Processing Systems Foundation, Inc.: 2017; pp 5099–5108.

(44) Ilse, M.; Tomczak, J. M.; Welling, M. Attention-Based Deep Multiple Instance Learning. arXiv Prepr. arXiv1802.04712 2018.

(45) Lee, J.; Lee, Y.; Kim, J.; Kosior, A. R.; Choi, S.; Teh, Y. W. Set Transformer: A Framework for Attention-Based Permutation-Invariant Neural Networks. In International Conference on Machine Learning; PMLR: 2018; pp 3744–3753.

(46) Yang, J.; Zhang, Q.; Ni, B.; Li, L.; Liu, J.; Zhou, M.; Tian, Q. Modeling Point Clouds with Self-Attention and Gumbel Subset Sampling. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; IEEE: 2019; pp 3323–3332.

(47) Vassani, A.; Brain, G.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. In Advances in neural information processing systems; Neural Information Processing Systems Foundation, Inc.: 2017; pp 5998–6008.

(48) Kingma, D. P.; Ba, J. Adam: A Method for Stochastic Optimization. arXiv Prepr. arXiv1412.6980 2014.

(49) Fox, T. G. Influence of Diluent and of Copolymer Composition on the Glass Temperature of a Polymer System. Bull. Am. Phys. Soc. 1956, 1, 123.

(50) Wiff, D. R.; Altieri, M. S.; Goldfarb, I. J. Predicting Glass Transition Temperatures of Linear Polymers, Random Copolymers, and Cured Reactive Oligomers from Chemical Structure. J. Polym. Sci., Polym. Phys. Ed. 1985, 23, 1165–1176.

(51) Rogers, D.; Hahn, M. Extended-Connectivity Fingerprints. J. Chem. Inf. Model. 2010, 50, 742–754.

(52) Landrum, G. RDKit: Open-source Cheminformatics (http://www.rdkit.org).

(53) Gonzalez, J. GPyOpt: A Bayesian Optimization Framework in Python. 2016.

(54) Chemaxon (http://Chemaxon.Com).

(55) Todeschini, R.; Consonni, V. Handbook of Molecular Descriptors; Wiley-VCH: New York, Chichester, Brisbane, Singapore, Toronto, 2008.

(56) Cherkasov, A.; Ban, F.; Santos-Filho, O.; Thorsteinson, N.; Fallahi, M.; Hammond, G. L. An Updated Steroid Benchmark Set and Its Application in the Discovery of Novel Nanomolar Ligands of Sex Hormone-Binding Globulin. J. Med. Chem. 2008, 51, 2047–2056.

(57) Otsuka, S.; Kuwajima, I.; Hosoya, J.; Xu, Y.; Yamazaki, M. PolYinfo: Polymer Database for Polymeric Materials Design. In 2011 International Conference on Emerging Intelligent Data and Web Technologies; IEEE: 2011; pp 22–29. DOI: 10.1109/EIDWT.2011.13.

(58) Langner, S.; Häse, F.; Perea, J. D.;Stubhan, T.; Hauch, J.; Roch, L. M.; Heumuelle, T.; Aspuru-Guzik, A.; Brabec, C. J. Beyond Ternary OPV: High-Throughput Experimentation and Self-Driving Laboratories Optimize Multicomponent Systems. Adv. Mater. 2020, 32, 1907801.

(59) Yamada, H.; Liu, C.; Wu, S.; Koyama, Y.; Ju, S.; Shiomi, J.; Morikawa, J.; Yoshida, R. Predicting Materials Properties with Little Data Using Shotgun Transfer Learning. ACS Cent. Sci. 2019, 1717.