https://doi.org/10.48047/AFJBS.7.4.2025.1391-1397

African Journal of Biological Sciences

Journal homepage: http://www.afjbs.com

Research Paper Open Access

TEST THE ACCURACY OF THE MESSAGE PASSING NEURAL NETWORKS DEEP LEARNING MODEL IN PREDICTING MOLECULAR PROPERTIES

Nguyen Thanh Tung[1], Mai Thi Hao[1], Ngo Thi Ngoc Diu[2*]

[1] Institute of Engineering Technology, Thu Dau Mot University, Vietnam
[2] Institute of Information Technology, Digital Transformation, Thu Dau Mot University, Vietnam
[*]Corresponding author: diuntn@tdmu.edu.vn

Volume 7, Issue 4,Apr 2025

Received: 15 Feb 2025

Accepted: 05 Mar 2025

Published: 05 Apr 2025

doi:10.48047/AFJBS.7.4.2025.1391-1397

ABSTRACT

Density Functional Theory (DFT) is the most important foundation in computational quantum chemistry. DFT provides a solid theoretical foundation for predicting the physical and chemical properties of molecules in materials. Although quantum computing methods based on DFT give very accurate results, they require huge computational resources. Therefore, high-throughput applications, which require the calculation of material properties for a large number (tens or hundreds of thousands) of molecules, cannot utilize DFT. Recent advances in deep learning, especially message-passing neural networks (MPNNs), offer a promising alternative for large-scale molecular property predictions. This project aims to evaluate the accuracy and effectiveness of the MPNNs deep learning model compared to DFT in predicting molecular properties. We use tests like R-Squared (0.87÷0.99), Pearson's R (0.97÷0.99), and Spearman's Rho (0.91÷0.99) to look at the predictions of 12 physicochemical properties for about 13,000 molecules in the test set. This shows that the potential for applying MPNNs to building a database system of new and doped materials is huge in the future.

1. INTRODUCTION

DFT is a quantum mechanical method [1] that is most commonly used to study the electronic structure of many-particle systems [2]. DFT simplifies the complex many-electron problem by using an electron density function instead of a wave function. The use of electron density functionals makes calculations of molecules with lots of electrons more feasible. DFT is a precise tool for predicting molecular structures, reaction energies, electronic properties, etc., providing insight into chemical reactivity and material properties [3]. However, DFT has main limitations that limit the wide applicability of this method such as: The high computational costs prevent the DFT method from scaling

up for large molecular systems, such as those in biology. school is very large in size); These exchange correlation functions may be accurate for this material but not for this material, affecting DFT accuracy other materials); DFT requires huge computational resources, so its use is limited for high-throughput screening of large data sets [4-7].

MPNNs are a class of deep learning models designed for structured data, particularly graphs [8]. They are commonly used in applications such as molecular modeling, social network analysis, and recommendation systems [9].

The application of the MPNNs method includes two main phases [10]: the Message Passing Phase and the Readout Phase. The initial phase is the crucial stage for the nodes in the graph to update information from neighboring nodes. This phase is repeated T times, each time consisting of three steps, including Calculate the message Mt, each edge in the graph transmits a message from a source vertex to a target vertex. The message is calculated based on the current state of the source vertex h_u^t, the current state of the target vertex h_v^t, and the edge's characteristic information e_{uv}. The general formula is as follows:

$$m_{uv}^t = M_t(h_u^t, h_v^t, e_{uv}) \qquad (1)$$

Where the message m_{uv}^t is sent from vertex u to v at step t.

The next step is to aggregate the message A_t, each vertex aggregates the message from all its neighboring vertices. This aggregation process typically uses functions such as sum, mean, or max, as shown in the following formula:

$$m_v^t = A_t(\{m_{uv}^t \mid u \in N(v)\}) \qquad (2)$$

Where it N(v) is the set of neighboring vertices of v. The third step is to update the state U_t. After aggregating the messages, the state of each vertex is updated based on the current state and the aggregated message according to formula (3):

$$h_v^{t+1} = U_t(h_v^t, m_v^t) \qquad (3)$$

The second phase is the Readout Phase. After completing T iterations, a readout function (R) is used to extract information from the graph or the vertices: for the graph classification problem: The R function usually calculates the sum or average of the states of all vertices. For the vertex classification problem: The R function only takes the states of specific vertices. We have the general formula (4) below:

$$y = R(\{h_v^T \mid v \in V\}) \qquad (4)$$

With v being the set of vertices in the graph. The ability to handle graph topologies is one of the MPNN method's highlights; it accurately simulates the relationships between vertices and edges. Because the functions M_t, A_t, U_t, and R can be customized, the flexibility is appropriate for a variety of graphs and situations. Scalability: utilizing effective synthesis techniques, it is simple to apply to enormous graphs.

Sumary, MPNN is a class of neural networks specifically designed for graph-structured data, such as molecules [11-21]. They work by passing messages between nodes (atoms) via edges (bonds), capturing the local chemical environment and allowing prediction of molecular properties. The main advantages of MPNN include: MPNN possesses the capability to manage extensive datasets and intricate molecular structures. Once trained, MPNN can make predictions quickly [22-26], providing a significant speed advantage over DFT.

2. METHOD

2.1 Research objectives

In this project, we apply the MPNN [27-31] deep learning model deployed in the Chemprop library to train the model for the QM9 data set including 12 quantities of physicochemical properties of organic compounds. The QM9 data set includes 113 thousand molecules divided into a training set (100 thousand) and a test set (13 thousand). We represent each molecule as the string SMILES. We collect the 12 predicted quantities from articles that have performed DFT calculations.

smiles	mu	alpha	homo	lumo	gap	r2	zpve	cv	u0	u298	h298	g298
C	0.00	13.21	-0.39	0.12	0.50	35.36	0.04	6.47	-40.48	-40.48	-40.48	-40.50
N	1.63	9.46	-0.26	0.08	0.34	26.16	0.03	6.32	-56.53	-56.52	-56.52	-56.54
O	1.85	6.31	-0.29	0.07	0.36	19.00	0.02	6.00	-76.40	-76.40	-76.40	-76.42
C#C	0.00	16.28	-0.28	0.05	0.34	59.52	0.03	8.57	-77.31	-77.31	-77.30	-77.33
C#N	2.89	12.99	-0.36	0.02	0.38	48.75	0.02	6.28	-93.41	-93.41	-93.41	-93.43
C=O	2.11	14.18	-0.27	-0.04	0.23	59.99	0.03	6.41	-114.48	-114.48	-114.48	-114.51

Table 1. Example data for a few molecules in the QM9 data set.

They include: mu (μ) dipole moment in chemistry; alpha (α) polarity of the molecule; homo (Highest Occupied Molecular Orbital), the orbital in which the electron has the highest energy in the molecule atom, but still fully occupied; lumo (Lowest Unoccupied Molecular Orbital), the orbital where the electron has the lowest energy in the molecule, but still not fully occupied; gap is the band gap energy level of the molecule; Borh glass; zpve is the molecular energy at 0 K; cv is the molecular energy at 00K; u298_atom is the molecular energy at 298.15 K; h298_atom is the molecular energy atom at 298.15 K; g298_atom (eV) is the free energy of the molecule at 298.15 K (See Table 1).

The goal of the project is to train the MPNN model for a training set of 100 thousand molecules and test the accuracy on a test set with 13 thousand molecules.

2.2 Research methods

The Chemprop library mainly uses a machine learning model based on graph convolutional neural networks (Graph Convolutional Networks, GCNs). More specifically, Chemprop uses a GCN specification called MPNNs. MPNNs are a type of deep learning model designed to work with graph data, where atoms are represented as nodes and chemical bonds are represented as edges in the graph. The Chemprop model training process is based on loss functions to optimize the parameters of the neural network:

- Regression Loss: Used when predicting continuous values solubility and binding energy. The universal loss function is Mean Squared Error (MSE), as in formula (5):

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \qquad (5)$$

In this case, yi represents the actual value and is the expected value.

- Classification Loss function: Used when predicting classification values whether a compound is biologically active or not. The universal loss function is Binary Cross-Entropy for binary classification problems, according to formula (6):

$$BCE = -\frac{1}{n}\sum_{i=1}^{n}[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)] \qquad (6)$$

Chemprop combines chemical features (descriptors) from molecular graphs with MPNNs to learn to predict desired properties of molecules. This process includes message passing between nodes in the graph to collect structural information and calculate the final features, then use these features to make predictions.

After installing the Python and Linux environments and installing the Chemprop library, prepare the necessary data, including files in train.py format running on the Python environment and data files train_100k.csv, and test files. look up test.csv

The code on the Chemprop (Python) interface we wrote consists of 2 steps, with the following form:

- Step 1: Declare the environment, data type and folder containing specific data and code:

```
import chemprop
import pandas as pd
arguments = [
```

```
    '--data_path', '/home/tungnt/AI_ThayDao/data/train_100k.csv',
    '--dataset_type', 'regression',
    '--save_dir', '/home/tungnt/AI_ThayDao/train_checkpoints/model_trained_100k'
]
args = chemprop.args.TrainArgs().parse_args(arguments)
chemprop.train.cross_validate(args=args, train_func=chemprop.train.run_training)
```

- Step 2: Train the model and specific code:

```
  arguments = [
   "--test_path", "/home/tungnt/AI_ThayDao/data/test.csv",
   "--preds_path", "/home/tungnt/AI_ThayDao/data/train_100k.csv",
   "--checkpoint_dir", "/home/tungnt/AI_ThayDao/train_checkpoints/model_trained_100k"
]
args = chemprop.args.PredictArgs().parse_args(arguments)
model_objects = chemprop.train.load_model(args=args)
preds = chemprop.train.make_predictions(args=args, smiles=smiles,
                                        model_objects=model_objects)
preds
```

The time to proceed in step 1, declaring the database, takes about 4 minutes, the model training time in step 2, for the train_100k.csv file and test.csv file lasts up to 34 hours.

3. RESULTS AND DISCUSSION

Through calculations from the model training process using the MPNNs method on the Python platform and the Chemprop library, we obtained the results in Table 2, showing the correlation between the original real parameters using the DFT method. that the QM9 data source has provided includes 12 physicochemical parameters of organic molecules.

Property	RMSR	R-Squared	Pearson's R	Spearman's Rho
mu	0.553	0.869	0.933	0.915
alpha	1.019	0.985	0.992	0.993
homo	0.005	0.955	0.978	0.968
lumo	0.005	0.988	0.994	0.993
gap	0.007	0.979	0.990	0.991
r2	43.567	0.976	0.988	0.985
zpve	0.002	0.996	0.998	0.997
cv	0.441	0.988	0.994	0.995
u0_atom	20.636	0.993	0.996	0.997
u298_atom	20.745	0.993	0.996	0.997
h298_atom	20.914	0.993	0.996	0.997
g298_atom	19.217	0.992	0.996	0.996

TABLE 2. Calculation results of correlation coefficients and errors between DFT and MPNN

According to the results in TABLE 2, it is shown that: i) The R-Squared correlation coefficient has a value from 0.87 to 0.99, showing that for every 100% of parameter deviations in the actual database (DFT), the model can explain and recognize 87% to 99% by machine learning according to the MPNNs model; ii) Pearson's R coefficient from 0.97 to 0.99, indicating that the degree of linear relationship between two numerical parameters according to DFT and MPNN methods is almost linear, gradually to completely linear (100%); iii) Spearman's Rho coefficient from 0.91 to 0.99 a statistical measure indicating the degree and direction of the monotonic relationship between two numerical parameters from DFT and MPNN calculations. Through the above three ways of considering the correlation, we find that the MPNN method we apply here is very effective and very reliable. Below are some graphs drawn from ML calculation results with each parameter comparing DFT and MPNN methods.

To understand and analyze the relationship between RMSE (Root Mean Square Error) and Pearson's R (Pearson correlation coefficient) from a graph containing data points, you need to clarify the basic concepts and their relationship. After that, based on the data from the graph, we can draw conclusions about the relationship between them.

RMSE measures the degree of deviation between the predicted value and the actual value. We have formula (5) derived into (7) below:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \qquad (7)$$

When the calculated RMSE value is smaller, the applied model fits the actual data better. Unnormalized RMSE depends on the unit of the output variable. Correlation coefficient Pearson's R measures the degree of linearity between two variables. We have formula (8):

$$R = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 . \sum_{i=1}^{n}(y_i - \bar{y})^2}} \qquad (8)$$

With x_i, y_i are two comparison variables, $\bar{x}_i, \bar{y}_i$ are the average values of the variables. When R = 1, it is a perfect positive linear relationship; R = −1 is a perfect negative linear relationship; R = 0 indicates no linear relationship. On the other hand, Pearson's R is not affected by the units of the variables. If R increases when RMSE decreases, it indicates that when the model predicts better (lower RMSE), the linear relationship between the prediction and the actual value is also stronger (higher Pearson's R). If there is no clear relationship, this may occur if the model performs well on nonlinear aspects or is influenced by noise.

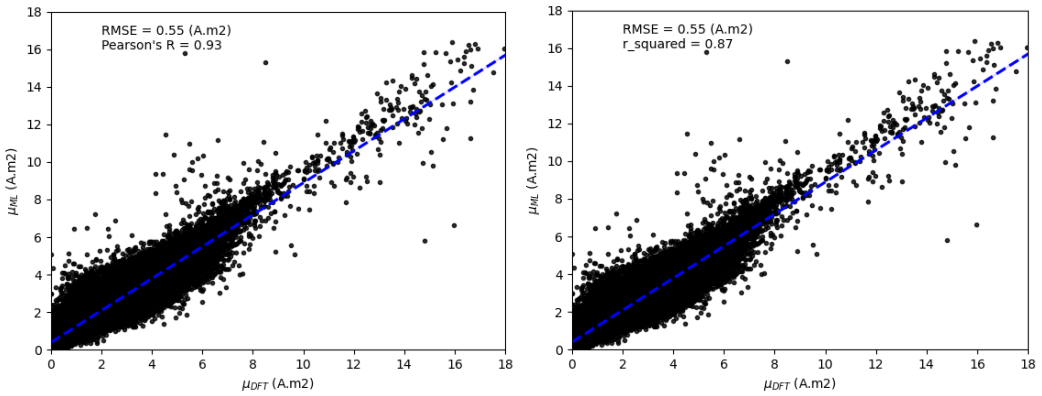

Fig 1. The relationship between RMSE, Pearson's R, r_squared and DFT method for μ

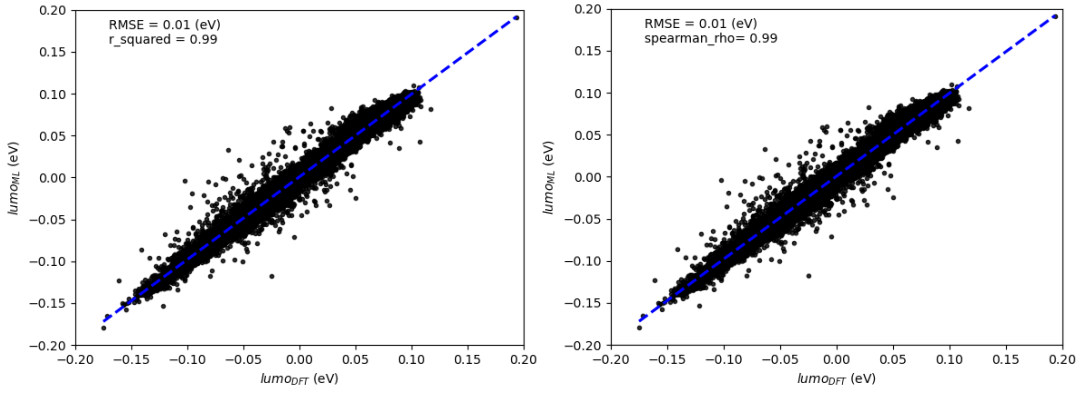

Fig 2. The relationship between RMSE, r_squared, spearman_rho and DFT method for LUMO

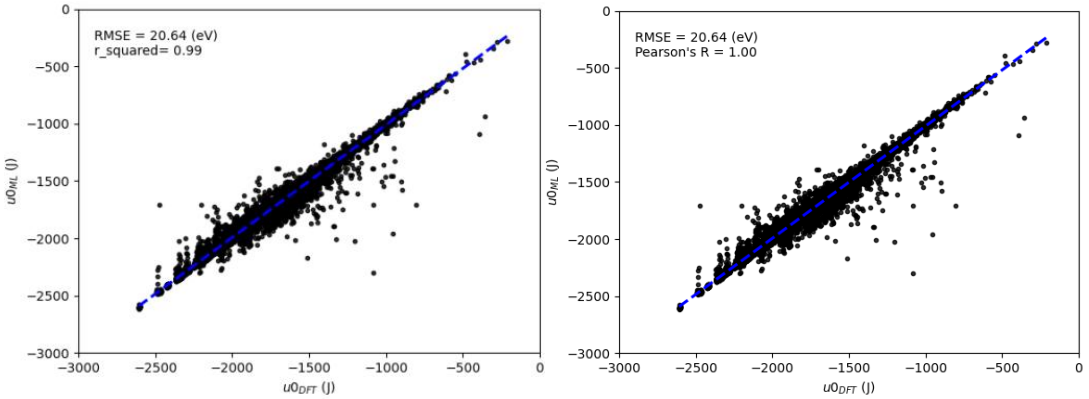

Fig 3. The relationship between RMSE, r_squared, Pearson's R and DFT method for u_0

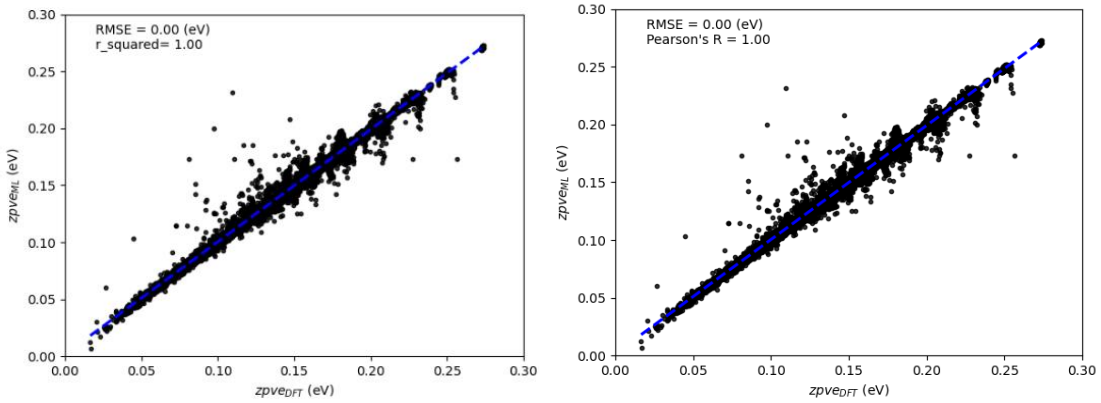

Fig 4. The relationship between RMSE, r_squared, Pearson's R and DFT method for ZPVE

Relatively different outcomes are shown by the association between RMSE and Pearson's R and r_squared based on the evaluation approach from formulas (7) and (8) and Figure 1. The RMSE value of 0.55 A.m2 indicates that the μ index is not accurate for the ML and DFT approaches, despite Pearson's R reaching 93% and r_squared at 87%.Considering the LUMO parameter shown in Figure 2, we see that the RMSE error value is 0.02 eV. The accuracy of r_squared and Spearman_rho, both at 99%, indicates that the ML method is close to DFT. Based on Fig 3, the RMSE is around 20.64 eV, indicating that the values from the DFT and ML methods do not differ much because the actual values are very large. The r-squared index of 99% and Pearson's R of 100% show that the accuracy between the two methods, ML and DFT, is very high, and the data is very convergent. Fig 4 shows that RMSE = 0 means that there is no linear relationship between the DFT data and MPNNs machine learning. As a result, the unit of the variable has no effect on the Pearson's R index. Nevertheless, examining the two graphs in Figure 4, the convergence points match Pearson's R and r_squared, and both are equal to 1, suggesting that the DFT technique and ML have a high degree of accuracy.

4. CONCLUSION

Through the results obtained from the project conducted in a short time, it shows that the application of ML in training models applied in the field of chemistry when connecting to the Chemprop library on the Python platform is very effective from the correlation calculation, it has high accuracy between real data and after applying MPNN, opening up many future research prospects for other models such as finding the bandgap energy Eg of substances newly doped based on known data from experiments or DFT and MD simulations. This method is less expensive in terms of time, money and human resources compared to experiments as well as DFT or MD simulations, showing that this is an effective research trend that is very necessary to be applied in the near future.

ACKNOWLEDGEMENTS

Thank you for the funding for this research from Thu Dau Mot University (TDMU)

REFERENCES

1. Sumit Sharma et al, "Chapter 1 - Introduction to Molecular Dynamics, Molecular Dynamics Simulation of Nanocomposites Using BIOVIA Materials Studio, Lammps and Gromacs", *Micro and Nano Technologies*, 2019, 1-38. https://doi.org/10.1016/B978-0-12-816954-4.00001-2
2. A.A. Vasilchenko et al, Density-functional study of electronic structure of quantum dots in high magnetic fields, Journal of Magnetism and Magnetic Materials, 2023, 588 (B): 171519. https://doi.org/10.1016/j.jmmm.2023.171519
3. Manjeet Bhatia et al, An overview of conceptual-DFT based insights into global chemical reactivity of volatile sulfur compounds (VSCs), *Computational Toxicology*, 2024, 29: 100295 https://doi.org/10.1016/j.comtox.2023.100295
4. Thanh Tung Nguyen, Van Ngoc Hoang, Thi Phuong Thuy Huynh, Duy Khanh Nguyen, Van On Vo, "Diverse structural and electronic properties of carbon-substituted amchair silicene nanoribbons: A first-principles study", *Physica E*, 2022; 142: 115309 https://doi.org/10.1016/j.physe.2022.115309
5. Hoang Van Ngoc, Nguyen Thanh Tung, Duy Khanh Nguyen, Vo Van On, J.F. Rivas-Silva, Gregorio H. Cocoletzi, D.M. Hoat, "First-principles calculations to investigate Structural, electronic, and optical properties of MgF2 monolayer in 1T-phase and 2H-phase using hybrid functional", *Chemical Physics*, 2022; 557: 111473. https://doi.org/10.1016/j.chemphys.2022.111473
6. Thanh Tung Nguyen, Hoang Van Ngoc, Huynh Thi Phuong Thuy, Duy Khanh Nguyen, Vo Van On, "Applying first principles to study the structural,

electronic, and magnetic properties of Pr adsorbed ASiNRs", *Thu Dau Mot University Journal of Science*, 2021; 3: 4
https://doi.org/10.37550/tdmu.EJS/2021.04.257

7. Thanh Tung Nguyen, "Structural and electronic properties of silveradsorbed silicene nanoribbons: A first-principles study", *Journal of Physics: Conference Series 2199*, 2022: 012011.
https://iopscience.iop.org/article/10.1088/1742-6596/2199/1/012011
8. Miru Tang et al, Application of message passing neural networks for molecular property prediction, *Current Opinion in Structural Biology*, 2023, 81: 102616.
https://doi.org/10.1016/j.sbi.2023.102616
9. Jie Zhou et al, Graph neural networks: A review of methods and applications, AI Open, 2020, 1: 57-81.
https://doi.org/10.1016/j.aiopen.2021.01.001
10. Jiali Li et al, AI Applications through the Whole Life Cycle of Material Discovery, Matter, 2020, 3(2): 393-432.
https://doi.org/10.1016/j.matt.2020.06.011
11. Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; Dahl, G. E. "Neural Message Passing for Quantum Chemistry". *Proceedings of the International Conference on Machine Learning*, *2017*: 1263−1272.
https://arxiv.org/pdf/1704.01212
12. Gasteiger, J.; Groß, J.; Günnemann, S, "Directional Message Passing for Molecular Graphs". Proceedings of the International Conference on Learning Representations, 2003.
https://doi.org/10.48550/arXiv.2003.03123
13. Zhang, S.; Liu, Y.; Xie, L, "Molecular Mechanics-Driven Graph Neural Network with Multiplex Graph for Molecular Structure Machine Learning for Molecules", *Workshop at Neur IPS*, 2020.
https://doi.org/10.48550/arXiv.2011.07457
14. Vermeire, F. H.; Chung, Y.; Green, W. H, "Predicting SolubilityLimits of Organic Solutes for a Wide Range of Solvents and Temperatures". *J. Am. Chem. Soc*. 2022; 144: 10785−10797.
https://doi.org/10.1021/jacs.2c01768
15. Dobbelaere, M. R.; Ureel, Y.; Vermeire, F. H.; Tomme, L.;Stevens, C. V.; Van Geem, K. M, "Machine Learning for Physicochemical Property Prediction of Complex Hydrocarbon Mixtures". *Ind. Eng. Chem. Res*, 2022; 61: 8581−8594.Journal of Chemical Information and Modeling pubs.acs.org/jcim Application Notehttps.
https://doi.org/10.1021/acs.iecr.2c00442
16. Dobbelaere, M. R.; Plehiers, P. P.; Van de Vijver, R.; Stevens, C.V.; Van Geem, K. M, "Learning Molecular Representations for Thermochemistry Prediction of Cyclic Hydrocarbons and Oxygen-ates". *J. Phys. Chem. A*, 2021; 125: 5166−5179.
https://doi.org/10.1021/acs.jpca.1c01956
17. Rittig, J. G.; Ben Hicham, K.; Schweidtmann, A. M.; Dahmen,M.; Mitsos, A,"Graph Neural Networks for Temperature-Dependent Activity Coefficient Prediction of Solutes in Ionic Liquids". *Comput.Chem. Eng*. 2023; 171: 108153.
https://doi.org/10.48550/arXiv.2206.11776
18. Rittig, J. G.; Ritzert, M.; Schweidtmann, A. M.; Winkler, S.;Weber, J. M.; Morsch, P.; Heufer, K. A.; Grohe, M.; Mitsos, A.;Dahmen, M, "Graph Machine Learning for Design of High-Octane Fuels". *AIChE J*. 2023; 69: e17971.
https://doi.org/10.48550/arXiv.2206.00619
19. Fleitmann, L.; Ackermann, P.; Schilling, J.; Kleinekorte, J.; Rittig,J. G.; vom Lehn, F.; Schweidtmann, A. M.; Pitsch, H.; Leonhard, K.;Mitsos, A.; Bardow, A.; Dahmen, M, "Molecular Design of Fuels for Maximum Spark-Ignition Engine Efficiency by Combining Predictive Thermodynamics and Machine Learning". *Energ. Fuel*, 2023; 37: 2213−2229.
https://doi.org/10.1021/acs.energyfuels.2c03296
20. Stokes, J. M.; Yang, K.; Swanson, K.; Jin, W.; Cubillos-Ruiz, A.;Donghia, N. M.; MacNair, C. R.; French, S.; Carfrae, L. A.; Bloom-Ackermann, Z.; et al, "A Deep Learning Approach to Antibiotic Discovery". *Cell 2020*; 180: 688−702.
https://doi.org/10.1016/j.cell.2020.01.021
21. De Cao, N.; Kipf, T, MolGan, "An Implicit Generative Modelfor Small Molecular Graphs". *arXiv Preprint*, 2022.
https://doi.org/10.48550/arXiv.1805.11973
22. Dan, Y.; Zhao, Y.; Li, X.; Li, S.; Hu, M.; Hu, J, "GenerativeAdversarial Networks (GAN) Based Efficient Sampling of Chemical Composition Space for Inverse Design of Inorganic Materials". *NpjComput. Mater*. 2020; 6: 1−7.
https://doi.org/10.1038/s41524-020-00352-0
23. Gómez-Bombarelli, R.; Wei, J. N.; Duvenaud, D.; Hernández-Lobato, J. M.; Sánchez-Lengeling, B.; Sheberla, D.; Aguilera-Iparraguirre, J.; Hirzel, T. D.; Adams, R. P.; Aspuru-Guzik, A, "Automatic Chemical Design using a Data-Driven Continuous Representation of Molecules". *ACS Cent. Sci*. 2018; 4: 268−276.
https://doi.org/10.1021/acscentsci.7b00572
24. Wei, J. N.; Duvenaud, D.; Aspuru-Guzik, A, "Neural Networks for the Prediction of Organic Chemistry Reactions". *ACS Cent. Sci.*2016; 2: 725−732.
https://doi.org/10.1021%2Facscentsci.6b00219
25. Segler, M. H.; Waller, M. P, "Neural-Symbolic MachineLearning for Retrosynthesis and Reaction Prediction". *Chem. Eur. J*,.2017; 23: 5966−5971.
https://doi.org/10.1002/chem.201605499
26. Coley, C. W.; Barzilay, R.; Jaakkola, T. S.; Green, W. H.;Jensen, K. F, "Prediction of Organic Reaction Outcomes using Machine Learning". *ACS Cent. Sci*, 2017; 3: 434−443.
http://dx.doi.org/10.1021/acscentsci.7b00064
27. Coley, C. W.; Green, W. H.; Jensen, K. F, "Machine Learning in Computer-Aided Synthesis Planning". *Acc. Chem. Res*, 2018; 51: 1281−1289.
https://doi.org/10.1021/acs.accounts.8b00087
28. Szymkuć , S.; Gajewska, E. P.; Klucznik, T.; Molga, K.; Dittwald,P.; Startek, M.; Bajczyk, M.; Grzybowski, B. A, "Computer-Assisted Synthetic Planning: The End of the Beginning". *Angew. Chem., Int. Ed*, 2016; 55: 5904−5937.
https://doi.org/10.1021/acs.accounts.8b00087
29. Segler, M. H.; Preuss, M.; Waller, M. P, "Planning Chemical Syntheses with Deep Neural Networks and Symbolic AI". *Nature*, 2018; 555: 604−610.
https://doi.org/10.1038/nature25978
30. Kayala, M. A.; Azencott, C.-A.; Chen, J. H.; Baldi, P, "Learningto Predict Chemical Reactions". *J. Chem. Inf. Model*, 2011; 51: 2209−2222.
https://doi.org/10.1021/ci200207y
31. Kayala, M. A.; Baldi, P, "Reaction Predictor: Prediction of Complex Chemical Reactions at the Mechanistic Level using MachineLearning*". J. Chem. Inf. Model.* 2012; 52: 2526−2540.
http://dx.doi.org/10.1021/ci3003039