Machine Learning: Science and Technology

PAPER • **OPEN ACCESS**

Neural message passing on high order paths

To cite this article: Daniel Flam-Shepherd *et al* 2021 *Mach. Learn.: Sci. Technol.* **2** 045009

View the article online for updates and enhancements.

You may also like

- A Model for Periodic Nonlinear Electric Field Structures in Space Plasmas
M.N.S. Qureshi, Shi Jian-Kui and Liu Zhen-Xing
- Radio Bridge Structure and Its Application to Estimate the Mach Number and Ambient Gas Temperature of Powerful Sources
Greg F. Wellman, Ruth A. Daly and Lin Wan
- Analysis of Electrical Cell-to-Cell Communication Using the Aggregate of Model Cells
Issei Kasai, Yuki Kitazumi, Kenji Kano et al.

This content was downloaded from IP address 223.236.224.249 on 04/07/2023 at 12:03

IOP Publishing Mach. Learn.: Sci. Technol. 2 (2021) 045009 https://doi.org/10.1088/2632-2153/abf5b8

MACHINE LEARNING
Science and Technology

PAPER

Neural message passing on high order paths

Daniel Flam-Shepherd[1,4,*], Tony C Wu[1,4,*], Pascal Friederich[1,2] and Alan Aspuru-Guzik[1,3,4]

[1] University of Toronto, Toronto, Canada
[2] Karlsruhe Institute of Technology, Toronto, Canada
[3] CIFAR, Karlsruhe, Germany
[4] Vector Institute, Toronto, Canada
[*] Authors to whom any correspondence should be addressed.

E-mail: danielfs@cs.toronto.edu **and** tonyc.wu@utoronto.ca

Keywords: graph neural networks, molecular property prediction, molecular geometry

OPEN ACCESS

RECEIVED
17 November 2020

REVISED
18 March 2021

ACCEPTED FOR PUBLICATION
7 April 2021

PUBLISHED
19 July 2021

Original Content from this work may be used under the terms of the Creative Commons Attribution 4.0 licence.

Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

Abstract

Graph neural networks have achieved impressive results in predicting molecular properties, but they do not directly account for local and hidden structures in the graph such as functional groups and molecular geometry. At each propagation step, graph neural networks aggregate only over first order neighbours and can only learn about important information contained in subsequent neighbours as well as the relationships between those higher order connections—over many propagation steps. In this work, we generalize graph neural nets to pass messages and aggregate across higher order paths. This allows for information to propagate over various levels and substructures of the graph. We demonstrate our model on a few tasks in molecular property prediction.

1. Introduction and motivation

Graph neural networks (GNNs) are a powerful tool for representation learning across different domains involving relational data such as molecules [1] or social and biological networks [2]. These models learn node embeddings in a message passing framework [3] by passing and aggregating node and edge feature information across the graph using neural networks. The learned node representations can then be used for any downstream procedure such as node or graph classification or regression. In particular, GNNs were shown to predict molecular properties with density-functional theory level accuracy but 10^5 times faster [3].

Another model, learns a similar embedding of atom types—SchNet [4] which is specifically designed to model atomistic systems by making use of continuous-filter convolutional layers to accurately predict a range of properties across chemical space for molecules and materials. These classes of deep learning models have lead to a revolution in molecular property prediction.

However, current GNN models still suffer from limitations as they only propagate information across neighbouring edges and pooling that information into final node embeddings [1, 5]. This means that, in most models, nodes only learn about the larger neighbourhood surrounding them over many propagation steps. This makes it difficult for GNNs to learn higher order graph structure and impossible to learn in a single propagation layer. However, such long range correlations are important for many domains, in particular, when learning chemical properties that depend on rings, branches, functional groups or molecular geometry.

It would be advantageous to develop a model that can account for long range molecular dependencies directly in a single propagation layer as there are many molecular features and substructures that arise in larger molecular neighbourhoods than a single atom and its neighbours. Consider figure 1, notice how much larger the neighbourhood of the atom gets when you consider second and third order neighbours.

The only way to directly account for higher order graph properties is to pass messages over additional neighbours in every propagation layer of the GNN. This work focuses on generalizing message passing neural networks to accomplish this.

© 2021 The Author(s). Published by IOP Publishing Ltd

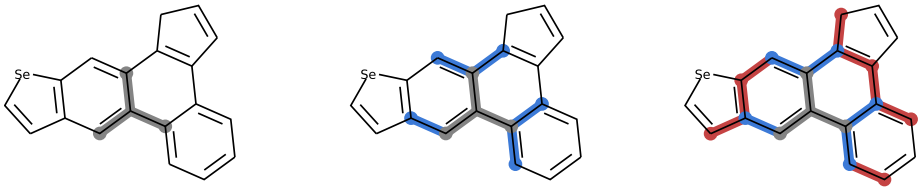


Figure 1. The neighbourhood of one atom comprised of (a) 1st (b) 2nd and (c) 3rd order neighbours for a molecule randomly sampled from the CEP dataset.

1.1. Motivations

There are many factors pertaining to molecular graphs that motivate the development of our model. In this section we discuss, in more depth, the limitations of GNNs with respect to specific aspects of molecules that motivate our model. These include molecular substructures like rings and functional groups, molecular geometry as characterized by internal coordinates as well as stereochemistry.

Molecular substructures play an important role in determining molecular properties for example functional groups influence the chemical reactions a molecule undergoes. By only aggregating over neighbours, GNNs cannot learn about these larger substructures in a single propagation layer. On the other hand, by passing messages over larger neighbourhoods, in every layer we could directly learn about these structures. Furthermore, We could directly indicate if the path that a message is travelling on contains a simple functional group like alcohol (ROH) or passes through a larger functional group. For example, atoms in the neighbourhood of a functional group could receive a message along a path indicating a functional group is in the neighbourhood.

Molecular geometry is the three dimensional arrangement of atoms in a molecule and influences several properties, including the reactivity, polarity and biological activity of the molecule. An important application of GNNs is predicting quantum mechanical properties of molecules, which are heavily dependent on the geometry of the molecule. The 3D configuration of a molecule can be fully specified by (1) bond lengths—the distance between two bonded atoms, (2) bond angles—the angle formed between three neighbouring atoms, and (3) dihedral angles between four consecutive atoms. In fact the potential energy is typically modelled as a sum of terms involving each of these three. Current GNN approaches to quantum chemistry incorporate neighbouring geometry by using bond distances as edge features [3], but do not directly account for the relative orientation of neighbouring atoms and bonds—a framework that could do so would be advantageous.

Stereochemistry involves the relative spatial arrangement of atoms in molecules, specifically, stereoisomers—which are molecules with the same discrete graph but different three-dimensional orientation of atoms. For example, enantiomers—non-superimposable mirror images of molecules and cis-trans isomers, that only differ through the rotation of a functional group. Even if they use interatomic distances as edge features, GNNs will have limited ability to distinguish stereoisomers, since these molecules only differ through the relative orientation of atoms. In general, at every propagation step, GNNs should learn representations over each node's extended neighbourhood to encode the relationships between nodes in that neighbourhood.

1.2. Approach and contributions

We generalize message passing neural networks (MPNNs) to aggregate across larger neighbourhoods by passing messages along simple paths of higher order neighbours. We describe the general framework in section 3. We experiment with various molecular property prediction task and a node classification task in citation networks. Our specific contributions are two-fold:

- We devise a simple extension to any message passing neural network to learn representations over larger node neighbourhoods within each propagation layer by simply augmenting the message function to aggregate over additional neighbours.
- By summing over additional neighbours we enable the use of path features such as bond angles for paths of length two and dihedral angles for paths of length three and thus encoding the full molecular geometry and orientation, so that MPNNs can distinguish isomers.

2. Related work and background

2.1. Background

Message passing neural networks operate on graphs G with n nodes each with feature vector $\mathbf{x}_v \in \mathbb{R}^f$ that specify what kind of atom the node is, among other possible features. There are $n \times n$ edge feature vectors $\boldsymbol{e}_{vw} \in \mathbb{R}^e$ that specify what kind of bond type atoms v, w have. The forward pass has two phases, a message passing phase and a readout phase.

The message passing phase runs for T propagation steps and is defined in terms of message functions M_t and node update functions U_t. During the message passing phase, hidden states $\boldsymbol{h}_v^t$ at each node in the graph are updated based on messages $\boldsymbol{m}_v^{t+1}$ according to:

$$\boldsymbol{m}_v^{t+1} = \sum_{w \in \mathcal{N}_v} M_t(\boldsymbol{h}_v^t, \boldsymbol{h}_w^t, \boldsymbol{e}_{vw}), \quad \boldsymbol{h}_v^{t+1} = U_t(\boldsymbol{h}_v^t, \boldsymbol{m}_v^{t+1}), \quad \mathbf{y} = \texttt{Readout}(\{\boldsymbol{h}_v^T, \boldsymbol{x}_v\}_{v \in G}).$$

We have $\boldsymbol{x}_v = \boldsymbol{h}_v^0$. The message node v receives aggregates over its neighbours $\mathcal{N}_v$, in this case, by simple summation. We then readout predictions $\mathbf{y}$ based on final node embeddings.

2.2. Related work

The first graph neural network model was proposed by [6] and many variants have been recently proposed [5, 7, 8]. Our focus is on the general framework of neural message passing from [3]. We review relevant GNN models and their use in Molecular Deep learning in this section.

2.2.1. Molecular deep learning

Recently GNNs have superseded machine learning methods involving hand-crafted feature representation, on predicting molecular properties for large datasets [3]. For example, neural fingerprints generalizes standard molecular fingerprints with a differentiable one that achieves better predictive accuracy [1]. Another model, SchNet [4] defines a continuous-filter convolutional neural network for modelling quantum interactions and achieves state of the art results.

2.2.2. Higher order GNNs

Recent work has generalized graph convolution networks (GCNs) [8] to higher order structure by repeatedly mixing feature representations of neighbours at various distances [9], or casting GCNs into a general framework inspired by the path integral formulation of quantum mechanics [10]. Both of these works are based on powers of the adjacency matrix and do not account directly for the relationship between higher order neighbours. Another work [11] proposes k-dimensional GNNs in order to take higher order graph structures at multiple scales into account. GNNs and higher order GNNs do not incorporate the relationship between higher order neighbours, which would allow for features that are dependent on that relationship, namely 'path features'.

2.2.3. Path augmented transformer

Another model based on the transformer architecture [12] accounts for long range dependencies in molecular graphs by augmenting edge feature tensor to include some (shortest) path features like bond type, conjugacy, inter-atomic distance and ring membership.

2.2.4. Structured transformer

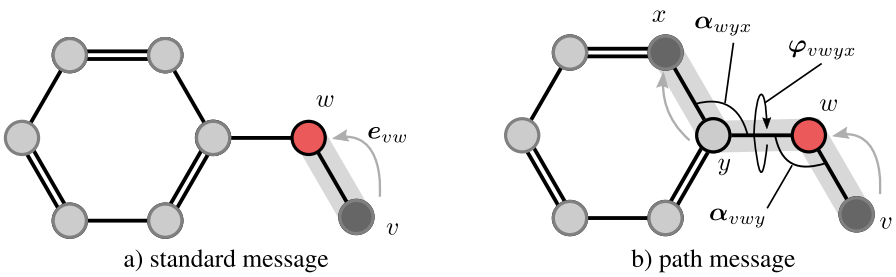
A few graph neural networks recently proposed have incorporated directional information. The first [13] builds a model for proteins that considers the local change in the coordinate system for each atom in the chain.

2.2.5. 3D GCN

Cho *et al* [14] build a three-dimensional graph convolutional network, for molecular properties and biochemical activities prediction using 3D molecular graph by augmenting the standard GCN layer with the relative atomic position vector.

2.2.6. Directional message passing

Klicpera *et al* [15] embeds the messages passed between atoms such that each message is associated with a direction in coordinate space and are rotationally equivariant since the associated directions rotate with the molecule. Their message passing scheme transforms messages based on the angle between them in order to encode direction.



a) standard message b) path message

Figure 2. Message function and path features for (a) standard MPNN and (b) MPNN passing messages on paths with length 3 in a molecule with path features involving molecular geometry. Light grey is carbon. Red is oxygen. Dark grey is the initial/end atom in the path. The molecule is anisole, or methoxybenzene.

Other work has incorporated attention and edge memory schemes to the existing message passing neural network framework [16] as well message passing directed over bonds to construct embeddings [17].

3. Neural message passing on paths

We extend the message passing framework by propagating information from every node's higher order neighbour instead of aggregating messages from only nearest neighbours. The message passing phase is augmented such that hidden states $\boldsymbol{h}_v^t$ at each node in the graph are updated based on messages over all simple paths up to length ℓ from its neighbourhood:

$$\boldsymbol{m}_v^{t+1} = \sum_{\boldsymbol{p}\in\mathcal{P}_\ell^v} M_t(\boldsymbol{h}_v^t, \boldsymbol{p}) = \sum_{v_1\in\mathcal{N}_v} \sum_{\substack{v_2\in\mathcal{N}_{v_1} \\ v_2\neq v}} \cdots \sum_{\substack{v_\ell\in\mathcal{N}_{v_{\ell-1}} \\ v_\ell\neq v_{\ell-2},\ldots,v}} M_t(\boldsymbol{h}_v^t, \boldsymbol{p}_{v_1\to v_\ell}), \tag{1}$$

where we define $\boldsymbol{p}$ to be a path in $\mathcal{P}_\ell{}^v$, which is the set of all simple paths starting from node v with length ℓ and $\boldsymbol{p}_{v_1\to v_\ell}$ to be path features along path $\boldsymbol{p}$ from node v_1 to node v_ℓ. We only sum over simple paths, excluding loops and multiple inclusions of the same node.

3.1. Path features

For graphs with a large number of nodes and edges, passing messages along paths becomes very expensive and, as in GraphSage [2], sampling a subset of paths of higher order neighbours is necessary. However, for molecules, where the number of neighbours is usually $\leqslant 4$ this is not necessary. Furthermore, one can include domain specific path features in the message function. We describe two examples of these path features below.

3.1.1. Molecular substructures

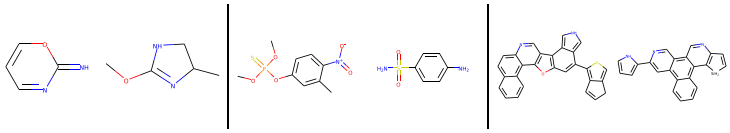
We can incorporate whether the path travels through a molecular substructure by considering paths of at least length 2, where we have a message function that sums over two neighbouring atoms $v\to w\to y$. Along with their node and edge features, the possible path features include ring features—ie one hot indication if any atoms are in (specific) rings as well as if the path is a functional group (ROH) or within a larger functional group:

$$\boldsymbol{m}_v^{t+1} = \sum_{w\in\mathcal{N}_v} \sum_{\substack{y\in\mathcal{N}_w \\ y\neq v}} M_t(\boldsymbol{h}_v^t, \boldsymbol{p}_{v\to y}) \quad \boldsymbol{p}_{v\to y} = \begin{bmatrix} \boldsymbol{h}_w^t & \boldsymbol{h}_y^t \\ \boldsymbol{e}_{vw} & \boldsymbol{e}_{wy} \end{bmatrix}. \tag{2}$$

3.1.2. Molecular geometry

Considering paths of length 3, where we have a message function that sums over three neighbouring atoms $v\to w\to y\to x$. Along paths of length three additional features include two bond angles $\boldsymbol{\alpha}_{vwy}$& $\boldsymbol{\alpha}_{wyx}$ and the dihedral angle $\boldsymbol{\varphi}_{vwyx}$ between the planes defined by the pairs of atoms (v, w) and (y, x). Effectively, messages passed over three consecutive neighbours contain information about the entire molecular geometry (see figure 2):

$$\boldsymbol{m}_v^{t+1} = \sum_{w\in\mathcal{N}_v} \sum_{\substack{y\in\mathcal{N}_w \\ y\neq v}} \sum_{\substack{x\in\mathcal{N}_y \\ x\neq w,v}} M_t(\boldsymbol{h}_v^t, \boldsymbol{p}_{v\to x}) \quad \boldsymbol{p}_{v\to x} = \begin{bmatrix} \boldsymbol{h}_w^t & \boldsymbol{h}_y^t & \boldsymbol{h}_x^t \\ \boldsymbol{e}_{vw} & \boldsymbol{e}_{wy} & \boldsymbol{e}_{yx} \\ \boldsymbol{\alpha}_{vwy} & \boldsymbol{\alpha}_{wyx} & \boldsymbol{\varphi}_{vwyx} \end{bmatrix}. \tag{3}$$



a) QM8 a) ESOL a) CEP

Figure 3. Molecules from the datasets considered.

4. Experiments

4.1. Datasets

We compare the performance of our model against a few baselines on a variety of molecular property prediction tasks involving different datasets of undirected molecular graphs with different sizes and distributions. One with moderately large graphs with 6–8 rings and two with much smaller molecules and 1–2 rings. Figure 3 has examples of molecules from the datasets in these tasks, they include:

- ESOL: Delaney [19] predicting the aqueous solubility of 1144 molecules.
- QM8: Ruddigkeit *et al* [20] predicting 16 electronic spectra values calculated using density functional theory for 21 786 organic molecules.
- CEP: the photovoltaic efficiency of 20 000 organic molecules from The Harvard Clean Energy Project [21].

4.2. Model design

For QM8 we use the Path MPNN with message given by equation (3) since molecular geometry is very important for the targets in QM8, the path features such as bond lengths, bond angles and dihedrals are computed on the fly using rdkit [22] given supplied atomic coordinates in the data. We use the message function from the interaction networks model in [23] $\boldsymbol{m}_{v\to x} = \tau(\texttt{Concat}[\boldsymbol{p}_{v\to x}])$ that passes the concatenated path features through a single layer neural net with relu activation and then uses graph attention [7] to aggregate incoming messages passed over paths:

$$M_t(\boldsymbol{h}_v^t, \boldsymbol{p}_{v\to x}) = a_{v\to x}\boldsymbol{m}_{v\to x} \quad \text{where} \quad a_{v\to x} = \frac{e^{\boldsymbol{a}^\top \boldsymbol{m}_{v\to x}}}{\sum_{v\to x} e^{\boldsymbol{a}^\top \boldsymbol{m}_{v\to x}}}, \tag{4}$$

where **a** is the attention weight vector and the summation is over the simple path $v \to x$ which is the same triple sum in (3). For ESOL and CEP we pass messages over paths of length two as specified by equation (2) in order to use path features for molecular substructures (which are important for the targets in these datasets as well), these features also computed using rdkit. For both models, the node update function concatenates incoming messages with the current node state and feeds it through a dense layer $U_t = \sigma([\boldsymbol{h}_v^t, \boldsymbol{m}_v^{t+1}])$. After propagation through message passing layers, we use the set2set model [24] in the same way as [3] as the readout function to combine the node hidden features into a fix-sized hidden vector.

For all models, we only use atom and bond type/distance as one-of-k hot encodings for for initial node and edge features $\boldsymbol{x}_v, \boldsymbol{e}_{uv}$, in addition to previously described path features. The models are trained using root mean squared error (RMSE) for loss. Model evaluation is done using mean absolute error (MAE) of the molecular properties in the QM8 dataset, RMSE for ESOL and percent for CEP. We use a 80%–10%–10% train, validation, test set split. We perform three runs with different randomizations and report the mean performance and standard deviation. We do not perform cross-validation.

4.3. Implementation details

The model was coded in pytorch [25] from scratch and trained on a GeForce RTX 2070 GPU which has 8 GB of ram. The Path MPNN model is memory intensive and the largest batch size we could use was 32 for the smaller dataset (QM8, ESOL) with 16 for the larger molecular graphs (CEP). Minimal hyperparameter optimization was needed for the path model to achieve superior performance. We used learning rate of 10^{-4} with the adam optimizer [26] training for 300 epochs for both the Path MPNN and baseline MPNN. We experimented with 1–3 path MPNN layers and 3–6 MPNN layers in the baseline but we found no improvement for increasing number of layers in both. For both the baseline MPNN and Path MPNN we used 128 hidden units for the node representation as well as in the message function's one layer NN with a ReLu non-linearity. In addition we use 1024 hidden units for the set2vec model with five unroll steps for both the Path MPNN and baseline MPNN. We only optimized hidden unit number and number of message passing layers along with learning rate.

Table 1. Mean and std error predictive accuracy on various dataset and baselines.

Dataset Units	QM8 MAE in eV ($\times 10^{-3}$)	ESOL RMSE in log mol1^{-1}	CEP Percent
Neural fingerprint [1]	13.80 ± 0.11	0.52 ± 0.07	1.43 ± 0.09
MPNN	11.30 ± 0.31	0.47 ± 0.03	1.37 ± 0.09
MolNet [18]	10.80 ± 0.30	0.58 ± 0.03	—
Path transformer [12]	10.20 ± 0.30	0.55 ± 0.06	—
Path MPNN	$\mathbf{8.70 \pm 0.06}$	$\mathbf{0.41 \pm 0.02}$	$\mathbf{1.23 \pm 0.08}$

Table 2. CORA test accuracy.

Model	Test accuracy
GCN [8]	81.5
MixHop [9]	81.9
PAN [10]	82.0
Path GCN	**82.4**

4.4. Baselines and results

We use the top performing model from Molecule Net [18] (Molnet) for each dataset. We also benchmark with the differentiable version of circular fingerprints from [1] (neural fingerprints). To highlight the importance of path features, we also compared the performance of a standard Message Passing neural net that uses three message passing layers. This uses the same message function, node update function and readout function as the Path model (except path length is one). The last benchmark is the path-augmented graph transformer network since this model is similarly built to model longer-range dependencies in molecular graphs. As can be seen in table 1, for QM8, ESOL and CEP, passing messages over paths leads to a substantial improvement in predictive accuracy.

5. Comparison with other higher order GNNs

In a separate experiment, we compare the path MPNN with other GNNs that use higher order neighbours and do not use edge features. We consider a standard task of semi-supervised node classification with the CORA citation network dataset [27].

The dataset contains sparse bag-of-words feature vectors for each document and a list of citation links between documents for undirected edges in the adjacency matrix. Each document has a class label. Altogether, the network has 2708 nodes and 5429 edges with 7 classes and 1433 features.

5.1. Model

We use the experimental setup of [8]. We sum over paths of length 3 while uniformly sampling a single second order and third order neighbour. Our base MPNN is a GCN [8] that has message function:

$$\boldsymbol{m}_v^{t+1} = \sum_{w \in \mathcal{N}_v} \hat{\mathbf{A}}_{vw} \boldsymbol{h}_v^t, \quad U_t = \sigma(\boldsymbol{m}_v^{t+1}), \tag{5}$$

where $\hat{\mathbf{A}}$ is the adjacency matrix and σ is a dense layer with sigmoid activation. There is no readout function necessary for pooling and a softmax layer maps the node representations to the prediction. For a citation network the path features are just the node features and edge features connecting v to nodes that are ℓ nodes away, i.e.

$$\boldsymbol{p}_{v_1 \to v_\ell} = \{\boldsymbol{h}_{v_1}^t, \boldsymbol{e}_{vv_1}, \ldots, \boldsymbol{h}_{v_\ell}^t, \boldsymbol{e}_{v_\ell v_{\ell-1}}\}.$$

5.2. Results

We compare with two other higher order GCN variants: Mixhop [9] and PAN [10]: Path integral graph convolution—both use powers of the adjacency to aggregate GCN layers of higher order neighbours. The results are displayed in table 2 and our model achieve similar accuracy to our baselines.

6. Conclusion and discussion

6.1. Limitations

In this work we only considered very simple message functions, in general, it is not straight forward to construct message function over paths. For example, the message function from [3], maps edge features to a

square matrix using a neural net—incorporating more neighbours and their edge and path features into this kind of message function introduces many design challenges. There are many graph types where the path MPNN would be roughly as useful as a standard MPNN, including most non chemical graph data. In general, we found no improvement in accuracy in passing messages over longer paths than relevant to the desired path features for example. The path MPNN has a large capacity and is very susceptible to overfitting, by using a fairly low learning rate we were able to overcome this and see comparable performance on training, validation and test sets. In further we hope to find methods to reduce the memory requirement of the Path model.

6.2. Complexity

For standard MPNNs, a single step of the message passing phase for a dense graph requires $\mathcal{O}(n^2d^2)$ floating point multiplications. Naturally for the Path models this is multiplied by the path length ℓ but is still roughly equivalent to ℓ propagation steps for the standard MPNN while having increased performance benefits.

We introduce a general GNN framework based on message passing over simple paths of higher order neighbours. This allows us to use path features in addition to node and edge features, which is very useful in molecular graphs, as many informative features are characterized by the paths between atoms. We benchmarked our framework on molecular property prediction tasks and a node classification task in citation networks.

Data availability statement

The data that support the findings of this study will be openly available following an embargo at the following URL/DOI: https://gitlab.com/verysure/path-mpnn. Data will be available from 31 March 2021.

Acknowledgments

A A-G acknowledges generous support from the Canada 150 Research Chair Program, Tata Steel, Anders G Froseth, and the Office of Naval Research. We acknowledge supercomputing support from SciNet.

ORCID iDs

Daniel Flam-Shepherd https://orcid.org/0000-0002-9568-3451
Pascal Friederich https://orcid.org/0000-0003-4465-1465
Alan Aspuru-Guzik https://orcid.org/0000-0002-8277-4434

References

[1] Duvenaud D, Maclaurin D, Aguilera-Iparraguirre J, Gómez-Bombarelli R, Hirzel T, Aspuru-Guzik A and Adams R P 2015 Convolutional networks on graphs for learning molecular fingerprints *Neural Information Processing Systems*
[2] Hamilton W, Ying Z and Leskovec J 2017 Inductive representation learning on large graphs *Advances in Neural Information Processing Systems* pp 1024–34
[3] Gilmer J, Schoenholz S S, Riley P F, Vinyals O and Dahl G E 2017 Neural message passing for quantum chemistry *Proc. 34th Int. Conf. Machine Learning* vol 70 (JMLR.org) pp 1263–72
[4] Schütt K T, Kindermans P-J, Sauceda H E, Chmiela S, Tkatchenko A and Müller K-R 2017 SchNet: a continuous-filter convolutional neural network for modeling quantum interactions (arXiv:1706.08566)
[5] Li Y, Tarlow D, Brockschmidt M and Zemel R 2015 Gated graph sequence neural networks (arXiv:1511.05493)
[6] Scarselli F, Gori M, Tsoi A C, Hagenbuchner M and Monfardini G 2008 The graph neural network model *IEEE Trans. Neural Netw.* **20** 61–80
[7] Veličković P, Cucurull G, Casanova A, Romero A, Lio P and Bengio Y 2017 Graph attention networks (arXiv:1710.10903)
[8] Kipf T N and Welling M 2016 Semi-supervised classification with graph convolutional networks (arXiv:1609.02907)
[9] Abu-El-Haija S, Perozzi B, Kapoor A, Harutyunyan H, Alipourfard N, Lerman K, Steeg G V and Galstyan A 2019 Mixhop: higher-order graph convolution architectures via sparsified neighborhood mixing (arXiv:1905.00067)
[10] Ma Z, Li M and Wang Y 2019 Pan: path integral based convolution for deep graph neural networks (arXiv:1904.10996)
[11] Morris C, Ritzert M, Fey M, Hamilton W L, Lenssen J E, Rattan G and Grohe M 2019 Weisfeiler and Leman go neural: higher-order graph neural networks *Proc. Conf. Artificial Intelligence* vol 33 pp 4602–9
[12] Chen B, Barzilay R and Jaakkola T 2019 Path-augmented graph transformer network (arXiv:1905.12712)
[13] Ingraham J, Garg V, Barzilay R and Jaakkola T 2019 Generative models for graph-based protein design *Advances in Neural Information Processing Systems* pp 15794–805
[14] Cho K, Bart Van Merriënboer, Bahdanau D and Bengio Y 2014 On the properties of neural machine translation: encoder–decoder approaches (arXiv:1409.1259)
[15] Klicpera J, Groß J and Günnemann S 2020 Directional message passing for molecular graphs *Int. Conf. Learning Representations*
[16] Withnall M, Lindelöf E, Engkvist O and Chen H 2020 Building attention and edge message passing neural networks for bioactivity and physical–chemical property prediction *J. Cheminformatics* **12** 1
[17] Yang K *et al* 2019 Analyzing learned molecular representations for property prediction *J. Chem. Inf. Model.* **59** 3370–88

[18] Wu Z, Ramsundar B, Feinberg E N, Gomes J, Geniesse C, Pappu A S, Leswing K and Pande V 2018 MoleculeNet: a benchmark for molecular machine learning *Chem. Sci.* **9** 513–30
[19] Delaney J S 2004 ESOL: estimating aqueous solubility directly from molecular structure *J. Chem. Inf. Comput. Sci.* **44** 1000–5
[20] Ruddigkeit L, van Deursen R, Blum L C and Reymond J-L 2012 Enumeration of 166 billion organic small molecules in the chemical universe database gdb-17 *J. Chem. Inf. Model.* **52** 2864–75
[21] Hachmann J, Olivares-Amaya R, Atahan-Evrenk S, Amador-Bedolla C, Sánchez-Carrera R S, Gold-Parker A, Vogt L, Brockway A M and Aspuru-Guzik Aan 2011 The harvard clean energy project: large-scale computational screening and design of organic photovoltaics on the world community grid *J. Phys. Chem. Lett.* **2** 2241–51
[22] Landrum G *et al* 2016 RDkit: open-source cheminformatics software (available at: www.rdkit.org/; https://github.com/rdkit/rdkit)
[23] Battaglia P *et al* 2016 Interaction networks for learning about objects, relations and physics *Advances in Neural Information Processing Systems* pp 4502–10
[24] Vinyals O, Bengio S and Kudlur M 2015 Order matters: sequence to sequence for sets (arXiv:1511.06391)
[25] Paszke A *et al* 2019 Pytorch: an imperative style, high-performance deep learning library *Advances in Neural Information Processing Systems 32* ed H Wallach, H Larochelle, A Beygelzimer, F d'Alché-Buc, E Fox and R Garnett (Curran Associates, Inc.) pp 8024–35
[26] Kingma D P and Ba J 2014 Adam: a method for stochastic optimization (arXiv:1412.6980)
[27] Sen P, Namata G, Bilgic M, Getoor L, Galligher B and Eliassi-Rad T 2008 Collective classification in network data *AI Mag.* **29** 93