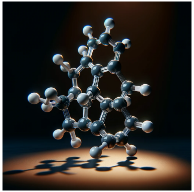
Predicting Atom Interactions with AI

Ludwig Schneider

Eric and Wendy Schmidt AI-Postdoctoral Fellowship
Pritzker School of Molecular Engineering
Data Science Institute University of Chicago

April 30, 2024



Moelcular Interaction: Non-Bonded I

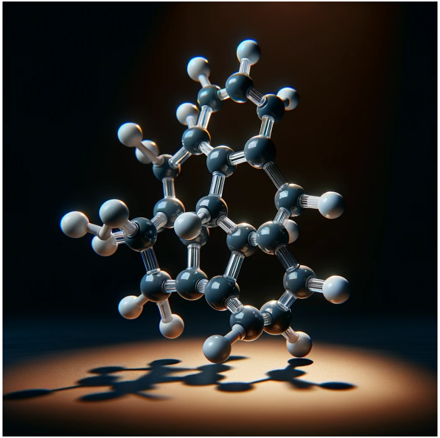
- Newtons Equation of Motion

$$m_i \frac{\mathrm{d}^2}{\mathrm{d}t^2}\vec{r}_i = \vec{F}_i$$

- determine forces $\vec{F}$
- OPLS: functional form for the forcefield

$$V_{\mathrm{nb}}(r_{ij}) = \varepsilon_{ij}\left(\frac{\sigma_{ij}}{r_{ij}^{12}} - \frac{\sigma_{ij}}{r_{ij}^{6}}\right) + \frac{q_i q_j e^2}{4\pi\varepsilon_0 r_{ij}}$$

$$\varepsilon_{ij} = \sqrt{\varepsilon_i \varepsilon_j} \quad \sigma_{ij} = \sqrt{\sigma_i \sigma_j}$$



Challenge: determine ε_i, σ_i, q_i, and m_i

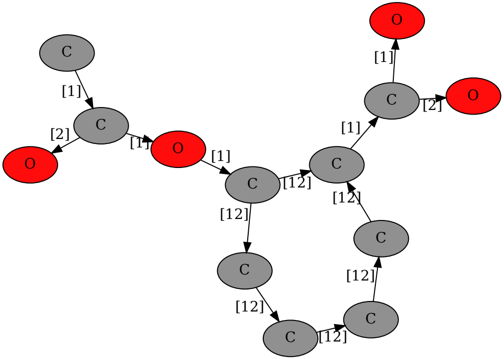
W. L. Jorgensen, D. S. Maxwell, and J. Tirado-Rives, Journal of the American Chemical Society **118**, 11225–11236 (1996)

Moelcular Interaction: Non-Bonded Interactions II

- determine ε_i, σ_i, q_i, and m_i
  - element (carbon, hydrogen)
  - molecular environment
- we know some values
- SMARTS based rules
- Quantum Mechanic simulations
- chemical space is vast, we want to interpolate

Your Challenge

Use ML to interpolate from known molecules!



W. L. Jorgensen, D. S. Maxwell, and J. Tirado-Rives, Journal of the American Chemical Society **118**, 11225–11236 (1996)

Details of the Basic Challenge

- 3000 molecular graphs
- known input parameters
- supervised learning possible
  1. predict non-bonded parameters
  2. predict uncertaintiy in interpolation
- you are free to choose your model
- you are free to choose tech stack
- make results permutation invariant

- `https://github.com/uchicago-dsi/ai-sci-hackathon-2024`
- `molecular/intro/talk.pdf` this slide deck
- `molecular/README.md` starting point
- `molecular/explain_graph_data.py` explanation of data set
- `molecular/data.json` training data
- molecular graph as `networkx` graphs

How to get Started?

1. Understand the Data

- visualize graphs
- transform to tech stack

Tech Stack Options

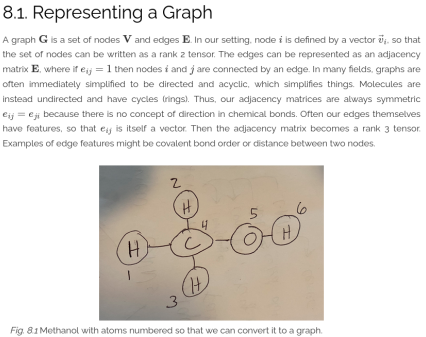
- Flax/JAX with Jraph
- PyTorch with Torch-Geometric
- Tensorflow with GNN

Tools

- visualization: https://graphviz.org/
- python: https://networkx.org/

- New to GNN?
- New to Molecular AI?
- Dmol introduction!

`https://dmol.pub/dl/gnn.html`



8.1. Representing a Graph

A graph $\mathbf{G}$ is a set of nodes $\mathbf{V}$ and edges $\mathbf{E}$. In our setting, node i is defined by a vector $\vec{v}_i$, so that the set of nodes can be written as a rank 2 tensor. The edges can be represented as an adjacency matrix $\mathbf{E}$, where if $e_{ij} = 1$ then nodes i and j are connected by an edge. In many fields, graphs are often immediately simplified to be directed and acyclic, which simplifies things. Molecules are instead undirected and have cycles (rings). Thus, our adjacency matrices are always symmetric $e_{ij} = e_{ji}$ because there is no concept of direction in chemical bonds. Often our edges themselves have features, so that e_{ij} is itself a vector. Then the adjacency matrix becomes a rank 3 tensor. Examples of edge features might be covalent bond order or distance between two nodes.

Fig. 8.1 Methanol with atoms numbered so that we can convert it to a graph.

How we Evaluate You!

- prepare a final presentation
- convince us your solution is best
- include model details and results
- 7 minutes to present
- 2 minutes for our questions

Wednesday 5PM: we give you additional data

- DO NOT TRAIN with this data
- Send your result back as soon as you can!
- `molecular/final_evaluation.py`
- report output in your presentation!

Example Test Data

- `molecular/validation_example.json`
- `molecular/permutation_example.json`
- `molecular/permutation_example_masked.json`
- Try it!

What if you solve the challenge too fast?

Traditional

- predict bond class for atom
- look-up table for parameters
- `ffbonded.itp`

GNN approach

- predict parameters directly
- bonds: edge features
- angles & dihedrals ?

- bonds: 2 atoms: 2 parameters

$$V_b(r_{ij}) = k_a(r_{ij} - b_0)^2$$

- angles: 3 atoms: 2 parameters

$$V_a(\theta) = k_\theta(\theta - \theta_0)^2 \quad \theta = \cos^{-1}(\vec{r}_{ij} \cdot \vec{r}_{jk} / (||\vec{r}_{ij}||||\vec{r}_{jk}||))$$

- dihedral: 4 atoms: 8 parameters

$$V_d(\phi) = \sum_{i=1,2,3,4} V_i/2[1 - (-1)^i \cos(i \cdot \phi - \phi_i)]$$

Happy Hacking!

Questions?