kaggle

Kaggle Winner Presentation (6th)

Google - Fast or Slow? Predict AI Model Runtime

https://www.kaggle.com/competitions/predict-ai-model-runtime

HENG CHER KENG
https://www.kaggle.com/hengck23

Agenda

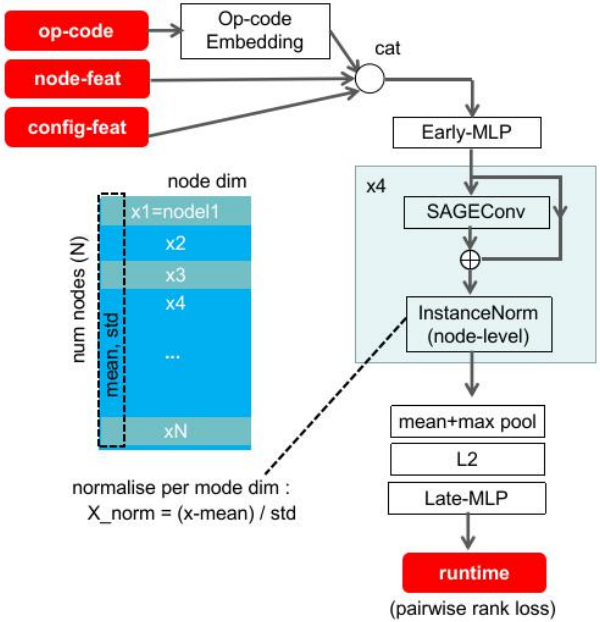
Agenda

1. Background
2. Summary
3. Feature selection & engineering
4. Training methods
5. Important findings
6. Simple model

Background

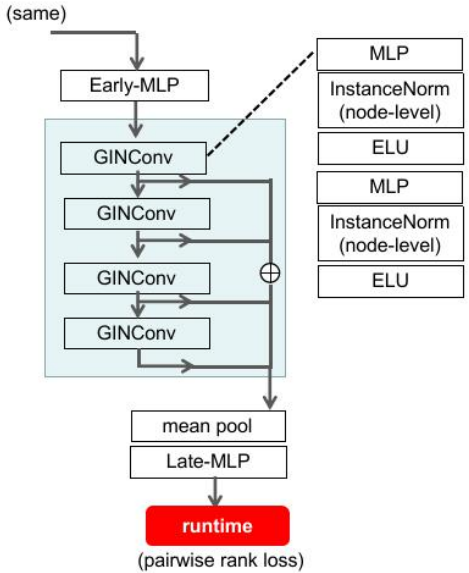
- Contract computer vision and deep learning algorithm engineer.
  - find fracture in x-ray images
  - implement visual slam for robotic navigation.

- Familiar with deep learning and build deep models in my work.
- Experiences in graph neural net GNN from previous Kaggle competitions.



Layout runtime prediction

Residual SAGEConv Net



Graph Isomorphism Net (GIN)

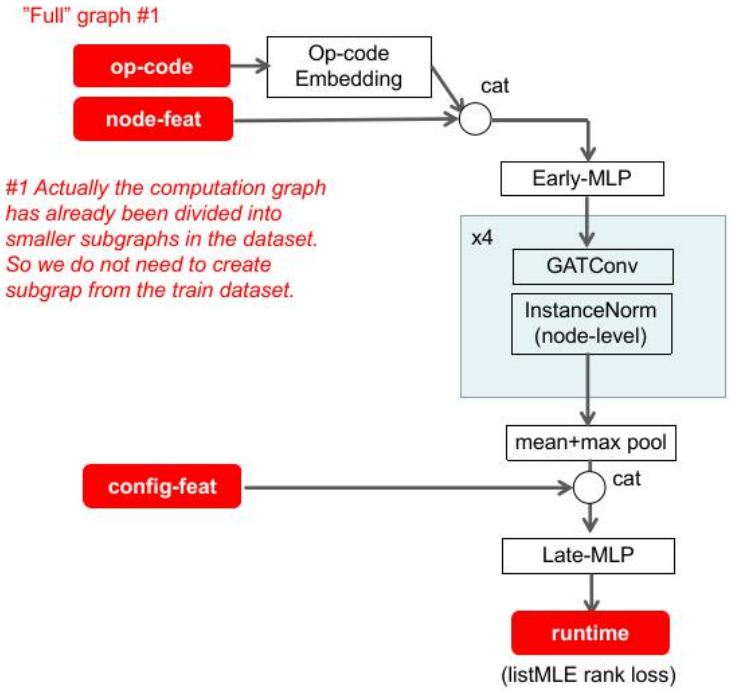
For layout runtime prediction:

- Reduce input graph to subgraph by including only the 5-hop neighbours from the configure node

- GNN model with 4-layer SAGE-conv[2] with residual shortcut

- GNN model with 4-layer GIN-conv[3]

- Graph instance normalisation[1] over node



Tile runtime prediction

GATConv Net

For tileruntime prediction:

- GNN model with 4-layer GAT-conv[4]

- Pytorch geometric used for building GNN.
- Training takes abopout 4 to 6 hours for one model in each collection, using Nvidia Quadro RTX 8000 GPU with memory 48 GB.

[1] "Learning Graph Normalization for Graph Neural Networks" - Yihao Chen
https://arxiv.org/abs/2009.11746

[2] "Inductive Representation Learning on Large Graphs" - William L. Hamilton
https://arxiv.org/abs/1706.02216

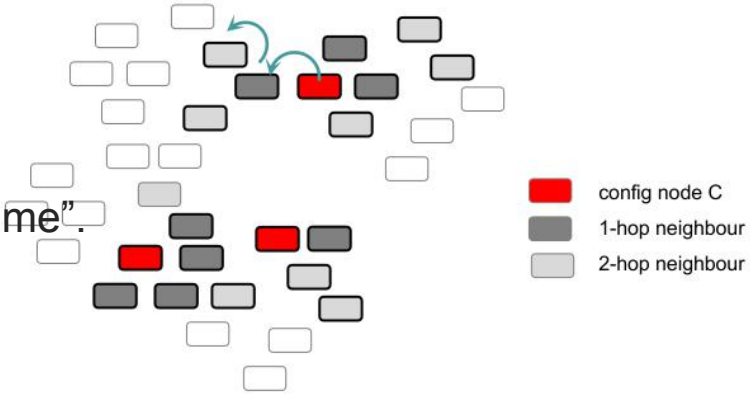
[3] "How Powerful are Graph Neural Networks?" - Keyulu Xu
https://arxiv.org/abs/1810.00826

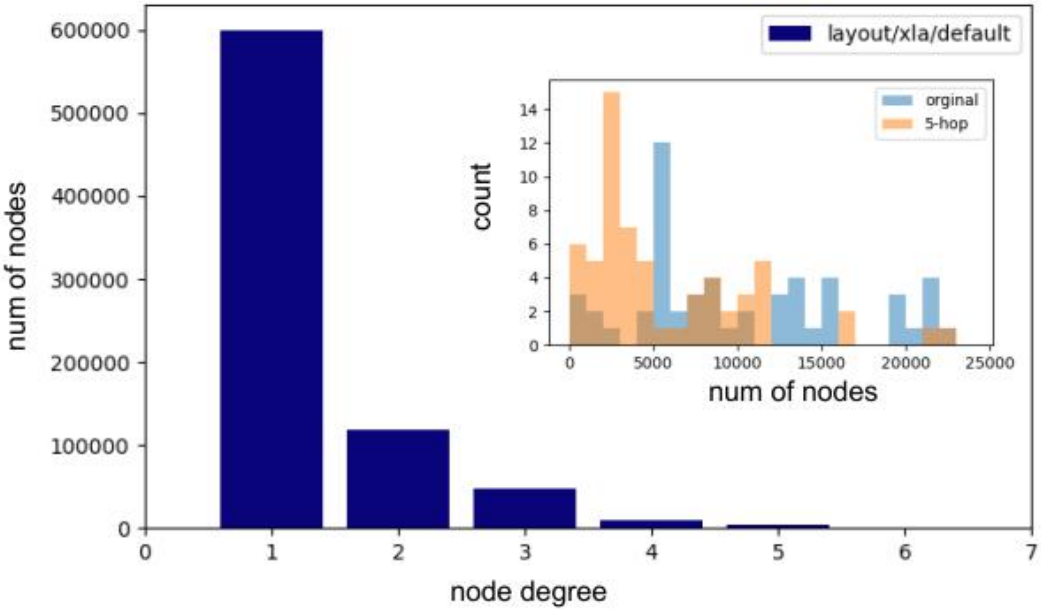
[4] "Graph Attention Networks" - Petar Veličković
https://arxiv.org/abs/1710.10903

Features Selection/ Engineering

5-hop neighbourhood subgraph as input:
- config node + their neighbours

- relative runtime ranking, just need to consider the "difference of two graph" to predict the "difference in runtime".





		XLA/Layout Default
graphsage GNN		kendall tau
	3-hop	0.41260
	4-hop	0.42503
	5-hop	0.45508
	6-hop	0.43508

- Ensemble should imporve results

		collection				
		NLP/Layout		XLA/Layout		XLA/Tile
		Default	Random	Default	Random	
[a]	4x-gatconv-listmle					0.97462
[b]	4x-graphsage-pair2	0.53938	0.92654	0.45508	0.67128	
[c]	4x-gin-pair2			0.45958		
[b]+[c]	ensemble			0.46952		
submission		0.53938	0.92654	0.46952	0.67128	0.97462
					avg	0.71627
					public lb	0.69424
					private lb	0.70549

For GIN Net, we don't have time to train for all layout prediction before the competition ends.

Training Methods

- For layout target kendall tau, use pairwise rank loss.
- Tile top5 slowndown target use listMLE

- ADAM optimizer with fixed lr 0.0005
- Use stochastic weight averaging SWA.

  final weights = average last 10 trained weights

  +0.01 improvement over best model

- batch size = 32, sample 80 to 100 configurations per subgraph

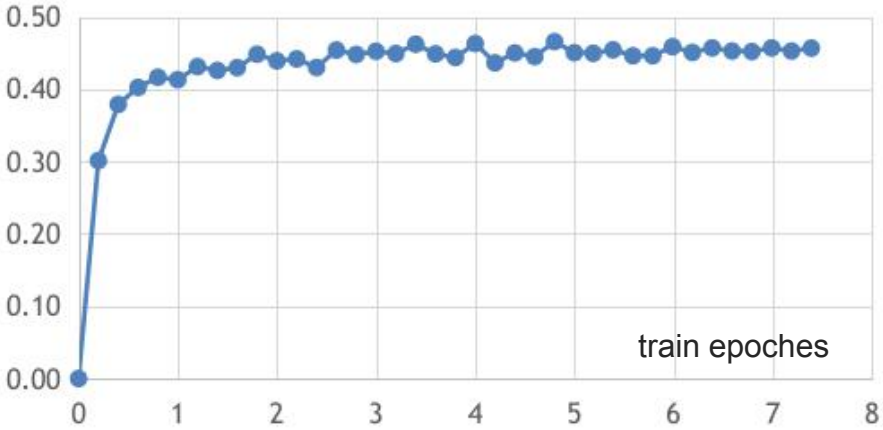
- gradient accumation

```
optimizer.zero_grad()

for b in range(batch_size):
        r = batch[r]
        loss = net(r)  # forward one subgraph
        scaler.scale(loss).backward() #backward accumuate gradient

scaler.step(optimizer) #update net parameters
scaler.update()
```

validation kendall tau (layout/xla/default)

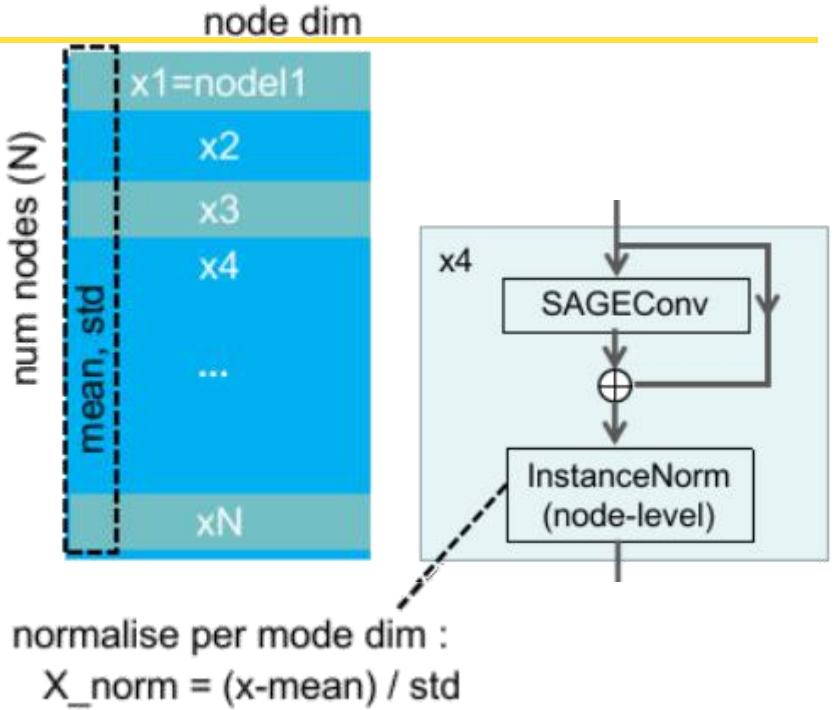


Important and Interesting Findings

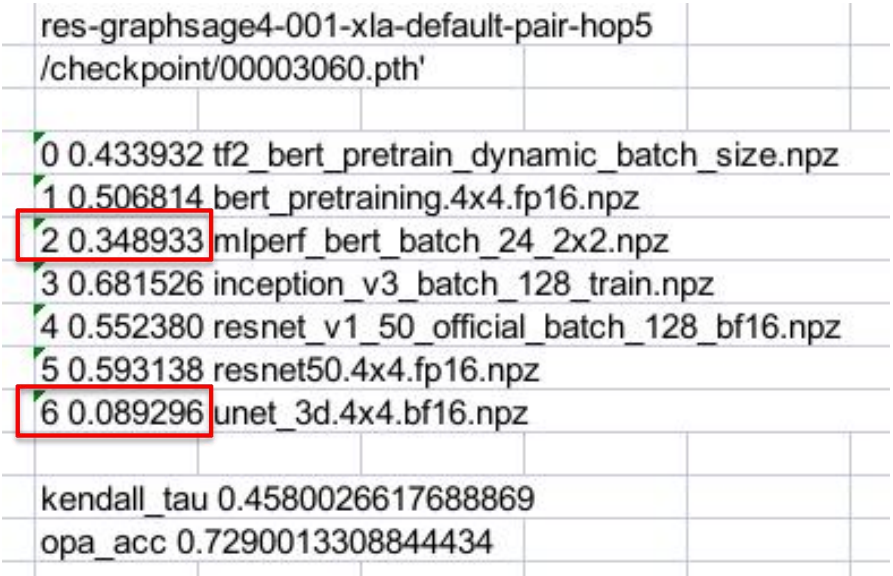
- Graph instance normalisation gives faster convergence and improve validation kendall tau by 0.02

- Attention pooling at graph readout for layout runtime prediction improves validation 0.01 for some cases

this solution is not selected becuase it did not improve public LB. (But private LB is better)



- validation kendall tau has large variance. Careful to select final submission to avoid “shakeup” for private LB.



Simple Model

- Possible to use even smaller 4-hop neighbourhood subgraph for better speed at the expense of 5 to 10% loss in accuracy

- Number of graph conv layers can also be reduced. We find num of layer = num of hop + 1

		XLA/Layout
		Default
graphsage GNN		kendall tau
	3-hop	0.41260
	4-hop	0.42503
	5-hop	0.45508
	6-hop	0.43508

Question and Answer


kaggle