Teaching Quantum Chemistry to a Deep Learning Model

Transformers for the many body Schrodinger Equation

Jorge Munoz Laredo, Angel Flores, Daniel Paredes

January 6, 2026

Outline

The Schrödinger equation

On 1926 Schrodinger derived his equatin:

$$\hat{H}\,\Psi = E\Psi \tag{1}$$

- Ψ is a complex value function called **wave function**.
- $\hat{H}$ is called the **Hamiltonian Operator**.

Hamiltonian

$$\hat{H} = \frac{\hat{\vec{P}}^2}{2m} + \hat{V} = -\frac{\hbar^2}{2m}\nabla^2 + \hat{V} \tag{2}$$

- Find the electrostatic potential V of the system.

Many-Body System

When considering n bodies, we have:

$$\hat{H}\psi(\mathbf{x}_1, \ldots, \mathbf{x}_n) = E\psi(\mathbf{x}_1, \ldots, \mathbf{x}_n) \tag{3}$$

With $\mathbf{x}_i = \{\mathbf{r}_i, \sigma\}$, where $\mathbf{r}_i \in \mathbb{R}^3$ is the position of each particle and $\sigma \in \{\uparrow . \downarrow\}$ is the so called spin.

Considerations

- Each particle interact with all the another particles.
- For atoms, consider all the protons, electrons and neutrons.
- Solution obey physical laws.

Setting up the Hamiltonian

The first step is obtain a practical form of the **Hamiltonian**.

- Kinetic energy: $T = -\frac{1}{2}\sum_{i=1}^{N} \nabla_i^2$.
- Electron-electron repulsion: $V_{ee} = \sum_{i<j} \frac{1}{r_{ij}}$.

$$\hat{H} = -\sum_{i=1}^{N} \frac{1}{2}\nabla_i^2 - \sum_{i=1}^{N} \frac{Z_I}{|\mathbf{r}_i - \mathbf{R}_I|} + \sum_{1\leq i<j\leq N} \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|} \tag{4}$$

Fermi-Dirac statistics

All the fermions follow the Fermi-Dirac Statistics, this is.

- Exchanging two electrons flips the wavefunction's sign: $\Psi(\dots i, j \dots) = -\Psi(\dots j, i \dots)$.

Slater Determinant

Enforce it using a determinant.

$$\psi = \begin{vmatrix} \phi_1^k(\mathbf{x}_1) & \dots & \phi_1^k(\mathbf{x}_n) \\ \vdots & & \vdots \\ \phi_n^k(\mathbf{x}_1) & \dots & \phi_n^k(\mathbf{x}_n) \end{vmatrix} \tag{5}$$

Where ϕ are called spin orbitals

Kato cusp conditions, Jastrow Factor

The potential are:

$$\sum \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|}$$

- Coulomb potentials cause a sharp cusp in Ψ when particles overlaps.

Jastrow Factor $\exp(\mathcal{J})$

In this work we are going to use this specific form:

$$\mathcal{J}_\theta(x) = \sum_{i<j;\sigma_i=\sigma_j} -\frac{1}{4}\frac{\alpha_{par}^2}{\alpha_{par} + |\mathbf{r}_i - \mathbf{r}_j|} + \sum_{i,j;\sigma_i\neq\sigma_j} -\frac{1}{2}\frac{\alpha_{anti}^2}{\alpha_{anti} + |\mathbf{r}_i - \mathbf{r}_j|} \quad (6)$$

Loss: Variational Principle

Variational principle states:

$$E[\Psi] = \frac{\langle \Psi \mid H \mid \Psi \rangle}{\langle \Psi \mid \Psi \rangle} \geq E_0$$

Minimizing $E[\Psi]$ drives the ansatz toward the ground state.

$$E[\Psi] = \mathcal{L}(\Psi_\theta) = \frac{\langle \Psi_\theta | \hat{H} | \Psi_\theta \rangle}{\langle \Psi_\theta | \Psi_\theta \rangle} = \frac{\int d\mathbf{R} \Psi^*(\mathbf{R}) \hat{H} \Psi(\mathbf{R})}{\int d\mathbf{R} \Psi^*(\mathbf{R}) \Psi(\mathbf{R})}$$

Define:

$$p_\theta(\mathbf{R}) = |\Psi_\theta(\mathbf{R})|^2 \frac{1}{\int d\mathbf{R}' \Psi_\theta^2(\mathbf{R}')} \wedge E_L(\mathbf{R}) = \Psi_\theta^{-1}(\mathbf{R}) \hat{H} \Psi_\theta(\mathbf{R})$$

Then:

$$\mathcal{L}_\theta = \mathbb{E}_{\mathbf{R} \sim p_\theta}[E_L(\mathbf{R})] \tag{7}$$

Variational Monte Carlo

Quantum Monte Carlo

With the samples $\mathbf{R}_1, \ldots, \mathbf{R}_M \sim p_\theta(\mathbf{R})$ we can make:

$$\mathcal{L}_\theta = \mathbb{E}_{x \sim p_\theta}[E_L(x)] \approx \frac{1}{M} \sum_{i=1}^{M} E_L(\mathbf{R}_k) \tag{8}$$

With:

$$E_L(\mathbf{R}_k) = \frac{\hat{H}\psi(\mathbf{R}_k)}{\psi(\mathbf{R}_k)} = -\frac{1}{2}\frac{\nabla^2\psi(\mathbf{R_k})}{\psi(\mathbf{R}_k)} + V(\mathbf{R}_k)$$

Obtain $\mathbf{R}_k \rightarrow$ Metropolis-Hastings Algorithm

Metropolis-Hastings Algorithm

Goal: Generate many samples $\mathbf{R} \sim \rho$, Requirement: $C\rho$

1. $\mathbf{X}_0 \in E$ arbitrary:
2. Propose $\mathbf{X}' = \mathbf{X}_0 + \eta$,where $\eta \sim q(\eta)$, (Normal Gaussian)
3. Compute the quantity:

$$A(\mathbf{X_0}, \mathbf{X}') = \min\left(1, \frac{\rho(\mathbf{X}')}{\rho(\mathbf{X}_0)}\right)$$

4. Generate a uniform number $U \in [0, 1]$. If: $U < A(\mathbf{X}_0, \mathbf{X}')$ then $\mathbf{X_1} = \mathbf{X}'$, otherwise try another $\mathbf{X}'$. Accept or decline.

Metropolis-Hastings Algorithm

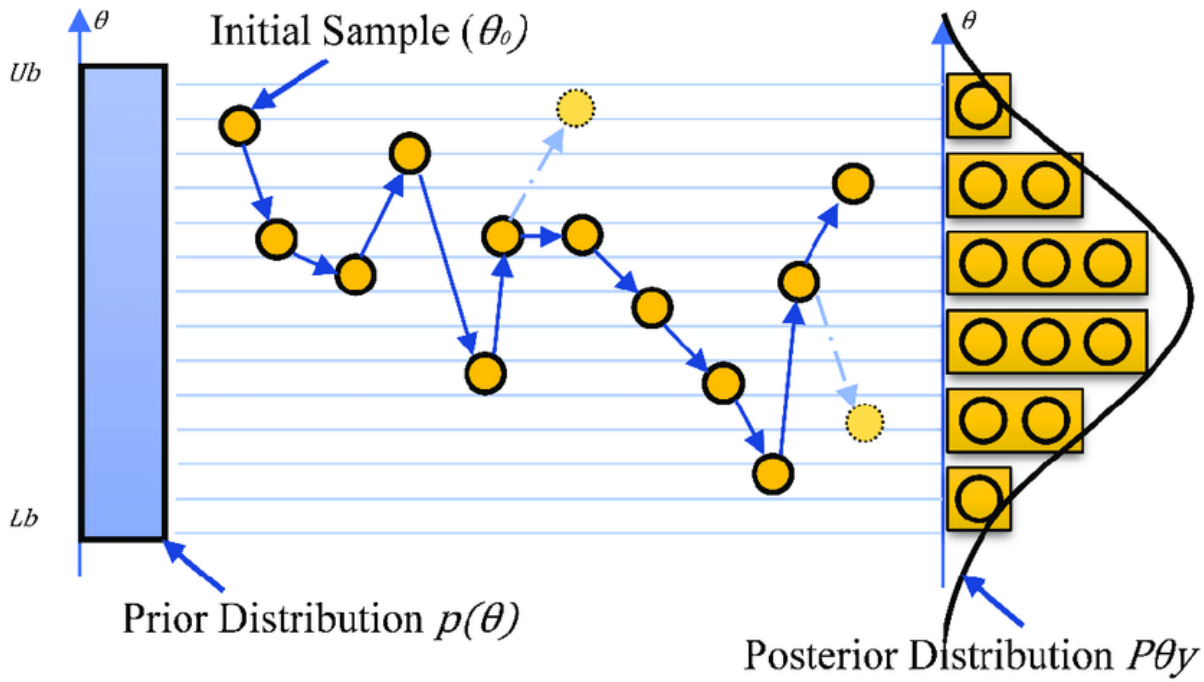


Figure 1: Metropolis-Hastings Walkers

The model is able to obtain its own data (no dataset $\mathcal{D}$), able to compute:

$$\mathcal{L}_\theta = \mathbb{E}_{\mathbf{R}\sim|\Psi_\theta|^2}[E_L(\mathbf{R})]$$

,

Loss function problems

$$\mathcal{L}_\theta = \mathbb{E}_{\mathbf{R}\sim|\Psi_\theta|^2}[E_L(\mathbf{R})]$$

,

- Three Backpropagations
- $|\Psi_\theta|^2$ changes over time.
- You are just optimizing just the **energy**, not the wave function itself.

Solution: **Log Derivative Trick**

$$\nabla_\theta \mathcal{L} = 2\mathbb{E}_{\mathbf{R}\sim\Psi^2}[(E_L(\mathbf{R}) - \mathbb{E}_p[E_L])\nabla_\theta \log \psi]$$

REINFORCE!

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau\sim\pi}[(R(\tau) - b)\nabla_\theta \log \pi_\theta(\tau)]$$

Outline

Transformer Architecture

Figure 2: Caption

Attention on the room

Multi Head Attention d the embedding dimension, n_h the number of attention heads, d_h the dimension per head, and $\mathbf{h}_t \in \mathbb{R}^d$ the hidden dimension. The learnable matrices are:

$$W^Q, W^K, W^V \in \mathbb{R}^{d_h n_h \times d}$$

$$\mathbf{k}_i = \mathbf{W}^k \mathbf{h}_i, \mathbf{q}_i = \mathbf{W}^q \mathbf{h}_i, \mathbf{v}_i = \mathbf{W}^v \mathbf{h}_i$$

$$[\mathbf{q_1}, \mathbf{q_2}, \ldots, \mathbf{q_{n_h}}] = \mathbf{q}$$
$$[\mathbf{k_1}, \mathbf{k_2}, \ldots, \mathbf{k_{n_h}}] = \mathbf{k}$$
$$[\mathbf{v_1}, \mathbf{v_2}, \ldots, \mathbf{v_{n_h}}] = \mathbf{v}$$

In the $i-th$ head:

$$\mathbf{o}_{t,i} = \sum_{j=1}^{t} \text{Softmax}\left(\frac{\mathbf{q}_{t,i}^T \mathbf{k}_{j,i}}{\sqrt{d_h}}\right) \mathbf{v}_{j,i} \tag{9}$$

$W^O \in \mathbb{R}^{d \times d_h n_h}$ the output projection matrix.

$$\mathbf{u}_t = W^O [\mathbf{o}_{t,1}; \mathbf{o}_{t,2}; \ldots; \mathbf{o}_{t,n_h}]$$

Outline

Fermi Net Architecture

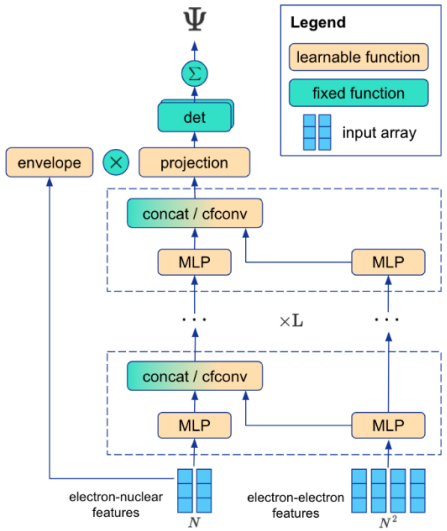


Figure 3: Fermi Net Architecture

- Learn the orbitals ϕ
- Learn the coefficients.

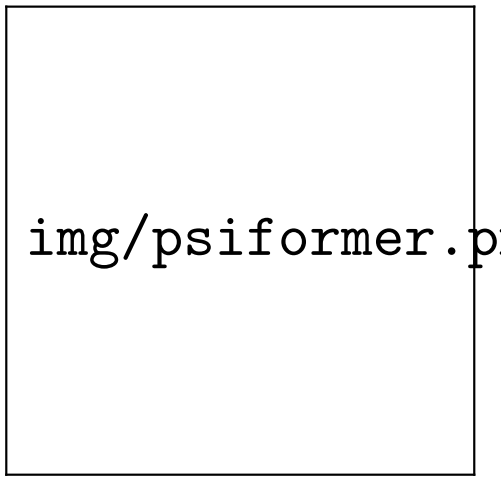
FermiNet baseline

Orbital

FermiNet computes:

$$\psi(\mathbf{r}_1^{\uparrow}, \ldots, \mathbf{r}_{n^{\downarrow}}^{\downarrow}) = \sum_k \omega_k \det\left[\phi_{ij}^{k\uparrow}\right] \det\left[\phi_{ij}^{k\downarrow}\right] \tag{10}$$

Psi Former Architecture



- Learn the Jastrow factor $\mathcal{J}$
- Learn the coefficients ω

Figure 4: Psi Former Architecture

Implementation : Laplacian Computation

Implementation : Determinant Stability

Implementation : Backpropagation on Markov Chain

Implementation : Optimizer Choice

Implementation : GPU Parallelization

Model Size

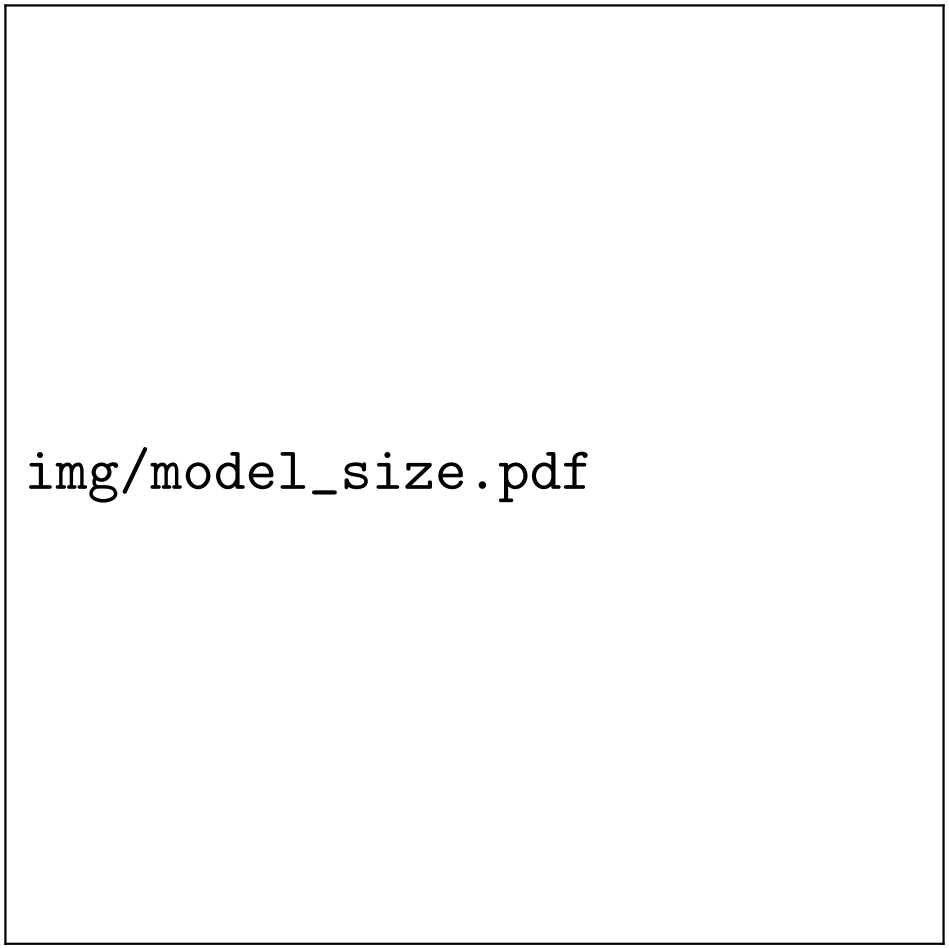


Figure 5: Model Size

Results: Convergence Curve

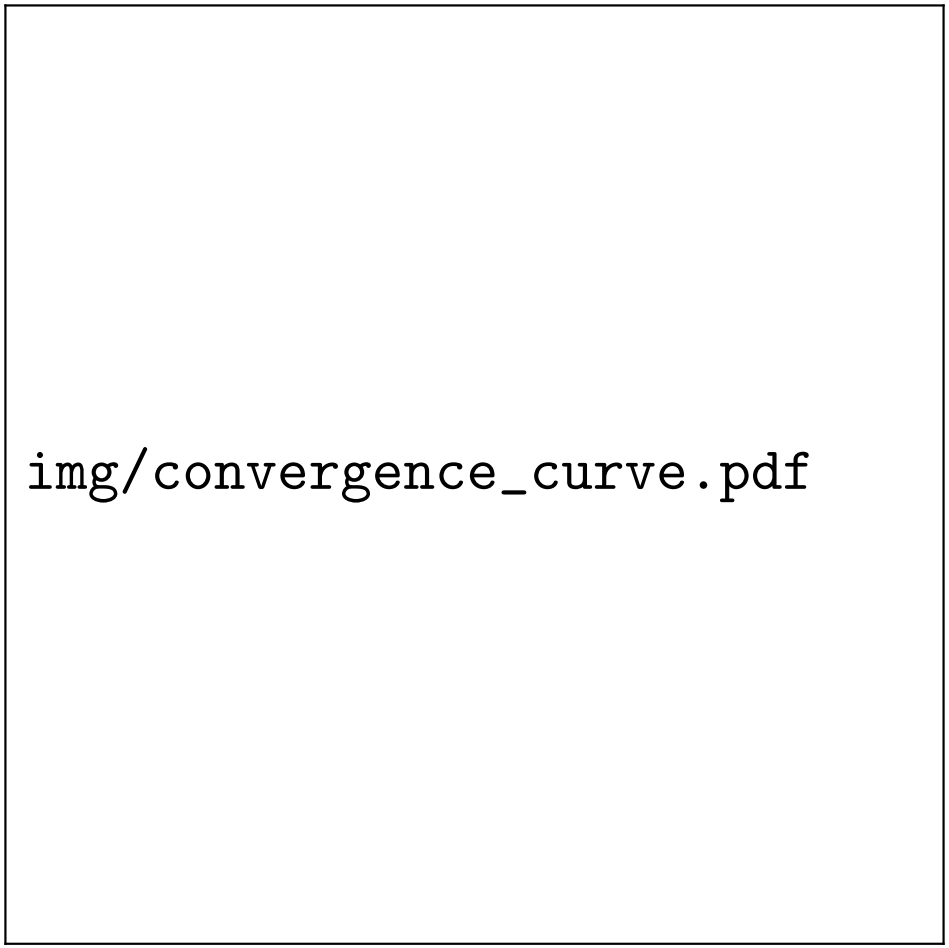


Figure 6: Convergence Curve

Results: Energy Estimates

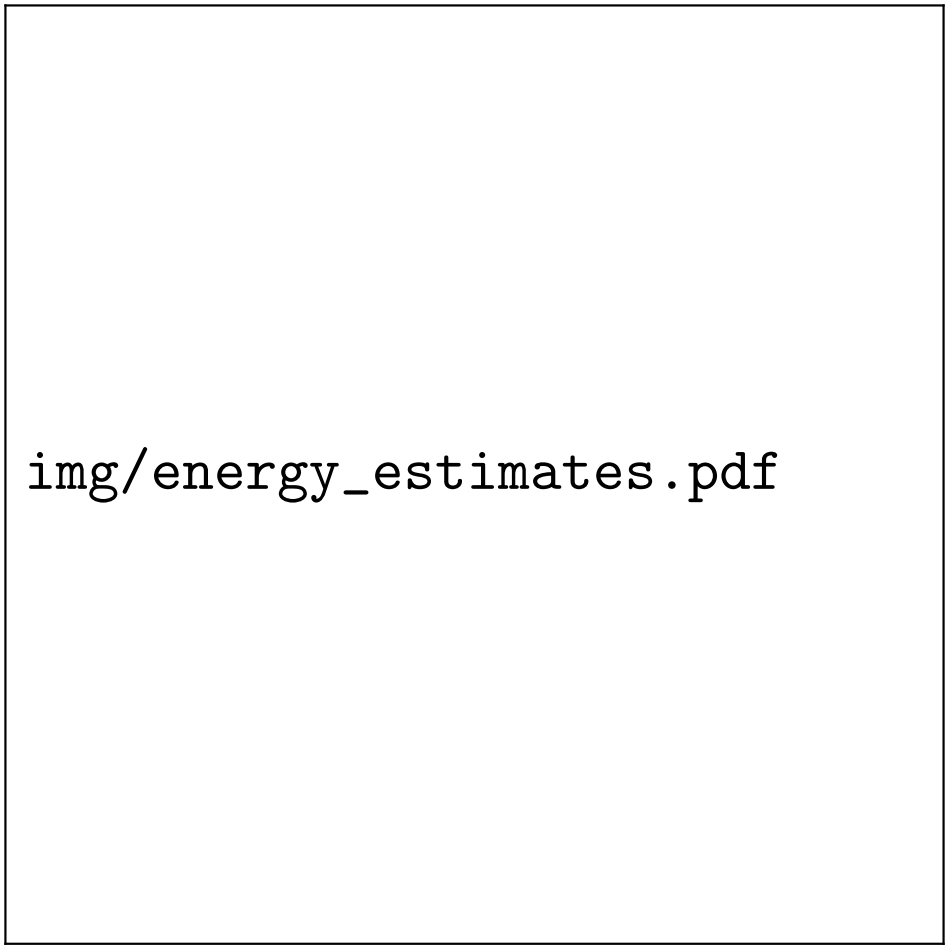


Figure 7: Energy Estimates

Improvements

- Use a better optimizer.
- Use a better model.
- Use a better sampling method.
- Use a better model.
- Use a better model.

Thanks!