Heterogeneous Graph Transformer for learning Compound-Ortholog links from the Kyoto Encyclopedia of Genes and Genomes

Nima Azbijari

Constructing KGs from KEGG

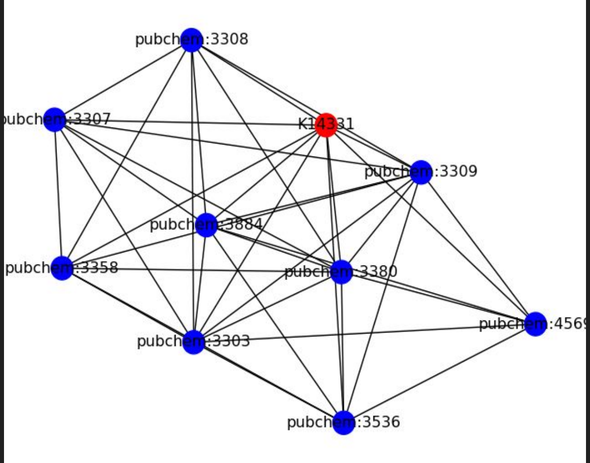
- KEGG has a lot of relation information between types
  - We can build graphs from this
- Graph built contains compounds (MACCS features from PubChem) and KEGG Orthologs
  - KEGG Orthologs (KOs) can be represented as a family of protein sequences related by function. We need a way to represent this as a vector.

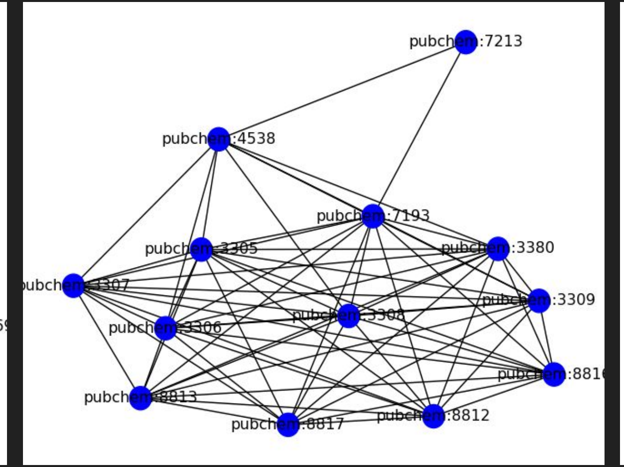
ESM-2 to embed protein sequences

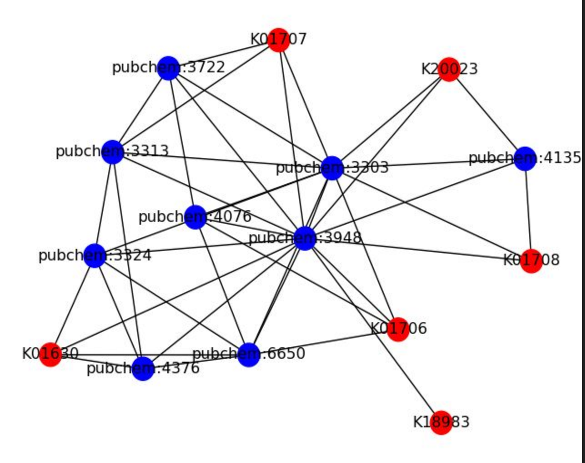
- Each KO has *N* protein sequences
  - We feed each sequence through ESM-2, a PLM, to get a protein embedding
  - Each KO is represented as and average vector over all proteins in that KO
- Now, we have features for compounds and features for KOs

Constructing the graph

- We connect compounds to KOs if the compound and KO share an enzyme (EC number in KEGG)
- By doing this we now have compound-compound edges and compound-KO edges
- cpd={ x=[4550, 167] }
- ko={ x=[2770, 1280] }
- (cpd, reacts, cpd)=109258
- (cpd, interacts, ko)=16629]



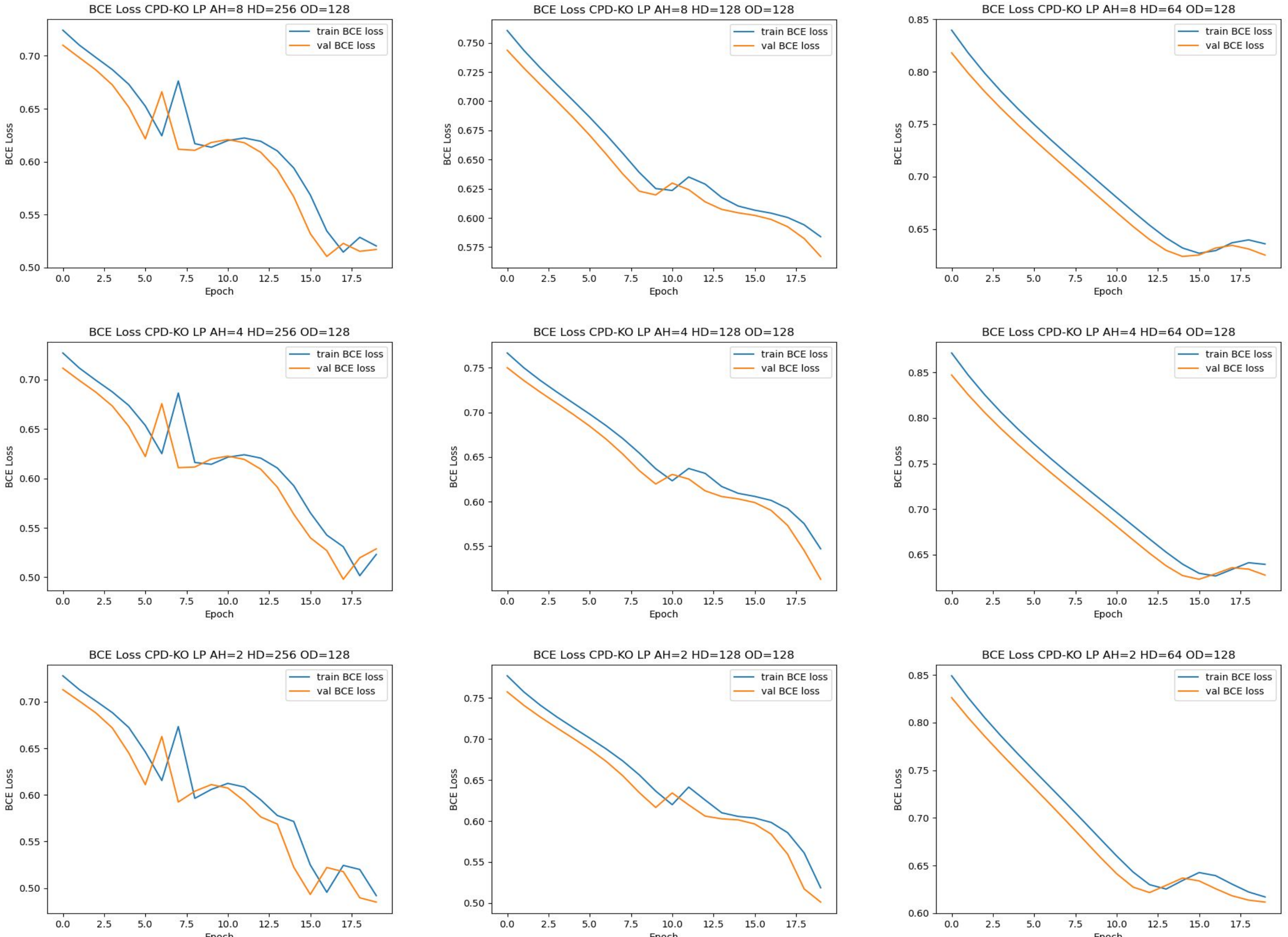




What are we asking?

- Can a GNN accurately capture relationships between KOs and compounds if we train it on a subset of compound-compound and compound-KO links?
- Our GNN is a Heterogeneous Graph Transformer with a link prediction head
  - Basically, get embeddings for each nodes and then do link prediction on these embeddings

Results


BCE Loss CPD-KO LP AH=8 HD=256 OD=128
BCE Loss CPD-KO LP AH=8 HD=128 OD=128
BCE Loss CPD-KO LP AH=8 HD=64 OD=128
BCE Loss CPD-KO LP AH=4 HD=256 OD=128
BCE Loss CPD-KO LP AH=4 HD=128 OD=128
BCE Loss CPD-KO LP AH=4 HD=64 OD=128
BCE Loss CPD-KO LP AH=2 HD=256 OD=128
BCE Loss CPD-KO LP AH=2 HD=128 OD=128
BCE Loss CPD-KO LP AH=2 HD=64 OD=128
train BCE loss
val BCE loss
BCE Loss
Epoch

Future Work

- Experiment with layers
- Add in metrics (F1, accuracy, AUC)
- Run on test set
- Add in unseen compounds