ABSTRACT: Predicting both accurate and reliable solubility values has long been a crucial but challenging task. In this work, surrogated model-based methods were developed to accurately predict the solubility of two molecules (solute and solvent) through machine learning and deep learning. The current study employed two methods: (1) converting molecules into molecular fingerprints and adding optimal physicochemical properties as descriptors and (2) using graph convolutional network (GCN) models to convert molecules into a graph representation and deal with prediction tasks. Then, two prediction tasks were conducted with each method: (1) the solubility value (regression) and (2) the solubility class (classification). The fingerprint-based method clearly demonstrates that high performance is possible by adding simple but significant physicochemical descriptors to molecular fingerprints, while the GCN method shows that it is possible to predict various properties of chemical compounds with relatively simplified features from the graph representation. The developed methodologies provide a comprehensive understanding of constructing a proper model for predicting solubility and can be employed to find suitable solutes and solvents.

INTRODUCTION

Solubility (molecular-level mixability via intermolecular interactions) applies to all areas of chemistry, geochemistry, pharmaceutical science, physics, and biochemistry. It is also of great importance in a variety of applications, including environmental prediction, agrochemical design, chemical process design, crystallography, autonomous robotic synthesis using flow chemistry, development of lithographic materials in the semiconductor industry, and drug design, biochemistry, early drug discovery stages, and protein ligand binding in the biomedical field. Specifically, aqueous solubility is essential in many industrial, pharmaceutical, and environmental applications because it is critical for the optimization of the initial process of finding drug candidates. Because all drugs in the body show medicinal effects in the form of aqueous solutions, a lower solubility reduces the medicinal effects and bioavailability.

While determining whether a material can be dissolved is important, solubility experiments, calculations, and predictions are not easy tasks. First, the complex and multidimensional nature of the melting process makes solubility prediction a difficult task. Various factors, such as chemistry, thermodynamics, kinetics, and morphology, need to be considered. In addition, it is almost impossible to collect all the data, which consists of hundreds of thousands of solute–solvent combinations. Although various statistical and machine learning approaches exist, they do not reduce the complexity of the computations because each related feature is either difficult to obtain or requires too many features. Therefore, a simple, accurate, and reliable experimental data-driven solubility prediction method will certainly be welcomed in various fields of research in which solubility prediction is critical.

Historically, three approaches have been used to predict solubility: (1) quantum mechanics, (2) the general solubility equation (GSE) and (3) machine learning. Quantum mechanical methods, such as the ADF COSMO-RS program, calculate the surface charge using density functional theory to obtain the chemical potential, which can be further used to compute thermodynamic equilibrium properties; however, quantum mechanical methods are expensive, and quantitative comparison with experimental results is somewhat difficult to achieve. In addition, a mismatch between the calculated physical properties and experimental data occurs when this method is utilized. The revised GSE utilizes Jørgensen and Duffy's computational method to reduce the absolute error and make the process simpler and more accurate than the original GSE. However, calculations must be performed for all solvent–solute combinations for validation, which is inefficient and requires considerable resources. As a solution, methods using machine learning and statistics have recently emerged. For example, the quantitative structure–property relationship (QSAR/QSPR) is used as a general methodology, and various data mining methodologies have been introduced in the field of solubility prediction. These methodologies have the advantage of providing efficient predictions of solubility because they do not require complex calculations; that is, they are computationally inexpensive. Furthermore, studies...
have been conducted using deep learning frameworks that utilize neural networks to build predictive models, particularly for aqueous solubility prediction. In addition, studies have been conducted to consider a variety of solute–solvent pairs rather than methods that make predictions with one fixed solvent. However, methods in previous studies are limited in their ability to predict solubility. They are unable to consider all solute–solvent combinations because, for most studies, the solvent in the training database is fixed. Furthermore, the models lack a comprehensive understanding of appropriate features for representing the solubility and the molecules. Each of these problems causes poor transferability and generality of the prediction model, which requires the time-consuming development of separate models for different systems.

In this study, two data-driven methodologies for predicting the solubility of solute–solvent sets are proposed: (1) the MFPCP method, a machine learning method in which the molecular fingerprint (MF) and physicochemical properties (PCP) are utilized as descriptors and (2) the GCN method, in which molecules are converted into graphs and a deep learning framework, a graph convolutional neural network (GCN), is used as a prediction model. To construct the training database, previously published solubility databases were merged and arranged. The MFPCP method focuses on computationally inexpensive physicochemical properties that provide the largest increase in the prediction accuracy when used as descriptors. There are several methods to represent molecules. For example, creating features using transformer and using mathematical features are also reasonable methods, but our study sought to use the most fundamental, simple, and clear method above all else. In addition, this study used a molecular fingerprint that best reflects structural simplicity since the purpose is to predict solubility. In addition, when predictive presentation is used, uncertainty would inevitably arise when making predictions for new molecular structures, so a molecular fingerprint, which reflects the molecular structure is used as a representation of the molecules.

While previous studies required numerous computationally expensive features such as HOMO, LUMO, LsolvHsolu, and LsolvHsolv, the MFPCP model only considers key physicochemical features that can be easily obtained. Recent works on predicting solubility through machine learning utilized molecular fingerprints in their work. Previous studies use machine learning/deep learning (ML/DL) methods including random forest (RF), support vector regression (SVR), LightGBM, LASSO and so on, and also deep learning to predict solubility. While the importance of solubility prediction has been emphasized, various studies have been on solubility prediction has been reported. However, previous studies are only focused on aqueous solubility.

In contrast to the structural or physicochemical descriptors used by the MFPCP model, the GCN methodology employs only a simplified molecular-input line-entry system (SMILES) to make predictions; therefore, no other descriptors are required. Moreover, GCNs can be improved by applying various algorithms. For example, DGraphDTA is a multi-input network for drug–target affinity prediction, and the combination of a GCN and graph attention networks further improves the model. Recently, the prediction of aqueous solubility using a graph-based message passing network, directed edge graph isomorphism network, and multilevel GCN has been reported. However, in the previous studies mentioned above, it is common to predict the solubility using an individual data set and a model for one solvent in the previous studies, so there is a limitation because sufficient data is required. Therefore, in this study, a methodology for predicting the solubility between solute–solvents of various combinations was proposed. We believe that the current work will contribute to overcoming the limitations of previous work by showing better typical prediction performance when working with pretrained solvents and in systems requiring complex computations and by predicting more accurate solubility in a simple and convenient way by using the following methodologies as predictive models.

**METHODS**

A flowchart for solubility prediction is shown in Figure 1. It briefly describes the database construction, types of descriptors, and surrogate models for regression and classification. The method to verify the performance of the models in this study is as follows: (1) The training and test set were divided 8:2 and performed 99 times with different configurations.

**Figure 1.** Flowchart of the whole process of solubility prediction. Two methods (Boosting Model, GCN) are used to predict the solubility or to predict whether the solvent material can well dissolve the solute material by three classes (good/fair/bad). Network architectures of GCN are introduced in the Methods. Directions and flow of the data construction process including obtaining property values is mentioned in the Supporting Information (Figure S2).
The solubility database used in this work integrated several individual data sets from different sources previously used in other studies. A list of these databases is provided in Table S1. Initially, the eight data sets had a total of 38992 data entries. Then these data were prioritized according to the number of solute–solvent combinations (because both methodologies proposed in this study use solute–solvent combination data), and deduplication was performed using the IUPAC International Chemical Identifier (InChIKey). Finally, from the initial 38992 data entries, a total of 17536 data entries were arranged, which contained 12849 solutes and 179 solvents, and were expressed using SMILES and InChIKey (Table S1). All molecules represented in SMILES in this data were converted to molecular fingerprints and graphs using the RDKit and PyTorch Geometric (PyG) packages, respectively. Figure S1 shows the distribution of the solubility (logS) values of each data set and the number of solutes and solvents. The most recurrent solutes were anthracene (0.55%), pyrene (0.44%), and acenaphthene (0.33%), while the most recurrent solvents were water (67.44%), ethanol (4.67%), and benzene (2.80%). The combined percentage of the top five solutes was less than 1%, but the combined percentage of the top five solvents, including water (11827), accounted for approximately 75% of the data set.

In addition to predicting solubility values (regression), solubility values were divided into three classes, and classification prediction was performed. Classification was performed for the following reasons. If the purpose is to determine whether substance A will dissolve well in substance B based on certain threshold values, predicting solubility through classification can be more intuitive for producing and analyzing results. Furthermore, predicting solubility through classification should work well because there was an imbalanced combination of solvents and solutes in the data set. The logS value was divided into three classes (low, medium, and high) based on the following criteria:

\[
f(x) = \begin{cases} 
  \text{high, if } x > -1 \\
  \text{medium, if } -3 \leq x \leq -1 \\
  \text{low, if } x < -3 
\end{cases}
\]

The distribution of the target value (logS) and class distribution of the data set are shown in Figure 2 and Figure S2. They show that the logS values were distributed mostly between −2.5 and 0.0 mol/L. In addition, because the class distribution was well balanced, the training process was facilitated. Furthermore, the boxplots in Figure S2 show the logS distribution of data01, data02, and data03, which indicates that the logS values had almost equal distributions in all three data sets. This is important because an imbalanced logS distribution in one of the data sets can lead to biased prediction results.

For the MFPCP method, physicochemical properties were added to the original data set, in addition to the molecular fingerprints. Because data loss occurred during property acquisition, three variations of the original data set were considered, which depended on the availability of each property. Finally, three databases each of which had different features for the prediction model were utilized. The data construction process is detailed in Figure S3, and the variations are named data01, data02, and data03. The goal was to obtain three property values: molecular weight (MW), volume, and melting point (MP).

First, data01 was equal to the original data set, in which no physicochemical properties were added (i.e., only molecular fingerprints were used as descriptors). Second, data02 was the data after obtaining MW and volume. Using RDKit, the data was removed if either property was not calculated, a total of 60 were deleted in the process, and the number of data01 and data02 is the same. Finally, data03 included all three properties along with MP. Data loss from obtaining MP values was considerably higher than that from obtaining the volume and MW values. Each property value and its Pearson correlation coefficient are shown in Figure S4. The R values of MW, volume, and MP according to logS were −0.34, −0.38, and −0.09, respectively. In particularly, MP did not appear to correlate with logS. The final three databases utilized each had different features for the prediction model (data01: MACCS fingerprint; data02: MACCS fingerprint, MW, volume; data03: MACCS fingerprint, MW, volume, MP).

| Data Set | Solute | Solvent | No. of Data |
|----------|--------|---------|-------------|
| data01   | 12849  | 179     | 17536       |
| data02   | 12849  | 179     | 17536       |
| data03   | 3942   | 161     | 6945        |

Figure 2. Data Identification: (a) logS distribution and (b) class distribution of the data set we used. (c) Table showing the number of solutes and solvents of each data and the total number of data. The table shows that the data size of data03 is small compared to the other two data sets due to data loss in the process of obtaining property values. Nevertheless, we use data03 since it contains all of the property values (MW, MP, volume).
MFPCP Method. In this method, molecular fingerprints and physicochemical properties were used as the descriptors. The most common way of representing molecules is the SMILES format. In this work, the chemical structures of the molecules were converted into forms that the computer could decode through a molecular fingerprinting methodology. Converting a molecule into a molecular fingerprint can be performed using various methods, but a 166-dimensional structure called Molecular ACCESS System (MACCS) keys was used in this study. The MACCS fingerprints can be readily obtained using RDKit and are directly converted from SMILES. The obtained fingerprints of the solutes and solvents were used as the descriptors in the predictive model.

Among the various data-driven machine learning methodologies, LightGBM was used. It was chosen because, as a boosting algorithm, it is capable of minimizing prediction error loss and reducing computation time. Because the scale of the three PCP features was different from that of the MACCS fingerprint, a standard scaler was implemented using scikit-learn. GridsearchCV was used for hyperparameter optimization. In particular, the complexity of the model was reduced by adjusting min_child_samples, max_depth, and num_leaves. More details, including the parameter values and code, are included in the Supporting Information.

The fingerprinting methodology was applied to the reference data set, which was obtained from a previous study, and the results from using different combinations of features as descriptors were compared (Table S2). Three physicochemical properties (MP, MW, and volume) were proven to be the most important features (Figure 3). Hence, they were selected as additional descriptors, and the selected properties were then added to the data set used in this study.

The following explains how each property was obtained. The MW and volume values were obtained from RDKit using SMILES. Unlike these, the MP value was not easily obtainable through the python package; therefore, a web crawler was used to scrape data directly from Chemspider, a chemical database site. Units were unified to Celsius, and those without MP data were removed in data03. Despite the considerable amount of data that was removed, data03 was used because the MP has a decisive impact on predicting solubility, as shown in Figure 3. However, because 1/3 of the data was omitted compared to data01 and data02, all three data sets were preserved and used in the prediction process to clearly compare performance differences due to the descriptors used. Various validations have been conducted to demonstrate that the physicochemical properties are effective in improving the performance.

Molecular Graph Representation. The idea of molecular graph representation lies in graph theory. When chemical compounds are expressed as graphs, atoms correspond to nodes, and the bonds between them correspond to edges without weight. A graph is represented as $G=(X, I)$, where $X$ is a node feature matrix and $I$ is an edge index matrix. The one-hot encoded node feature matrix ($N \times 75$, where $N$ is the number of nodes) consisted of 75 dimensions from the eight features of atoms (Table 1). Matrix $I$ was transformed into matrix $A$ for graph convolution. $A$ is an adjacency matrix, where $A_{ij} = 1$ if the $i$th and $j$th atoms have a bond and $A_{ij} = 0$ otherwise; that is, an $N \times N$ matrix, where $N$ is the number of nodes. The SMILES information from the database was transformed into molecular graphs using the RDKit and PyG packages.

![Figure 3](https://pubs.acs.org/acsomega/2c00697/image/2a.png)

**Figure 3.** Feature importance plot for (a) LightGBM regression models and (b) LightGBM classification models. Volume, MW, and MP seem to have high feature importance in both regression and classification tasks. (c) Description and dimension of the descriptors used in MFPCP method.

| Descriptor | Description | Dimension |
|------------|-------------|-----------|
| Molecular Fingerprint (MACCS) | MACCS (Molecular ACCcess System) keys | 166 |
| MP | Experimental Melting Point ($^\circ$C) | 1 |
| MW | Molecular Weight (g/mol) | 1 |
| Volume | Molar Volume of a molecule ($V_m$) | 1 |
Graph Convolutional Network. Graph neural networks (GNNs) have shown outstanding performance in various fields because there are no limitations on the size of graphs (consisting of nodes and edges) used as inputs to GNNs; hence, they provide a flexible format for extracting in-depth information from molecules. In chemistry, molecules are modeled as graphs, and their properties need to be verified for drug discovery. GCNs and GATs are widely used GNN models that have gradually been applied for purposes such as drug property prediction and molecular fingerprint generation, thus presenting a new paradigm for virtual screening that accelerates drug discovery with improved prediction accuracy. Furthermore, protein interface prediction using graph convolution suggests the promising potential of GCNs in drug development.

Solubility predictions require the conversion of solutes and solvents into graphs to obtain logS values or classes. Therefore, the multi-input GCN model in this study was composed of three types of layers: (1) graph convolution layers, (2) a merge layer, and (3) prediction layers. In the graph convolution...
layers, the features of each atom were updated by their neighborhoods. The detailed process flow, including the molecule-to-graph conversion in the GCN and construction of model structures, are shown in Figure 4. The graph convolution algorithms in this study were implemented using PyG (Figure S5) and GCNConv. Detailed formulas are described in the Supporting Information. After the graphs of the solutes and solvents passed through the convolution layers, the activations were concatenated in the merge layer. The merged activations then passed through the fully connected layers for prediction. The output of the GCN model was either the activations were concatenated in the merge layer. The prediction layers consisted of four fully connected layers, each similarly followed by the ReLU activation function, batch normalization, a global pooling layer, and a dropout layer with a rate of 0.1. The R squared, the features of each atom were updated by their neighborhoods. The detailed process flow, including the molecule-to-graph conversion in the GCN and construction of model structures, are shown in Figure 4. The graph convolution algorithms in this study were implemented using PyG (Figure S5) and GCNConv. Detailed formulas are described in the Supporting Information. After the graphs of the solutes and solvents passed through the convolution layers, the activations were concatenated in the merge layer. The merged activations then passed through the fully connected layers for prediction. The output of the GCN model was either the activations were concatenated in the merge layer. The prediction layers consisted of four fully connected layers, each similarly followed by the ReLU activation function, batch normalization, a global pooling layer, and a dropout layer with a rate of 0.1. The R-squared = 0.80 (±0.03), the R-squared value of 0.80 (±0.02) was obtained with mean absolute error (MAE) and root-mean-square error (RMSE) values of 0.65 (±0.03) and 0.96 (±0.05) mol/L, respectively. While a reasonable performance score was achieved with data02, the best score was achieved using all the properties (MACCS fingerprint, volume, MW, and MP). The R-squared value for data03 was 0.85 (±0.03) when the training to test set ratio was 8:2 and 99 different configurations (random selection) of the training set were used. Although data01 contained the largest number of data entries, it did not have the highest prediction accuracy (R^2 = 0.74 (±0.03)); this validates the importance of including meaningful features. The true vs predicted logS plots for all the data sets are shown in Figure Sa and Figure S7a. Interestingly, although the number of data entries was drastically reduced from 17536 to 6945 in the process of obtaining all three PCP values, the performance score of the regression task increased from 0.75 to 0.853 (average R^2) when significant features, such as volume, MW, and MP, are introduced.

For the classification task with data02, the receiver operating characteristic—area under the ROC curve (ROC-AUC) curve and the confusion matrix are shown in Figure 5b. The micro and macro F1 scores were used to evaluate the results of the multiclass classification task more precisely. The accuracy of 77.70% (±1.58%) with a micro F1 value of 0.78 (±0.02) confirmed the high prediction accuracy. It is interesting to note that, as previously observed in the regression task, the best accuracy of 84.16% (±2.56%) and F1 value of 0.84 (±0.03)

| Regression | Classification |
|------------|----------------|
| R-squared  | ROC-AUC Score  |
|            | Accuracy (%)    |
| MAE (mol/L)| 0.84 (±0.01)    |
| RMSE (mol/L)| 78.86 (±1.78)  |
|            | F1 Score        |
|            | Micro           |
|            | 0.79 (±0.02)    |
|            | Macro           |
|            | 0.79 (±0.02)    |

Figure 6. GCN results: (a) regression and (b) classification prediction results using GCN. The results below are the results of the prediction using data01. Results tested on all three data sets are available in the Supporting Information.
were achieved using data03, as shown in Figure S8. This means that, as previously discussed for the regression task, including all three physicochemical properties (MP, MW, and volume) increased the prediction capability despite the reduced size of the database. For reference, the results of all three data sets are shown in the Supporting Information; the regression results are shown in Figure S7, and the classification results are shown in Figure S8. Additionally, Figure S9 shows the ROC curves drawn using all three data sets.

In practice, the training data set is small, while the test data set is large. Therefore, it is important to verify whether the current model is applicable over a wide chemical space by determining how the prediction accuracy is affected by reducing the size of the training set. More specifically, the model was previously trained with 80% of the total data; therefore, the size of the training set was drastically reduced to 20%, and the size of the test sets was increased to 80% to ensure that the model performed sufficiently on a smaller amount of data. The performance variation from changing the portion of the test set from 20% to 80% (data02) is shown in Figure S10. The results clearly validate that the current methodology mostly preserves its prediction accuracy when the portion of the training set is greatly reduced. Specifically, the $R^2$ value was only slightly reduced to 0.75 ($\pm 0.01$) (a reduction of 78%) when only 20% of data02 was used for training ($R^2$ was 0.8 when the training set was 80% of the data set). This indicates that the chemical space to be explored can be widened without losing the prediction performance. It has also been shown that adding new physicochemical properties, such as the ones given in this work, results in higher performance when predicting solubility and that it is worth obtaining new property values at the expense of losing some parts of the database. From Figure S11, it can be seen that the MFPCP model exhibited almost no variation with the training or test data sets. This means that this model is sufficiently stable to be used in various cases.

Performance of the GCN Architecture. Since data01 and data02 are the same, when data01 was used for training, the GCN model had an $R^2$ value of 0.80 ($\pm 0.03$), MAE of 0.61 ($\pm 0.03$) mol/L, and RMSE of 0.96 ($\pm 0.06$) mol/L for regression and an average accuracy of 78.86% ($\pm 1.78\%$) for classification (Figure 6). For classification, the prediction accuracy for the high class (high solubility) was the highest at 81.96% ($\pm 2.78\%$), while the prediction accuracies for the low and medium classes were slightly lower at 81.86% ($\pm 4.42\%$) and 73.57% ($\pm 3.77\%$), respectively. This demonstrates that the current model can be employed to predict any class. Looking at the performance differences between data sets, $R^2$ and the average accuracy for data03 were 0.81 ($\pm 0.04$) and 80.4% ($\pm 3.60\%$), respectively (Figures S8 and S9). Therefore, consistent prediction performance was observed for both regression and classification, regardless of the data set. In addition, only the amount of data in the data set is relevant for the GCN model because it does not require any PCP values.

Compared to the results of the MFPCP method, the $R^2$ value of regression (Figure S7) for the GCN model (0.80) was 0.06 higher than that for the MFPCP model (0.74) using data01 (using only fingerprints). In contrast, using data03 with less data, the $R^2$ value for the MFPCP model (0.85) was 0.04 higher than that for the GCN model (0.81). Similarly, the classification accuracy (Figure S8) for the GCN model (78.86%) was 4.49% higher than that for the MFPCP model (74.37%) using data01. In contrast, using data03, the classification accuracy for the MFPCP model (84.16%) was 3.76% higher than that for the GCN model (80.40%). This suggests that the GCN model using only SMILES is more advantageous if there are more solubility data, while the MFPCP model is more advantageous when there is less data, if MFPCP data are available. Even if the data set is small, the MFPCP model can be highly accurate if the appropriate features are obtained. Therefore, the two models built in this study can be the best model for solubility prediction in a given environment, provided that they are appropriately used according to the scale of the database and the number of features available.

To evaluate the stability of the model, additional validation was conducted by varying the ratio of the training to test set from 8:2 to 2:8, as shown in Figure S12. When data01 was divided at a ratio of 2:8 (training to test set), the $R^2$ value was 0.72 ($\pm 0.02$), and the average accuracy was 72.96% ($\pm 1.16\%$). Both the $R^2$ value and the accuracy were reduced relative to those obtained using the 8:2 ratio (reductions of 0.08 and 5.90%, respectively). This demonstrates the advantage of the GCN model, which exhibits great potential for predicting vast amounts of unknown data by training with a relatively small database. Furthermore, another validation was conducted by randomly splitting the database (8:2 training to test set ratio) with different random states, as shown in Figure S13. The $R^2$ value remained between 0.77 and 0.83, and the average accuracy was between 77.02% and 80.59%. This confirms the stability of the current GCN model.

Many solubility prediction models focus only on a single solvent type. In contrast, the concatenated model, which combines the graphs for the solute and solvent, can predict the solubility of a solute in a target solvent with a small portion of its solubility data set. Moreover, the GCN model has the advantage that feature generation is relatively intuitive and does not require additional computation. Because converting chemical compounds into graphs only requires atom and structural information (such as SMILES), the complex process of obtaining experimental or computational descriptors is not necessary. Therefore, if a large amount of solubility data is available, it can be used immediately without additional calculations. As a result, the constructed GCN model is capable of predicting the solubility of any combination of solutes and solvents without additional descriptors. This suggests that the multi-input GCN model has great potential in predicting not only solubility but also many important properties of two targets that can be expressed as graphs.

Additional Validation. For the data construction, the case of using water as the solvent accounts for 67.44% of the total (Figure S1b). Hence, it can be argued that the performance of predicting the aqueous solubility of the two models predominantly determines the overall result. To verify this, the results of regression were further confirmed by reconstructing the data01, data02, and data03 excluding the water solvent. The number of data without water was 5709 (data01, data02) and 3305 (data03), respectively. It is critical to note that the prediction accuracy from both methods reaches high reliability even without water. MFPCP method showed an $R^2$ value of 0.675 ($\pm 0.02$), 0.721 ($\pm 0.023$), and 0.840 ($\pm 0.02$) and GCN showed $R^2$ value of 0.70 ($\pm 0.05$) and 0.74 ($\pm 0.08$), respectively, for each data set. GCN showed stable performance regardless of the data set, while in the case of MFPCP P, the prediction performance reached highest when data03 was used since more features are used. In addition, in order to
verify that the different performance between data set configurations was not because of decreased number of water but a decrease in the number of data, the results were confirmed by matching each number of databases while including water. The MFPCP method showed R² values of 0.68 (±0.02), 0.787 (±0.02), and 0.842 (±0.02) and for the GCN method and 0.74 (±0.02) and 0.76 (±0.06), respectively, for each data set. Such results are similar to that excluding water, so it can be confirmed that the great performance from the current model is only because of water but still be applicable to various solute–solvent combinations.

In addition, most of the solubility prediction results were performed with solvents only for one specific solute; thus, direct performance comparison with the current study is difficult. Therefore, this study additionally tried to compare the model performance by modifying the multi-input process of solute and solvent into one input process for the solute. Boobier et al.21 reported the results of predicting solubility in water, so we compared the performance of a single-input process by modifying the architectures for the same data set. As a result, the prediction accuracy for the modified MFPCP model is R² = 0.86, RMSE = 0.82, and for the GCN model with the modified architecture is R² = 0.87, RMSE = 0.86. Compared to the previous results, it is slightly lower performance (Boobier: R² = 0.93, RMSE = 0.71). In addition to water, the prediction results of ethanol, benzene, and acetone are summarized in Table S4. It shows that, overall, the performance between models does not exhibit a significant difference. It is essential to note that a previous study implemented the features which must be generated first with additional calculations from the density functional calculations. Such an approach requires a significant calculation time and resources. Furthermore, they obtained the results based on each solvent, meaning that the performance of the multiple solute–solvent combinations is not examined. Again, unlike them, our model does not require additional calculations for both MFPCP and GCN model, and the solute–solvent variety is considered. To summarize, our model adopted the advantage of using a not-resource-demanding descriptor and the convenience of graph generation while maintaining the functionality of assessing multi-input.

Moreover, as a multi-input model, it is expected that the models of this study will be able to predict unknown intermolecular solubility by learning the combinations of many different compounds. To further verify this, an experiment was conducted with a small number of solvents and GCN model: for data01, the test set is composed of solvents having a number of 20 or more and 45 or less, respectively, and the corresponding solvent is not present in the training set. As a result of regression, 18 out of 29 test sets showed R² > 0.80 (as can be seen in Figure S14). This suggests that solubility between unknown compounds can be still predicted using one of our models, as opposed to approaches in previous studies20,25,26,30–32 where individual models were trained with sufficient data for solvents.

CONCLUSION

In this study, we proposed two data-driven methodologies for solubility prediction. Previous efforts have mainly focused on predicting solubility in a single solvent or a small number of designated solvents. However, because numerous fields (such as flow chemistry) require knowledge of the relationships between various combinations of solutes and solvents, we designed a model that is not affected by the number of unique solutes. The experiment was conducted with a total of 179 solvents, and two methods were designed: MFPCP and GCN.

The MFPCP method demonstrates that adding physicochemical properties to the features used for predicting solubility has several benefits, including high performance. Simply using the molecular fingerprints of the solute and solvent produced moderate performance, but the performance improved when specific physicochemical properties were added to the features. A definite advantage of this model is its low computational cost because the process of obtaining the features is computationally inexpensive. In addition, it can be utilized over a broad chemical space because training with a small data set enables solubility predictions for much larger unknown data sets.

Deep learning, particularly GCNs, has attracted considerable attention in the field of cheminformatics. The GCN model proposed in this work considers the atom features and structures of various combinations of solutes and solvents. Because the GCN only requires simple structural descriptors, adding more data sets is relatively straightforward, unlike conventional feature-based machine learning models, which often require additional quantum mechanical calculations to improve their prediction accuracy. Furthermore, the GCN model can be improved by tuning the composition of its architecture and parameters. The GCN model developed in this study is an effective approach for solubility prediction and can assist in new material and drug development processes.

The methodologies proposed in this paper only require seconds or minutes of calculation time, making them faster and more effective than computational quantum mechanical modeling method, and they can consider a variety of combinations. In addition, the models are robust in that they are almost unaffected by how the data set is separated and reliably show no difference in performance. In summary, machine learning can be an effective tool for predicting the solubility of various combinations of solutes and solvents.

ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acsomega.2c00697.

Detailed information for machine learning and graph convolution algorithm; data distribution and construction process; supplementary prediction results for various cases (PDF)

AUTHOR INFORMATION

Corresponding Authors

Mi-Jeong Kim — Polymer Research Lab, Samsung Advanced Institute of Technology, Siwon, Gyeonggi-do 16678, Republic of Korea; Email: mj09.kim@samsung.com

Kyoungmin Min — School of Mechanical Engineering, Soongsil University, Seoul 06978, Republic of Korea; orcid.org/0000-0002-1041-6005; Email: kmin.min@ssu.ac.kr

Authors

Sumin Lee — Department of Industrial and Information Systems Engineering, Soongsil University, Seoul 06978, Republic of Korea
ACKNOWLEDGMENTS
This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (No. 2020R1F1A1066519). This work was also supported by Samsung Advanced Institute of Technology.

REFERENCES

(1) Ruether, F.; Sadowski, G. Modeling the Solubility of Pharmaceuticals in Pure Solvents and Solvent Mixtures for Drug Process Design. J. Pharm. Sci. 2009, 98 (11), 4205–4215.
(2) Abraham, M. H.; Le, J. The Correlation and Prediction of the Solubility of Compounds in Water Using an Amended Solvation Energy Relationship. J. Pharm. Sci. 1999, 88 (9), 868–880.
(3) Morelock, M. M.; Choi, L. L.; Bell, G. L.; Wright, J. L. Estimation and Correlation of Drug Water Solubility with Pharmacological Parameters Required for Biological Activity. J. Pharm. Sci. 1994, 83 (7), 948–952.
(4) Chandrasekar, A.; Kim, C.; Venkatram, S.; Ramprasad, R. A Deep Learning Solvent-Selection Paradigm Powered by a Massive Solvent/Nonsolvent Database for Polymers. Macromolecules 2020, 53 (12), 4764–4769.
(5) Papaneophytou, C. P.; Grigoudis, A. I.; McInnes, C.; Kontopidis, G. Quantification of the Effects of Ionic Strength, Viscosity, and Hydrophobicity on Protein-Ligand Binding Affinity. ACS Med. Chem. Lett. 2014, 5 (8), 931–936.
(6) Dj, L.; Fish, P. V.; Mano, T. Bridging Solubility between Drug Discovery and Development. Drug Discovery Today 2012, 17 (9–10), 486–495.
(7) Saviani, K. T.; Gajjar, A. K.; Saviani, J. K. Drug Solubility: Importance and Enhancement Techniques. ISRN Pharm. 2012, 2012, 195727.
(8) Hansen, C. M. Hansen Solubility Parameters: A User’s Handbook, 2nd ed.; CRC Press: Boca Raton, FL, 2007.
(9) Li, L.; Totton, T.; Frenkel, D. Computational Methodology for Solubility Prediction: Application to the Sparingly Soluble Solutes. J. Chem. Phys. 2017, 146 (21), 214110.
(10) Peterson, D. L.; Yalkowsky, S. H. Comparison of Two Methods for Predicting Aqueous Solubility. J. Inf. Comput. Sci. 2001, 41 (6), 1531–1534.
(11) Ran, Y.; He, Y.; Yang, G.; Johnson, J. L. H.; Yalkowsky, S. H. Estimation of Aqueous Solubility of Organic Compounds by Using the General Solubility Equation. Chemosphere 2002, 48 (5), 487–509.
(12) Ran, Y.; Yalkowsky, S. H. Prediction of Drug Solubility by the General Solubility Equation (GSE). J. Chem. Inf. Comput. Sci. 2001, 41 (2), 354–357.
(13) Delaney, J. S. ESOL: Estimating Aqueous Solubility Directly from Molecular Structure. J. Chem. Inf. Comput. Sci. 2004, 44 (3), 1000–1005.
(14) 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 (7), 1563–1575.
(15) Klamt, A. Conductor-like Screening Model for Real Solvents: A New Approach to the Quantitative Calculation of Solution Phenomena. J. Phys. Chem. 1995, 99 (7), 2224–2235.
(16) Klamt, A.; Eckert, F.; Hornig, M.; Beck, M. E.; Bürger, T. Prediction of Aqueous Solubility of Drugs and Pesticides with COSMO-RS. J. Comput. Chem. 2002, 23 (2), 275–281.
(17) Duchowicz, P. R.; Castro, E. A. QSAR Studies on AqueosSolubilities of Drug-like Compounds. Int. J. Mol. Sci. 2009, 10 (6), 2558–2577.
(18) Yu, X.; Wang, X.; Wang, H.; Li, X.; Gao, J. Prediction of Solubility Parameters for Polymers by a QSAR Model. QSAR Comb. Sci. 2006, 25 (2), 156–161.
(19) Deng, T.; Jia, G.-Z. Prediction of Aqueous Solubility of Compounds Based on Neural Network. Mol. Phys. 2020, 118 (2), e1600754.
(20) Cui, Q.-B.; Su, S.; Ni, B.; Zeng, X.; Tan, Y.; Chen, Y. D.; Zhao, H. Improved Prediction of Aqueous Solubility of Novel Compounds by Going Deeper with Deep Learning. Front. Oncol. 2020, 10, 121.
(21) Boobier, S.; Hose, D. R. J.; Blacker, A. J.; Nguyen, B. N. Machine Learning with Physicochemical Relationships: Solubility Prediction in Organic Solvents and Water. Nat. Commun. 2020, 11 (1), 5753.
(22) Chen, D.; Wei, G.; Pan, F. Extracting Predictive Representations from Hundreds of Millions of Molecules. Research Square 2021, 12, 10793.
(23) Wee, J.; Xia, K. Forman Persistent Ricci Curvature (FPRC)-Based Machine Learning Models for Protein-Ligand Binding Affinity Prediction. Brief. Bioinform. 2021, DOI: 10.1093/bib/bbab136.
(24) Xie, L.; Xu, L.; Kong, R.; Chang, S.; Xu, X. Improvement of Prediction Performance with Conjoint Molecular Fingerprint in Deep Learning. Front. Pharmacol. 2020, 11, 606668.
(25) Perryman, A. L.; Inoyama, D.; Patel, J. S.; Ekins, S.; Freundlich, J. S. Pruned Machine Learning Models to Predict Aqueous Solubility. ACS Omega 2020, 5 (27), 16562–16567.
(26) Lovrič, M.; Pavlovič, K.; Žuvela, P.; Spataru, A.; Lučić, B.; Kern, R.; Wong, M. W. Machine Learning in Prediction of Intrinsic Aqueous Solubility of Drug-like Molecules: Generalization, Complexity, or Predictive Ability? J. Chemom. 2021, 35 (7–8), e3349 DOI: 10.1021/cem.2c00003.
(27) Jiang, M.; Li, Z.; Zhang, S.; Wang, S.; Wang, X.; Yuan, Q.; Wei, Z. Drug—Target Affinity Prediction Using Graph Neural Network and Contact Maps. RSC Adv. 2020, 10 (35), 20701–20712.
(28) Kipf, T. N.; Welling, M. Semi-Supervised Classification with Graph Convolutional Networks. arXiv [cs.LG], 2016.
(29) Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; Bengio, Y. Graph Attention Networks. arXiv [stat.ML], 2017.
(30) Tang, B.; Kramer, S. T.; Fang, M.; Qiu, Y.; Wu, Z.; Xu, D. A Self-Attention Based Message Passing Neural Network for Predicting Molecular Lipophilicity and Aqueous Solubility. J. Cheminform. 2020, 12 (1), 15.
(31) Wieder, O.; Kuenemann, M.; Wieder, M.; Seidel, T.; Meyer, C.; Bryant, S. D.; Langer, T. Improved Lipophilicity and Aqueous Solubility Prediction with Composite Graph Neural Networks. Molecules 2021, 26 (20), 6185.
(32) Gao, P.; Zhang, J.; Sun, Y.; Yu, J. Accurate Predictions of Aqueous Solubility of Drug Molecules via the Multilevel Graph Convolutional Network (MGCN) and SchNet Architectures. Phys. Chem. Chem. Phys. 2020, 22 (41), 23766–23772.
(33) Sorkun, M. C.; Khetan, A.; Er, S. AqSolDB, a Curated Reference Set of Aqueous Solubility and 2D Descriptors for a Diverse Set of Compounds. *Sci. Data* 2019, 6 (1), 143.

(34) Cao, Y.; Shen, Y. Energy-Based Graph Convolutional Networks for Scoring Protein Docking Models. *Proteins* 2020, 88 (8), 1091–1099.

(35) Huuskonen, J. Estimation of Aqueous Solubility for a Diverse Set of Organic Compounds Based on Molecular Topology. *J. Chem. Inf. Comput. Sci.* 2000, 40 (3), 773–777.

(36) Solubility data https://www.gaylordchemical.com/products/product-grades/dmso/solubility-data/ (accessed 2022-03-16).

(37) Bradley, J.-C.; Guha, R.; Hooker, B.; Koch, S. J.; Lang, A.; Neylon, C.; Williams, A. J. Open Notebook Science Challenge Solubility Dataset, 2015.

(38) Landrum, G. RDKit https://www.rdkit.org (accessed 2022-03-6).

(39) Fey, M.; Lenssen, J. E. Fast Graph Representation Learning with PyTorch Geometric. *arXiv [cs.LG]*, 2019.

(40) Kim, S.; Chen, J.; Cheng, T.; Gindulyte, A.; He, J.; He, S.; Li, Q.; Shoemaker, B. A.; Tiessen, P. A.; Yu, B.; Zaslavsky, L.; Zhang, J.; Bolton, E. E. PubChem in 2021: New Data Content and Improved Web Interfaces. *Nucleic Acids Res.* 2021, 49 (D1), D1388–D1395.

(41) Durant, J. L.; Leland, B. A.; Henry, D. R.; Nourse, J. G. Reoptimization of MDL Keys for Use in Drug Discovery. *J. Chem. Inf. Comput. Sci.* 2002, 42 (6), 1273–1280, DOI: 10.1021/ci010132r.

(42) Pence, H. E.; Williams, A. ChemSpider: An Online Chemical Information Resource. *J. Chem. Educ.* 2010, 87 (11), 1123–1124.

(43) Nisius, B.; Bajorath, J. Molecular Fingerprint Recombination: Generating Hybrid Fingerprints for Similarity Searching from Different Fingerprint Types. *ChemMedChem.* 2009, 4 (11), 1859–1863.

(44) Cereto-Massagué, A.; Ojeda, M. J.; Valls, C.; Mulero, M.; Garcia-Vallvé, S.; Pujadas, G. Molecular Fingerprint Similarity Search in Virtual Screening. *Methods* 2015, 71, 58–63.

(45) Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.-Y.; Research, M.; University, P.; Redmond, M. LightGBM: A highly efficient gradient Boosting Decision Tree https://proceedings.neurips.cc/paper/2017/file/644f44a102fe848669bd9eb6b76fa-Paper.pdf (accessed 2022-03-16).

(46) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Müller, A.; Not hamstring, J.; Louppe, G.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; Duchesnay, E. Scikit-Learn: Machine Learning in Python. *arXiv [cs.LG]*, 2012.

(47) Lim, J.; Ryu, S.; Park, K.; Choe, Y. J.; Ham, J.; Kim, W. Y. Predicting Drug-Target Interaction Using a Novel Graph Neural Network with 3D Structure-Embedded Graph Representation. *J. Chem. Inf. Model.* 2019, 59 (9), 3981–3988.

(48) Zhou, Y.; Zheng, H.; Huang, X.; Hao, S.; Li, D.; Zhao, J. Graph Neural Networks: Taxonomy, Advances, and Trends. *ACM Trans. Intell. Syst. Technol.* 2022, 13 (1), 1–54.

(49) Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; Yu, P. S. A Comprehensive Survey on Graph Neural Networks. *IEEE Trans. Neural Netw. Learn. Syst.* 2021, 32 (1), 4–24.

(50) Kearnes, S.; McCloskey, K.; Berndl, M.; Pande, V.; Riley, P. Molecular Graph Convolutions: Moving beyond Fingerprints. *J. Comput. Aided Mol. Des.* 2016, 30 (8), 595–608.

(51) Liu, K.; Sun, X.; Jia, L.; Ma, J.; Xing, H.; Wu, J.; Gao, H.; Sun, Y.; Boulnois, F.; Fan, J. Chemi-Net: A Molecular Graph Convolutional Network for Accurate Drug Property Prediction. *Int. J. Mol. Sci.* 2019, 20 (14), 3389.

(52) Fout, A.; Byrd, J.; Shariat, B.; Ben-Hur, A. Protein interface prediction using graph convolutional networks https://proceedings.neurips.cc/paper/2017/file/5077783927f2ec27f37ba40fbd17f6e-Paper.pdf (accessed 2022-03-16).