EquiformerV2: Improved Equivariant Transformer for Scaling to Higher-Degree Representations

Yi-Lun Liao[1], Brandon Wood[2], Abhishek Das[2]*, Tess Smidt[1]*
[1]Massachusetts Institute of Technology [2]FAIR, Meta AI *Equal contribution

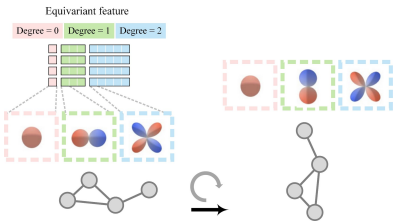
github.com/atomicarchitects/equiformer_v2

Motivation and Contribution

(1) Equivariant networks such as Equiformer have shown the importance of incorporating 3D-related inductive biases in learning representations of 3D atomistic systems.
(2) Tensor products, the core operations in equivariant networks, have high complexity and become the bottleneck in scaling degrees, L_{max}, of equivariant representations.
(3) eSCN reduces SO(3) convolutions built from tensor products to SO(2) linear operations, enabling higher L_{max}.
(4) We revisit the design of equivariant Transformers and propose EquiformerV2, the improved equivariant Transformer for scaling to higher L_{max}.

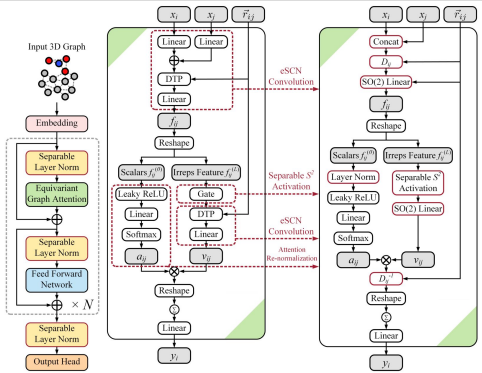
Proposed Method

Equivariant features (e.g., Euclidean vectors) consist of vectors of different degrees L, with maximum degree L_{max}, and those of different L rotate at different speeds when input graphs are rotated.



We start with Equiformer and have the following modifications:
(1) We use eSCN convolutions, which reduce tensor products to SO(2) linear operations, to efficiently incorporate vectors of higher degrees and scale to higher L_{max}.
(2) To better leverage the power of higher degees, we propose three architectural improvements – attention re-normalization, separable S^2 activation and separable layer normalization.

Overall Architecture of EquiformerV2

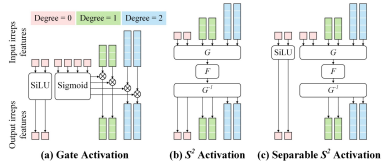


Attention Re-Normalization

$$z_{ij} = w_a^T \text{LeakyReLU}(\text{LN}(f_{ij}^{(0)})) \text{ and } a_{ij} = \text{softmax}_j(z_{ij}) = \frac{exp(z_{ij})}{\sum_{k \in \mathbb{N}(i)} exp(z_{ik})}$$

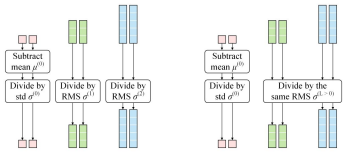
We apply an additional layer normalization (LN) before the non-linear functions when calculating attention weights.

Separable S^2 Activation



(a) Gate Activation (b) S^2 Activation (c) Separable S^2 Activation

Separable Layer Normalization



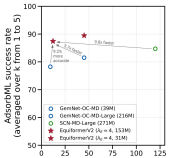
(a) Equivariant Layer Normalization (b) Separable Layer Normalization

Result

OC20

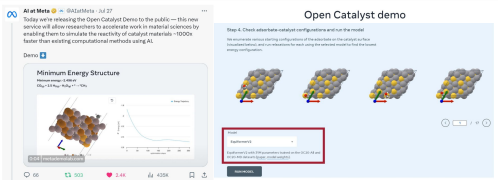
(1) The proposed architectural improvements are necessary to better leverage the power of higher degrees.

Index	Attention re-normalization	Activation	Normalization	Epochs	forces	energy
1	✗	Gate	LN	12	21.85	286
2	✓	Gate	LN	12	21.86	279
3	✓	S^2	LN	12	didn't converge	
4	✓	Sep. S^2	LN	12	20.77	285
5	✓	Sep. S^2	SLN	12	20.46	285
6	✓	Sep. S^2	LN	20	20.02	276
7	✓	Sep. S^2	SLN	20	19.72	278
8	eSCN baseline			12	21.3	294



(2) EquiformerV2 achieves the state-of-the-art results on OC20, OC22 and AdsorbML and is used in Open Catalyst Demo.

Training set	Model	Number of parameters ↓	Training time (GPU-days) ↓	Throughput Samples / GPU sec. ↑	S2EF validation Energy MAE (meV) ↓	S2EF validation Force MAE (meV/Å) ↓	S2EF test Energy MAE (meV) ↓	S2EF test Force MAE (meV/Å) ↓	IS2RS test AFbT (%) ↑	IS2RS test ADwT (%) ↑	IS2RE test Energy MAE (meV) ↓
OC20 S2EF-All+MD	GemNet-OC-L-E (Gasteiger et al., 2022)	56M	640	3.5	239	22.1	230	21.0	-	-	-
	GemNet-OC-L-F (Gasteiger et al., 2022)	216M	765	3.2	252	20.0	241	19.0	40.6	60.4	-
	GemNet-OC-L-F+E (Gasteiger et al., 2022)	-	-	-	-	-	-	-	-	-	348
	SCN L=6 K=16 (4 $\times_{SO2}$ 2 heads) (Zitnick et al., 2022)	168M	414	-	-	-	228	17.8	43.3	64.9	328
	SCN L=8 K=20 (Zitnick et al., 2022)	271M	1280	-	-	-	237	17.2	43.6	67.5	321
	eSCN L=6 K=20 (Passaro & Zitnick, 2023)	200M	568	2.9	243	17.1	228	15.6	50.3	66.7	323
	EquiformerV2 (L_{max} = 4, 31M)	31M	705	7.1	232	16.3	228	15.5	47.6	68.3	315
	EquiformerV2 (L_{max} = 2, 153M)	153M	1368	1.8	**230**	**14.6**	227	**13.8**	**55.4**	**69.8**	**311**
	EquiformerV2 (L_{max} = 4, 153M)	153M	1571	1.8	**227**	15.0	**219**	14.2	54.4	69.4	**309**



QM9

(1) Unlike larger datasets like OC20 S2EF, the performance gain of using higher degrees on smaller datasets is smaller.
(2) When trained with Noisy Nodes, the performance gain becomes larger, and EquiformerV2 can match the performance of some models pre-trained on 30× larger datasets.

Model	Task Units	α a_0^3	$\Delta\varepsilon$ meV	ε_{HOMO} meV	ε_{LUMO} meV	μ D	C_v cal/mol K	G meV	H meV	R^2 a_0^2	U meV	U_0 meV	ZPVE meV
Equiformer (Liao & Smidt, 2023)		.046	30	15	14	.011	.023	7.63	6.63	.251	6.74	6.59	1.26
Equiformer (Liao & Smidt, 2023) + NN†		.040	26.4	13.7	13.0	.011	**.020**	5.49	4.61	.235	4.81	4.61	1.18
GNS-TAT + NN (Zaidi et al., 2023)‡		.047	25.7	17.3	17.1	.021	.022	7.41	6.42	0.65	6.39	6.39	1.080
GNS-TAT + NN + pretraining (Zaidi et al., 2023)‡		.040	**22.0**	14.9	14.7	.016	**.020**	6.90	5.79	0.44	5.76	5.76	**1.018**
EquiformerV2		.050	29	14	13	.010	.023	7.57	6.22	.186	6.49	6.17	1.47
EquiformerV2 + NN		**.039**	24.2	**12.2**	**11.4**	**.009**	**.020**	**5.34**	**4.24**	**.182**	**4.28**	**4.34**	1.21

Table 12: Mean absolute error results on QM9 test set when trained with Noisy Nodes (Godwin et al., 2022). † denotes that the results are produced by this work. ‡ denotes using different data partitions. "NN" denotes Noisy Nodes, and "pretraining" denotes pretraining on PCQM4Mv2 dataset (Nakata & Shimazaki, 2017). The performance gain from Equiformer to EquiformerV2 becomes larger when trained with Noisy Nodes.