Step1: ReadMe Page Selection

README path: dgl/examples/pytorch/bgrl/README.md

This DGL example implements the GNN experiment proposed in the paper Large-Scale Representation Learning on Graphs via Bootstrapping. For the original implementation, see here(URL).

Requirements

The codebase is implemented in Python 3.8. For version requirement of packages, see below.

- dgl 0.8.3
- numpy 1.21.2 ...

Step2: Task Mining

Inductive Task

```
python main.py \
    --dataset coauthor_cs  \
    --graph_encoder_layer 1024 \
    --drop_edge_p 0.3 \
    --feat_mask_p 0.25 \
    --lr 5e-2
```

Dataset

Dataset summary:

Dataset	Task	Nodes	Edges	Features
WikiCS	Transductive	11,701	216,123	300
Amazon Computers	Transductive	13,752	245,861	767
Amazon Photos	Transductive	7,650	119,081	745
Coauthor CS	Transductive	18,333	81,894	6,805

Step3: Argument Extraction

Dataset: wiki_cs, amazon_photos, amazon_computers, coauthor_cs ...
Graph Encoder Layer: 64, 128, 256, 512, 1024 ...
Learning Rate: 1e-3, 5e-3, 1e-2, 5e-2, 1e-1, 5e-1 ...

Step4: Multi-Argument Combination

Argument combination 1
wiki_cs, 128, , 5e-3
.....
Argument combination N
amazon_photos, 256, 5e-1

→ Selected Argument :
wiki_cs,
128
5e-3

Step5: Diverse Instruction Construction

In Selected Argument:

Human Instruction: I have a dataset entitled wiki_cs that I'm eager to utilize for training purposes. Specifically, I'm aiming to employ this dataset to train a cutting-edge **BGRL** Model. To achieve optimal performance, I intend to utilize graph encoder layer 128, alongside a learning rate (lr) of 0.005. Could you kindly assist me in crafting the necessary code to accomplish this task?"

Step6: Ground Truth Code Construction

```
python main.py \
    --dataset wiki_cs  \
    --graph_encoder_layer 128 \
    --drop_edge_p 0.3 \
    --feat_mask_p 0.25 \
    --lr 5e-3
```