Graph Nets for Partial Charge Prediction

Yuanqing Wang
Memorial Sloan Kettering Cancer Center
New York, N.Y. 10065 USA
yuanqing.wang@choderalab.org

Josh Fass
Memorial Sloan Kettering Cancer Center
New York, N.Y. 10065 USA
josh.fass@choderalab.org

Chaya D. Stern
Memorial Sloan Kettering Cancer Center
New York, N.Y. 10065 USA
chaya.stern@choderalab.org

Kun Luo
Uli Statistical Learning
Shenzhen, Guangdong 518057, P.R. China
k@uli.ai

John D. Chodera*
Memorial Sloan Kettering Cancer Center
New York, N.Y. 10065
john.chodera@choderalab.org

Abstract

Atomic partial charges are crucial parameters for Molecular Dynamics (MD) simulations, molecular mechanics calculations, and virtual screening, as they determine the electrostatic contributions to interaction energies. Current methods for calculating partial charges, however, are either slow and scale poorly with molecular size (quantum chemical methods) or unreliable (empirical methods). Here, we present a new charge derivation method based on Graph Nets—a set of *update* and *aggregate* functions that operate on molecular topologies and propagate information thereon—that could approximate charges derived from Density Functional Theory (DFT) calculations with high accuracy and an over 500-fold speed up.

1 Introduction

Molecular machine learning, with its significance in drug and materials design gradually being realized by the community, has been maturing rapidly. Recently, the focus has shifted from embedding molecules in a fixed-dimensional feature space to fit architectures borrowed from neighboring fields toward developing representations that respect the combinatorial, discrete nature of molecular structures. Among the newly developed algorithms are a plethora of graph-based methods [6, 9, 4, 3, 12, 13, 14, 11, 12], which represent molecules as graphs, atoms as nodes, and bonds as edges. Generally speaking, such representation makes it possible for the operations on modelled molecules to preserve the permutation invariance and equivariance with regard to chemically equivalent atoms.

Despite the promising results achieved by these architectures in predicting per-molecule attributes (hydration free energies, binding affinities, lipohilicities, etc.), little attention has been paid to per-atom and per-bond regression tasks. This is drastically different from the general applications of graph learning, where per-node and per-edge problems are far more common. The prediction of atomic partial charges, we believe, could serve as an interesting pivotal task: As commercially available compound libraries now exceed 10^9 molecules [8], there is a significant need for fast methods for determining high-quality partial charges for small organic molecules for virtual screening.

---

*Corresponding Author

In addition, next-generation force fields that consistently treat small molecules and biomolecules would greatly benefit from a fully consistent approach to determining charges for biopolymers (both naturally occurring and covalently modified) that exceed sizes practical for current quantum chemical calculations without requiring fragmentation and capping. This task may be particularly suited to machine learning methods due to the relative ease of generating an arbitrarily large number of partial charge training examples derived from high-level quantum chemistry.

Our model was built on top of the full-block Graph Net structure [1], where edges and the global node are attributed and connected to the nodes. Maximal generality and flexibility are provided by this framework so that, with certain choices of update and aggregate functions, the network could specialize into either a Message Passing Neural Net[6] or Graph Convolutional Network[9]. We made contributions by designing and tuning its component functions while preserving the permutation equivariance end-to-end, as well as enforcing a *net charge constraint*—the sum of the atomic partial charges equal the known total charge of the molecule—as a tractable quadratic program. We also optimized the batching pipeline for molecular graphs. This architecture is able to estimate charges derived from DFT calculations with an RMSE of 0.0223 e on a subset of molecules from ChEMBL [5] curated by Bleiziffer et al. [2]. Also, within the dataset, the prediction accuracy does not decrease as the size of the system increases. We therefore believe that this method, with results in agreement with quantum chemical calculations and computation time similar to empirical methods, has the potential to replace existing charge derivation schemes widely used in MD calculations and virtual screening approaches. Compared to the Random Forest model proposed by Bleiziffer et al. [2], our model does not depend on rule-based fingerprints and is differentiable everywhere, and could therefore be optimized concurrently with forcefield parameters.

This work is part of a project that uses graph representations in molecular learning, termed GIMLET (Graph Inference on MoLEcular Topology). The infrastructure was implemented in Python 3.6 with no dependencies other than TensorFlow 2.0 [10], in order to ensure the pipelines can be integrated into TensorFlow computation graphs, and executed in a parallel and distributed fashion. The code is open source (`https://github.com/choderalab/gimlet`) and available under the MIT License.

2 Architecture

2.1 Molecules as graphs

A *graph* can be defined as a tuple of three sets:

$$\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{U}\} \tag{1}$$

where $\mathcal{V}$ is the set of the vertices (or nodes) (atoms), $\mathcal{E}$ the set of edges (bonds), and $\mathcal{U} = \{\mathbf{u}\}$ the universal attribute. We model the molecules as node-, edge-, and globally attributed, undirected graphs, whose nodes are atoms and edges are bonds. In this notation, the connectivity is included in $\mathcal{E}$. In our implementation though, a vector $\mathbf{a}$ denoting a sequence of atomic numbers and a upper-triangular adjacency matrix A is sufficient to describe the topology of the graph.

2.2 Graph Nets

There are three stages in both the training and inference process: initialization, propagation, and readout, which, in our setting, are governed by sets of learnable functions. *Initialization stage* is the phase in which the data is abstracted as graph entities. Driven by the idea that the chemical environment of atoms could be realized in rounds of message passing, we featurize the atoms simply using one-hot encodings of the element identity, which is then fed into a feedforward neural network to form the vertex attributes at $t = 0$,

$$\mathbf{v}_{ij}^{(-1)} = \begin{cases} 1, & \text{atom } i \text{ is element } j; \\ 0, & \text{elsewhere,} \end{cases}, \mathbf{v}^{(0)} = \text{NN}_{v_0}(\mathbf{v}^{(-1)}) \in \mathbb{R}^{N_{\text{atoms}} \times d_v}, \tag{2}$$

where $i \in \{1, 2, ..., N_{\text{atoms}}\}$ and $j \in \{1, 2, ..., N_{\text{elements}}\}$, and d_v is the fixed dimension of node attributes. Edge attributes are initialized as the output of a feedforward network which takes the bond order (averaged over resonance structures) as input; global attributes are initialized as zeros as placeholder.

$$\mathbf{e}^{(0)} = \text{NN}_{e_0}(\text{BO}) \in \mathbb{R}^{N_{\text{bonds}} \times d_e}, \mathbf{u}^{(0)} = \mathbf{0}^{1 \times d_u}, \tag{3}$$

where d_e is the hidden dimension of edges and d_u is the hidden dimension of global attributes. In *propagation stage*, the framework we adopted follows a formalism by Battaglia et al,[1] where, in each round of message passing, the attributes of nodes, edges, and the graph as a whole, $\mathbf{v}$, $\mathbf{e}$, and $\mathbf{u}$ are updated by trainable functions in the following order:

$$\mathbf{e}_k^{(t+1)} = \phi^e(\mathbf{e}_k^{(t)}, \sum_{i \in \mathcal{N}_k^e} \mathbf{v}_i, \mathbf{u}^{(t)}), \qquad \text{(edge update)} \qquad (4)$$

$$\bar{\mathbf{e}}_i^{(t+1)} = \rho^{e \to v}(E_i^{(t+1)}), \qquad \text{(edge to node aggregate)} \qquad (5)$$

$$\mathbf{v}_i^{(t+1)} = \phi^v(\bar{\mathbf{e}}_i^{(t+1)}, \mathbf{v}_i^{(t)}, \mathbf{u}^{(t)}), \qquad \text{(node update)} \qquad (6)$$

$$\bar{\mathbf{e}}^{(t+1)} = \rho^{e \to u}(E^{(t+1)}), \qquad \text{(edge to global aggregate)} \qquad (7)$$

$$\bar{\mathbf{v}}^{(t+1)} = \rho^{v \to u}(V^{(t)}), \qquad \text{(node to global aggregate)} \qquad (8)$$

$$\mathbf{u}^{(t+1)} = \phi^u(\bar{\mathbf{e}}^{(t+1)}, \bar{\mathbf{v}}^{(t+1)}, \mathbf{u}^{(t)}), \qquad \text{(global update)} \qquad (9)$$

where $E_i = \{\mathbf{e}_k, k \in \mathcal{N}_i^v\}$ is the set of attributes of edges connected to a specific node, E is the set of attributes of all edges, V is the set of attributes of all nodes, and $\mathcal{N}^v$ and $\mathcal{N}^e$ denote the set of indices of entities connected to a certain node or a certain edge, respectively. ϕ^e, ϕ^v, and ϕ^u are update functions that take the *environment* of the an entity as input and update the attribute of the entity, which could be stateful (Recurrent Neural Networks) or not; $\rho^{e \to v}$, $\rho^{e \to u}$, and $\rho^{v \to u}$ are aggregate functions that aggregate the attributes of multiple entities into an *aggregated* attribute which shares the same dimension with each entity. Although in this work, the definition of edges is limited to that connect exactly two nodes (bonds connecting two atoms), we could expand the notion of edges to include *hyperedges*, to connect more than two nodes (angles and torsions).

Finally, after a designated number of rounds of propagation (message passing), in the *readout stage*, $t = T$, a readout function f^r that takes the entire trajectory as input summarizes the information and yields the final output of desired dimensionality,

$$\hat{y} = f^r(\{\{\mathbf{v}^{(\mathbf{t})}, \mathbf{e}^{(t)}, \mathbf{u}^{(t)}\}, t = 1, 2, ..., T\}). \qquad (10)$$

2.3 Graph Batching

The number of nodes (atoms) in molecule graphs varies greatly and is usually much smaller than, say, the number of individuals in a social graph. For efficient backpropagation, especially on GPUs, molecule graphs need to be *combined* into larger ones, rather than *partitioned* or *padded* to the same size. This could be achieved by concatenating the attribute vectors of graphs and merging their adjacency matrices of graphs as

$$\widetilde{A}_{kl} = \begin{cases} (\{A\}_j)_{k - \sum\limits_{m < j} |\mathcal{V}_m|, l - \sum\limits_{m < j} |\mathcal{V}_m|}, \text{where} \sum\limits_{m < j} |\mathcal{V}_m| \le k, l < \sum\limits_{m < j+1} |\mathcal{V}_m|; \\ 0, \text{ elsewhere.} \end{cases} \qquad (11)$$

After choosing an appropriate batch size, which is the first dimension of $\widetilde{\mathcal{V}}$ and $\widetilde{\mathcal{A}}$, we repeat this process until another addition of small graph into the batch would result in $\sum\limits_i |\mathcal{V}_i|$ greater than the batch size, upon which the adjacency and the concatenated attributes will be padded to the batch size and another batch will be initialized.

2.4 Determination of atomic partial charges respecting a net charge constraint

One of the challenges in predicting atomic partial charges is to satisfy the constraint that their sum should equal to the total charge of the molecule:

$$\sum_i \hat{q}_i = \sum_i q_i = Q, \qquad (12)$$

where Q is the total (net) charge of the molecule, which could be positive, negative, or zero. Naively, we could either not explicitly encode this constraint and let the model "learn" it, or, as in Bleiziffer et al. [2], redistribute charge necessary to cancel any "excess charge" evenly to all atoms. Experimentally, none of these methods achieved satisfactory results when used with our model (with no constraint,

RMSE is around 0.280 e.) We instead adopted a trick proposed by Gilson et al. [7] and use our model to instead predict the first- and second-order derivatives of the potential energy E w.r.t. the atomic partial charge, which happens to correspond to the *electronegativity* e_i and *hardness* s_i of the atom in its chemical environment.

$$e_i \equiv \frac{\partial E}{\partial q_i}, s_i \equiv \frac{\partial E^2}{\partial^2 q_i}. \tag{13}$$

This problem could thus be formulated as follows: we use the graph net to make a prediction of the electronegativity and hardness, $\{\hat{e}_i, \hat{s}_i\}$, and the partial charges could be yielded by minimizing the second-order Taylor expansion of the potential energy contributed by atomic charges:

$$\{\hat{q}_i\} = \operatorname*{argmin}_{q_i} \sum_i \hat{e}_i q_i + \frac{1}{2} \hat{s}_i q_i^2, \tag{14}$$

subject to 12. Fortunately, using Lagrange multipliers, the solution to 14 could be given analytically by:

$$\hat{q}_i = -e_i s_i^{-1} + s_i^{-1} \frac{Q + \sum_i e_i s_i^{-1}}{\sum_j s_j^{-1}}, \tag{15}$$

whose Jacobian and Hessian are trivially easy to calculate. As a result, the prediction of $\{\hat{e}_i, \hat{s}_i\}$ could be optimized end-to-end using backpropagation.

3 Results and Discussion

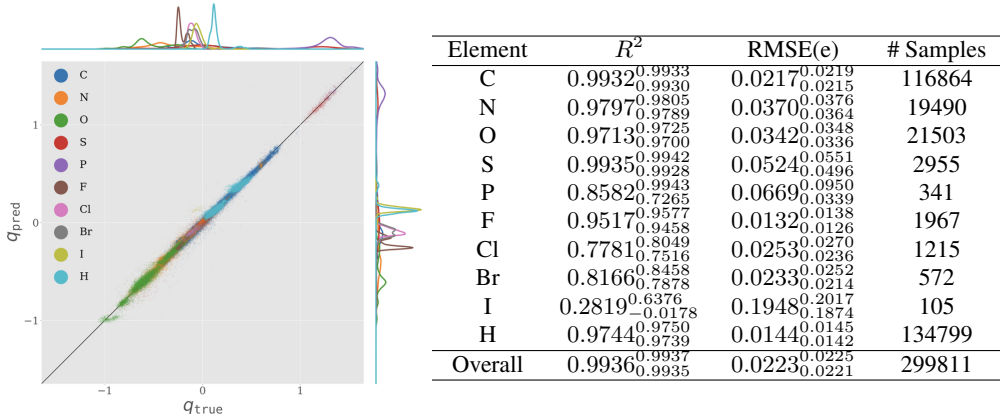| Element	R^2	RMSE(e)	# Samples
C	$0.9932^{0.9933}_{0.9930}$	$0.0217^{0.0219}_{0.0215}$	116864
N	$0.9797^{0.9805}_{0.9789}$	$0.0370^{0.0376}_{0.0364}$	19490
O	$0.9713^{0.9725}_{0.9700}$	$0.0342^{0.0348}_{0.0336}$	21503
S	$0.9935^{0.9942}_{0.9928}$	$0.0524^{0.0551}_{0.0496}$	2955
P	$0.8582^{0.9943}_{0.7265}$	$0.0669^{0.0950}_{0.0339}$	341
F	$0.9517^{0.9577}_{0.9458}$	$0.0132^{0.0138}_{0.0126}$	1967
Cl	$0.7781^{0.8049}_{0.7516}$	$0.0253^{0.0270}_{0.0236}$	1215
Br	$0.8166^{0.8458}_{0.7878}$	$0.0233^{0.0252}_{0.0214}$	572
I	$0.2819^{0.6376}_{-0.0178}$	$0.1948^{0.2017}_{0.1874}$	105
H	$0.9744^{0.9750}_{0.9739}$	$0.0144^{0.0145}_{0.0142}$	134799
Overall	$0.9936^{0.9937}_{0.9935}$	$0.0223^{0.0225}_{0.0221}$	299811

Figure 1 (Left): Predicted vs true partial charge of atoms in held-out test set color-coded by element types. A kernel density estimate of the distribution of charges for each element are plotted on the axes.

Table 1 (Right): R^2 and RMSE of the prediction and number of data points in held-out test set. The 95% confidence interval is also annotated.

We tested our model on a dataset consisting of 350 259 molecules in ChEMBL database, selected by Bleiziffer et al. [2] The reference charges are also calculated by Bleiziffer et al. [2] using DFT with dielectric permittivity $\epsilon = 4$. We randomly split the training and test set with 80:20 ratio. Random search on a limited hyperparameter space was conducted for hyperparameter tuning, with the hyperparameter set with highest 5-fold cross validation results chosen. On the test set, the error between the true and predicted value, RMSE ≈ 0.02 e, is roughly within the difference between DFT and AM1-BCC calculations, whereas it takes around 0.03 seconds to calculate the charges for a single molecule, which is approximately 500 times faster than AM1-BCC methods. We therefore argue that such method has the potential to replace AM1-BCC in calculating the charges for small molecules for MD simulation. Moreover, within the dataset (where the largest molecule has 63 atoms), we observed no positive correlation between the prediction error and the number of atoms in the molecule, indicating potential scalability of this model.

One potential problem of training our model on this dataset is that, charges yielded from DFT calculations are also dependent on the conformation of the molecule, although the conformation used

in the calculation is thought to be of lowest energy, thus is deterministic with regard to the topology. Currently, we are using AM1-BCC ELF10 methods to calculate partial charges that could be seen as independent of conformation. Apart from charges, such calculations could also yield per-bond (per-edge), and per-molecule (per-graph) features, namely partial bond orders such as Wiberg bond order and formation energy. We plan to continue this study by carrying out experiments to predict these features independently, jointly as multitask learning, and predicting some with others given.

4 Acknowledgments

5 Disclosures

JDC is a current member of the Scientific Advisory Board of OpenEye Scientific Software. YW and KL are co-founders and equity holders of Uli Statistical Learning. The Chodera laboratory receives or has received funding from multiple sources, including the National Institutes of Health, the National Science Foundation, the Parker Institute for Cancer Immunotherapy, Relay Therapeutics, Entasis Therapeutics, Silicon Therapeutics, EMD Serono (Merck KGaA), AstraZeneca, Vir Biotechnology, XtalPi, the Molecular Sciences Software Institute, the Starr Cancer Consortium, the Open Force Field Consortium, Cycle for Survival, a Louis V. Gerstner Young Investigator Award, and the Sloan Kettering Institute. A complete funding history for the Chodera lab can be found at `http://choderalab.org/funding`.

6 Disclaimers

The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

References

[1] Peter W. Battaglia et al. "Relational inductive biases, deep learning, and graph networks". In: *CoRR* abs/1806.01261 (2018). arXiv: 1806.01261. URL: http://arxiv.org/abs/1806.01261.

[2] Patrick Bleiziffer, Kay Schaller, and Sereina Riniker. "Machine Learning of Partial Charges Derived from High-Quality Quantum-Mechanical Calculations". In: *Journal of Chemical Information and Modeling* 58.3 (2018). PMID: 29461814, pp. 579–590. DOI: 10.1021/acs.jcim.7b00663. eprint: https://doi.org/10.1021/acs.jcim.7b00663. URL: https://doi.org/10.1021/acs.jcim.7b00663.

[3] Chi Chen et al. "Graph Networks as a Universal Machine Learning Framework for Molecules and Crystals". In: *Chemistry of Materials* 31.9 (2019), pp. 3564–3572. DOI: 10.1021/acs.chemmater.9b01294. eprint: https://doi.org/10.1021/acs.chemmater.9b01294. URL: https://doi.org/10.1021/acs.chemmater.9b01294.

[4] Evan N. Feinberg et al. "Spatial Graph Convolutions for Drug Discovery". In: *CoRR* abs/1803.04465 (2018). arXiv: 1803.04465. URL: http://arxiv.org/abs/1803.04465.

[5] Anna Gaulton et al. "ChEMBL: a large-scale bioactivity database for drug discovery". In: *Nucleic Acids Research* 40.D1 (Sept. 2011), pp. D1100–D1107. ISSN: 0305-1048. DOI: 10.1093/nar/gkr777. eprint: http://oup.prod.sis.lan/nar/article-pdf/40/D1/D1100/16955876/gkr777.pdf. URL: https://doi.org/10.1093/nar/gkr777.

[6] Justin Gilmer et al. "Neural Message Passing for Quantum Chemistry". In: *CoRR* abs/1704.01212 (2017). arXiv: 1704.01212. URL: http://arxiv.org/abs/1704.01212.

[7] Michael K. Gilson, Hillary S. R. Gilson, and Michael J. Potter. "Fast Assignment of Accurate Partial Atomic Charges: An Electronegativity Equalization Method that Accounts for Alternate Resonance Forms". In: *Journal of Chemical Information and Computer Sciences* 43.6 (2003). PMID: 14632449, pp. 1982–1997. DOI: 10.1021/ci034148o. eprint: `https://doi.org/10.1021/ci034148o`. URL: `https://doi.org/10.1021/ci034148o`.

[8] Torsten Hoffmann and Marcus Gastreich. "The next level in chemical space navigation: going far beyond enumerable compound libraries". In: *Drug discovery today* (2019).

[9] Steven Kearnes et al. "Molecular graph convolutions: moving beyond fingerprints". In: *Journal of computer-aided molecular design* 30.8 (2016), pp. 595–608.

[10] Martιén Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: `https://www.tensorflow.org/`.

[11] Benjamin Nebgen et al. "Transferable Dynamic Molecular Charge Assignment Using Deep Neural Networks". In: *Journal of Chemical Theory and Computation* 14.9 (2018). PMID: 30064217, pp. 4687–4698. DOI: 10.1021/acs.jctc.8b00524. eprint: `https://doi.org/10.1021/acs.jctc.8b00524`. URL: `https://doi.org/10.1021/acs.jctc.8b00524`.

[12] Prakash Chandra Rathi, R. Frederick Ludlow, and Marcel L. Verdonk. "Practical High-Quality Electrostatic Potential Surfaces for Drug Discovery Using a Graph-Convolutional Deep Neural Network". In: *Journal of Medicinal Chemistry* 0.0 (0). PMID: 31553186, null. DOI: 10.1021/acs.jmedchem.9b01129. eprint: `https://doi.org/10.1021/acs.jmedchem.9b01129`. URL: `https://doi.org/10.1021/acs.jmedchem.9b01129`.

[13] Andrew E. Sifain et al. "Discovering a Transferable Charge Assignment Model Using Machine Learning". In: *The Journal of Physical Chemistry Letters* 9.16 (2018). PMID: 30039707, pp. 4495–4501. DOI: 10.1021/acs.jpclett.8b01939. eprint: `https://doi.org/10.1021/acs.jpclett.8b01939`. URL: `https://doi.org/10.1021/acs.jpclett.8b01939`.

[14] Oliver T. Unke and Markus Meuwly. "PhysNet: A Neural Network for Predicting Energies, Forces, Dipole Moments, and Partial Charges". In: *Journal of Chemical Theory and Computation* 15.6 (2019). PMID: 31042390, pp. 3678–3693. DOI: 10.1021/acs.jctc.9b00181. eprint: `https://doi.org/10.1021/acs.jctc.9b00181`. URL: `https://doi.org/10.1021/acs.jctc.9b00181`.