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

Agnieszka Pocha[1], **Tomasz Danel**[1,2], **Łukasz Maziarka**[1,2]
[1] Jagiellonian University, [2] Ardigen
agnieszka.pocha@doctoral.uj.edu.pl

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.

1 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 the chemical bonds. Since graph-based models were shown to outperform models based on molecular fingerprints [Duvenaud et al., 2015], the interest in GCNs increased, which resulted in proposing new models for molecular property prediction [Coley et al., 2017, Gilmer et al., 2017, Schütt et al., 2018, Yang et al., 2019, Klicpera et al., 2020, Maziarka et al., 2020].

The community's focus is on developing new methods, ignoring the issue what atomic representation is used. Therefore, the atomic representations vary between different models. For instance, Gilmer et al. [2017] use one-hot encoding of atom type alone, whereas Coley et al. [2017] only encode the 10 most common atom types with one-hot vectors but add information about atom's number of heavy neighbours, number of hydrogen neighbors, aromaticity, formal charge and whether the atom is in a ring. Liu et al. [2019] 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. [2019] 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 as one must take into consideration that the differences in performance might arise not only from the choices concerning the architecture but also the choices about the representation being used.

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 vanilla GCNs and evaluate them on multiple datasets. We analyse the obtained results and show which molecules are most difficult to predict for GCNs with a given representation. We use MACCS keys [Durant et al., 2002], to analyze the substructures which presence in molecules leads to difficulties in their property prediction by GCN and examine whether these patterns are shared by models which use different representations.

Machine Learning for Molecules Workshop at NeurIPS 2020. https://ml4molecules.github.io

2 Data and Methods

In this section, we first describe representations used in our experiments. Next, we share details on the GCN architecture and model selection method. Finally, we describe the datasets chosen for evaluation.

2.1 Atom representations.

We have chosen five commonly used atom features and considered graph representations that used all, none, exactly one or exactly four of them. All representations additionally use one-hot encoded atom types. The details are given in Table 1.

Table 1: 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	✓						✓	✓	✓	✓	✓	

2.2 Model

We used graph neural network implementation based on Kipf and Welling [2016]. The best performing architectures were found using random search. All the neural networks consisted of graph convolutional layers followed by dense layers, and varied by: number of convolutional layers, number of channels in each convolutional layer, number of dense layers, size of dense layers, dropout, batch-norm, learning rate, batch size, and learning rate scheduler. Number of channels in convolutional layers and the size of hidden layers were equal for all the models. The detailed description of the hyperparameter space can be found in the appendix A. The models were trained for 750 epochs using Adam and MSE loss.

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

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.

2.3 Datasets

For evaluation we have chosen 4 datasets that represent a wide range of molecular property prediction tasks. Moreover, for ESOL dataset we used two different types of splitting the data, namely random split and scaffold split [Bemis and Murcko, 1996], to examine if the choice of splitting method affects the performance of models trained with different representations.

QM9 is a dataset for predicting quantum properties. 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 298.15K (unit: Hartree)).

ESOL is a water solubility prediction dataset. We report results on both random and scaffold split.

HUMAN and **RAT** are datasets for metabolic stability prediction from Podlewska and Kafel [2018]. 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 was used. The stability values were expressed in hours

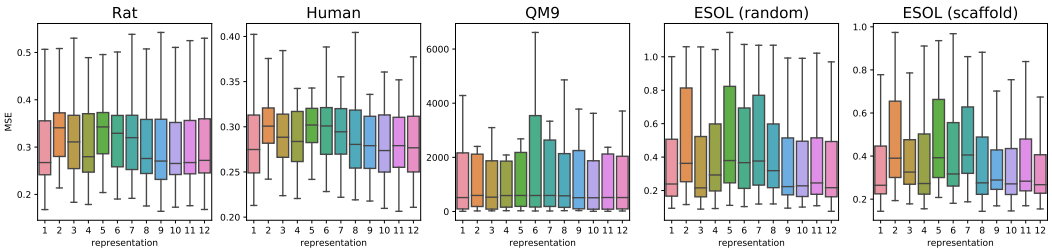


Figure 1: Distribution of mean square error on the test set of all models trained with the selected representation.

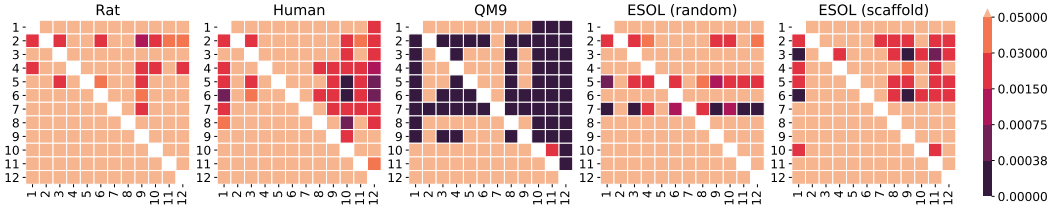


Figure 2: P-values of one-tailed Wilcoxon tests between the best models trained on each representation. The value in i-th row and j-th column corresponds to the alternative hypothesis saying that the median squared error of i-th representation is greater than the median of j-th representation (superior representations have darker columns, and inferior ones have darker rows). The darkest cells are statistically significant with Bonferroni correction.

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.

3 Results

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. The exact values obtained by all models can be found in the appendix B in Table 3 and the rank plot for all representations in figure 5.

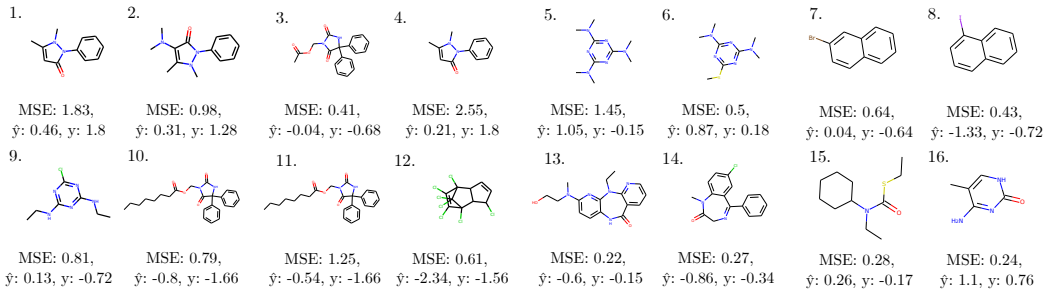
In order to systematically study the error distributions, we ran Wilcoxon tests for pairwise representation comparisons. The p-values of one-sided tests are plotted in Figure 2. We observe that many representations are equivalent even before applying the Bonferroni correction ($p \geq 0.05$), e.g. in 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). 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. First, 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). Second, 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. Third, 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).

Qualitative analysis. Figure 3 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 the given representation, we calculated margin between the mean error of this representation and the highest mean error of the remaining representations.

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 the information about ring inclusion often makes mistakes for the compounds with non-aromatic rings or nitrogens in rings.



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

Figure 3: 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 (standardized).

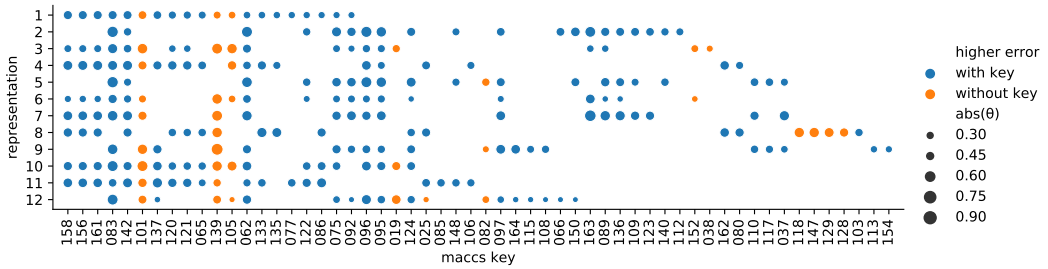


Figure 4: θ values of 20 keys with highest absolute value for each representation separately on ESOL (scaffold) test set molecules.

To examine whether molecular substructures can be connected to higher or lower errors made by models trained with specific representations we picked the best architecture found in random search for each representation and calculated MACCS representation of the test set molecules. Afterwards, for each representation r and for each MACCS key k we calculated $\theta_k^r = \frac{MSE_{k_0}^r - MSE_{k_1}^r}{max(MSE_{k_0}^r, MSE_{k_1}^r)}$, $MSE_{k_i}^r$ being an average mean squared error on molecules containing substructure encoded by key k ($MSE_{k_1}^r$) or not containing it ($MSE_{k_0}^r$). Keys encoding substructures which were not present in at least 10% of the molecules were dropped from the analysis.

In Figure 4 we present θ values of 20 keys with the highest absolute θ for ESOL (scaffold). The SMARTS of the shown keys can be found in Table 4 in the appendix C. As can be seen, several keys are shared by most of the representations and with similar values of θ, suggesting that their presence or absence influences the model's error irrespectively of the representation used. However, there are several substructures that are more difficult for specific representations, e.g. representation 9 (no information about hydrogens) incorrectly predicts compounds containing oxygen (MACCS 164), which can be caused by different solubility of compounds with hydroxy (-OH) and carbonyl (=O) groups. Similar results for QM9 dataset can be found in the appendix C.

4 Conclusions

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

Acknowledgements

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

References

Guy W Bemis and Mark A Murcko. The properties of known drugs. 1. molecular frameworks. *Journal of medicinal chemistry*, 39(15):2887–2893, 1996.

Connor W Coley, Regina Barzilay, William H Green, Tommi S Jaakkola, and Klavs F Jensen. Convolutional embedding of attributed molecular graphs for physical property prediction. *Journal of chemical information and modeling*, 57(8):1757–1772, 2017.

Joseph L Durant, Burton A Leland, Douglas R Henry, and James G Nourse. Reoptimization of mdl keys for use in drug discovery. *Journal of chemical information and computer sciences*, 42(6): 1273–1280, 2002.

David K Duvenaud, Dougal Maclaurin, Jorge Iparraguirre, Rafael Bombarell, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P Adams. Convolutional networks on graphs for learning molecular fingerprints. In *Advances in neural information processing systems*, pages 2224–2232, 2015.

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. *arXiv preprint arXiv:1704.01212*, 2017.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Johannes Klicpera, Janek Groß, and Stephan Günnemann. Directional message passing for molecular graphs. *arXiv preprint arXiv:2003.03123*, 2020.

Ke Liu, Xiangyan Sun, Lei Jia, Jun Ma, Haoming Xing, Junqiu Wu, Hua Gao, Yax Sun, Florian Boulnois, and Jie Fan. Chemi-net: a molecular graph convolutional network for accurate drug property prediction. *International journal of molecular sciences*, 20(14):3389, 2019.

Łukasz Maziarka, Tomasz Danel, Sławomir Mucha, Krzysztof Rataj, Jacek Tabor, and Stanisław Jastrzębski. Molecule attention transformer. *arXiv preprint arXiv:2002.08264*, 2020.

Sabina Podlewska and Rafał Kafel. Metstabon—online platform for metabolic stability predictions. *International journal of molecular sciences*, 19(4):1040, 2018.

Kristof T Schütt, Huziel E Sauceda, P-J Kindermans, Alexandre Tkatchenko, and K-R Müller. Schnet–a deep learning architecture for molecules and materials. *The Journal of Chemical Physics*, 148(24):241722, 2018.

Kevin Yang, Kyle Swanson, Wengong Jin, Connor Coley, Philipp Eiden, Hua Gao, Angel Guzman-Perez, Timothy Hopper, Brian Kelley, Miriam Mathea, et al. Analyzing learned molecular representations for property prediction. *Journal of chemical information and modeling*, 59(8): 3370–3388, 2019.

A Experimental details

The details of the grid space are given in Table 2.

Table 2: Hyperparameters considered in our experiments

hyperparameter	values considered
number of convolutional layers	1, 3, 5
number of channels in convolutional 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 the epochs, decrease after 80% of the epochs

B Additional results for quantitative analysis

In Table 3 one can see detailed results for all datasets and all representations. Figure 5 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.

Table 3: Average test mean squared error of models trained with different representations.

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	$\mathbf{0.37 \pm 0.29}$	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	0.308 ± 0.084	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	$\mathbf{0.297 \pm 0.093}$	$\mathbf{0.289 \pm 0.078}$	$\mathbf{34040 \pm 65270}$	0.38 ± 0.29	$\mathbf{0.36 \pm 0.21}$
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	$\mathbf{0.36 \pm 0.20}$

C Additional results for qualitative analysis

In Table 4 we present SMARTS of the MACCS keys for ESOL (scaffold) dataset, shown in Figure 4.

Figure 6 shows molecules with the highest mean errors of predictions for QM9 dataset for all representations jointly and for three selected ones. In Figure 7 we present θ values of 20 keys with highest absolute θ for QM9 dataset. The SMARTS of the shown keys can be found in Table 5.

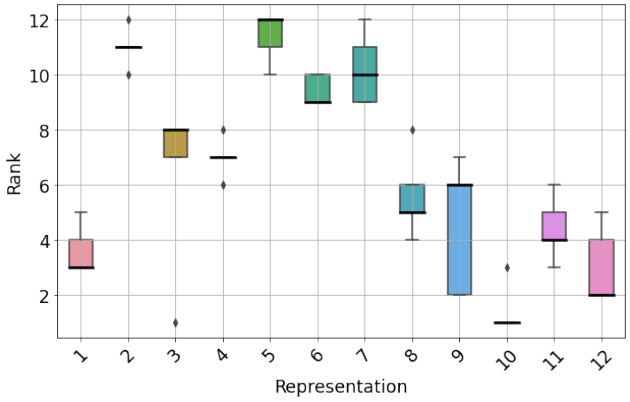
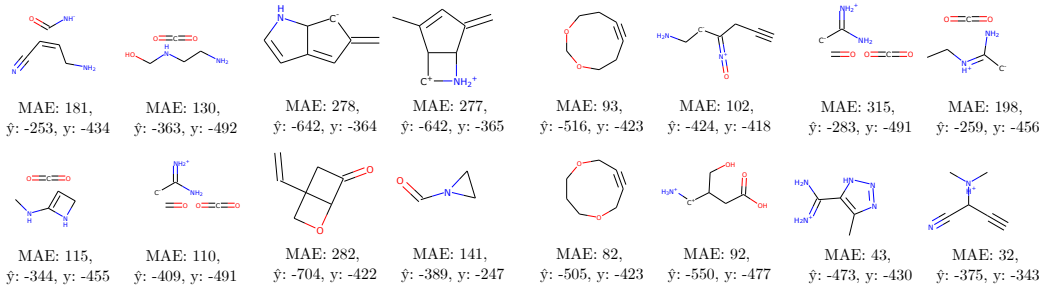


Figure 5: Rankings obtained for every given representation on all datasets. The median ranking is marked by a bold line.



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

Figure 6: 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).

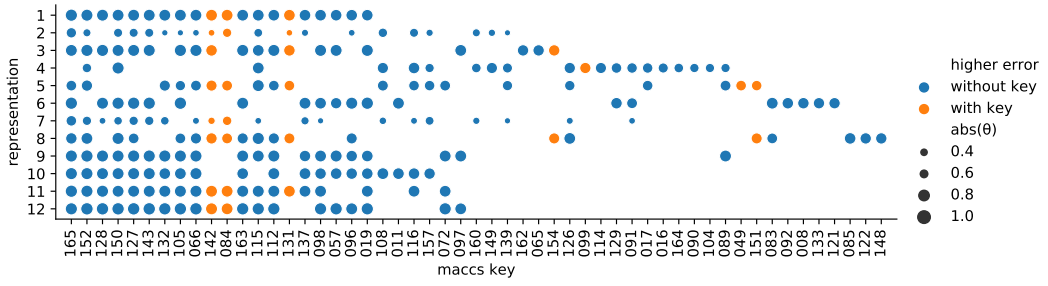


Figure 7: θ values of 20 keys with highest absolute value for each representation separately on QM9 test set molecules.

Table 4: SMARTS for MACCS keys shown in Figure 4.

key	SMARTS	key	SMARTS
19	*1~*~*~*~*~*~*~1	25	[#7]~[#6](~[#7])~[#7]
37	[#7]~[#6](~[#8])~[#7]	38	[#7]~[#6](~[#6])~[#7]
62	*@*!@*@*	65	c:n
66	[#6]~[#6](~[#6])(~[#6])~*	75	*!@[#7]@*
77	[#7]~*~[#7]	80	[#7]~*~*~*~[#7]
82	*~[CH2]~[!#6;!#1;!H0]	83	[!#6;!#1]1~*~*~*~*~1
85	[#6]~[#7](~[#6])~[#6]	86	[C;H2,H3][!#6;!#1][C;H2,H3]
89	[#8]~*~*~*~[#8]	92	[#8]~[#6](~[#7])~[#6]
95	[#7]~*~*~[#8]	96	*1~*~*~*~*~1
97	[#7]~*~*~*~[#8]	101	[$([R]@1@[R]@[R]@[R]@[R] @[R]@[R]@[R]1), $([R]@1@[R]@[R]@[R]@[R] @[R]@[R]@[R]@[R]1), $([R]@1@[R]@[R]@[R]@[R] @[R]@[R]@[R]@[R]@[R]1), $([R]@1@[R]@[R]@[R]@[R] @[R]@[R]@[R]@[R]@[R]@[R]1), $([R]@1@[R]@[R]@[R]@[R] @[R]@[R]@[R]@[R]@[R]@[R]@[R]1), $([R]@1@[R]@[R]@[R]@[R] @[R]@[R]@[R]@[R]@[R]@[R]@[R]@[R]1), $([R]@1@[R]@[R]@[R]@[R] @[R]@[R]@[R]@[R]@[R]@[R]@[R]@[R]@[R]1)]
103	Cl	105	*@*(@*)@*
106	[!#6;!#1]~*(~[!#6;!#1])~[!#6;!#1]	108	[CH3]~*~*~*~[CH2]~*
109	*~[CH2]~[#8]	110	[#7]~[#6]~[#8]
112	*~*(~*)(~*)~*	113	[#8]!:*:*
115	[CH3]~*~[CH2]~*	117	[#7]~*~[#8]
118	[$(*~[CH2]~[CH2]~*),$(*1~[CH2]~[CH2]1)]	120	[!#6;R]
121	[#7;R]	122	*~[#7](~*)~*
123	[#8]~[#6]~[#8]	124	[!#6;!#1]~[!#6;!#1]
128	[$(*~[CH2]~*~*~*~[CH2]~*), $([R]1@[CH2;R]@[R]@[R]@[R]@[CH2;R]1), $(*~[CH2]~[R]1@[R]@[R]@[CH2;R]1), $(*~[CH2]~*~[R]1@[R]@[CH2;R]1)]	129	[$(*~[CH2]~*~*~[CH2]~*), $([R]1@[CH2]@[R]@[R]@[CH2;R]1), $(*~[CH2]~[R]1@[R]@[CH2;R]1)]
133	*@*!@[#7]	135	[#7]!:*:*
136	[#8]=*	137	[!C;!c;R]
139	[O;!H0]	140	[#8]
142	[#7]	147	[$(*~[CH2]~[CH2]~*),$([R]1@[CH2;R]@[CH2;R]1)]
148	*~[!#6;!#1](~*)~*	150	*!@*@*!@*
152	[#8]~[#6](~[#6])~[#6]	154	[#6]=[#8]
156	[#7]~*(~*)~*	158	[#6]-[#7]
161	[#7]	162	a
163	*1~*~*~*~*~*~1	164	[#8]

Table 5: SMARTS for MACCS keys shown in Figure 7.

key	smarts	key	smarts
8	[!#6;!#1]1~*~*~*~1	11	*1~*~*~*~1
16	[!#6;!#1]1~*~*~1	17	[#6]#[#6]
19	*1~*~*~*~*~*~*~1	49	[!+0]
57	[#8R]	65	c:n
66	[#6]~[#6](~[#6])(~[#6])~*	72	[#8]~*~*~[#8]
83	[!#6;!#1]1~*~*~*~*~1	84	[NH2]
85	[#6]~[#7](~[#6])~[#6]	89	[#8]~*~*~*~[#8]
90	[$([!#6;!#1;!H0]~*~*~[CH2]~*), $([!#6;!#1;!H0;R]1@[R]@[R]@[CH2;R]1), $([!#6;!#1;!H0]~[R]1@[R]@[CH2;R]1)]	91	[$([!#6;!#1;!H0]~*~*~*~[CH2]~*), $([!#6;!#1;!H0;R]1@[R]@[R]@[R]@[CH2;R]1), $([!#6;!#1;!H0]~[R]1@[R]@[R]@[CH2;R]1), $([!#6;!#1;!H0]~*~[R]1@[R]@[CH2;R]1)]
92	[#8]~[#6](~[#7])~[#6]	96	*1~*~*~*~*~1
97	[#7]~*~*~*~[#8]	98	[!#6;!#1]1~*~*~*~*~*~1
99	[#6]=[#6]	104	[!#6;!#1;!H0]~*~[CH2]~*
105	*@*(@*)@*	108	[CH3]~*~*~*~[CH2]~*
112	*~*(~*)(~*)~*	114	[CH3]~[CH2]~*
115	[CH3]~*~[CH2]~*	116	[$([CH3]~*~*~[CH2]~*),$([CH3]~*1~*~[CH2]1)]
121	[#7;R]	122	*~[#7](~*)~*
126	*!@[#8]!@*	127	*@*!@[#8]
128	[$(*~[CH2]~*~*~*~[CH2]~*), $([R]1@[CH2;R]@[R]@[R]@[R]@[CH2;R]1), $(*~[CH2]~[R]1@[R]@[R]@[CH2;R]1), $(*~[CH2]~*~[R]1@[R]@[CH2;R]1)]	129	[$(*~[CH2]~*~*~*~[CH2]~*), $([R]1@[CH2]@[R]@[R]@[CH2;R]1), $(*~[CH2]~[R]1@[R]@[CH2;R]1)]
131	[!#6;!#1;!H0]	132	[#8]~*~[CH2]~*
133	*@*!@[#7]	137	[!C;!c;R]
139	[O;!H0]	142	[#7]
143	*@*!@[#8]	148	*~[!#6;!#1](~*)~*
149	[C;H3,H4]	150	*!@*@*!@*
151	[#7;!H0]	152	[#8]~[#6](~[#6])~[#6]
154	[#6]=[#8]	157	[#6]-[#8]
160	[C;H3,H4]	162	a
163	*1~*~*~*~*~*~1	164	[#8]
165	[R]		