

Article

Graph Neural Networks with Multiple Feature Extraction Paths for Chemical Property Estimation

Sho Ishida, Tomo Miyazaki * , Yoshihiro Sugaya  and Shinichiro Omachi 

Graduate School of Engineering, Tohoku University, Sendai 9808579, Japan; sho0849@iic.ecei.tohoku.ac.jp (S.I.); sugaya@iic.ecei.tohoku.ac.jp (Y.S.); machi@ecei.tohoku.ac.jp (S.O.)

* Correspondence: tomo@tohoku.ac.jp

Abstract: Feature extraction is essential for chemical property estimation of molecules using machine learning. Recently, graph neural networks have attracted attention for feature extraction from molecules. However, existing methods focus only on specific structural information, such as node relationship. In this paper, we propose a novel graph convolutional neural network that performs feature extraction with simultaneously considering multiple structures. Specifically, we propose feature extraction paths specialized in node, edge, and three-dimensional structures. Moreover, we propose an attention mechanism to aggregate the features extracted by the paths. The attention aggregation enables us to select useful features dynamically. The experimental results showed that the proposed method outperformed previous methods.

Keywords: chemical property estimation; graph neural networks; molecular data; multiple feature extraction



Citation: Ishida, S.; Miyazaki, T.; Sugaya, Y.; Omachi, S. Graph Neural Networks with Multiple Feature Extraction Paths for Chemical Property Estimation. *Molecules* **2021**, *26*, 3125. <https://doi.org/10.3390/molecules26113125>

Academic Editor: James Gauld

Received: 5 March 2021

Accepted: 21 May 2021

Published: 24 May 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Each molecule has its unique chemical properties. Estimation of the chemical properties is the first step in the field of drug discovery. Reagent testing is a standard estimation method. However, its process requires long time and equipment cost. Machine learning methods have been widely studied to reduce time and cost.

Most machine learning methods transform molecules into feature vectors and estimate chemical properties using a neural network. There is a high correlation between molecular structure and chemical properties. For example, molecules with benzene rings have a sweet aroma and flammability, and hydroxy groups (OH groups) are readily soluble in water. Therefore, feature extraction of molecular structures is essential in the estimation of chemical properties using machine learning. Since chemical properties depend on an essential structure, a flexible feature extraction method is necessary. A general feature extraction method is Molecular Fingerprints [1–4], which transform a molecular structure into a one-hot vector of the presence or absence of specific structures designed by humans. However, the specific structures are hard for modification according to chemical properties since experts need to change the specific structure of Molecular Fingerprints.

Recently, feature extraction using graph convolutional neural networks [5] has been attracting attention as a learnable feature extraction method. As shown in Figure 1, the graph represents the molecule using nodes (atoms) and edges (bonds). Node features are extracted by updating their features and neighboring node features. The node feature propagates to further nodes by the number of update processes. Besides, the update is based on a neural network. Thus, a feature extraction model can learn the essential substructures in a molecule according to the chemical characteristics of the estimation target. Various models using graph convolutional neural networks have been developed [6,7]. The weave model [6] extracts edge features to consider relationships between nodes. The 3DGCN model used relative coordinates between nodes to extract features of three-dimensional structures [7].

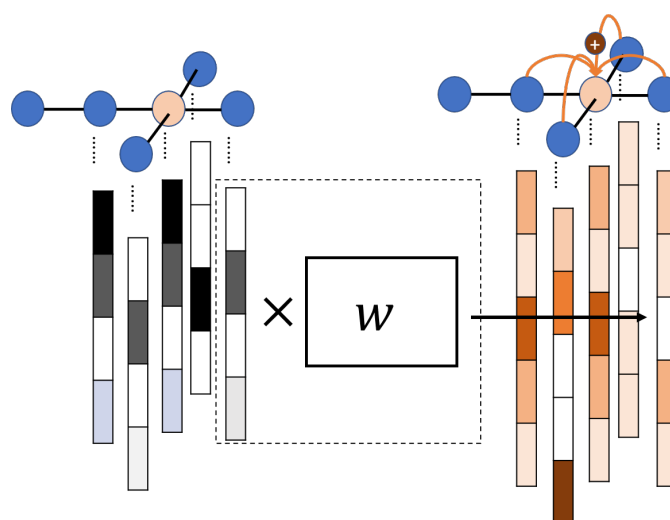


Figure 1. Graph convolutional neural network. Node features are updated with weights w (node features after arrows in the figure). Then, the node of interest (orange) is updated with the features of its neighbors.

Graph convolutional neural networks worked well in a classification problem, such as active or inactive. However, there is room for improvement in the regression problem due to its extensive estimation range. Furthermore, substructures can be different for target properties. Thus, it is essential to consider multiple structural features simultaneously.

In this paper, we propose a method for chemical property estimation of molecules using multiple structural features. Specifically, we integrate feature extraction paths that consider nodes, edges, and three-dimensional structures, respectively. For more flexible feature extraction, we utilize an attention mechanism to select useful features dynamically.

2. Related Work

In the estimation of chemical properties by machine learning, the estimator uses feature vectors extracted from molecules. Molecular Fingerprint is a method for extracting feature vectors from molecules [1–3]. This method uses a one-hot vector to represent the presence or absence of human-designed molecular structures. An improved method is Extended Connectivity Fingerprints, or ECFP [4]. ECFP extracts the presence or absence of subgraphs within molecular radius as a feature vector. However, these methods only consider pre-designed molecular structures and, thus, cannot extract features according to the chemical properties of the target.

A flexible feature extraction method has been developed using machine learning. Duvenaud et al. used neural networks to refine the features of ECFP [8]. Recently, the graph convolutional neural network [5] has attracted much attention. Graph convolutional neural networks sequentially update node features using the features of their neighborhood nodes. Finally, all the node features are merged into a one-dimensional feature vector, resulting in a feature vector of a molecule. In addition to estimating molecular properties, graph convolutional neural networks are used in a wide range of fields, including language processing [9–11], human motion estimation [12–14], graph similarity estimation [15,16], and class identification [17,18].

For the estimation of chemical properties, various models exist [19–24]. Directed graphs are used to reduce computation and update node features [22,23]. Edge features are extracted in Reference [6,25]. Relative coordinates between nodes are used to extract features of three-dimensional structures [7]. There are methods that learn the importance of node features [26,27]. However, the aforementioned methods specialized in a specific molecular structure, such as edge and three-dimensional structures. In this study, we propose to integrate three feature extraction methods [5–7] to simultaneously extract

multiple molecular structures. Furthermore, we dynamically select features using an attention mechanism to improve the estimation performance.

3. Materials and Methods

We propose a graph convolutional neural network that integrates three different approaches of feature extraction. Depending on the chemical properties of the estimation target, we need to extract different features. Therefore, we simultaneously extract node features, edge features, and three-dimensional features. Furthermore, we use attention to calculate the importance of each feature dynamically. As shown in Figure 2, the proposed method extracts features using multiple paths (node features, edge features, and three-dimensional features) and aggregates each feature. Firstly, we extract features through each path. Then, we form a molecular feature by aggregating the features. The proposed method enables us to consider various structures of the molecule by extracting features through multiple paths.

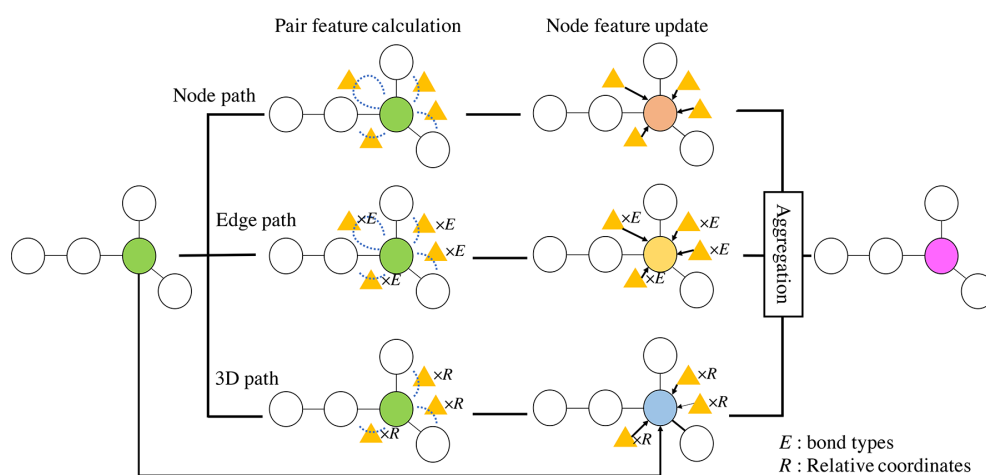


Figure 2. Overview of the proposed method. For simplicity, we illustrated the flow of feature update for a single attention node (green circle). Firstly, we generate pair features (yellow triangles) representing the relationship between the node and its neighbors. We use bond types and relative coordinates to extract edge relationships and three-dimensional structures. Then, we update the node features using the pair features (orange, yellow, and blue circles). Finally, we aggregate each path's features of the attention node to obtain the node features, which are the output of this layer (purple circles). We repeat the above processes for feature updates.

3.1. Node Feature Extraction Path

This path extracts node features using relationships between nodes. Let $H_i^t \in \mathbb{R}^{M \times 1}$ represents a M -dimensional feature vector of node i at t -th update round. We produce the pair feature $P_{ij} \in \mathbb{R}^{M \times 1}$ between node i and j by Equation (1), where σ represents the rectified linear unit, and \parallel is the concatenation operation. A weight $W_{np} \in \mathbb{R}^{M \times 2M}$ and a bias $B_{np} \in \mathbb{R}^{M \times 1}$ are learning parameters. Subsequently, we update the node features H_i as Equation (2). $N(i)$ is the set of neighboring nodes of node i . The weight is $W_n \in \mathbb{R}^{M \times M}$, and the bias is $B_n \in \mathbb{R}^{M \times 1}$.

$$P_{ij} = \sigma(W_{np}(H_i^t \parallel H_j^t) + B_{np}), \quad (1)$$

$$H_i^{t+1} = \sigma\left(\sum_{j \in N(i)} W_n P_{ij} + B_n\right). \quad (2)$$

3.2. Edge Feature Extraction Path

The extraction path of edge features takes into account the edge relationships between nodes. The atoms in a molecule can have various bonds, such as single bonds and double

bonds. We incorporate these bond types into the feature extraction to consider the molecular structure's connectivity.

We use the five bonds: single bonds, double bonds, triple bonds, aromatic bonds, and bonds to themselves. Let N represent the number of atoms. We represent the bonds using the edge parameter $E \in \mathbb{R}^{N \times N}$ to describe the connectivity types. A naive parameter for E is using categorical values, such as 1 for the single bond. Inspired by Reference [25], we learn parameters to represent the bonds rather than merely representing the bonds using five categorical values. As shown in Figure 3, we create five adjacency matrices and learn the edge parameters using convolutional filter. The convolution filter has a kernel size of 1, and the number of channels is 5.

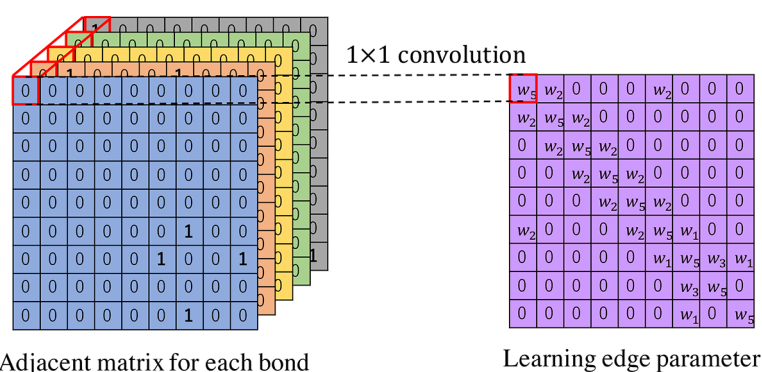


Figure 3. Edge parameters.

We obtain a pair feature P_{ij} by Equation (3) using E_{ij} , the (i, j) th element of E . Note that E_{ij} is a scalar value. Then, we update the feature H_i by Equation (4). The learning parameters are weight $W_{ep} \in \mathbb{R}^{M \times 2M}$, $W_e \in \mathbb{R}^{M \times M}$, bias $B_{ep}, B_e \in \mathbb{R}^{M \times 1}$. We take the molecular bonds into account by multiplying E and the paired features.

$$P_{ij} = \sigma(E_{ij}W_{ep}(H_i^t \parallel H_j^t) + B_{ep}), \quad (3)$$

$$H_i^{t+1} = \sigma\left(\sum_{j \in N(i)} W_e P_{ij} + B_e\right). \quad (4)$$

3.3. Three-Dimensional Feature Extraction Path

We incorporate three-dimensional structural information into feature updates based on Reference [7]. Let (x_i, y_i, z_i) represent the absolute coordinate of node i , we calculate the relative coordinate $R(x)_{ij} = x_i - x_j$ of the x -coordinate. Likewise, we obtain relative coordinate in y and z , $R(y)_{ij} = y_i - y_j$ and $R(z)_{ij} = z_i - z_j$.

We calculated the pair feature P_{ij} using the relative coordinates R as defined in Equation (5). Then, we obtain the intermediate feature Q_i by accumulating the pair features as in Equation (6). However, Q_i exclude node feature H_i due to $R_{ii} = 0$. Therefore, as shown in Equation (7), we propose to concatenate H_i and Q_i .

$$P_{ij} = \sigma\left(\sum_{k \in (x,y,z)} R(k)_{ij}W_{tp}(H_i^t \parallel H_j^t) + B_{tp}\right), \quad (5)$$

$$Q_i = \sigma\left(\sum_{j \in N(i)} W_{tq}P_{ij} + B_{tq}\right), \quad (6)$$

$$H_i^{t+1} = \sigma(W_t(H_i^t \parallel Q_i)). \quad (7)$$

There is a drawback in relative coordinates. The difference of relative coordinates is affected by translation and rotation. For further improvements, it is promising to use distance between atoms.

3.4. Feature Aggregation

We propose to extract more useful features by merging the features extracted through the paths. We integrate the three features using attention to dynamically select important features for each node. We integrate the features as Equation (8), where H^{node} , H^{edge} , and H^{3d} represent features extracted by the paths. Where α_i^{node} represents an attention for H_i^{node} at node i , which is defined in Equation (9). We used the softmax function to obtain α . Inspired by Reference [26], we calculate e_i^{node} , e_i^{edge} , e_i^{3d} for each feature by Equation (10). We use the initial feature H_i^{init} of the node i and H^p , $p \in \{\text{node, edge, 3d}\}$.

$$H_i = \sigma(W_{\text{agg}} \sum_{p \in \{\text{node, edge, 3d}\}} \alpha_i^p H_i^p), \quad (8)$$

$$\alpha_i^p = \text{softmax}(e_i^p) = \frac{\exp(e_i^p)}{\sum_{k \in \{\text{node, edge, 3d}\}} \exp(e_i^k)}, \quad (9)$$

$$e_i^k = W_{\text{att}}(\sigma(H_i^{\text{init}} W_{\text{init}}) \parallel \sigma(H_i^k W_k)). \quad (10)$$

3.5. Details of the Proposed Model

We illustrated the structure of the proposed model in Figure 4. The proposed model extracts features using the paths and aggregation, which are composed of graph convolutional neural networks. Then, we sum up the features along to each dimension to produce a molecular feature vector. Finally, we estimate chemical properties by applying a fully connected layer.

We adopted two-stage training. Specifically, we independently trained each path. Then, we fixed the paths and trained the aggregation layer and the fully connected layer. We used the mean square error (MSE) loss for training. We followed Reference [7] to determine initial features, resulting in 60 dimensions feature vectors. The batch size was set to 16.

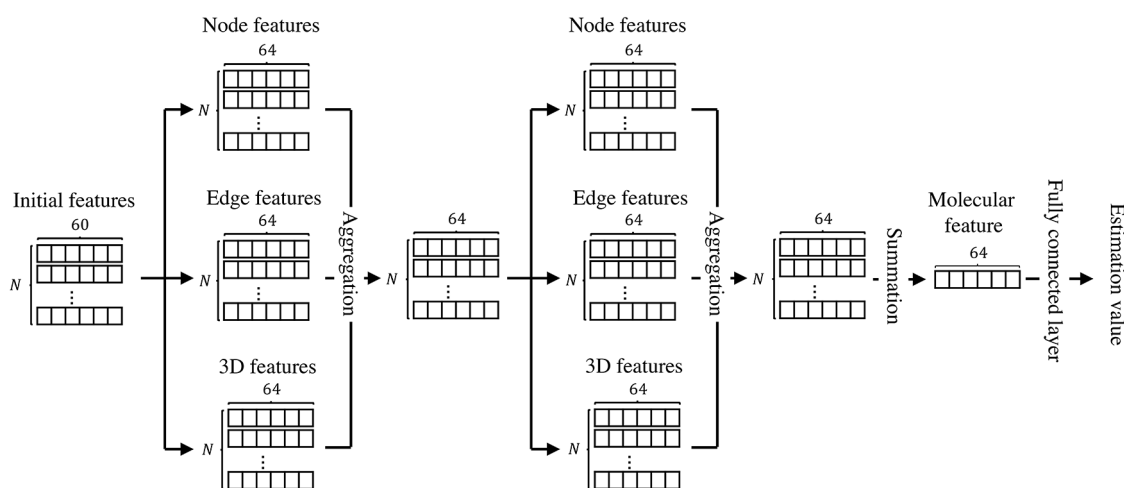


Figure 4. The structure of the model. The initial features are the 60-dimensional features.

3.6. Datasets and Metrics

We mainly used two datasets in the experiments: Freesolv and ESOL. Each of these datasets has been compiled in Reference [28] and is widely used as a dataset to evaluate methods for estimating chemical properties. Freesolv is a dataset for estimating the free energy of hydration of molecules and contains 1128 molecules. ESOL is a dataset for estimating solubility and contains 643 molecules. Overall, Freesolv and ESOL are regression tasks, which directly predict the values. Besides, we used four datasets for verification of the proposed method. We summarized the datasets in Table 1. QM8 has four excited state properties calculated by three different methods. Thus, 12 properties in total.

Table 1. Summary of the dataset used in the experiments.

Dataset	#Mols	Category	Task
Freesolv	1128	Physical chemistry	Regression for water solubility
ESOL	643	Physical chemistry	Regression for hydration free energy
QM7	7160	Quantum mechanism	Regression for Atomization energy
QM8	21,786	Quantum mechanism	Regression for excited state properties
BACE	1513	Biophysics	Classification for inhibitors of β -secretase 1
BBBP	2039	Physiology	Classification for blood-brain barrier penetration

We randomly split the dataset to 8:1:1 for training data, validation data, and test data. We evaluated the proposed method and the comparison methods for 10 trials. We calculated the average of the metrics over the trials. The evaluation metrics is Mean Absolute Error (MAE). The smaller MAE is better.

4. Results

4.1. Comparison Methods

As comparison methods, we used the graph convolutional neural network (GCN) [5], the Weave model [6], and the 3DGCN [7]. Broadly, the comparison methods extract node features (GCN), edge features (Weave), and three-dimensional features (3DGCN), respectively. We set the number of updating layers to two in the proposed method and the comparison methods for equivalent comparison. In addition, the summation is used for producing molecular features as same as the proposed method. The main difference between the comparison methods and the proposed method is the number of feature extraction paths. The comparison methods have a single path. In contrast, the proposed method has multiple paths to consider node, edge, and three-dimensional structure simultaneously.

4.2. Main Results

We trained models until they converged. We stopped training if the loss is no longer improving for ten successive epochs. We defined no improvement if improvements are less than 0.0001. Figure 5 shows typical loss curves of the proposed method in training and validation. The loss curves of the validation also converged. Thus, there was no overfitting. The models successfully converged. The proposed model has 143,286 parameters. Compared to 135 M and 11.4 M parameters in VGG-16 and ResNet-18 models, the number of parameters is significantly small. Therefore, the numbers of data points in Freesolv and ESOL are satisfactory for the proposed method.

We showed the numerical results in Table 2. GCN was better among the comparison method. Moreover, the proposed method outperformed the comparison methods on all datasets. The proposed method successfully learned the essential features. Thus, the results showed the effectiveness of the multiple feature extraction for chemical property regression.

Table 2. Averages of MAE over 10 trials (Bold and underline are the best and the second-best, respectively).

	GCN	Weave	3DGCN	Proposed
Freesolv	0.764	0.817	<u>0.743</u>	0.717
ESOL	<u>0.503</u>	0.665	0.531	0.498

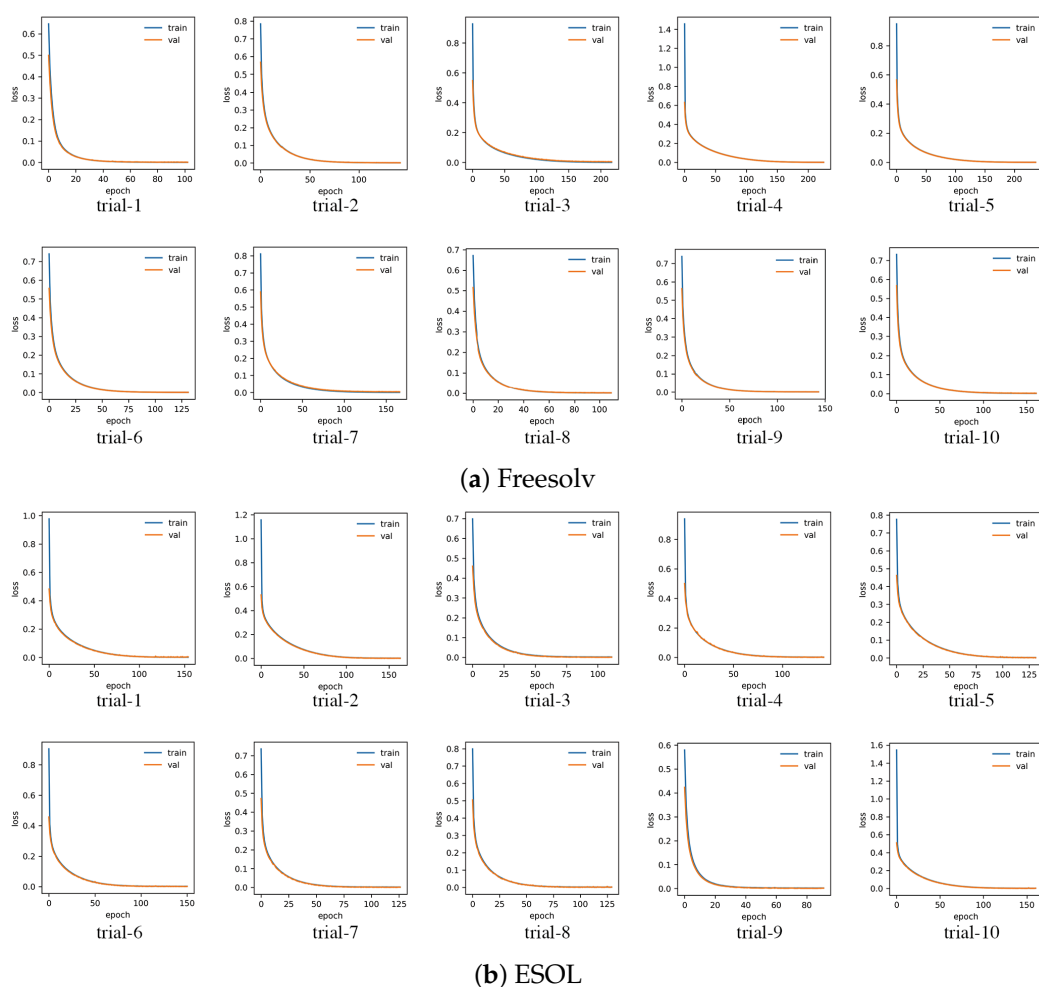


Figure 5. Loss curves of the proposed model.

4.3. Results on Quantum Mechanism

We conducted experiments using QM7 and QM8 datasets. We trained models until they converged. The results are shown in Table 3. The results show that the proposed method outperformed the comparison methods at ten tasks. In addition, the proposed method was the second-best on the rest tasks. Thus, we verified the effectiveness of the proposed method in various tasks.

Table 3. Averages of MAE over 10 trials on quantum mechanism datasets (Bold and underline are the best and the second-best, respectively).

	GCN	Weave	3DGCN	Proposed
QM7	<u>10.75</u>	12.22	12.89	9.13
QM8 (E1-CC2)	0.00846	0.00608	0.00651	<u>0.00611</u>
QM8 (E2-CC2)	0.0099	<u>0.0080</u>	0.0081	0.0077
QM8 (f1-CC2)	0.0180	0.0161	<u>0.0148</u>	0.0141
QM8 (f2-CC2)	0.0346	0.0327	<u>0.0324</u>	0.0303
QM8 (E1-PBE0)	0.0082	0.0064	0.0071	<u>0.0066</u>
QM8 (E2-PBE0)	0.00945	0.00705	0.00822	<u>0.00712</u>
QM8 (f1-PBE0)	0.0154	<u>0.0120</u>	0.0124	0.0114
QM8 (f2-PBE0)	0.0291	<u>0.0259</u>	0.0261	0.0247
QM8 (E1-CAM)	0.0074	<u>0.0061</u>	0.0065	0.0058
QM8 (E2-CAM)	0.0084	<u>0.0065</u>	0.0071	0.0063
QM8 (f1-CAM)	0.0166	0.0132	<u>0.0127</u>	0.0123
QM8 (f2-CAM)	0.0308	0.0268	<u>0.0275</u>	0.0259

4.4. Evaluation on Aggregation Approaches

We carried out experiments to discuss the effectiveness of the feature aggregation. Besides the attention, there can be various aggregation approaches, such as concatenation, summing, and maximum. We define them in Equations (11)–(13).

$$H_i = \sigma(W_{\text{concat}}(H_i^{\text{node}} \parallel H_i^{\text{edge}} \parallel H_i^{\text{3d}})), \quad (11)$$

$$H_i = \sigma(W_{\text{sum}}(H_i^{\text{node}} + H_i^{\text{edge}} + H_i^{\text{3d}})), \quad (12)$$

$$H_i = \sigma(W_{\text{max}} \max(H_i^{\text{node}}, H_i^{\text{edge}}, H_i^{\text{3d}})). \quad (13)$$

Table 4 shows the results. The attention and the concatenation aggregations were the best on Freesolv and ESOL, respectively. All the aggregations achieved accurate estimation results. Thus, the proposed method has capability to aggregate different features using various approaches.

Table 4. Averages of MAE for aggregation approaches.

	Concat	Sum	Max	Attention
Freesolv	0.666	0.663	0.703	0.639
ESOL	0.472	0.478	0.488	0.484

4.5. Impacts of Feature Extraction Paths

We conducted experiments to clarify the impacts of feature extraction paths on the datasets. We built models with one-, and two-paths. Specifically, one-path models have a single path of node, edge, and three-dimensional features, respectively. The two-path models have node and edge paths, node and three-dimensional paths, and edge and three-dimensional paths, respectively. We used the attention aggregation in the two-path models.

The results are shown in Table 5. In Freesolv, the two-path models outperformed the one-path models. Furthermore, the proposed model was the best at 0.639. Likewise, two-path models were superior to the one-path models on ESOL. Overall, multiple features were significant in chemical property estimation.

Table 5. Average of MAE for path combinations.

	Path Combinations						
Node	✓			✓	✓		✓
Edge		✓		✓		✓	✓
3D			✓		✓	✓	✓
Freesolv	0.710	0.702	0.864	0.640	0.676	0.685	0.639
ESOL	0.483	0.498	0.538	0.477	0.476	0.482	0.484

4.6. Results in Classification Tasks

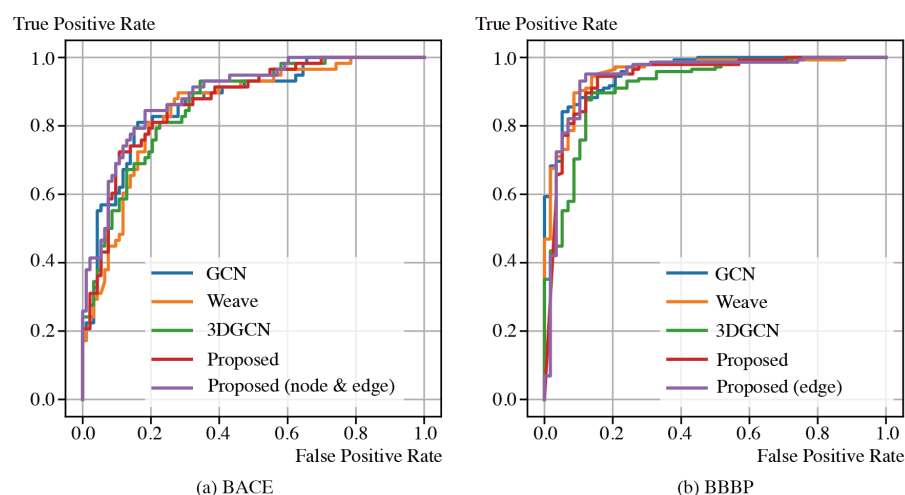
We conducted classification experiments on BACE and BBBP. In addition, we trained models until they converged. We used four metrics: Accuracy, Recall, Precision, and F-score. Tables 6 and 7 show the results. The two path model of the proposed method achieved the best results on BACE at all metrics. In addition, the proposed method was the best at precision and the second-best at the other metrics on BBBP. The ROC curves in Figure 6 shows the significant performance of the proposed method. These results show that the effectiveness of the proposed method in a classification task.

Table 6. Classification results on BACE (Bold and underline are the best and the second-best, respectively).

	GCN	Weave	3DGCN	Proposed	Proposed (Node & Edge)
Accuracy	<u>0.807</u>	0.751	0.774	0.799	0.811
Recall	<u>0.779</u>	0.714	0.726	0.782	0.782
Precision	<u>0.777</u>	0.739	0.749	0.763	0.783
F-score	<u>0.778</u>	0.726	0.737	0.773	0.782

Table 7. Classification results on BBBP (Bold and underline are the best and the second-best, respectively).

	GCN	Weave	3DGCN	Proposed	Proposed (Edge)
Accuracy	0.886	0.871	0.873	0.874	<u>0.884</u>
Recall	0.942	0.915	0.923	0.922	<u>0.928</u>
Precision	0.912	<u>0.917</u>	0.912	0.915	0.922
F-score	0.927	0.916	0.918	0.918	<u>0.925</u>

**Figure 6.** ROC curves.

4.7. Verification of Edge Parameters

We carried out experiments to verify the effect of edge parameter E , a learning parameter in Equation (3). We compared two modes on the basis of the edge path model. One used edge parameters learned by convolution with kernel sizes. The other adopted the fixed edge parameter, e.g., categorical values, one for the self node, 2 for single bonds, 3 for double bonds, 4 for triple bonds, 5 for aromatic bonds. We adopted 50 epochs for training. The results are shown in Table 8. The model using the learned edge parameters improved on all the datasets. Therefore, the proposed method learned an optimal representation of edge types.

Table 8. Averages of MAE using fixed and learned edge parameters.

	Fixed	Conv-1	Conv-3	Conv-5	Conv-7
Freesolv	1.181	0.799	0.872	0.957	0.927
ESOL	0.706	0.525	0.598	0.585	0.654

4.8. Effect of the Self Node in Three-Dimensional Features

We evaluated the effect of the self node in the three-dimensional feature. We used the model of the extraction path of three-dimensional features. Then, we compared the model

with and without the self node. Specifically, we defined the model without self node as Equation (14) instead of Equation (7).

$$H_i^{3d} = \sigma(W_{tq}Q_i). \quad (14)$$

If we omit the self nodes, the self nodes were not considered when aggregating the pair features and updating the node features. The experimental results are shown in Table 9. There were specific improvements by the self node. Therefore, we confirmed that the performance could improve by incorporating the self nodes into three-dimensional features.

Table 9. Average of MAE with and without self node features.

	w/o Self	w/ Self
Freesolv	1.525	0.864
ESOL	0.726	0.538

4.9. Attention Visualization

We conducted experiments to confirm the capability of dynamic determination for the attention values in the proposed method by visualizing each node's attentions α_i . According to Equation (9), the summation of α_i among the paths is normalized to one. Thus, we directly illustrated α using bar charts. The visualization results are shown in Figure 7. The various attention values were assigned to each node. The result shows that the proposed method flexibly determined attention for each node.

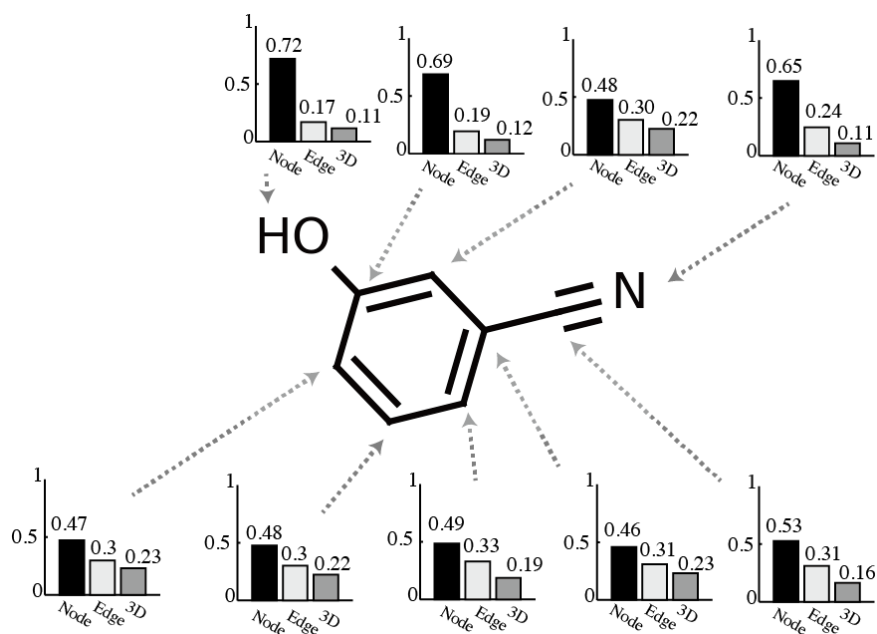


Figure 7. Visualization results of attention values.

5. Conclusions

In this study, we proposed a method for chemical property estimation in molecules. The proposed method uses multiple paths to extract features focusing on specific structures, such as node relationship, edge relationship, and three-dimensional structure in a molecule. Furthermore, we proposed to obtain more useful features by aggregating multiple features by selecting essential features dynamically. Compared to existing methods that focus on only one structure, the experimental results showed that the proposed method outperformed the comparison methods in regression tasks. Therefore, multiple feature extraction can improve the performance of chemical property estimation in molecules.

Author Contributions: Conceptualization, S.I. and T.M.; methodology, S.I.; software, S.I.; validation, S.I. and T.M.; formal analysis, S.I.; investigation, S.I. and T.M.; resources, S.I.; data curation, S.I.; writing—original draft preparation, S.I. and T.M.; writing—review and editing, Y.S. and S.O.; visualization, S.I.; supervision, Y.S. and S.O.; project administration, S.O.; funding acquisition, T.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by JSPS KAKENHI Grant Numbers 19K11848.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets are available at <http://moleculenet.ai/datasets-1> (accessed on 14 May 2021).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Carhart, R.E.; Smith, D.H.; Venkataraghavan, R. Atom pairs as molecular features in structure-activity studies: definition and applications. *J. Chem. Inf. Comput. Sci.* **1985**, *25*, 64–73. [[CrossRef](#)]
2. Nilakantan, R.; Bauman, N.; Dixon, J.S.; Venkataraghavan, R. Topological torsion: A new molecular descriptor for SAR applications. Comparison with other descriptors. *J. Chem. Inf. Comput. Sci.* **1987**, *27*, 82–85. [[CrossRef](#)]
3. Gedeck, P.; Rohde, B.; Bartels, C. QSAR - How Good Is It in Practice? Comparison of Descriptor Sets on an Unbiased Cross Section of Corporate Data Sets. *J. Chem. Inf. Model.* **2006**, *46*, 1924–1936. [[CrossRef](#)] [[PubMed](#)]
4. Rogers, D.; Hahn, M. Extended-Connectivity Fingerprints. *J. Chem. Inf. Model.* **2010**, *50*, 742–754. [[CrossRef](#)] [[PubMed](#)]
5. Kipf, T.N.; Welling, M. Semi-Supervised Classification with Graph Convolutional Networks. In Proceedings of the 5th International Conference on Learning Representations, Toulon, France, 24–26 April 2017.
6. 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. [[CrossRef](#)] [[PubMed](#)]
7. Cho, H.; Choi, I.S. Enhanced Deep-Learning Prediction of Molecular Properties via Augmentation of Bond Topology. *ChemMedChem* **2019**, *14*, 1604–1609. [[CrossRef](#)] [[PubMed](#)]
8. Duvenaud, D.K.; Maclaurin, D.; Iparraguirre, J.; Bombarell, R.; Hirzel, T.; Aspuru-Guzik, A.; Adams, R.P. Convolutional Networks on Graphs for Learning Molecular Fingerprints. In Proceedings of the Annual Conference on Neural Information Processing Systems 2015, Montreal, QC, Canada, 7–12 December 2015; Volume 28, pp. 2224–2232.
9. Johnson, J.; Gupta, A.; Li, F. Image Generation from Scene Graphs. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1219–1228.
10. Li, K.; Zhang, Y.; Li, K.; Li, Y.; Fu, Y. Visual Semantic Reasoning for Image-Text Matching. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27–28 October 2019; pp. 4653–4661.
11. Wang, Z.; Liu, X.; Li, H.; Sheng, L.; Yan, J.; Wang, X.; Shao, J. CAMP: Cross-Modal Adaptive Message Passing for Text-Image Retrieval. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27–28 October 2019; pp. 5763–5772.
12. Zhang, X.; Xu, C.; Tian, X.; Tao, D. Graph Edge Convolutional Neural Networks for Skeleton-Based Action Recognition. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *31*, 3047–3060. [[CrossRef](#)] [[PubMed](#)]
13. Li, M.; Chen, S.; Chen, X.; Zhang, Y.; Wang, Y.; Tian, Q. Actional-Structural Graph Convolutional Networks for Skeleton-Based Action Recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27–28 October 2019; pp. 3590–3598.
14. Qi, M.; Li, W.; Yang, Z.; Wang, Y.; Luo, J. Attentive Relational Networks for Mapping Images to Scene Graphs. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 3952–3961.
15. Bai, Y.; Ding, H.; Bian, S.; Chen, T.; Sun, Y.; Wang, W. SimGNN: A Neural Network Approach to Fast Graph Similarity Computation. In Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, Melbourne, Australia, 11–15 February 2019; pp. 384–392.
16. Li, G.; Müller, M.; Thabet, A.; Ghanem, B. DeepGCNs: Can GCNs Go As Deep As CNNs? In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27–28 October 2019; pp. 9266–9275.
17. Kim, J.; Kim, T.; Kim, S.; Yoo, C.D. Edge-Labeling Graph Neural Network for Few-Shot Learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 11–20.
18. Li, A.; Luo, T.; Lu, Z.; Xiang, T.; Wang, L. Large-Scale Few-Shot Learning: Knowledge Transfer With Class Hierarchy. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 7205–7213.

19. Pope, P.E.; Kolouri, S.; Rostami, M.; Martin, C.E.; Hoffmann, H. Explainability Methods for Graph Convolutional Neural Networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 10764–10773.
20. Gong, L.; Cheng, Q. Exploiting Edge Features for Graph Neural Networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 9203–9211.
21. 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, Sydney, Australia, 6–11 August 2017; Volume 70, pp. 1263–1272.
22. Song, Y.; Zheng, S.; Niu, Z.; Fu, Z.h.; Lu, Y.; Yang, Y. Communicative Representation Learning on Attributed Molecular Graphs. In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20, Yokohama, Japan, 7–15 January 2021; pp. 2831–2838.
23. Yang, K.; Swanson, K.; Jin, W.; Coley, C.; Eiden, P.; Gao, H.; Guzman-Perez, A.; Hopper, T.; Kelley, B.; Mathea, M.; et al. Analyzing Learned Molecular Representations for Property Prediction. *J. Chem. Inf. Model.* **2019**, *59*, 3370–3388. [[CrossRef](#)] [[PubMed](#)]
24. Miyazaki, T.; Omachi, S. Structural Data Recognition With Graph Model Boosting. *IEEE Access* **2018**, *6*, 63606–63618. [[CrossRef](#)]
25. Shang, C.; Liu, Q.; Chen, K.S.; Sun, J.; Lu, J.; Yi, J.; Bi, J. Edge Attention-based Multi-Relational Graph Convolutional Networks. *arXiv* **2018**, arXiv:1802.04944.
26. Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; Bengio, Y. Graph Attention Networks. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018.
27. Chen, B.; Barzilay, R.; Jaakkola, T. Path-Augmented Graph Transformer Network. *arXiv* **2019**, arXiv:cs.LG/1905.12712.
28. Wu, Z.; Ramsundar, B.; Feinberg, E.; Gomes, J.; Geniesse, C.; Pappu, A.S.; Leswing, K.; Pande, V. MoleculeNet: A benchmark for molecular machine learning. *Chem. Sci.* **2018**, *9*, 513–530. [[CrossRef](#)] [[PubMed](#)]