Supplementary materials: Crystals with Transformers on Graphs, for predictions of crystal material properties

1 Architecture of CrysToGraph

Figures in this section are same as Fugure 4 in the main text. Limited by space in the main text, we keep the figures in the main text compact and leave the large version here in Figure S1 - S2.

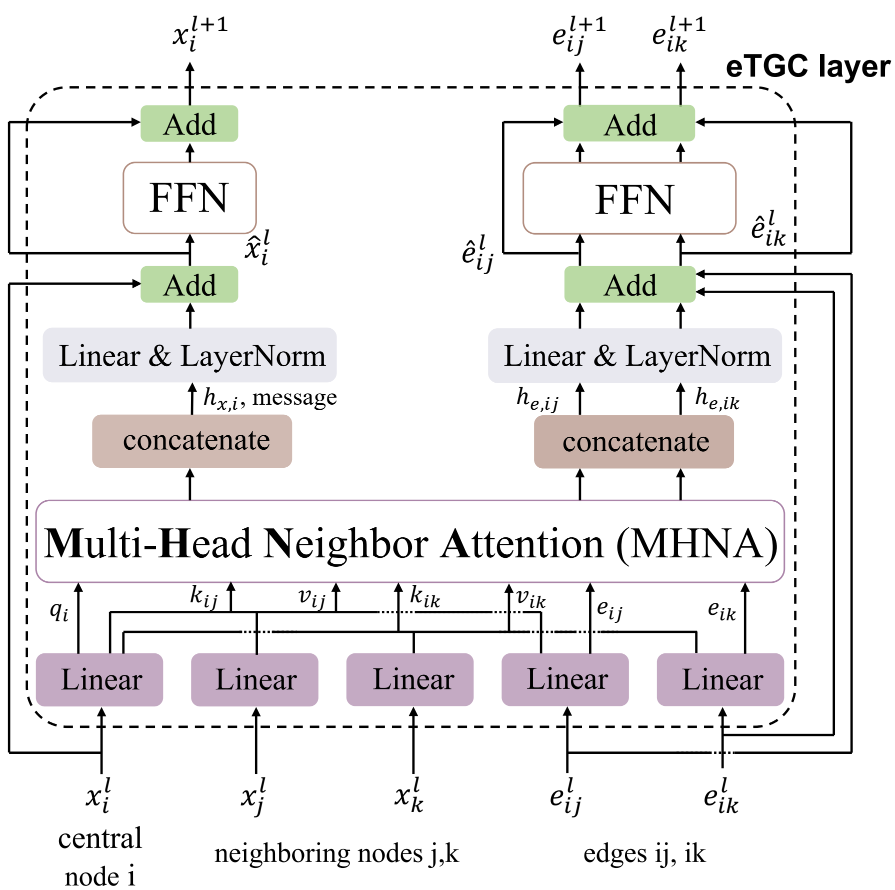
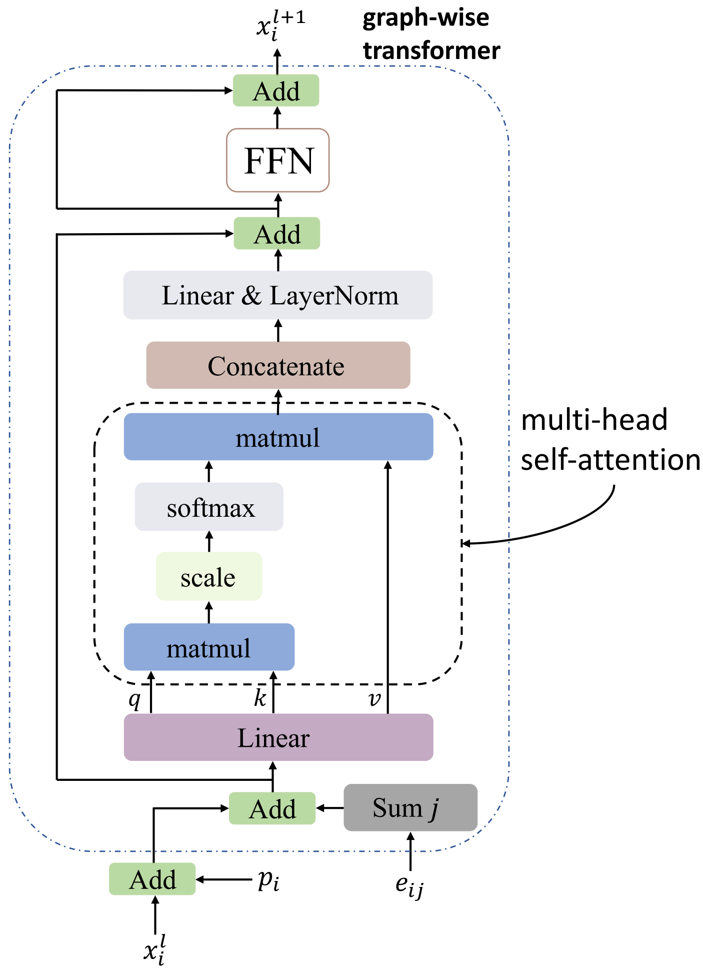
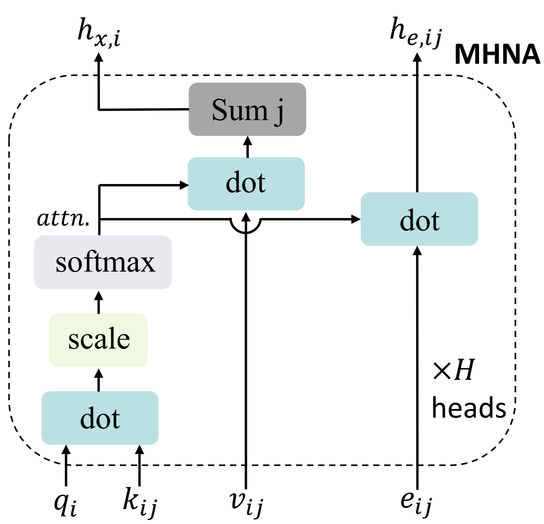


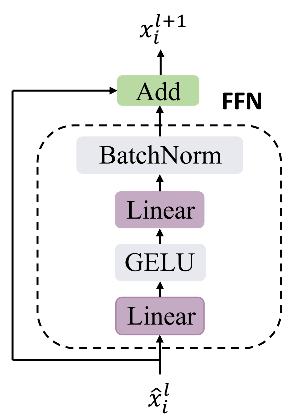
Figure S1: Structure of an eTGC layer.



(a) GwT



(b) MHNA in eTGC



(c) FFN

Figure S2: Structure of (a) a GwT layer, (b) multi-head neighbor attention in eTGC layers, (c) FFN in eTGC layers and GwT layers.

2 Masked Atom Pretraining

In the pretraining phase for atom representations, we introduced a masked atom prediction task. In this task, a specified percentage of atoms within each graph are masked. Specifically, 15% of the atoms in each graph are subjected to masking operations. Among these, 80% are substituted with a designated mask token, while 10% are replaced with randomly selected tokens, and the remaining 10% are left unchanged. In instances where the number of nodes in a graph is insufficient to maintain the masking rate below 15%, the crystal structure is expanded in all three dimensions. With graphs constructed as aforementioned, we trained a CGCNN model on these constructed graphs to predict the types of masked atoms. The loss curve is in Figure S3.

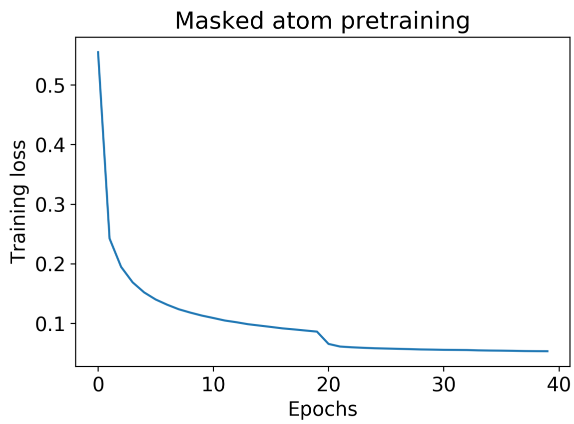


Figure S3: The loss curve in atom masked pretraining. The learning rate decrease by 10 at epoch 20.

Following the pretraining of atom embeddings, we concatenated the machine-learnt embeddings with the manually curated CGCNN atom embeddings.

3 Layers of FFNN

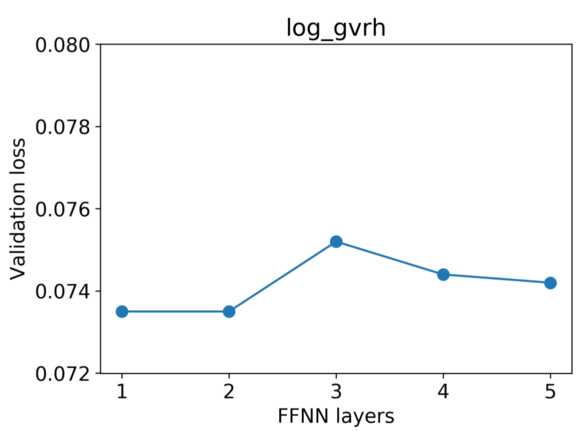


Figure S4: Various depth of FFNN, trained on `log_gvrh` dataset.

We can see in Figure S4, deeper FFNN impedes the overall performance. The ideal depth of FFNN is 1 or 2 layer.

4 Learning Rate

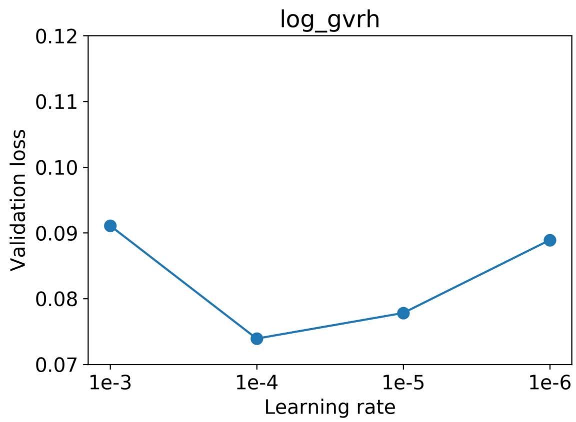


Figure S5: Same model trained in various learning rate.

Shown in Figure S5, the optimal learning rate is 1e-4, which is generalized to other experiments in this work.

5 Weight Decay

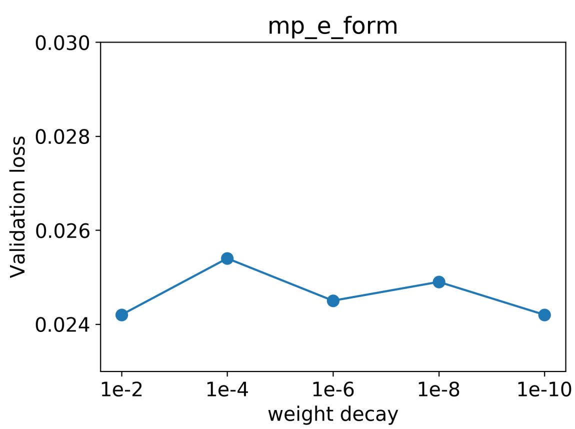


Figure S6: Models trained with various weigh decay penalty.

The weight decay can be regarded as a derivative of L2 regularization. Our model is excessive in parameter, however, the performance does not change much when the weight decay penalty varies, as shown in Figure S6.