Comparison of Atom Representations in Graph Neural Networks for Molecular Property Prediction

Agnieszka Pocha*, Tomasz Danel*†, Sabina Podlewska‡, Jacek Tabor* and Łukasz Maziarka*†
*Faculty of Mathematics and Computer Science, Jagiellonian University, Kraków, Poland
†Ardigen, Kraków, Poland
‡Maj Institute of Pharmacology, Polish Academy of Sciences, Kraków, Poland
Email: agnieszka.pocha@doctoral.uj.edu.pl, lukasz.maziarka@ii.uj.edu.pl

2021 International Joint Conference on Neural Networks (IJCNN) | 978-1-6654-3900-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/IJCNN52387.2021.9533698

***Abstract*—Graph neural networks have recently become a standard method for analysing chemical compounds. In the field of molecular property prediction, the emphasis is now put on designing new model architectures, and the importance of atom featurisation is oftentimes belittled. When contrasting two graph neural networks, the use of different atom features possibly leads to the incorrect attribution of the results to the network architecture. To provide a better understanding of this issue, we compare multiple atom representations for graph models and evaluate them on the prediction of free energy, solubility, and metabolic stability. To the best of our knowledge, this is the first methodological study that focuses on the relevance of atom representation to the predictive performance of graph neural networks.**

***Index Terms*—Graph neural networks, Molecular property prediction**

I. Introduction

Graph convolutional neural networks (GCNs) are state-of-the-art models for predicting molecular properties. As the input, they use molecular graphs in which vertices represent atoms and edges represent the chemical bonds. Since graph-based models were shown to outperform models based on molecular fingerprints [1], the interest in GCNs increased, which resulted in proposing new models for molecular property prediction [2]–[10].

Since the beginning, the main focus of the deep learning community has been placed on developing better machinery for processing the graph data. For instance [11] introduce the attention mechanism for GCNs and [12] introduce a dummy super node – an artificial node connected to all nodes in the graph which is responsible for learning the graph-level representation. At the same time, authors of new methods oftentimes neglect the impact of the used atomic representation. Therefore, atoms are represented in a different way for each new graph-based model, which may lead to the unfair attribution of the achieved results solely to the developed processing methods.

There is a need for systematic comparison of graph representations which seems independent from the choice of architecture. In this work, we compare different atomic representations using simple GCNs [13] and evaluate them on multiple datasets.

The work of A. Pocha and Ł. Maziarka was supported by the National Science Centre (Poland) grant no. 2019/35/N/ST6/02125. The work of T. Danel was supported by the National Science Centre (Poland) grant no. 2020/37/N/ST6/02728.

Initially, we study the influence of single atomic features in isolation and show that some of them encode more useful information than others. Subsequently, we examine few exemplary representations used in literature and show their differences in terms of accuracy and generalisation gap.

Furthermore, we analyse the results qualitatively by showing molecules which are most difficult to predict for GCNs with a given representation. Finally, we use t-SNE [14] to show that molecules with the highest prediction error tend to cluster together.

The code used to run the presented experiments is available at github.com/gmum/graph-representations.

II. Related work

Two main components of molecular property prediction are the representation of chemical compounds and the algorithm used to calculate the property values. The classical machine learning methods which were used to find the relationship between chemical structures of molecules and their properties used simple 1D molecular descriptors, e.g. lipophilicity or electron density, which were plugged into machine learning models to create predictions of more complex molecular properties [15]. Shortly afterwards, these descriptors were replaced by features derived from the structure of molecules.

A prominent example of structural compound descriptors are molecular fingerprints, which are typically a mapping from chemical substructures to numeric feature vectors. The vectors constructed in this way can become an input to machine learning models, e.g. random forests, support vector machines, or neural networks, in order to find quantitative structure-property relationships (QSPR).

ECFP fingerprint [16] is one of the most commonly used fingerprints in this setup [17], [18]. To calculate this molecular representation, the algorithm uses a hash function that encodes fragments contained in the molecule. The crucial part of this encoding is the atomic representation that makes the individual atoms in fragments distinguishable. As their representation, Rogers and Hahn [16] use the number of non-hydrogen neighbours, the valence minus the number of hydrogens, the atomic number, the atomic mass, the atomic charge, the number of attached hydrogens, and inclusion in rings.

With the development of recurrent neural networks for natural language processing SMILES [19], a string representation of

978-0-7381-3366-9/21/$31.00 ©2021 IEEE

a molecular graph, became a frequent choice for both molecular property prediction [20] and molecule generation [21]–[24].

Currently, the neural representations of molecules are displacing molecular fingerprints as graph neural networks can learn a molecular representation that is tailored to the prediction task. The problem of the selection of atomic representation remains relevant also for these state-of-the-art methods as they require atom features and molecular graph topology at the input. The atomic representations diverge, starting from the first works on GCNs. For example, Kearnes *et al.* [25] use atom types, chirality, formal and partial charge, ring sizes, hybridization, hydrogen bonding, and aromaticity. Gilmer *et al.* [3] use one-hot encoding of the atom type alone, whereas Coley *et al.* [2] only encode 10 most common atom types along with the number of atom heavy neighbours, the number of hydrogen neighbours, aromaticity, formal charge and inclusion in a ring. Liu *et al.* [26] expand the one-hot representation to 23 most common atom types and add information about vdW and covalent radius of the atom. However, they do not use information about atom neighbourhood. Yang *et al.* [5] extend one-hot encoding to 100 dimensions and add information about atom's chirality, atomic mass, hybridization and number of bonds the atom is involved in. Moreover, some of the models additionally use bond vector representations.

A huge diversity of atomic representations makes it difficult to compare performance between different models. One must take into consideration that the differences in performance might arise not only from the choices concerning the architecture, but also the representation being used. To the best of our knowledge, there does not exist any extensive study of atomic representations in graph neural networks, so currently the choice of the used atomic features is subjective. In this study, we indicate the importance of the atomic features and measure their impact on the graph-based models performance.

III. Data and Methods

In this section, we first describe the representations used in our experiments and follow with the details of the GCN architecture and the model selection method. Next, we briefly summarise the statistical methods used to anylyse the results and finally, we describe the datasets chosen for evaluation.

A. Atom representations

We represent a molecule with N atoms as an undirected graph $\mathcal{G} = (X, A)$, where $X \in \mathbb{R}^{N \times D}$ is the atomic representation matrix, $A \in \mathbb{R}^{N \times N}$ is the graph adjacency matrix, and D is the number of atomic features. We chose five commonly used atom features: the atom type (one-hot encoded atom symbol), the number of heavy (non-hydrogen) atom neighbours, the number of attached hydrogens, formal charge, inclusion in a ring, and aromaticity. We consider 4 representation groups:

- using all the atomic features,
- using only one-hot encoded atom types,
- using exactly one atomic feature besides the atom type,
- using all atomic features but one.

The details are given in Table I.

TABLE I: Features included in each of the 12 atom representations.

	1	2	3	4	5	6	7	8	9	10	11	12
atom type	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
neighbors	✓		✓						✓	✓	✓	✓
hydrogens	✓			✓				✓		✓	✓	✓
formal charge	✓				✓			✓	✓		✓	✓
in ring	✓					✓		✓	✓	✓		✓
aromatic	✓						✓	✓	✓	✓	✓	

B. Model

We use graph neural network implementation based on [13]. Namely, we use the following graph convolution formula:

$$H^{(l+1)} = D^{-\frac{1}{2}} \hat{A} D^{-\frac{1}{2}} H^{(l)} W^{(l)}, \tag{1}$$

where $\hat{A} = A + I$ is the graph adjacency matrix including self-loops, $D_{ii} = \sum_j \hat{A}_{ij}$, $H^{(l)}$ is the node representation matrix in the l-th layer, and $W^{(l)}$ is a trainable weight matrix. The node representation at the input to the first layer is the atomic representation matrix ($H^{(0)} = X$).

a) Model selection: The best performing architectures were found using random search. All neural networks consist of graph convolutional layers followed by dense layers, and vary by: number of convolutional layers, number of channels in each convolutional layer, number of dense layers, size of dense layers, dropout, batchnorm, learning rate, batch size, and learning rate scheduler. The number of channels in convolutional layers and the size of hidden layers are equal in all models. A detailed description of the hyperparameter space can be found in Table II. All models were trained for 750 epochs using Adam and MSE loss.

TABLE II: Hyperparameters considered in our experiments

hyperparameter	values considered
number of conv. layers	1, 3, 5
number of channels in conv. layers	16, 64, 256
number of dense layers	1, 3
size of dense layers	16, 64, 256
dropout	0.0, 0.2
batchnorm	True, False
batchsize	8, 32, 128
learning rate	.01, .001, .0001, .00001, .000001
scheduler	no scheduler, decrease after 50% of epochs, decrease after 80% of epochs

We use the same set of 100 randomly sampled hyperparameter configurations for all the datasets. Each architecture was trained three times to accommodate for variance resulting from random initialisation.

C. Statistical methods

To compare atom features, we picked the best architecture found in random search for each representation. We performed one- and two-tailed Wilcoxon tests with Bonferroni correction to analyze the differences between representations.

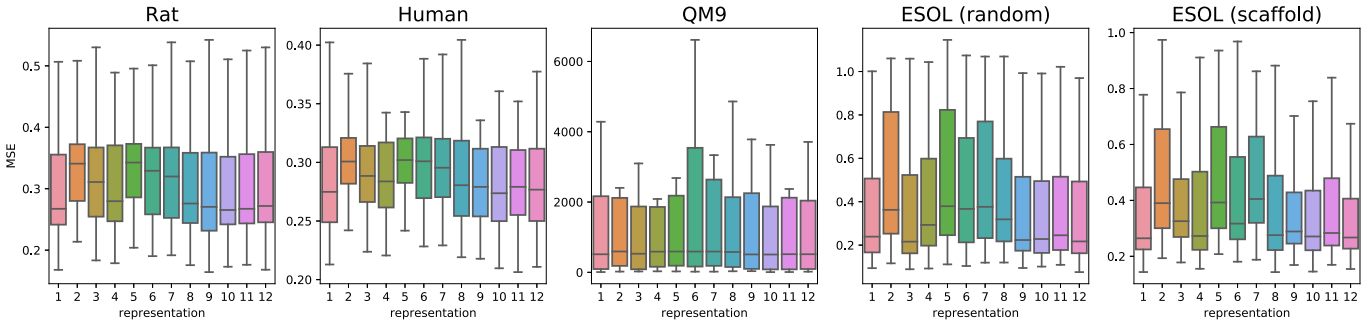


Fig. 1: Distribution of mean square error on the test set of all models trained with the selected representation.

TABLE III: Average test mean squared error of all models trained with different representations (no model selection).

representation	rat	human	qm9-random	esol-random	esol-scaffold
1	0.301 ± 0.102	0.291 ± 0.090	34100 ± 65370	0.39 ± 0.29	0.37 ± 0.21
2	0.339 ± 0.087	0.314 ± 0.075	35990 ± 65780	0.49 ± 0.29	0.47 ± 0.21
3	0.322 ± 0.101	0.303 ± 0.085	34820 ± 65340	**0.37 ± 0.29**	0.41 ± 0.19
4	0.311 ± 0.104	0.299 ± 0.073	35140 ± 65300	0.42 ± 0.29	0.38 ± 0.22
5	0.342 ± 0.092	0.315 ± 0.078	35740 ± 65670	0.49 ± 0.30	0.47 ± 0.20
6	0.329 ± 0.101	0.310 ± 0.083	35210 ± 65410	0.46 ± 0.29	0.42 ± 0.21
7	0.326 ± 0.100	0.309 ± 0.087	35930 ± 65640	0.48 ± 0.28	0.47 ± 0.19
8	0.303 ± 0.098	0.295 ± 0.083	34460 ± 65300	0.44 ± 0.28	0.37 ± 0.22
9	0.299 ± 0.104	0.295 ± 0.083	34670 ± 65290	0.38 ± 0.28	0.38 ± 0.20
10	**0.297 ± 0.093**	**0.289 ± 0.078**	**34040 ± 65270**	0.38 ± 0.29	**0.36 ± 0.21**
11	0.300 ± 0.097	0.294 ± 0.084	34250 ± 65360	0.39 ± 0.28	0.37 ± 0.20
12	0.302 ± 0.099	0.290 ± 0.080	34060 ± 65340	0.38 ± 0.29	**0.36 ± 0.20**

D. Datasets

For evaluation we chose four datasets that represent a wide range of molecular property prediction tasks. For the ESOL dataset, we use two different methods of splitting the data, random split and scaffold split [27], to examine if the choice of the splitting method affects the performance of models trained with different representations. The datasets used in our experiments are:

a) QM9: a dataset for predicting quantum properties [28]. We randomly sampled 5K molecules for training, 1k molecules for validation, and 10% of the dataset (13K molecules) for the test set. The models were trained to predict g298 (Free energy at 298.15 K (unit: Hartree)).

b) ESOL: a water solubility prediction dataset of 1128 samples [29]. We report results on both random split from [7] and 80-10-10 scaffold split.

c) HUMAN and RAT: datasets for metabolic stability prediction from [30]. Only records with the source being 'Liver', 'Liver microsome', or 'Liver microsomes' were used, resulting in 3578 and 1819 samples, respectively. In case of multiple measurements for the same molecule, the median of the measurements is used. The stability values are expressed in hours and log scaled. 10% of data was left out for testing and the remaining samples were divided into 5 cross-validation folds using random stratified split.

IV. Results

A. Quantitative analysis

In Figure 1, we compare the performance of models trained with different representations. Datasets and representations are on the x-axis and on the y-axis the distribution of mean square error on the test set of all models trained with the selected representation.

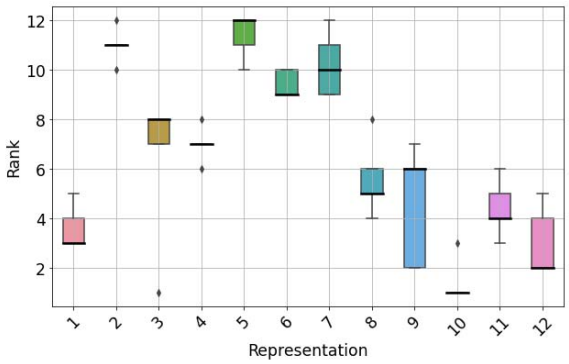


Fig. 2: Rankings obtained for every given representation on all datasets. The median ranking is marked with a bold line.

In Table III, one can see detailed results and Figure 2 presents the box plot with rankings obtained by the representations on all datasets. The best scores are obtained by models trained with representation 10 (no formal charge), which beat models trained with other representations in 4 out of 5 tasks.

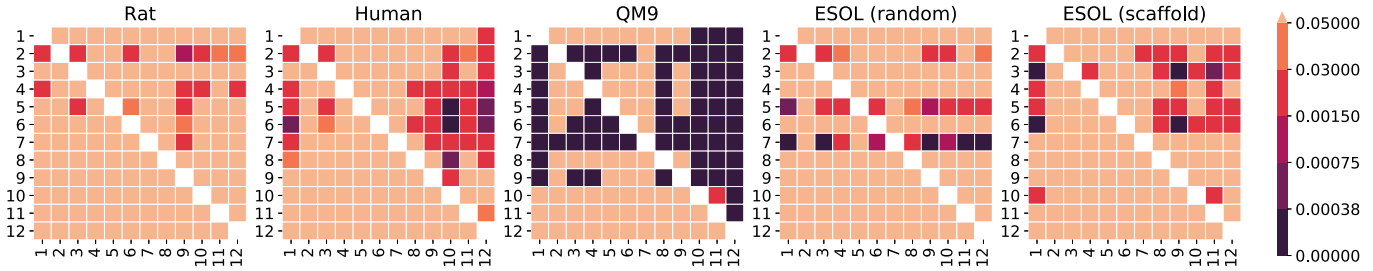
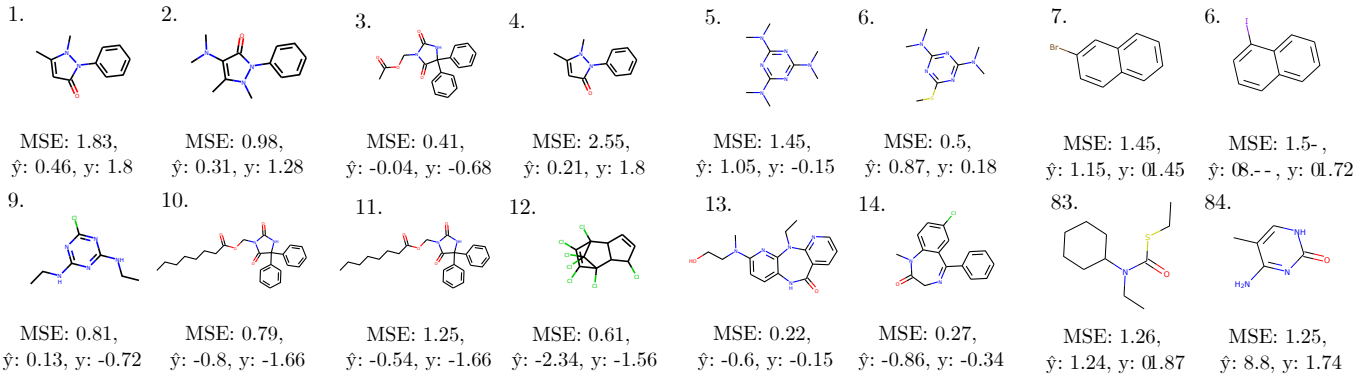


Fig. 3: P-values of one-tailed Wilcoxon tests between the best models trained on each representation. The value in i-th row and j-th column corresponds to the alternative hypothesis saying that the median squared error of i-th representation is greater than the median of j-th representation (superior representations have darker columns, and inferior ones have darker rows). The darkest cells are statistically significant with Bonferroni correction.



(a) overall (b) repr-2 (empty) (c) repr-11 (no rings) (d) repr-12 (no aromatic)

Fig. 4: The worst predicted molecules in the ESOL (scaffold) dataset. Plots show compounds with the highest MSE in all representations (a), and MSE higher than in other representations (b-d); $\hat{y}$ is the average predicted value, and y is the true value (standardised).

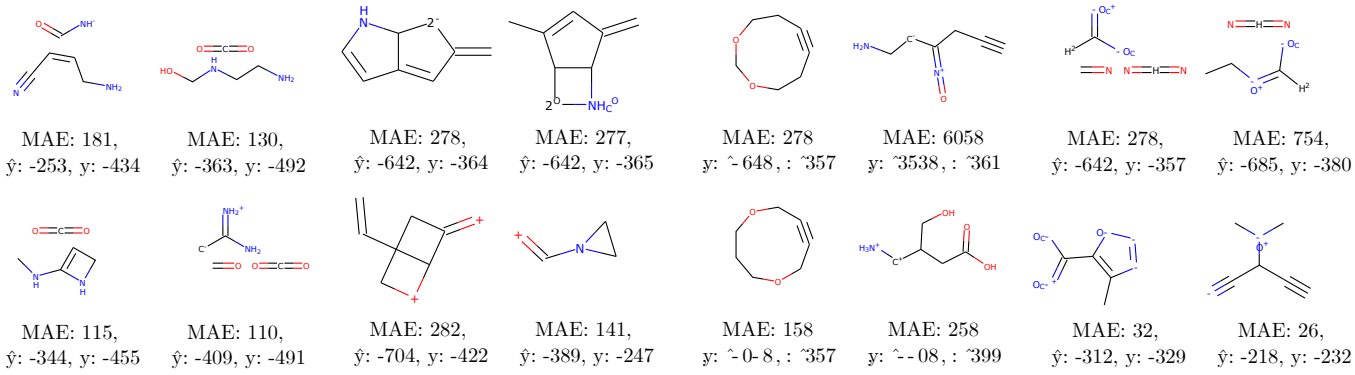
In order to systematically study the error distributions, we ran Wilcoxon tests for pairwise representation comparisons. The p-values of one-sided tests are plotted in Figure 3. We observe that many representations are equivalent even before applying the Bonferroni correction ($p \geq 0.05$), e.g. in RAT the lowest p-value is above the level of significance ($p \geq 0.002$ in a two-tailed Wilcoxon test, while the significant differences should be below 0.05 / 66 pairwise tests). The differences between representations are most apparent in QM9, which is the biggest dataset in the comparison ($p \leq {}^{0.05}/_{66}$ in all two-tailed Wilcoxon tests besides the ones between representations 3-5, 5-9, and 10-11).

There are several patterns that can be noted in the heatmaps.

1) Atom representations with almost full set of features are usually comparable with each other (bright area in the bottom left corner) and better than nearly empty feature vectors (dark area in the top right corner).
2) There are features that perform significantly worse than others when used alone, e.g. including only aromaticity (repr. 7) yields almost as poor results as using no atom features in QM9 and ESOL with a random split. On the other hand, adding information about heavy neighbors and hydrogens (repr. 3 and 4) gives the biggest performance boost across all datasets.
3) Removing features related to aromaticity (repr. 12), inclusion in a ring (repr. 11), and formal charges (repr. 10) can improve model quality, compared with the full representation (repr. 1).

B. *Qualitative analysis*

Figure 4 shows molecules with the highest mean errors of solubility prediction for all representations jointly and for three selected ones. To pick molecules that are predicted worse by a given representation, we calculated a margin between the mean error of this representation and the highest mean error of the remaining representations. To put it more precisely, for each compound we calculate predictions using the best model for each representation $\hat{y}_1, \ldots, \hat{y}_{12}$ and compare these predictions with the true label y. Next, we sort the compounds by the



(a) overall (b) repr-4 (hydrogens) (c) repr-8 (no neighbors) (d) repr-9 (no hydrogens)

Fig. 5: The worst predicted molecules in the QM9 dataset. Plots show compounds with the highest MAE in all representations (a) and MAE higher than in other representations (b-d); $\hat{y}$ is the average predicted value, and y is the true value (standardized).

following value:

$$m_i = \max\left(0, \varepsilon(y - \hat{y}_i) - \max_{\substack{j=1,\ldots,12 \\ i \neq j}} \varepsilon(y - \hat{y}_j)\right), \quad (2)$$

where $\varepsilon : \mathbb{R} \to \mathbb{R}_+$ is an error function (e.g. MSE or MAE), and m_i is the error margin of the compound for the i-th representation.

We observe that using only the topological graph information and no atom features besides the atom type produces similar structures to those that are on average worst predicted by all representations. For instance, the molecule with a long aliphatic chain (molecule 11) is predicted as more soluble probably because the model with no atom features cannot differentiate between saturated and unsaturated chains. Similarly, the compound with a cyclohexane ring (molecule 15) could be predicted as more soluble due to the lack of aromaticity information – the aromatic counterpart of the cyclohexane, a benzene, is more soluble in water. Also, we note that the representation without information about ring inclusion often makes mistakes for compounds with non-aromatic rings or nitrogens in rings.

Similar results for QM9 dataset can be found in Figure 5. In these selected representations we again observe recurring patterns. For example, representation 9, which does not contain information about attached hydrogens, poorly predicts compounds with nitrogen cations. Similarly, representation 8, which misses the information about the number of heavy neighbours, obtains the highest error values for branched structures with carbocations or carbanions.

To confirm that some representations are more prone to committing errors where certain patterns appear in the molecular structure, in Figure 6 we plotted a t-SNE map of QM9 compounds. For each representation, we selected at most 500 compounds with the highest error margin over other representations in the testing set. Additionally, the error margin was averaged over top 5 models for each representation

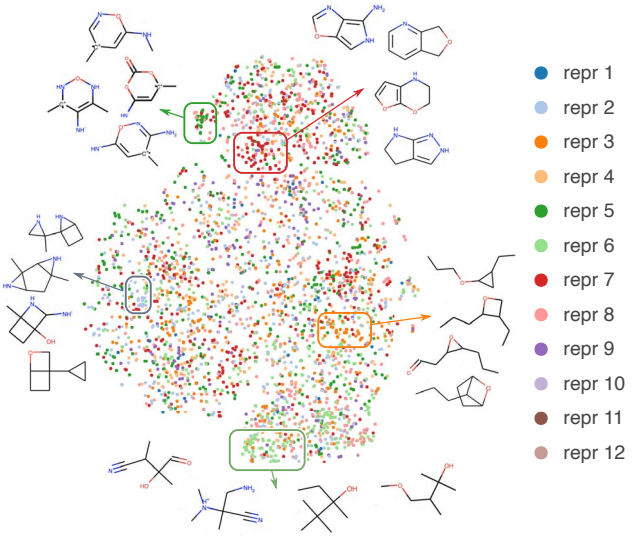


Fig. 6: t-SNE map of QM9 compounds coloured by the representation with the highest prediction error (MAE). The algorithm uses the ECFP fingerprints and Tanimoto distance.

to make the resulting map less dependent on the single training runs. Next, we calculated ECFP fingerprints to encode chemical structure of the molecules. This representation was used as an input to the t-SNE algorithm [14] with Tanimoto metric used for calculating distances in the fingerprint space. In the plot, each colour corresponds to the representation with the highest prediction error. To find compound clusters, we used the DBSCAN clustering algorithm [31] with $\epsilon = 4$. In Figure 6, we plotted 5 out of 27 found clusters along with 4 compounds sampled from each of them.

We observe that small clusters of one colour form in the t-SNE map. These clusters correspond to structural motifs that confuse the model which was trained with the given representation. This observation suggests that some repre-

sentations fail to predict certain structural patterns due to inductive biases. For example, representation 6 (only inclusion in rings) tends to make errors for the structures with many branches and no rings (depicted at the bottom of Figure 6). Interestingly, representation 5 incorrectly predicts a group of compounds with a carbocation in a ring even though it contains information about formal charge. A similar cluster corresponds to representation 7 (only aromaticity) that makes mistakes predicting fused bicyclic compounds which are partly aromatic.

C. Literature representations

The goal of this section is to put our analysis in a wider context by comparing different representations present in literature when used with the same architecture. To this end we repeat our experiments with additional representations, namely:

L1) 33 dimensional representation from [26].
L2) 38 dimensional representation from [32].
L3) 58 dimensional representation from [1].
L4) 127 dimensional representation from [5].

We also note that representation 1 (full) from the previous section was used in [2]. The details of all representations are given in Table IV. All of these representations include information whether an atom is in an aromatic system, and the type of the atom – though the number of encoded atomic types ranges from 12 in representations 1 and L2 up to 100 in representation L4. Moreover, most of the representations include information about atom's (heavy) neighbours and formal charge.

For each representation, we report the validation and test results of the model with the best result on the validation set. The results are presented in Table V and rank-plots are depicted in Figure 7. Additionally, we include results of representation 10 (no formal charge) which on average performed best in the previous section. All experiments were conducted using the same settings as in the previous section.

Representation 1 obtained the best test scores on almost all of the datasets despite having the smallest size among the literature representations under analysis. More interestingly, it also consistently achieves a much lower generalisation gap than the remaining literature representations. Representation 10, which differs only by dismissing information about formal charge, consistently achieves similar but poorer results, and the generalisation gap also slightly increases. We see this as an indication of good generalisation properties of this representation.

Representations L1 and L2 despite being similar in size to representation 1 (respectively, 33 and 38 versus 26) usually achieve weaker results and with a much higher generalisation gap. Therefore, we conclude that the generalisation properties cannot be attributed solely to the size of the representation.

Not surprisingly the results on ESOL with scaffold split are consistently lower than on ESOL random and with a higher generalisation gap which should be attributed to the method of the data split.

TABLE IV: Different featurisation methods used in the literature.

representation 1 [2]	
size	description
12	one-hot vector specifying the type of atom
6	number of heavy neighbours as one-hot vector
5	number of hydrogen atoms as one-hot vector
1	formal charge
1	is in a ring
1	is in aromatic system

representation L1 [26]	
size	description
23	one-hot vector specifying the type of atom
2	vdW radius and covalent radius of the atom
6	for each size of ring (3-8) the number of rings that include this atom
1	is in aromatic system
1	electrostatic charge of this atom

representation L2 [32]	
size	description
12	one-hot vector specifying the type of atom
6	number of heavy neighbours as one-hot vector
5	number of hydrogen atoms as one-hot vector
6	implicit valence as one-hot vector
1	is in aromatic system
1	number of radical electrons
5	hybridisation type as one-hot vector
1	formal charge
1	Gasteiger partial charge

representation L3 [1]	
size	description
44	one-hot vector specifying the type of atom
6	number of heavy neighbours as one-hot vector
5	number of hydrogen atoms as one-hot vector
6	implicit valence as one-hot vector
1	is in aromatic system

representation L4 [5]	
size	description
100	one-hot vector specifying the type of atom
6	number of bonds the atom is involved in as one-hot vector
5	formal charge as one-hot vector
4	chirality as one-hot vector
5	number of hydrogen atoms as one-hot vector
5	hybridisation type as one-hot vector
1	is in aromatic system
1	atomic mass

Overall, we see that a single representation can consistently outperform others on multiple tasks though usually the differences are not strongly emphasised.

V. Conclusions

In this study, we examine the influence of atomic representations on the predictive performance of graph neural networks. We show that the choice of atom features used in the representation results in an improved or reduced performance of the trained models and we confirm the significance of the arising differences by one-tailed Wilcoxon test. The differences are most pronounced in case of the QM9 dataset.

The presented results indicate that the choice of atom features is task-specific though some general conclusions can be drawn.

TABLE V: Average valid and test mean squared error of models trained with different representations from literature. Best mean test results are in bold.

	Rat		Human		QM9		ESOL (scaffold)		ESOL (random)	
Representation	valid	test	valid	test	valid	test	valid	test	valid	test
10	$.254 \pm .02$	$.247 \pm .03$	$.134 \pm .01$	$\mathbf{.208 \pm .01}$	$2.667 \pm .76$	9.698 ± 1.47	$.110 \pm .01$	$.201 \pm .01$	$.085 \pm .01$	$.123 \pm .01$
1 [2]	$.226 \pm .02$	$\mathbf{.214 \pm .01}$	$.135 \pm .00$	$.219 \pm .01$	$4.531 \pm .19$	$\mathbf{9.193 \pm 1.18}$	$.107 \pm .01$	$\mathbf{.166 \pm .02}$	$.086 \pm .01$	$\mathbf{.115 \pm .01}$
L1 [26]	$.221 \pm .02$	$.286 \pm .02$	$.146 \pm .00$	$.229 \pm .01$	$2.813 \pm .77$	20.544 ± 6.85	$.111 \pm .01$	$.238 \pm .02$	$.078 \pm .01$	$.133 \pm .01$
L2 [32]	$.202 \pm .01$	$.239 \pm .02$	$.147 \pm .00$	$.215 \pm .01$	$2.087 \pm .16$	17.52 ± 4.74	$.115 \pm .01$	$.216 \pm .02$	$.073 \pm .01$	$.141 \pm .01$
L3 [1]	$.204 \pm .02$	$\mathbf{.214 \pm .02}$	$.147 \pm .00$	$.228 \pm .01$	3.308 ± 1.19	30.036 ± 3.21	$.103 \pm .01$	$.225 \pm .02$	$.076 \pm .01$	$.117 \pm .01$
L4 [5]	$.207 \pm .04$	$.224 \pm .02$	$.146 \pm .00$	$.224 \pm .01$	3.831 ± 1.18	25.967 ± 3.54	$.105 \pm .01$	$.223 \pm .02$	$.078 \pm .01$	$.134 \pm .01$

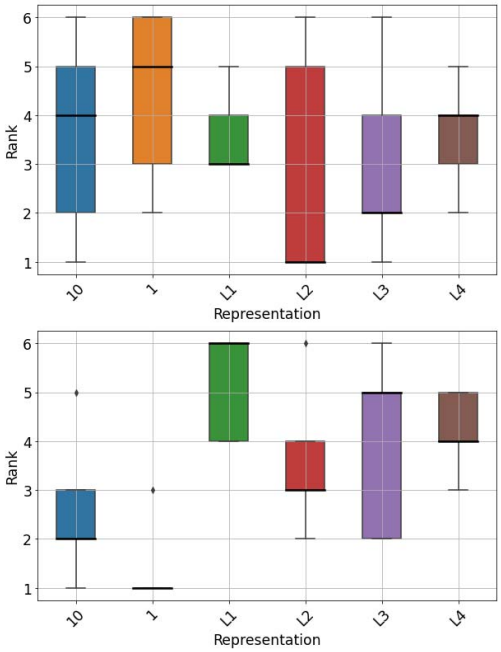


Fig. 7: Rankings obtained for each representation on all datasets for validation (left) and test (right) datasets. The median ranking is marked with a bold line.

Not surprisingly representations with more features tend to give better results. However, removing features related to aromaticity, inclusion in a ring, and formal charges can improve the model quality. On the other hand, adding information about heavy neighbours and hydrogens gives the biggest performance boost across all datasets.

The qualitative analysis suggests that the committed errors can be attributed to the absence of information about atom features which were not included in the model's representation.

Comparing representations used in literature reveals that a single representation can consistently outperform others though the differences are not strongly emphasised. We also take a look at generalisation properties of selected representations and conclude that they cannot be attributed solely to the size of the representation.

To the best of our knowledge, this is the first methodological study that focuses on the relevance of atom representation to the predictive performance of graph neural networks.

REFERENCES

[1] D. K. Duvenaud, D. Maclaurin, J. Iparraguirre, R. Bombarell, T. Hirzel, A. Aspuru-Guzik, and R. P. Adams, "Convolutional networks on graphs for learning molecular fingerprints," in *Advances in neural information processing systems*, 2015, pp. 2224–2232.

[2] C. W. Coley, R. Barzilay, W. H. Green, T. S. Jaakkola, and K. F. Jensen, "Convolutional embedding of attributed molecular graphs for physical property prediction," *Journal of chemical information and modeling*, vol. 57, no. 8, pp. 1757–1772, 2017.

[3] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," *arXiv preprint arXiv:1704.01212*, 2017.

[4] K. T. Schütt, H. E. Sauceda, P.-J. Kindermans, A. Tkatchenko, and K.-R. Müller, "Schnet–a deep learning architecture for molecules and materials," *The Journal of Chemical Physics*, vol. 148, no. 24, p. 241722, 2018.

[5] K. Yang, K. Swanson, W. Jin, C. Coley, P. Eiden, H. Gao, A. Guzman-Perez, T. Hopper, B. Kelley, M. Mathea *et al.*, "Analyzing learned molecular representations for property prediction," *Journal of chemical information and modeling*, vol. 59, no. 8, pp. 3370–3388, 2019.

[6] J. Klicpera, J. Groß, and S. Günnemann, "Directional message passing for molecular graphs," *arXiv preprint arXiv:2003.03123*, 2020.

[7] Ł. Maziarka, T. Danel, S. Mucha, K. Rataj, J. Tabor, and S. Jastrzębski, "Molecule attention transformer," *arXiv preprint arXiv:2002.08264*, 2020.

[8] T. Danel, P. Spurek, J. Tabor, M. Śmieja, Ł. Struski, A. Słowik, and Ł. Maziarka, "Spatial graph convolutional networks," in *International Conference on Neural Information Processing*. Springer, 2020, pp. 668–675.

[9] Y. Rong, Y. Bian, T. Xu, W. Xie, Y. Wei, W. Huang, and J. Huang, "Self-supervised graph transformer on large-scale molecular data," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[10] Y. Song, S. Zheng, Z. Niu, Z.-H. Fu, Y. Lu, and Y. Yang, "Communicative representation learning on attributed molecular graphs," in *IJCAI*, 2020.

[11] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.

[12] J. Li, D. Cai, and X. He, "Learning graph-level representation for drug discovery," *arXiv preprint arXiv:1709.03741*, 2017.

[13] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[14] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008.

[15] H. Kubinyi, "Qsar and 3d qsar in drug design part 1: methodology," *Drug discovery today*, vol. 2, no. 11, pp. 457–467, 1997.

[16] D. Rogers and M. Hahn, "Extended-connectivity fingerprints," *Journal of chemical information and modeling*, vol. 50, no. 5, pp. 742–754, 2010.

[17] A. L. Perryman, T. P. Stratton, S. Ekins, and J. S. Freundlich, "Predicting mouse liver microsomal stability with "pruned" machine learning models and public data," *Pharmaceutical research*, vol. 33, no. 2, pp. 433–449, 2016.

[18] O. Laufkötter, N. Sturm, J. Bajorath, H. Chen, and O. Engkvist, "Combining structural and bioactivity-based fingerprints improves prediction performance and scaffold hopping capability," *Journal of cheminformatics*, vol. 11, no. 1, pp. 1–14, 2019.

[19] D. Weininger, "Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules," *Journal of Chemical Information and Computer Sciences*, vol. 28, no. 1, pp. 31–36, 1988. [Online]. Available: https://pubs.acs.org/doi/abs/10.1021/ci00057a005

[20] S. Jastrzębski, D. Leśniak, and W. M. Czarnecki, "Learning to smile (s)," *arXiv preprint arXiv:1602.06289*, 2016.

[21] M. J. Kusner, B. Paige, and J. M. Hernández-Lobato, "Grammar variational autoencoder," in *International Conference on Machine Learning*. PMLR, 2017, pp. 1945–1954.
[22] M. Olivecrona, T. Blaschke, O. Engkvist, and H. Chen, "Molecular de-novo design through deep reinforcement learning," *Journal of cheminformatics*, vol. 9, no. 1, pp. 1–14, 2017.
[23] M. Popova, O. Isayev, and A. Tropsha, "Deep reinforcement learning for de novo drug design," *Science advances*, vol. 4, no. 7, p. eaap7885, 2018.
[24] R. Gómez-Bombarelli, J. N. Wei, D. Duvenaud, J. M. Hernández-Lobato, B. Sánchez-Lengeling, D. Sheberla, J. Aguilera-Iparraguirre, T. D. Hirzel, R. P. Adams, and A. Aspuru-Guzik, "Automatic chemical design using a data-driven continuous representation of molecules," *ACS central science*, vol. 4, no. 2, pp. 268–276, 2018.
[25] S. Kearnes, K. McCloskey, M. Berndl, V. Pande, and P. Riley, "Molecular graph convolutions: moving beyond fingerprints," *J. Comput. Aid. Mol. Des.*, vol. 30, no. 8, pp. 595–608, 2016.
[26] K. Liu, X. Sun, L. Jia, J. Ma, H. Xing, J. Wu, H. Gao, Y. Sun, F. Boulnois, and J. Fan, "Chemi-net: a molecular graph convolutional network for accurate drug property prediction," *International journal of molecular sciences*, vol. 20, no. 14, p. 3389, 2019.
[27] G. W. Bemis and M. A. Murcko, "The properties of known drugs. 1. molecular frameworks," *Journal of medicinal chemistry*, vol. 39, no. 15, pp. 2887–2893, 1996.
[28] L. Ruddigkeit, R. Van Deursen, L. C. Blum, and J.-L. Reymond, "Enumeration of 166 billion organic small molecules in the chemical universe database gdb-17," *Journal of chemical information and modeling*, vol. 52, no. 11, pp. 2864–2875, 2012.
[29] J. S. Delaney, "Esol: estimating aqueous solubility directly from molecular structure," *Journal of chemical information and computer sciences*, vol. 44, no. 3, pp. 1000–1005, 2004.
[30] S. Podlewska and R. Kafel, "Metstabon—online platform for metabolic stability predictions," *International journal of molecular sciences*, vol. 19, no. 4, p. 1040, 2018.
[31] M. Ester, H.-P. Kriegel, J. Sander, X. Xu *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise." in *Kdd*, vol. 96, no. 34, 1996, pp. 226–231.
[32] X. Li, X. Yan, Q. Gu, H. Zhou, D. Wu, and J. Xu, "Deepchemstable: Chemical stability prediction with an attention-based graph convolution network," *Journal of chemical information and modeling*, vol. 59, no. 3, pp. 1044–1049, 2019.