SchNet – An interpretable atomistic neural network

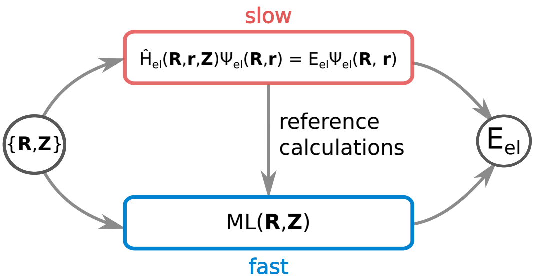
Kristof T. Schütt

Berliner Zentrum für MASCHINELLES LERNEN

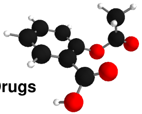
Technische Universität Berlin

The challenges in a nutshell

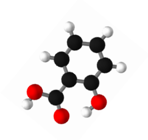
Quantum chemistry is too slow.



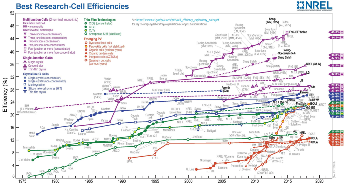
Chemical compound space is too big.



Drugs



Food processing



Solar cells **Batteries**

Solar fuel **Catalysis**

The Deep Tensor Neural Network (DTNN) framework

1. Embed atom types

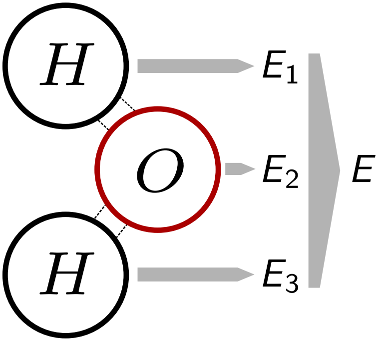
$$\mathbf{x}_i^{(0)} = \mathbf{A}_{Z_i} \in \mathbb{R}^d$$

2. Add interactions

$$\mathbf{x}_i^{(t+1)} = \mathbf{x}_i^{(t)} + \sum_{j \neq i} \mathbf{v}^{(t)} \left(\mathbf{x}_j^{(t)}, \|\mathbf{r}_i - \mathbf{r}_j\| \right)$$

3. Predict via atom-wise contributions:

$$\hat{E} = \sum_{i=1}^{n_{\text{atoms}}} e(\mathbf{x}_i^{(T)})$$



*Schütt, Arbabzadah, Chmiela, Müller, Tkatchenko (2017), Nature Communications **8**, 13890*

The Deep Tensor Neural Network (DTNN) framework

1. Embed atom types

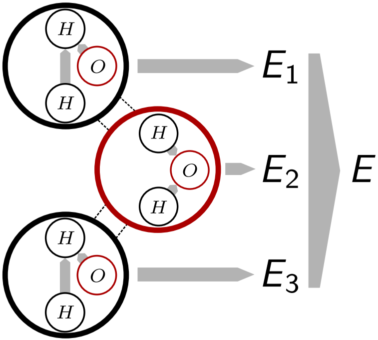
$$\mathbf{x}_i^{(0)} = \mathbf{A}_{Z_i} \in \mathbb{R}^d$$

2. Add interactions

$$\mathbf{x}_i^{(t+1)} = \mathbf{x}_i^{(t)} + \sum_{j \neq i} \mathbf{v}^{(t)} \left(\mathbf{x}_j^{(t)}, \|\mathbf{r}_i - \mathbf{r}_j\| \right)$$

3. Predict via atom-wise contributions:

$$\hat{E} = \sum_{i=1}^{n_{\text{atoms}}} e(\mathbf{x}_i^{(T)})$$



Schütt, Arbabzadah, Chmiela, Müller, Tkatchenko (2017), Nature Communications ***8****, 13890*

The Deep Tensor Neural Network (DTNN) framework

1. **Embed atom types**

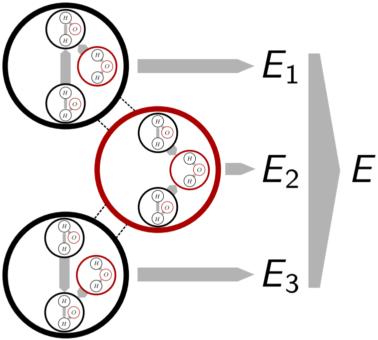
$$\mathbf{x}_i^{(0)} = \mathbf{A}_{Z_i} \in \mathbb{R}^d$$

2. **Add interactions**

$$\mathbf{x}_i^{(t+1)} = \mathbf{x}_i^{(t)} + \sum_{j \neq i} \mathbf{v}^{(t)} \left(\mathbf{x}_j^{(t)}, \|\mathbf{r}_i - \mathbf{r}_j\| \right)$$

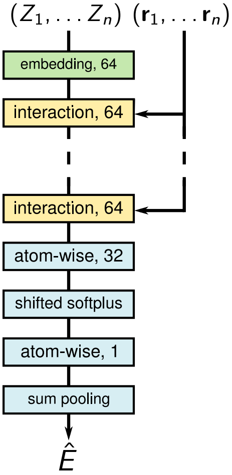
3. **Predict via atom-wise contributions:**

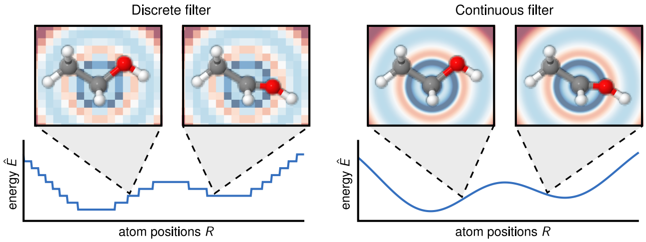
$$\hat{E} = \sum_{i=1}^{n_{\text{atoms}}} e(\mathbf{x}_i^{(T)})$$



*Schütt, Arbabzadah, Chmiela, Müller, Tkatchenko (2017), Nature Communications **8**, 13890*

SchNet – a continuous-filter CNN for atomistic systems



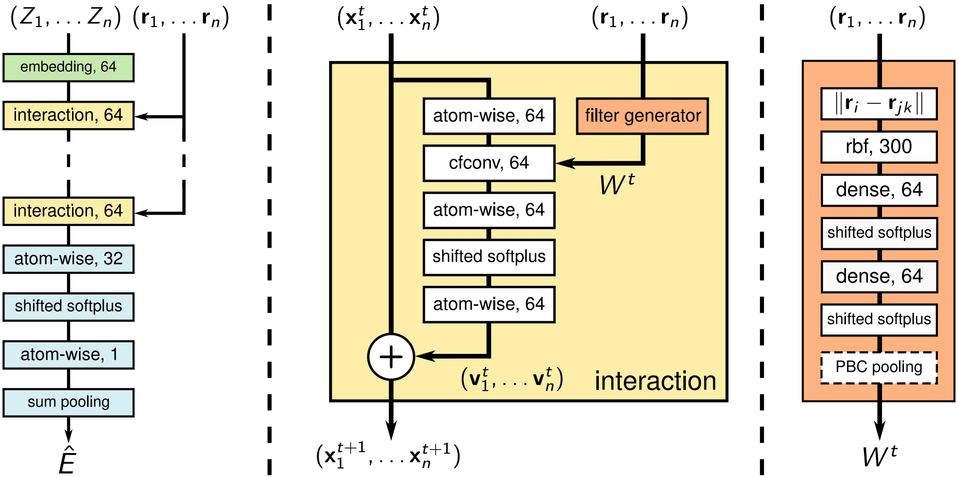


$$(\mathbf{x} * W)(\mathbf{r}_i) = \sum_{j=1}^{N_{\text{atom}}} \mathbf{x}_j^{(t)} \circ \underbrace{W^{(t)}_{[\mathbf{r}_i - \mathbf{r}_j]}}_{\text{parameter tensor}}$$

$$(\mathbf{x} * W)(\mathbf{r}_i) = \sum_{j=1}^{N_{\text{atom}}} \mathbf{x}_j^{(t)} \circ \underbrace{W^{(t)}(\mathbf{r}_i - \mathbf{r}_j)}_{\text{neural network}}$$

Schütt, Kindermans, Sauceda, Chmiela, Tkatchenko, Müller (2017). NeurIPS 30.

SchNet – a continuous-filter CNN for atomistic systems



Schütt, Kindermans, Sauceda, Chmiela, Tkatchenko, Müller (2017). NeurIPS 30.

Property-specific output layers

	Total energy U_0	**Dipole moment** $\|\mu\|$
SchNet output layer	$E = \sum_i E(\mathbf{x}_i)$	$\mu = \sum_i q(\mathbf{x}_i)(\mathbf{r}_i - \mathbf{r}_0)$

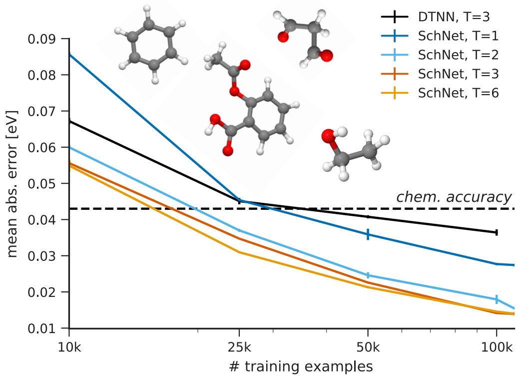
QM9 – 110k ref. calculations – *mean abs. errors*

SchNet (T=6, SGDR)	**9.5 meV**	**0.017 Debye**
HIP-NN[1]	11.1 meV	–
Message-passing NN[2]	19.5 meV	0.030 Debye

[1] N. Lubbers, J.S. Smith, K. Barros (2018). Hierarchical modeling of molecular energies using a deep neural network. The Journal of Chemical Physics, 148(24), 241715.
[2] J. Gilmer, S.S. Schoenholz, P.F. Riley, O. Vinyals, G.E. Dahl. Neural-Message Passing for Quantum Chemistry (2017). ICML.

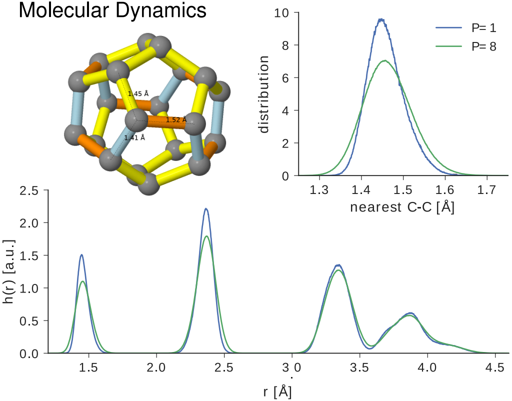
SchNet accelerates quantum simulations

Chemical Compound Space



Predictions of organic molecules at
chemical accuracy

Molecular Dynamics

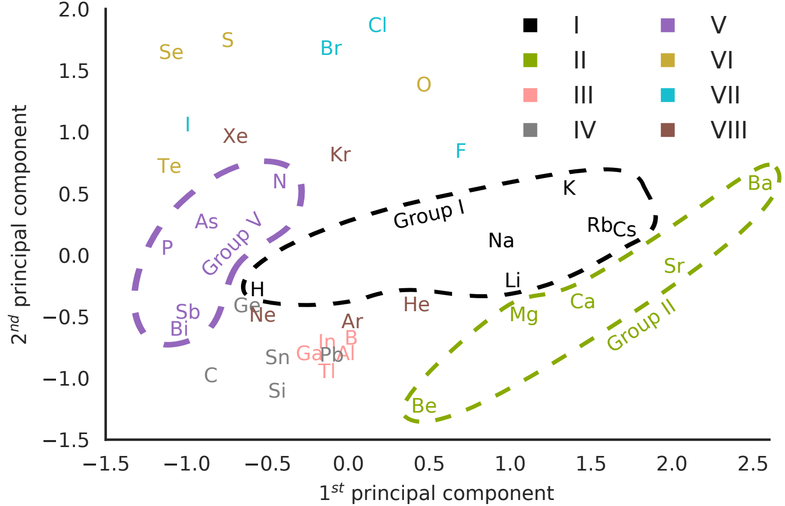


1.25ns PIMD trajectory of the fullerene C_{20}
7 years ⇒ 7 hours

ANI-1 data, 10.1M: 23.9 meV

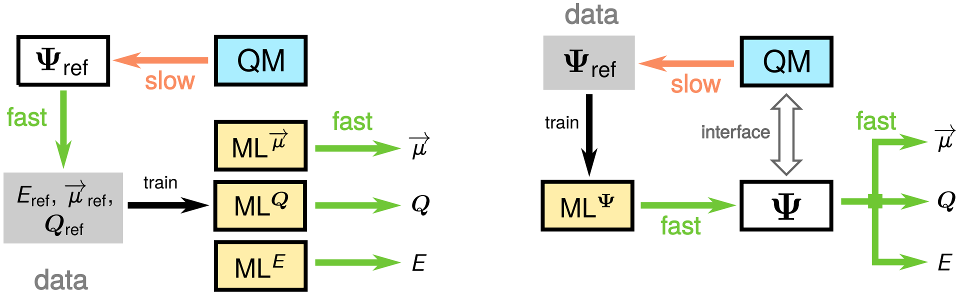
*Schütt, Sauceda, Kindermans, Tkatchenko, Müller (2018). JCP **148** (24); Schütt et al (2018). JCTC, **15** (1).*

Learning the periodic table of elements

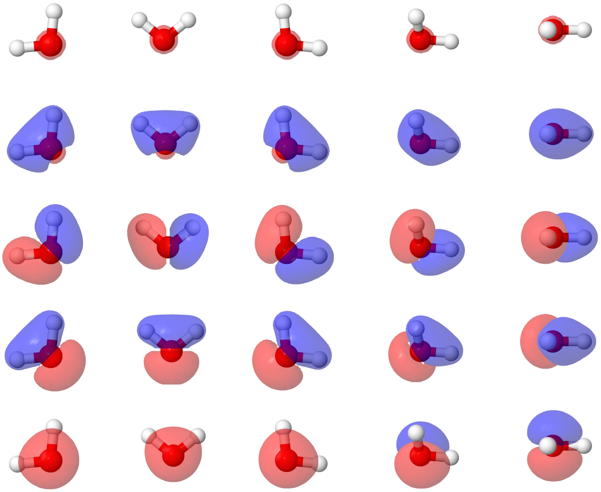


Trained on 60k bulk crystals from the Materials Project.

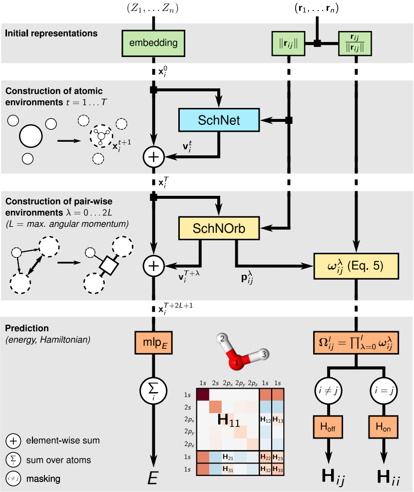
Can we obtain all properties at once?

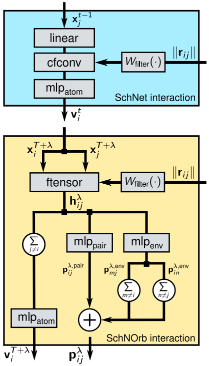


Rotational equivarience of molecular orbitals



SchNOrb – SchNet for Orbitals



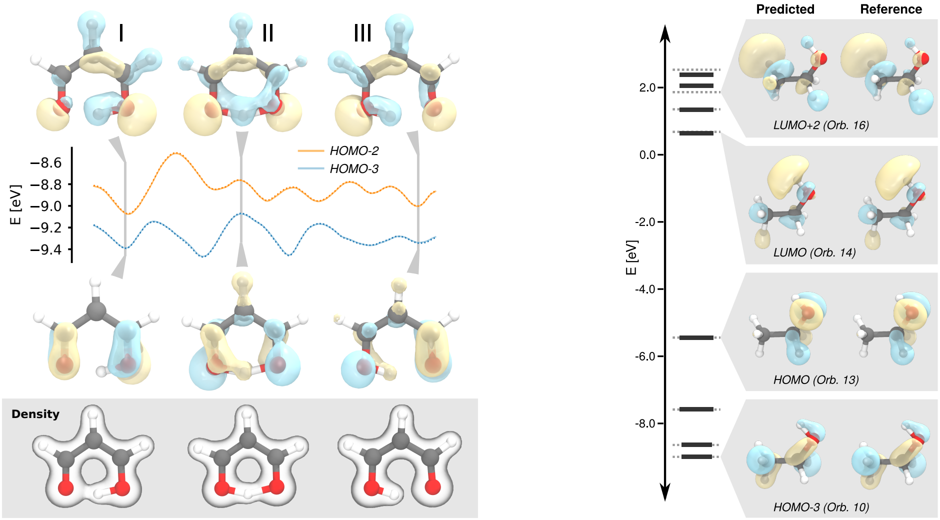


$$\Omega^l_{ij} = \prod_{\lambda=0}^{l} \omega^\lambda_{ij} \quad \text{with } 0 \le l \le 2L \tag{4}$$

$$\omega^\lambda_{ij} = \begin{cases} \mathbf{p}^\lambda_{ij} \otimes \mathbb{1}_D & \text{for } \lambda = 0 \\ \left[\mathbf{p}^\lambda_{ij} \otimes \frac{\mathbf{r}_{ij}}{\|\mathbf{r}_{ij}\|}\right] \mathbf{W}^\lambda & \text{for } \lambda > 0 \end{cases}, \tag{5}$$

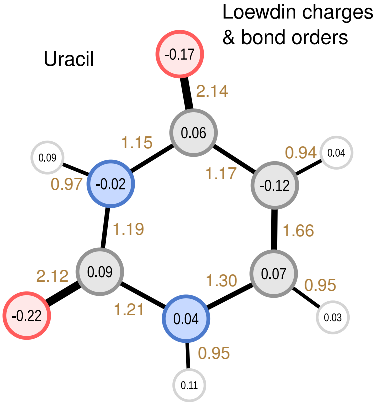
K. T. Schütt, M. Gastegger, A. Tkatchenko, K.-R. Müller, R. J. Maurer. arXiv:1906.10033. 2019

SchNOrb – SchNet for Orbitals

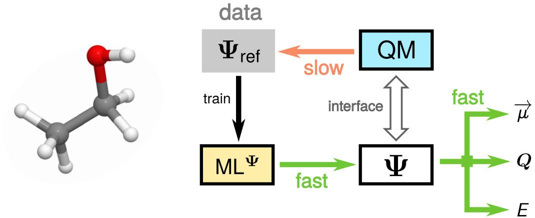


K. T. Schütt, M. Gastegger, A. Tkatchenko, K.-R. Müller, R. J. Maurer. arXiv:1906.10033. 2019

SchNOrb – Predicting derived properties



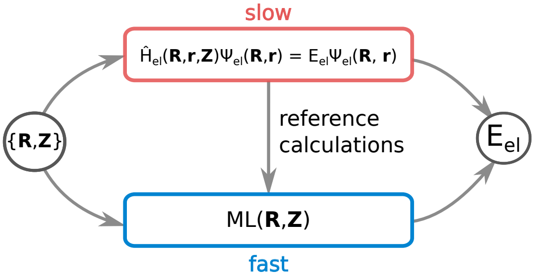
Energy component	MAE [meV]	MAE [%]
HF	21.4	0.01
MP2 correlation	83.1	17.19
HF + MP2	92.6	0.06



K. T. Schütt, M. Gastegger, A. Tkatchenko, K.-R. Müller, R. J. Maurer. arXiv:1906.10033. 2019

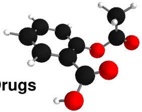
Challenges

Quantum chemistry is too slow.

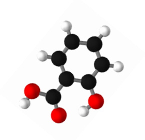


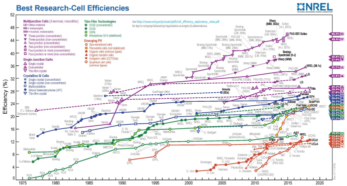
✓

Chemical compound space is too big.



Drugs





Solar cells **Batteries**

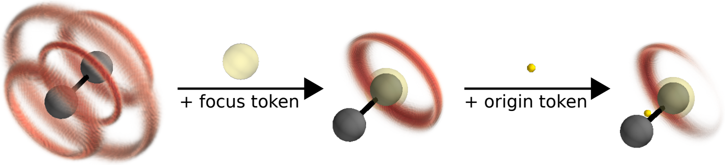
Food processing **Solar fuel** **Catalysis**

Symmetry-adapted generation for the target discovery of molecules

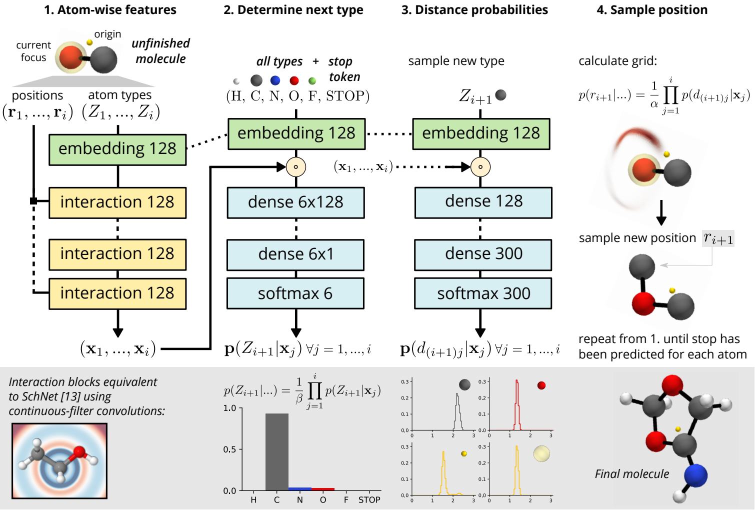
Goal: Draw from distribution of molecules in equilibirum $p(\mathbf{R}, \mathbf{Z})$.

$$p(\mathbf{R}_{\leq n}, \mathbf{Z}_{\leq n}) = \prod_{i=1}^{n} p\left(\mathbf{r}_i, Z_i | \mathbf{R}_{\leq i-1}, \mathbf{Z}_{\leq i-1}\right)$$

$$\begin{aligned} p\left(\mathbf{r}_{n+1}, Z_{n+1} | \mathbf{R}_{\leq n}, \mathbf{Z}_{\leq n}\right) &= p(\mathbf{r}_{n+1} | Z_{n+1}, \mathbf{R}_{\leq n}, \mathbf{Z}_{\leq n})\, p(Z_{n+1} | \mathbf{R}_{\leq n}, \mathbf{Z}_{\leq n}) \\ &\approx \left[\frac{1}{\alpha} \prod_{i=0}^{n} p(d_{n+1,i} | Z_{n+1}, \mathbf{R}_{\leq n}, \mathbf{Z}_{\leq n})\right] p(Z_{n+1} | \mathbf{R}_{\leq n}, \mathbf{Z}_{\leq n}) \end{aligned}$$

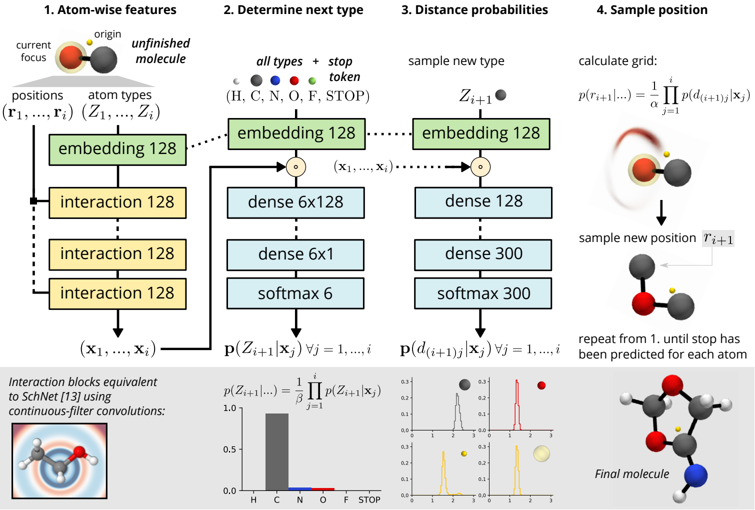


G-SchNet



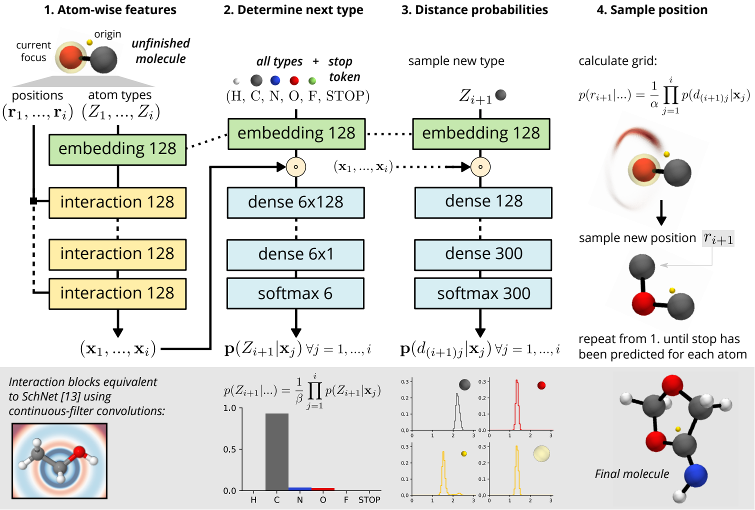
N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

G-SchNet



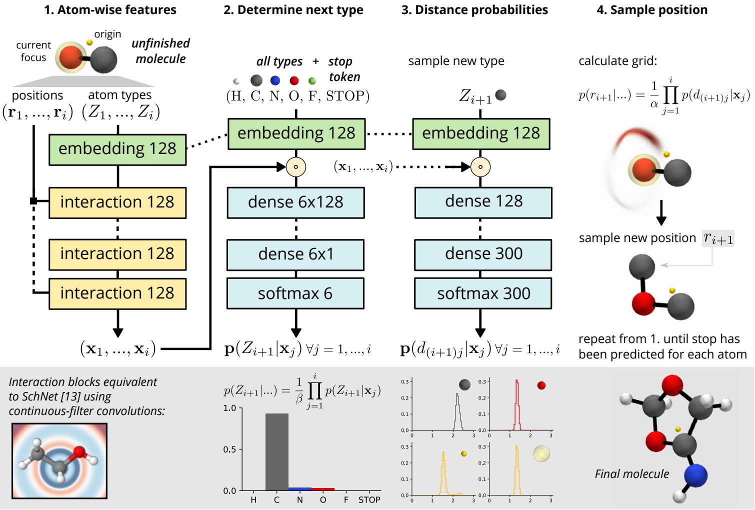
N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

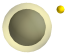
G-SchNet



N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

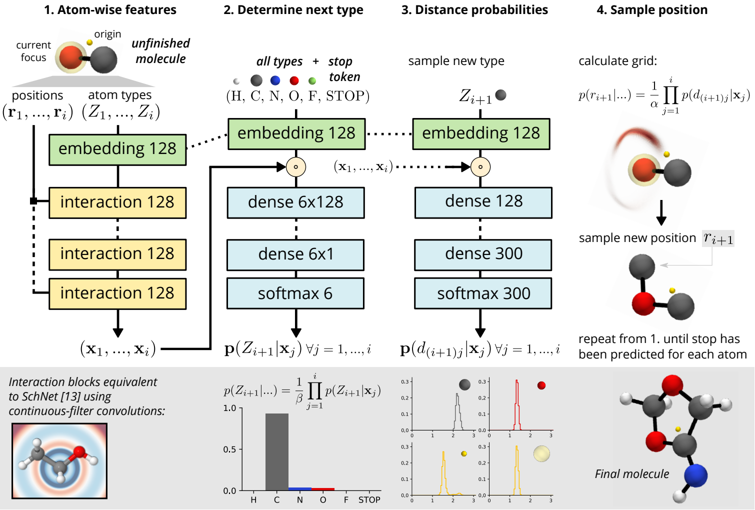
G-SchNet

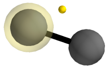




N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

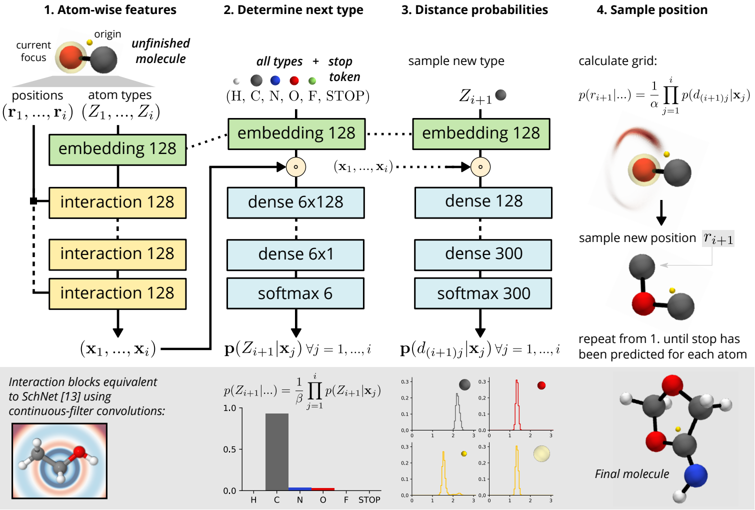
G-SchNet

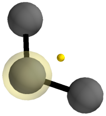




N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

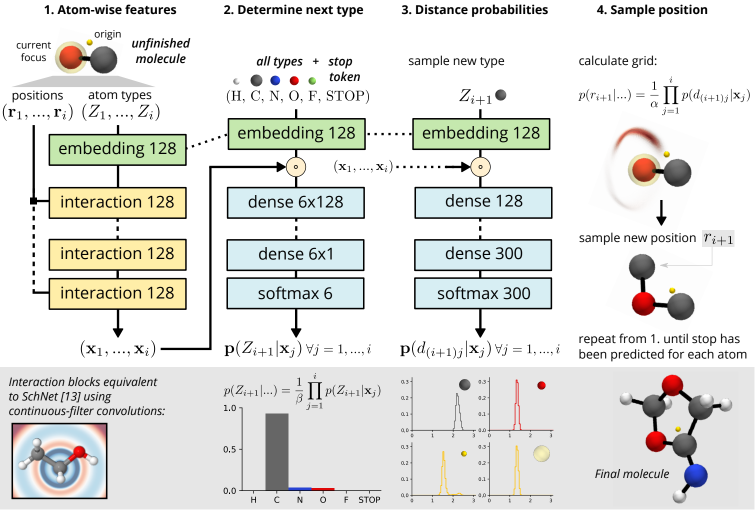
G-SchNet

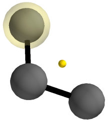




N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

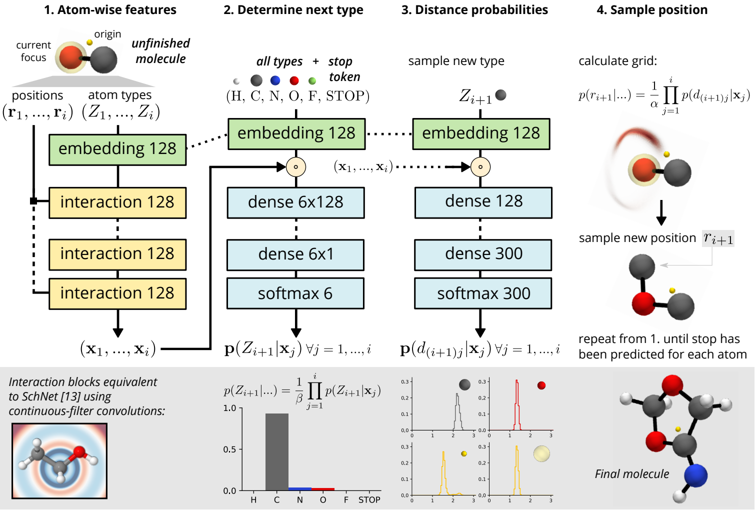
G-SchNet

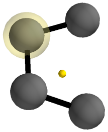




N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

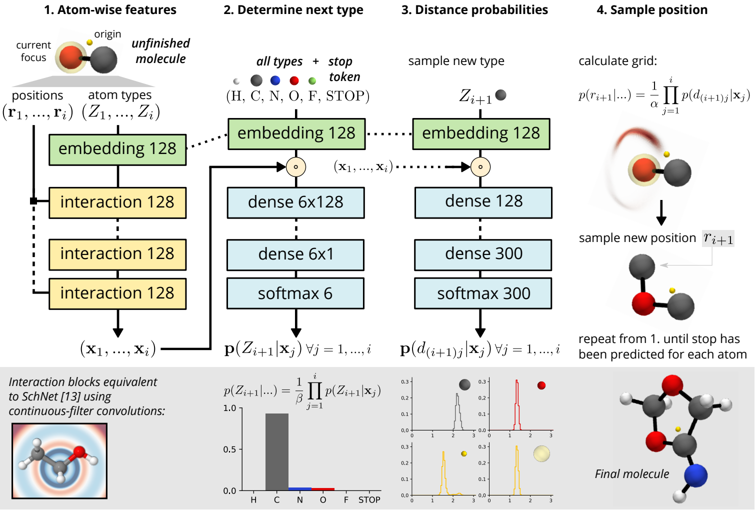
G-SchNet

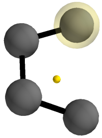




N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

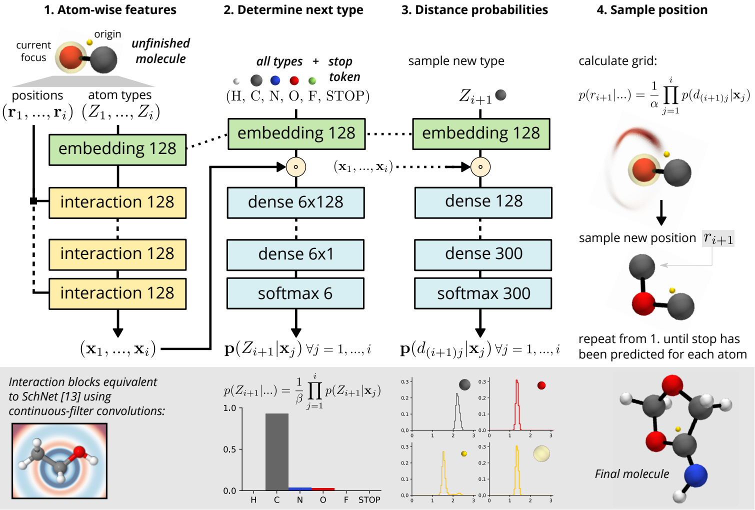
G-SchNet

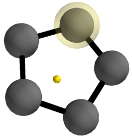




N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

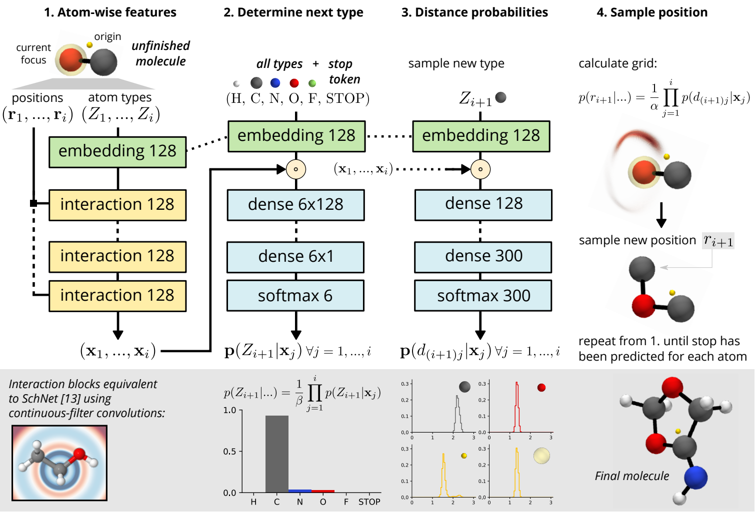
G-SchNet

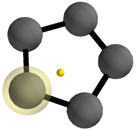




N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

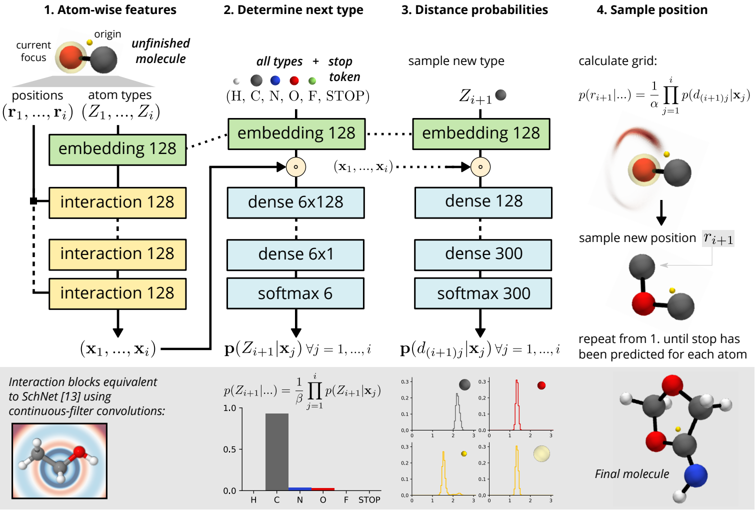
G-SchNet

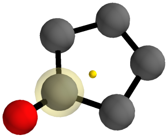




N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

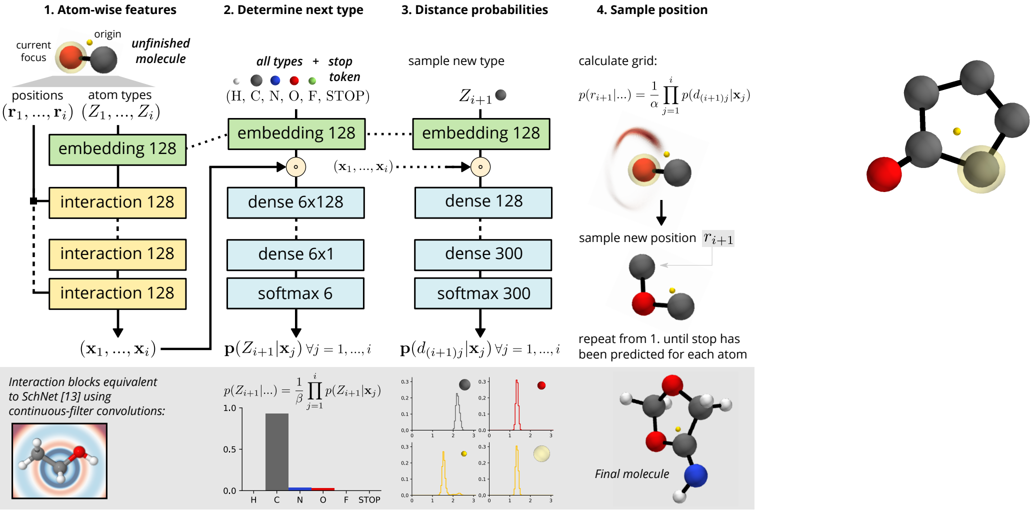
G-SchNet





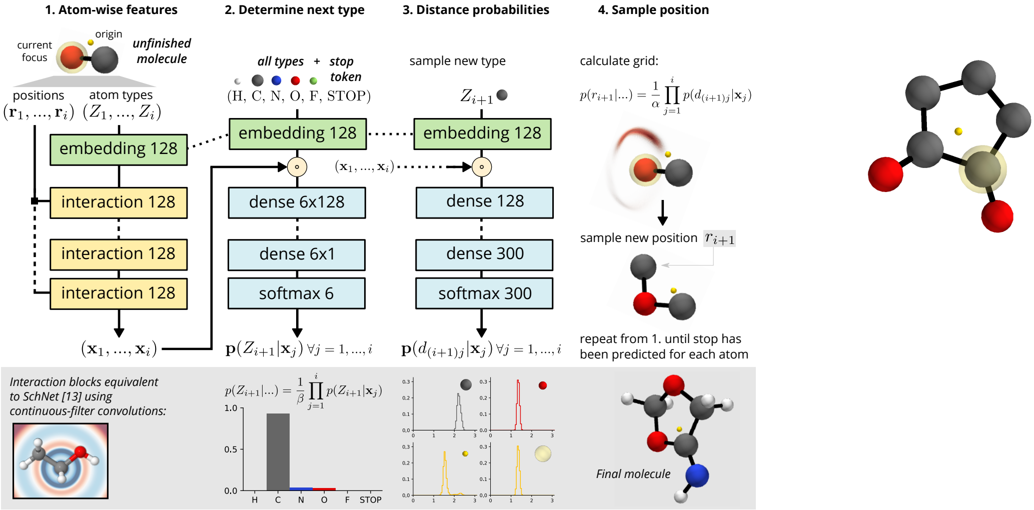
N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

G-SchNet



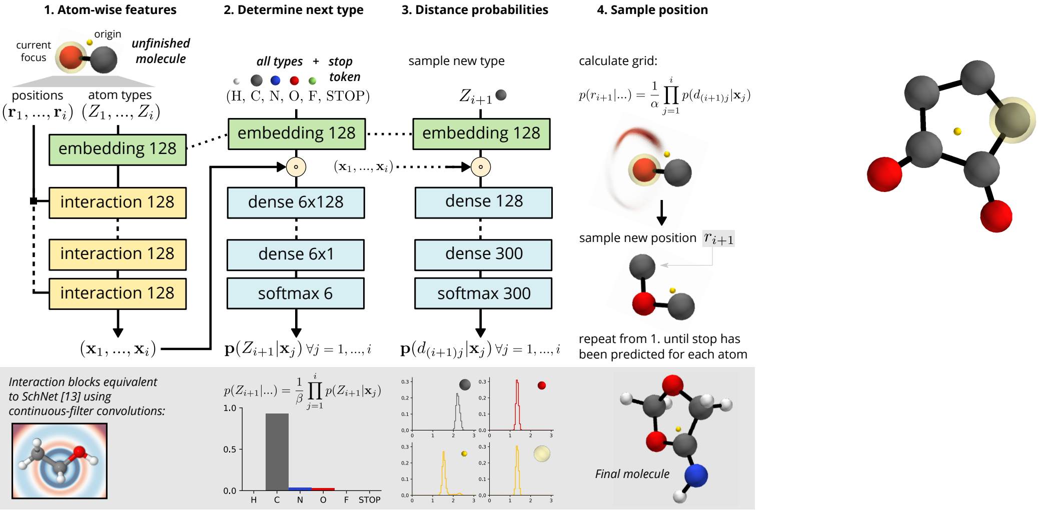
N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

G-SchNet



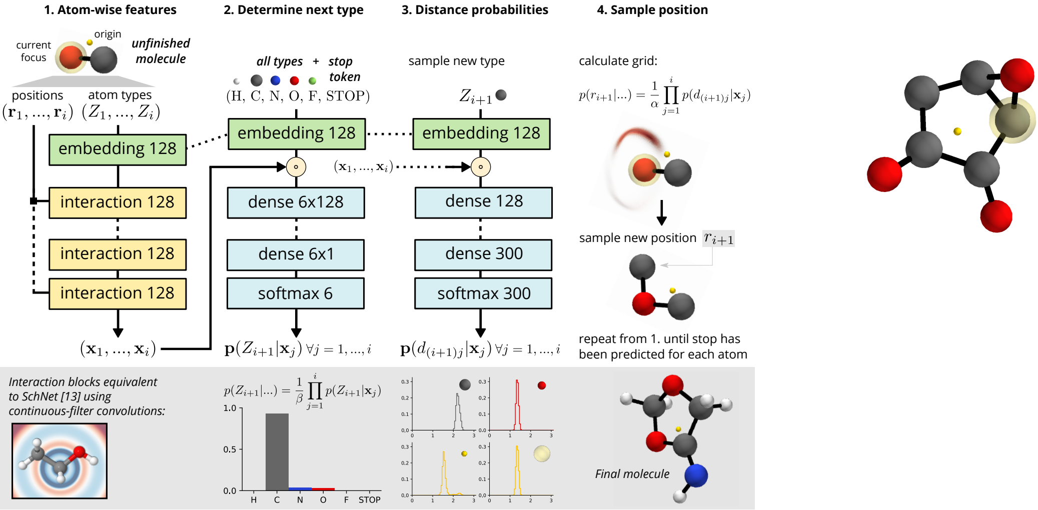
N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

G-SchNet



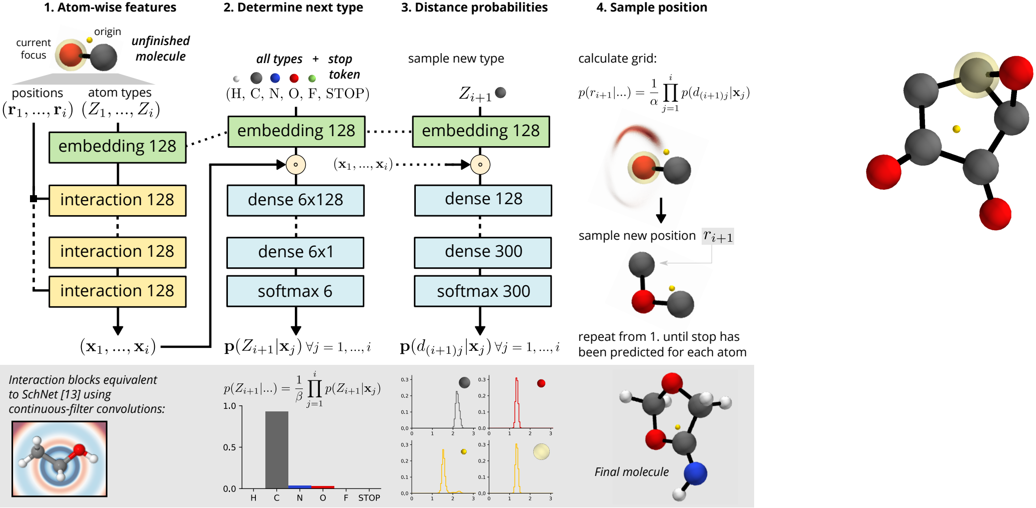
N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

G-SchNet



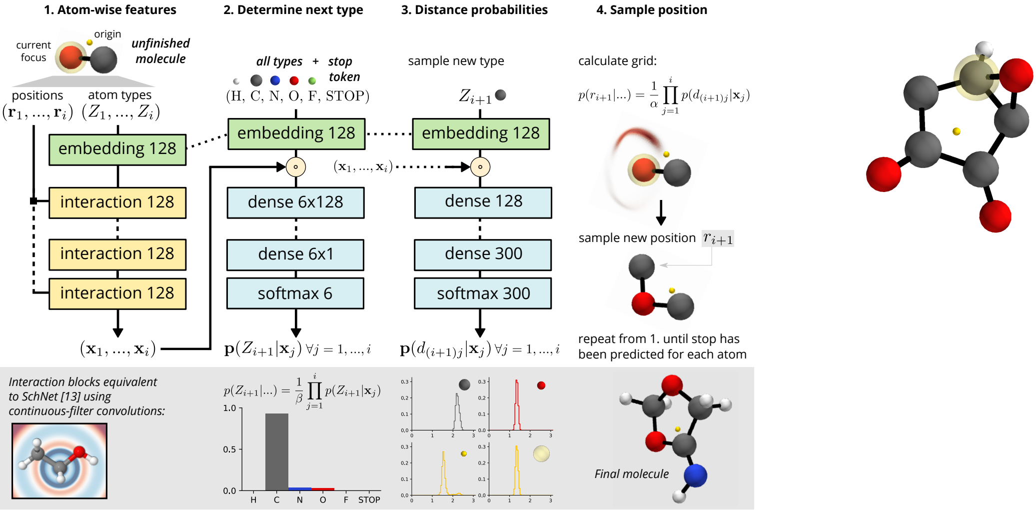
N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

G-SchNet



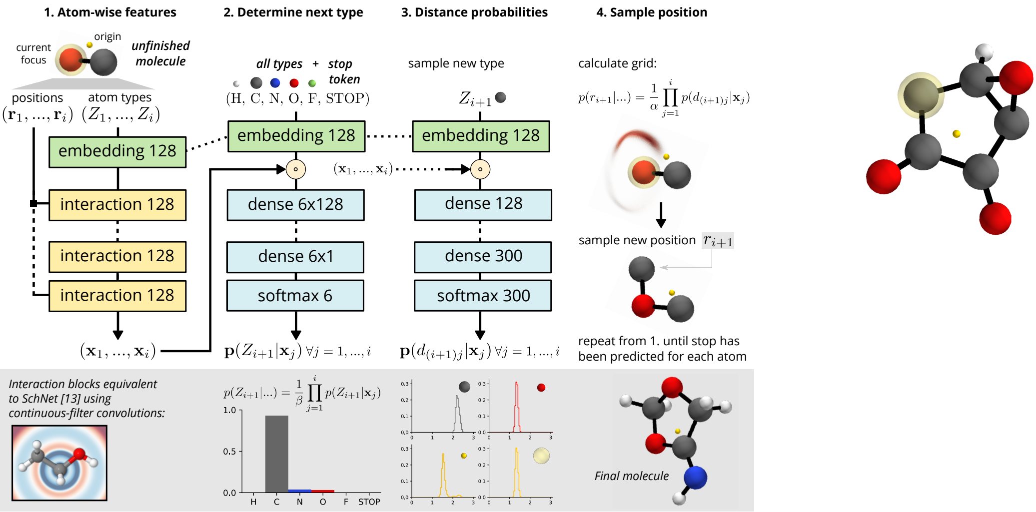
N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

G-SchNet



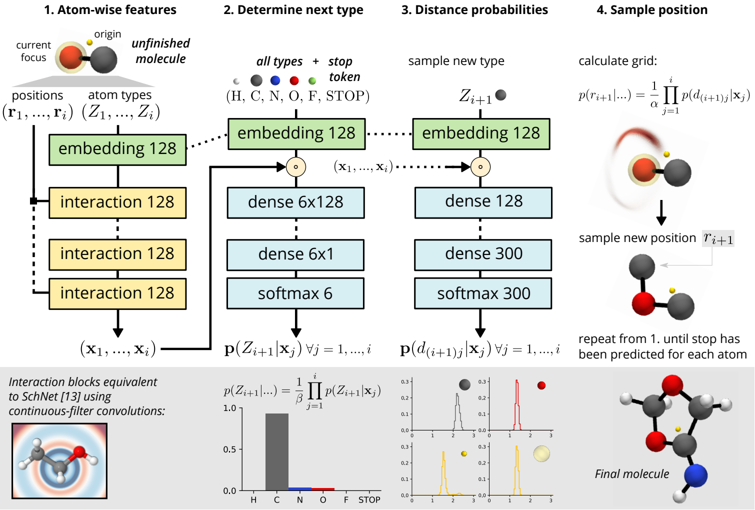
N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

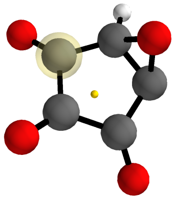
G-SchNet



N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

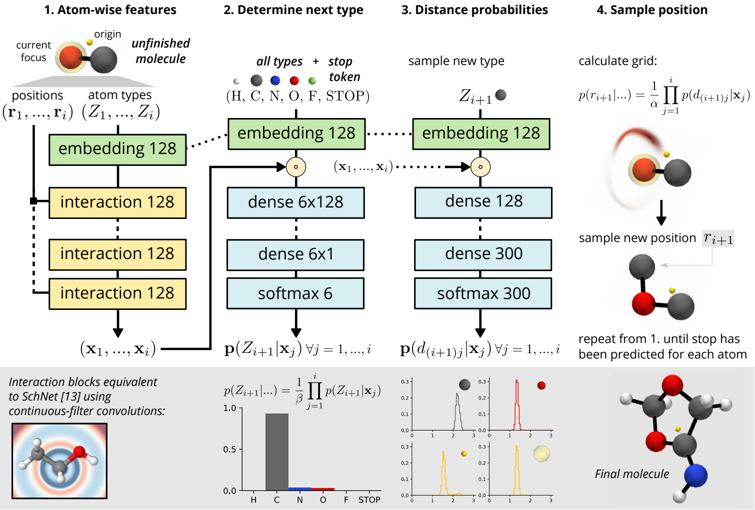
G-SchNet

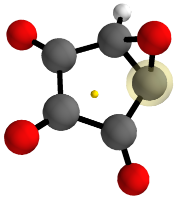




N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

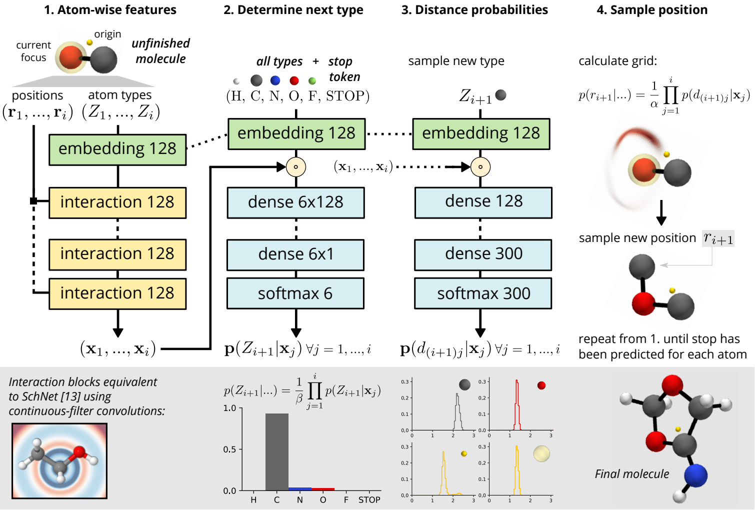
G-SchNet

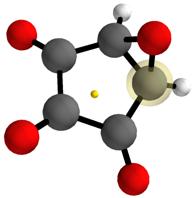




N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

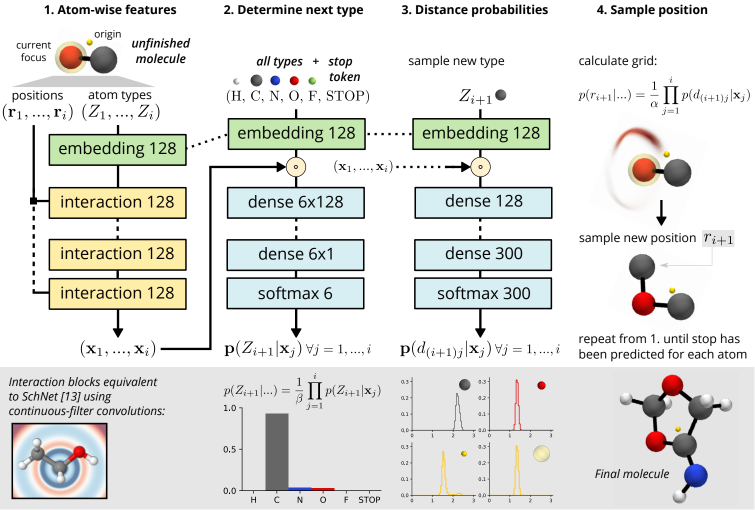
G-SchNet

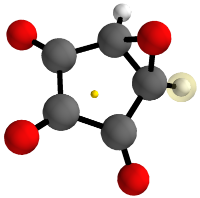




N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

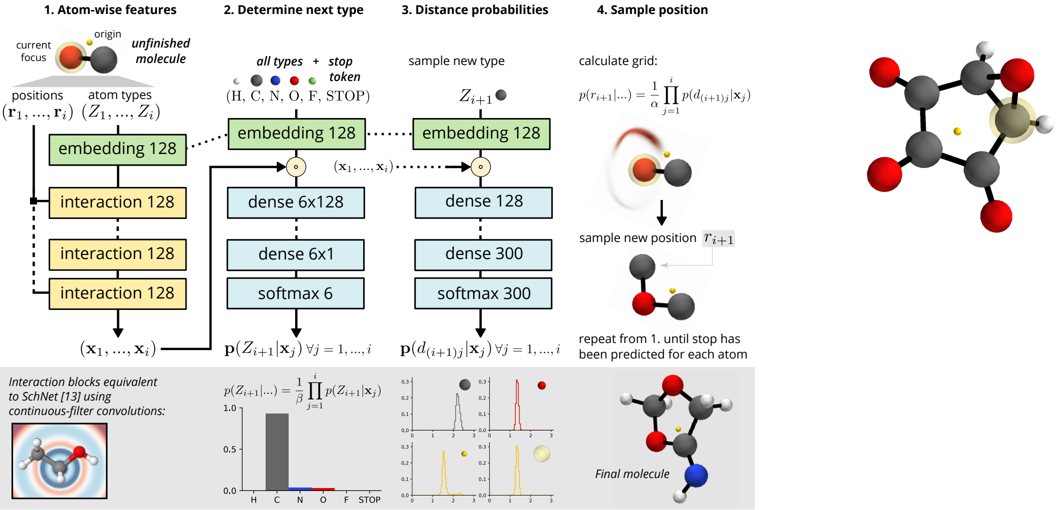
G-SchNet





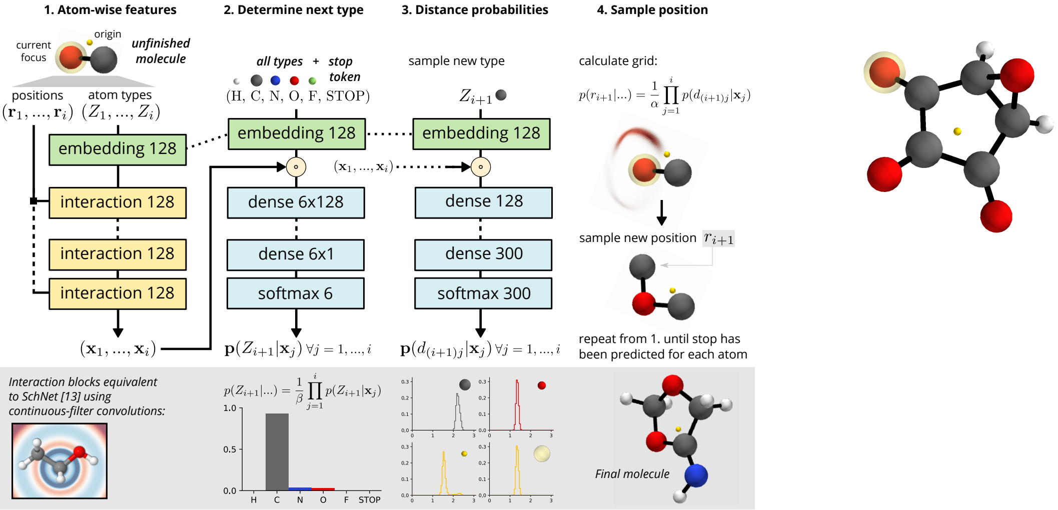
N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

G-SchNet



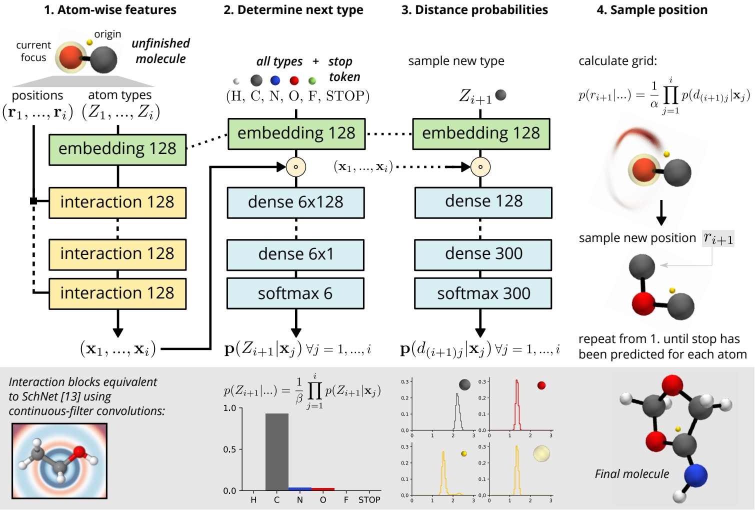
N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

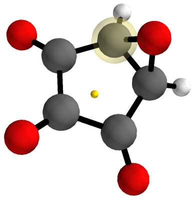
G-SchNet



N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

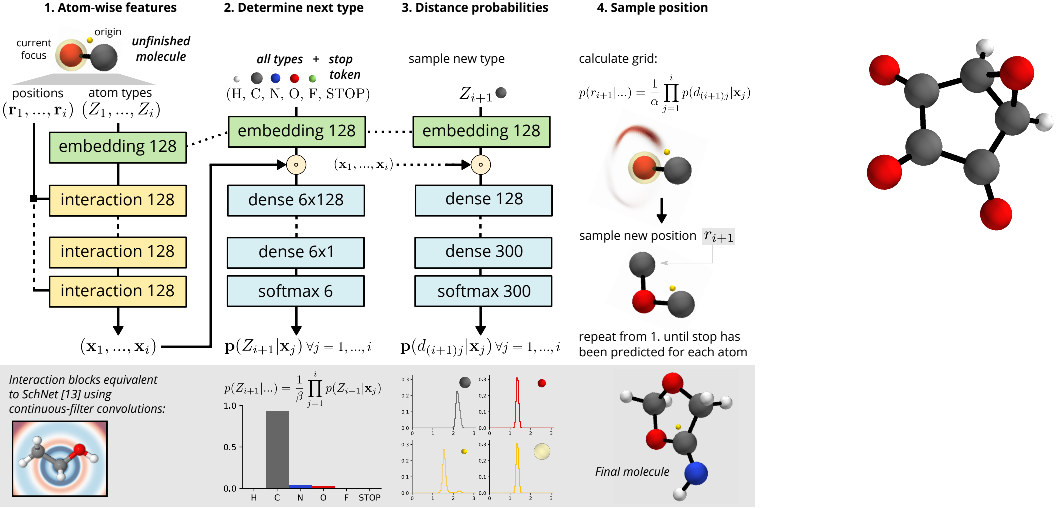
G-SchNet





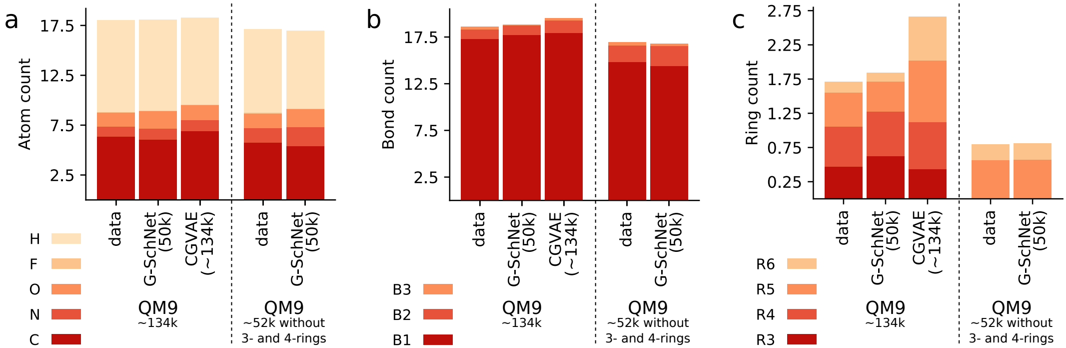
N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

G-SchNet



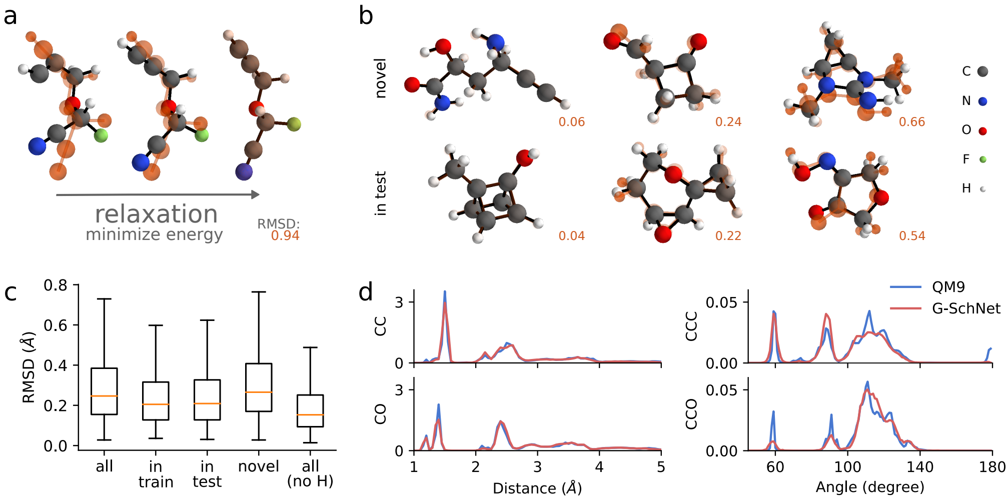
N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

Matching the distribution of QM9



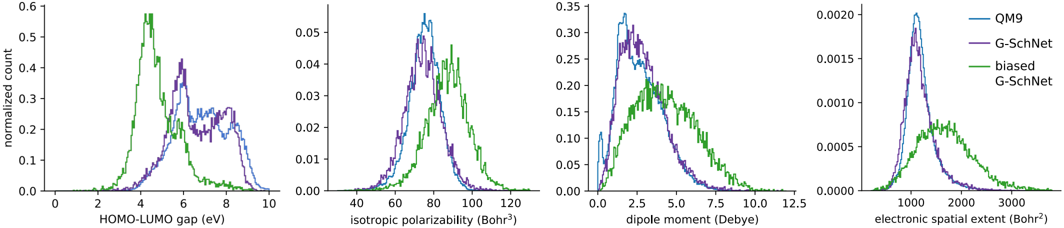
N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

Generation of equilibrium geometries



N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

Targeted generation of molecules

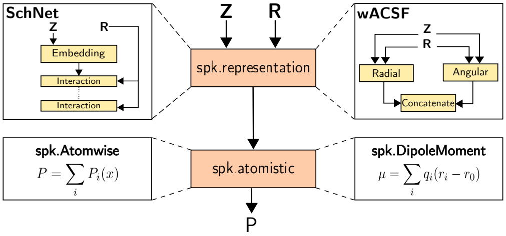


Generated molecules available: `http://www.quantum-machine.org/datasets/`

N. W. A. Gebauer, M. Gastegger, K. T. Schütt. arXiv:1906.00957, NeurIPS 2019 (accepted).

SchNetPack: A Deep Learning Toolbox For Atomistic Systems

```python
import schnetpack as spk
import schnetpack.atomistic as atm
import schnetpack.representation as rep
import torch
from torch.optim import Adam
import torch.nn.functional as F
from schnetpack.datasets import *

# load qm9 dataset and download if necessary
data = QM9("qm9/", properties=[QM9.U0])

# split in train and val
train, val, test = data.create_splits(10000,
                                      1000)
loader = spk.AtomsLoader(train,
                         batch_size=100,
                         num_workers=4)
val_loader = spk.AtomsLoader(val)

# create model
reps = rep.SchNet()
output = atm.Atomwise()
model = atm.AtomisticModel(reps, output)

# create trainer
opt = Adam(model.parameters(), lr=1e-4)
loss = lambda b,p: F.mse_loss(p["y"],b[QM9.U0])
trainer = spk.Trainer("output/", model, loss,
                      opt, loader, val_loader)

# start training
trainer.train(torch.device("cpu"))
```



SchNetPack

www.quantum-machine.org/schnetpack

K.T. Schütt et al (2018). J. Chem. Theory Comput. 10.1021/acs.jctc.8b00908

Thank you!

Collaborators:
N.W.A. Gebauer, M. Gastegger, H.E. Sauceda, P.-J. Kindermans, P. Kessel, K.A. Nicoli, S. Chmiela, R. Maurer, A. Tkatchenko, K.-R. Müller

Berliner Zentrum für MASCHINELLES LERNEN

BERLIN BIG DATA CENTER