Insights into Chemistry: Explainable AI with Group Contribution in Graph Neural Networks

G. Cathoud[1], V. R. Somnath[2], L. Macedo[1], and K. Jorner[3]

[1] Department of Informatics Engineering, University of Coimbra, Portugal
[2] Department of Computer Science, ETH Zurich, Switzerland
[3] Department of Chemistry and Applied Biosciences, ETH Zurich, Switzerland
kjell.jorner@chem.ethz.ch

Abstract. Recent advancements in Graph Neural Networks (GNNs) have demonstrated significant potential in the realm of chemistry, offering robust performance for various predictive tasks. However, achieving high accuracy is only part of the challenge; understanding the mechanisms driving these models is equally critical for their reliable application. Although numerous general-purpose GNN explainers are available, leveraging domain-specific knowledge can significantly improve the design of explanations for chemical applications. In our research, we introduced an explainability framework grounded in the well-established principle of group contributions. Using this approach, we were able to explain the model's prediction without sacrificing accuracy. Our findings suggest that different GNN models may learn distinct patterns from the molecules. Moreover, by implementing a customized loss function, we successfully guided the models' learning process to align with the expected group contributions, all while preserving overall model performance.

Keywords: Machine Learning · Graph Neural Networks (GNNs) · chemical properties regression · explainable AI (XAI) · group contributions

1 Introduction

The ability to accurately predict chemical properties has been a long-standing goal in both academic research and industrial applications. In recent years, artificial intelligence (AI) and machine learning (ML) have shown significant promise in the field of chemistry [10,3]. Among these advancements, Graph Neural Networks (GNNs) have emerged with tremendous potential in this field, as they naturally align with the intrinsic graph structure of a molecule. Numerous GNN-based models have achieved state-of-the-art accuracy in chemical predictions [6].

However, as models become more accurate, other important debates have emerged. Critics argue that ML models often capture mere correlations in data rather than true chemical principles. Given the complexity and non-linear nature of these models, their internal mechanisms can be difficult for humans to interpret. Therefore, it is not only important to focus on model accuracy and performance but also to understand how these models function internally.

In this context, Explainable Artificial Intelligence (XAI) is of paramount importance. XAI focuses on developing techniques that clarify the functioning of models and explain their predictions. In chemistry, XAI serves three key roles. First, it builds trust among skeptical users by providing transparency. Second, by revealing the inner workings of models, it allows experts to leverage their chemical knowledge to enhance and refine these models. Third, it supports informed decision-making and helps validate the results derived from model predictions. Although various XAI methods have been developed to explain GNN models [22,13,11], many are too general and require customization for specific model applications. A critical discussion centers on the potential benefits of developing explainers that utilize domain-specific knowledge to craft more relevant explanations. Additionally, this can help establish a more accurate ground truth for what an explanation should entail. Some researchers have begun to explore this avenue by developing explainers that incorporate chemical expertise [20,21,1].

In our study, we revisited the established concept of group contribution (GC) methods [8] in chemistry to elucidate GNN models. GC methods decompose molecules into groups and calculate the properties of the molecule by summing the contributions from each group. Typically, these contributions are estimated using linear regression, a simple and transparent approach. Since these groups represent parts of the molecule familiar to chemists, they are intuitively understood within the chemical community. By modifying the aggregation approach of GNN models, we were able to extract group contributions alongside the predictions. Our goal was to implement these changes with minimal alterations to the original models, maintaining their accuracy while simultaneously providing interpretability. Figure 1 illustrates our approach, showing how these modifications were integrated into the existing model framework.

2 Related work

Rasmussen et al. [16] employed perturbation methods to determine the contributions of different molecular fragments to the predicted octanol-water partition coefficient (LogP). They benchmarked the contributions derived from Crippen's LogP model, a type of GC method, against those obtained from ML models. Inspired by their approach, we expanded this methodology to other chemical properties, such as the enthalpy of formation and the HOMO-LUMO gap.

Chen et al. [7] employed the GC concept to develop their model using a GNN architecture based on 2D molecular graphs. They integrated GC principles and benchmarks to improve model accuracy. While their study concentrated on creating a new model based on 2D graphs, our focus has been on providing explanations for state-of-the-art models that leverage 3D molecular structures.

Wu et al. [21] adopted a masking strategy to evaluate the contributions of various molecular fragments, including BRICS substructures, Murcko substructures, and functional groups. These masks helped identify substructures that significantly affect model predictions. Although masking is effective for determining the importance of different graph nodes, our method takes a different

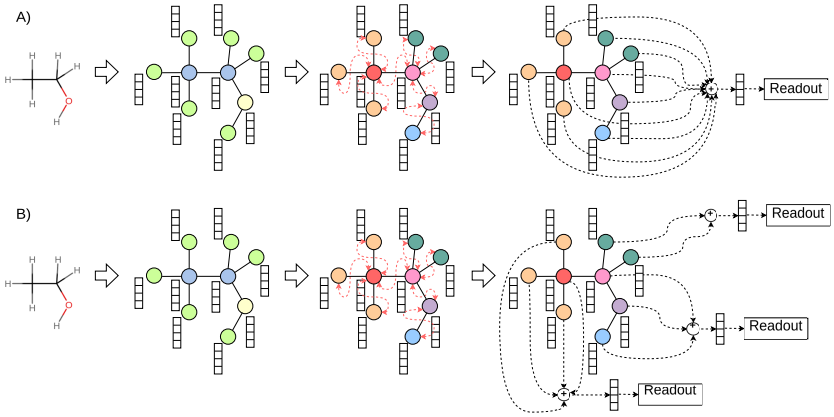

Fig. 1. Schematic of a classical GNN approach (A) and our approach (B). In both approaches, a molecule is represented using a graph where each node is associated with a vector. During the message-passing process, the vectors are updated based on the neighborhood information. In the classical GNN approach (A), all vectors are aggregated after the message-passing process. In our approach (B), the vectors are aggregated based on the information about the groups.

approach by avoiding input modifications. Instead, our aim was to derive explanations directly within the prediction process itself.

Aouichaoui et al. [4] have developed a model that learns embeddings at three levels: node, group, and junction tree. They combined these embeddings and used a multilayer perceptron for predictions. Their junction tree level allowed them to explain predictions and determine the influence of different groups. Rather than employing a multi-level approach, we directly use readouts to obtain scalar values for each group and sum them to produce the final prediction.

Walter et al. [19] used an attention-based approach to identify which molecular parts contribute the most to a given prediction. Their work focused on fingerprint-based models for classification tasks. In contrast, our research centers on GNN models for regression tasks, emphasizing distinct methodological and application focuses.

Our work introduces a novel application of GC methods to explain existing GNNs. The key innovation lies in modifying the GNN's aggregation mechanism to enhance interpretability without sacrificing model performance. This is particularly important, as traditional XAI tools often compromise model accuracy by employing simpler surrogate models or decomposing inputs to evaluate component influences. In our approach, we retained the original message-passing and readout layers of the models, making changes only to the aggregation logic.

3 Proposed approach

3.1 Group Aggregation

GNNs typically consist of message-passing layers that propagate information between nodes, followed by a readout phase to produce a final output. Before the readout, the embeddings of the nodes in the graph are aggregated. Typically, they are either summed or averaged all together altogether.

In our approach, however, we modify this process to aggregate embeddings based on predefined group information. Each group is defined by a central atom and its neighboring atoms. During aggregation, we sum the embeddings of all neighboring atoms with the embeddings of the central atom within each group.

The resulting vector for each group is then passed to the readout layer. This process generates a scalar value for each group, representing its contribution to the overall prediction. The final molecular property prediction is achieved by summing these scalar group contributions (see Fig. 1).

3.2 Group definition

The definition of groups is a critical step in our approach, as it determines how node embeddings will be aggregated. Inspired by Benson's work [5], we defined a group as a central atom and its adjacent neighbors, with a minimum requirement of two neighboring atoms.

3.3 Explanations

After obtaining the contributions of each group, we used heatmaps to visualize these contributions across the molecular structure. Each heatmap highlights the relative importance of different groups.

4 Experiments

4.1 Data

We utilized the QM9 dataset [15] with a focus on predicting the enthalpy of formation (ΔH_f) and the HOMO-LUMO gap ($\Delta\epsilon$). The ΔH_f values were corrected using reference atomic energies as already done by others [2]. The targets were scaled using the mean absolute deviation (MAD) and the median following [17,14]. Data partitioning was performed based on Bemis-Murcko scaffolds, resulting in 70% of the data allocated for training, 15% for validation, and 15% for testing.

4.2 GNN Models

We employed the SchNet [18] and EGNN [17] models, which leverage 3D molecular information, maintain desirable properties such as equivariance, and have been effectively used in various material science applications. Although newer models are available, our objective was to utilize well-established methods to demonstrate proof-of-concept. We anticipate that the observed results would extend to more recent models as well. Given our usage of the message-passing and readout layers from other models, we expect scalability and computational costs to be associated with these operations. The aggregation step is straightforward and efficient, as adjacency information for the groups is precomputed. We employed the original hyperparameters from the respective studies. For each model type, we created two variants: the original and the groups variant. In the groups variant, embeddings were grouped according to Benson groups, and a scalar value was derived for each group, as previously explained. In the original variant, the embeddings were aggregated using a summation operation.

4.3 Regression Models

We also performed linear regression using the ridge method [9] with the group counts within each molecule serving as features. The bias of the ridge models was set to zero. In this case, the group contributions were considered as the coefficients of the ridge regression. A 5-fold cross-validation was applied to determine the optimal hyperparameter for the ridge regression.

4.4 Standard Training and Testing

The models were trained using the Adam optimizer, incorporating a cosine annealing learning rate scheduler and a weight decay of 1×10^{-16}. The initial learning rate was set at 5×10^{-4}. The mean absolute error (MAE) was employed as the loss function for both training and validation phases. Models were validated at 20-epoch intervals. We implemented a checkpointing mechanism with a patience of 10 validation cycles, ceasing training if the validation loss did not improve over 10 validations. The maximum number of epochs was set at 1000.

4.5 Custom Loss

In addition to the standard training procedure, we also trained the models using a custom loss function composed of two elements: the difference between the prediction and the target value, and the cumulative differences between the model's group contributions and reference group contributions (see Equation 1 below). These elements were weighted by an α parameter. The group contribution values from the ridge regression models served as the reference.

$$Loss_{custom} = (1 - \alpha)MAE_{pred.} + \alpha MAE_{groups} \tag{1}$$

Training began with an α value of 0.5, which was linearly decreased to 0. Due to the dynamic nature of α, checkpointing was not used during this phase, and models were trained for 1000 epochs without interruption. The purpose of this approach was to initially bias the model with group contribution information and gradually shift focus solely to minimizing prediction error.

4.6 XAI Plots

We used the RDKit cheminformatics toolkit [12] to generate molecular drawings overlaid with heatmaps created using the group contributions attributed to the central atom of each group.

4.7 Code Availability

The code developed for this study is available at https://github.com/g-cathoud/GNNXGroup

5 Results and discussion

5.1 Accuracy of the Models

The linear regression models performed reasonably well for such a simple model, particularly for predicting ΔH_f ($R^2 = 0.915$). However, their performance for $\Delta\epsilon$ was comparatively lower ($R^2 = 0.815$), likely due to the non-local distribution of the HOMO and LUMO orbitals in the molecules influencing $\Delta\epsilon$. The grouping method divides molecules into segments, favoring the prediction of localized properties like ΔH_f.

GNN models significantly improved accuracy compared to the regression models. For ΔH_f, GNN models achieved an R^2 value of 0.998 across all cases. For $\Delta\epsilon$, GNN models also outperformed the ridge regression models, although predicting $\Delta\epsilon$ remained more challenging than ΔH_f, with R^2 values ranging from 0.917 to 0.934.

When comparing our results for the original GNN models with those reported by the original authors, we found that our MAE was twice as high for ΔH_f and four times as high for $\Delta\epsilon$. This discrepancy was expected, as we used a scaffold split, while the original authors employed a random split. In our experiments, SchNet slightly outperformed EGNN for both ΔH_f and $\Delta\epsilon$.

Interestingly, the different aggregation schemes did not impact the overall model accuracy for SchNet and EGNN. The R^2 values remained nearly constant, and the MAE varied by a maximum of only 7 meV for ΔH_f and 13 meV for $\Delta\epsilon$. Notably, for $\Delta\epsilon$, the grouping method yielded better results. Typically, XAI techniques tend to reduce model accuracy, but our approach offers the significant advantage of enhancing interpretability without sacrificing model performance.

Table 1. Accuracy metrics obtained with the test set for the different models (O - original model, G - model with group aggregation)

ΔH_f	RMSE / meV (↓)	MAE / meV (↓)	R^2 (↑)
R.R.	303	235	0.915
EGNN (O)	43	27	0.998
EGNN (G)	50	30	0.998
SchNet (O)	41	22	0.998
SchNet (G)	46	26	0.998
$\Delta\epsilon$	RMSE / meV (↓)	MAE / meV (↓)	R^2 (↑)
R.R.	514	401	0.815
EGNN (O)	348	222	0.917
EGNN (G)	343	214	0.919
SchNet (O)	337	211	0.922
SchNet (G)	311	198	0.934

5.2 Explainability of the models

Using the group contributions from the different models, we were able to generate XAI plots (see examples in Fig. 2). A general visual analysis revealed a strong alignment in the group contributions between the ridge regression and SchNet for both target properties. In contrast, the agreement between the group contributions from EGNN and ridge regression was notably lower, particularly for ΔH_f.

We also computed the cosine similarity and MAE between the regression and GNN model contributions, as detailed in Table 2. The results align with the insights gained from the visual inspection, as the contributions from the ridge regression and SchNet models exhibited high agreement, with an average cosine similarity of 0.70 for ΔH_f and 0.62 for $\Delta\epsilon$ and an average MAE of 0.25 for ΔH_f and 0.27 for $\Delta\epsilon$.

In contrast, the EGNN models showed much lower cosine similarity values, especially for ΔH_f. This suggests that although EGNN and SchNet achieve similar predictive accuracy, their underlying learned patterns are different. The high agreement between SchNet and the ridge regression model, which is based on Benson's approach, indicates that these models share some chemical insights. However, EGNN seems to be capturing different features, inviting further investigation into the unique aspects identified by EGNN and their potential contributions to chemical understanding.

5.3 Custom loss

Regarding the models trained with the custom loss function, the results indicate that the accuracy of the EGNN in predicting ΔH_f decreased. This result underscores the notion that the EGNN may be learning different patterns in

Table 2. Average cosine similarity (CS) and mean absolute error (MAE) between the contributions of the ridge regression (R.R.) models and the GNN models with group aggregation. Results obtained using the standard training procedure.

Target	Model comparison	μ_{CS} (↑)	μ_{MAE} / ev (↓)
	Regular loss		
ΔH_f	EGNN vs. R.R.	-0.17	0.32
	SchNet vs. R.R.	0.70	0.25
$\Delta\epsilon$	EGNN vs. R.R.	0.51	0.23
	SchNet vs. R.R.	0.62	0.27

this context and does not align with the learning process of the ridge regression model. Consequently, efforts to align the learning with the reference contributions resulted in a decline in the model's performance. This is important because, although the ridge regression method is based on an established methodology from the literature, the EGNN appears to learn distinct and potentially novel patterns.

In terms of the accuracy of the other models, comparisons between Tables 1 and 3 show that there was minimal variation. Notably, the SchNet model demonstrated a slight improvement in accuracy when using the custom loss. This confirms that employing the custom loss function does not impair the accuracy of the models. Instead, it suggests that the custom loss can be beneficial, particularly for models like SchNet.

Table 3. Accuracy metrics obtained with the test set for the different models with group aggregation and using the custom loss.

ΔH_f	RMSE / meV (↓)	MAE / meV (↓)	R^2 (↑)
EGNN	101	38	0.990
SchNet	41	20	0.998
$\Delta\epsilon$	RMSE / meV (↓)	MAE / meV (↓)	R^2 (↑)
EGNN	343	222	0.919
SchNet	308	199	0.935

When considering the alignment between the group contributions from the GNN models and the reference group contributions, the use of the custom loss function resulted in significantly greater agreement. The averages of the alignment metrics are presented in Table 4.

Examining the values in Table 4, it is evident that the agreement is much higher, with the values for the average cosine similarity reaching as high as 0.99 for ΔH_f and 0.96 for $\Delta\epsilon$. Yet, the high average MAE regarding EGNN when predicting ΔH_f further underscores the model's resistance to aligning with the

Table 4. Average cosine similarity (CS) and mean absolute error (MAE) between the contributions of the ridge regression (R.R.) models and the GNN models with group aggregation. These results were obtained by training the models using a custom loss function.

Target	Model comparison	μ_{CS} (↑)	μ_{MAE} / ev (↓)
ΔH_f	EGNN vs. R.R.	0.99	0.52
	SchNet vs. R.R.	0.99	0.04
$\Delta\epsilon$	EGNN vs. R.R.	0.94	0.07
	SchNet vs. R.R.	0.96	0.07

group contributions derived from ridge regression. In other cases, the average MAE remained very low.

Reviewing the examples in Fig. 3, it is clear that the agreement is significantly higher across models. These results are particularly significant because, while the models demonstrated similar accuracy for most cases, the incorporation of the custom loss function enabled the alignment of the model's learning with established chemical intuition.

6 Conclusions

Our study demonstrates that altering the aggregation mechanism in GNN models can significantly improve their explainability without sacrificing performance. This advancement is crucial for providing interpretable predictions while maintaining high accuracy. Our findings also indicate a strong alignment in group contributions between the ridge regression and SchNet models, suggesting that these models capture common fundamental chemical principles. In contrast, the EGNN model's group contributions diverge considerably from those of the ridge regression, indicating that EGNN may be identifying different patterns. This discrepancy invites further exploration into the origins of these differences and the potential insights they might offer.

Moreover, the use of a custom loss function demonstrates that it is feasible to align a model's learning process with specific chemical intuitions, potentially enhancing or at least maintaining model accuracy. The application of group contributions is particularly advantageous as it allows for the integration of established values from the literature, thereby improving the model's alignment with recognized chemical knowledge.

In summary, this study sets the stage for developing more interpretable GNN models in the field of chemistry. The ability to align model learning with established chemical principles while maintaining accuracy underscores the significant potential of combining advanced machine learning techniques with domain-specific knowledge. Future research should explore additional models and datasets to further validate and extend these findings.

Acknowledgments. This work was supported by the Young Talents Fellowship granted by the National Centre of Competence "Sustainable chemical processes through catalyst design" (NCCR Catalysis); by the Portuguese Recovery and Resilience Plan through project C645008882-00000055, Center for Responsible AI; and by national funds through FCT – Foundation for Science and Technology, I.P. (grant number UI/BD/153496/2022), within the scope of the project CISUC (UID/CEC/00326/2020).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. An, H., Liu, X., Cai, W., Shao, X.: Explainable Graph Neural Networks with Data Augmentation for Predicting p K_a of C–H Acids. Journal of Chemical Information and Modeling **64**(7), 2383–2392 (Apr 2024). https://doi.org/10.1021/acs.jcim.3c00958
2. Anderson, B., Hy, T.S., Kondor, R.: Cormorant: Covariant Molecular Neural Networks (2019). https://doi.org/10.48550/ARXIV.1906.04015
3. Anstine, D.M., Isayev, O.: Generative Models as an Emerging Paradigm in the Chemical Sciences. Journal of the American Chemical Society **145**(16), 8736–8750 (Apr 2023). https://doi.org/10.1021/jacs.2c13467
4. Aouichaoui, A.R.N., Fan, F., Mansouri, S.S., Abildskov, J., Sin, G.: Combining Group-Contribution Concept and Graph Neural Networks Toward Interpretable Molecular Property Models. Journal of Chemical Information and Modeling **63**(3), 725–744 (Feb 2023). https://doi.org/10.1021/acs.jcim.2c01091
5. Benson, S.W.: Thermochemical Kinetics: Methods for the Estimation of Thermochemical Data and Rate Parameters. Wiley (1968)
6. Bihani, V., Pratiush, U., Mannan, S., Du, T., Chen, Z., Miret, S., Micoulaut, M., Smedskjaer, M.M., Ranu, S., Krishnan, N.M.A.: EGraFFBench: Evaluation of Equivariant Graph Neural Network Force Fields for Atomistic Simulations (2023). https://doi.org/10.48550/ARXIV.2310.02428
7. Chen, L.Y., Hsu, T.W., Hsiung, T.C., Li, Y.P.: Deep Learning-Based Increment Theory for Formation Enthalpy Predictions. The Journal of Physical Chemistry A **126**(41), 7548–7556 (Oct 2022). https://doi.org/10.1021/acs.jpca.2c04848
8. Gani, R.: Group contribution-based property estimation methods: Advances and perspectives. Current Opinion in Chemical Engineering **23**, 184–196 (Mar 2019). https://doi.org/10.1016/j.coche.2019.04.007
9. Hoerl, A.E., Kennard, R.W.: Ridge Regression: Biased Estimation for Nonorthogonal Problems. Technometrics **42**(1), 80–86 (Feb 2000). https://doi.org/10.1080/00401706.2000.10485983
10. Janet, J.P., Kulik, H.J.: Machine Learning in Chemistry. ACS In Focus, American Chemical Society, Washington, DC, USA (May 2020). https://doi.org/10.1021/acs.infocus.7e4001
11. Kakkad, J., Jannu, J., Sharma, K., Aggarwal, C., Medya, S.: A Survey on Explainability of Graph Neural Networks (2023). https://doi.org/10.48550/ARXIV.2306.01958
12. Landrum, G., Tosco, P., Kelley, B., Rodriguez, R., Cosgrove, D., Vianello, R., sriniker, gedeck, Jones, G., NadineSchneider, Kawashima, E., Nealschneider, D., Dalke, A., Swain, M., Cole, B., Turk, S., Savelev, A., Vaucher, A., Wójcikowski, M.,

Take, I., Scalfani, V.F., Walker, R., Ujihara, K., Probst, D., guillaume godin, Pahl, A., Lehtivarjo, J., Berenger, F., jasondbiggs, strets123: Rdkit/rdkit: 2024_03_1 (Q1 2024) Release. Zenodo (May 2024). https://doi.org/10.5281/ZENODO.591637
13. Li, Y., Zhou, J., Verma, S., Chen, F.: A Survey of Explainable Graph Neural Networks: Taxonomy and Evaluation Metrics (2022). https://doi.org/10.48550/ARXIV.2207.12599
14. Pinsky, E., Klawansky, S.: MAD (about median) vs. quantile-based alternatives for classical standard deviation, skewness, and kurtosis. Frontiers in Applied Mathematics and Statistics **9**, 1206537 (Jun 2023). https://doi.org/10.3389/fams.2023.1206537
15. Ramakrishnan, R., Dral, P.O., Rupp, M., Von Lilienfeld, O.A.: Quantum chemistry structures and properties of 134 kilo molecules. Scientific Data **1**(1), 140022 (Aug 2014). https://doi.org/10.1038/sdata.2014.22
16. Rasmussen, M.H., Christensen, D.S., Jensen, J.H.: Do machines dream of atoms? Crippen's logP as a quantitative molecular benchmark for explainable AI heatmaps (Dec 2022). https://doi.org/10.26434/chemrxiv-2022-gnq3w-v2
17. Satorras, V.G., Hoogeboom, E., Welling, M.: E(n) Equivariant Graph Neural Networks (2021). https://doi.org/10.48550/ARXIV.2102.09844
18. Schütt, 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 (2017). https://doi.org/10.48550/ARXIV.1706.08566
19. Walter, M., Webb, S.J., Gillet, V.J.: Interpreting Neural Network Models for Toxicity Prediction by Extracting Learned Chemical Features. Journal of Chemical Information and Modeling **64**(9), 3670–3688 (May 2024). https://doi.org/10.1021/acs.jcim.4c00127
20. Wellawatte, G.P., Seshadri, A., White, A.D.: Model agnostic generation of counterfactual explanations for molecules. Chemical Science **13**(13), 3697–3705 (2022). https://doi.org/10.1039/D1SC05259D
21. Wu, Z., Wang, J., Du, H., Jiang, D., Kang, Y., Li, D., Pan, P., Deng, Y., Cao, D., Hsieh, C.Y., Hou, T.: Chemistry-intuitive explanation of graph neural networks for molecular property prediction with substructure masking. Nature Communications **14**(1), 2585 (May 2023). https://doi.org/10.1038/s41467-023-38192-3
22. Yuan, H., Yu, H., Gui, S., Ji, S.: Explainability in Graph Neural Networks: A Taxonomic Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence pp. 1–19 (2022). https://doi.org/10.1109/TPAMI.2022.3204236

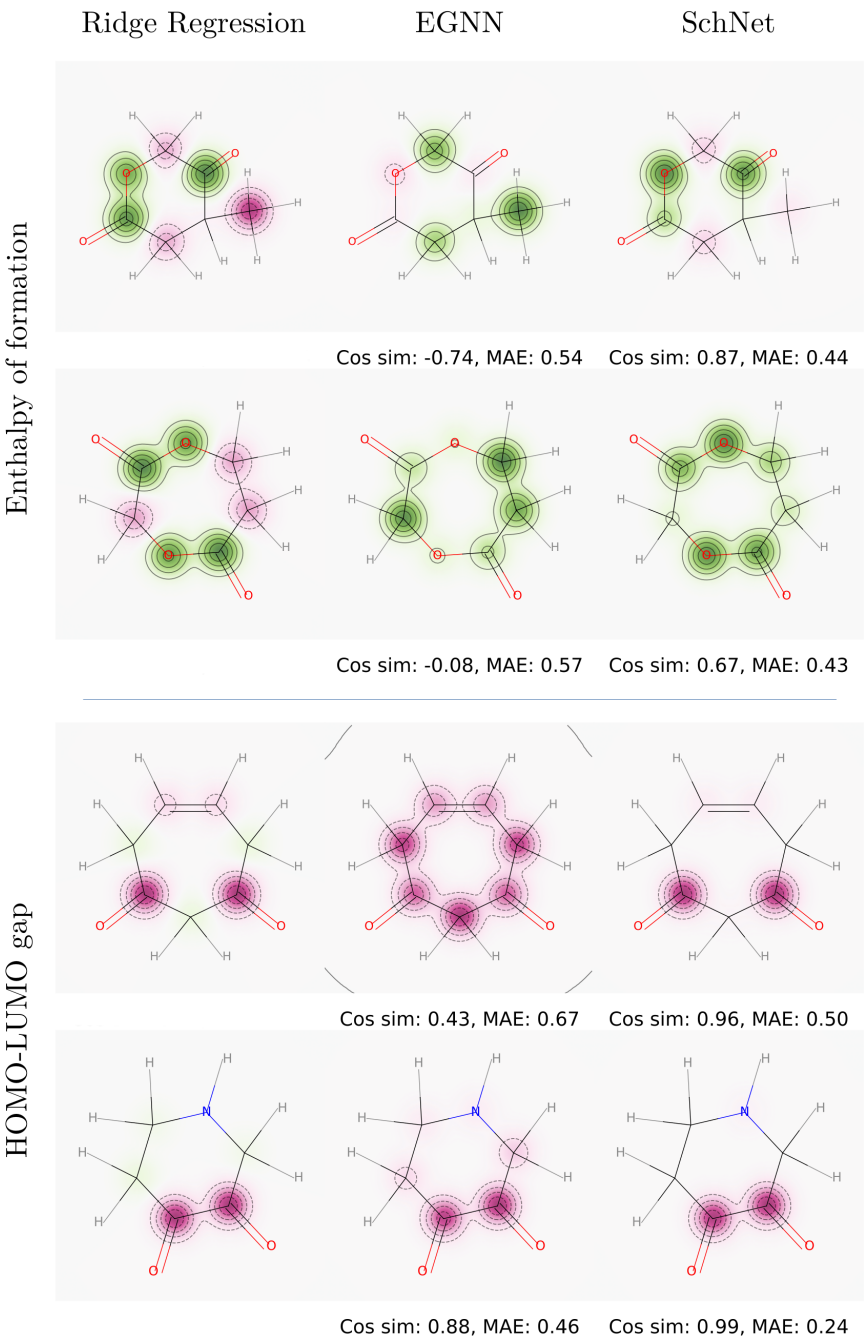

Fig. 2. Example of heatmaps obtained with the group contributions for the ridge regression and the GNN models with group aggregation. In this case, Cos sim stands for cosine similarity. Green colors represent positive values, while pink colors represent negative values.

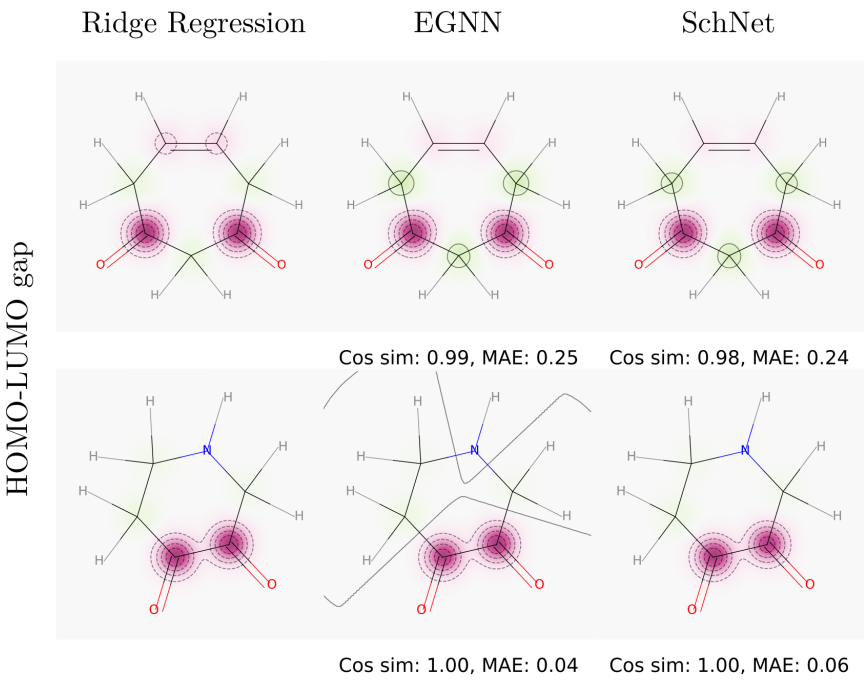

Fig. 3. Example of heatmaps obtained with the group contributions for the ridge regression and the GNN models with group aggregation. Results obtained with custom loss. Again, Cos sim stands for cosine similarity. Green colors represent positive values, while pink colors represent negative values.