Summary for Sales Team

James Xu

11th December 2022

Overview

In this task we examined briefly three models for a binary classification task taken from Therapeutics Data Commons and benchmarked their performances.

Choice of Dataset

We chose the dataset **CYP P450 2C19 Inhibition, Veith et al.** This dataset was to do with the inhibition of CYP2C19 gene, but this had no reason to why it was chosen. The following reasons made this dataset desirable:

1. The dataset is relatively large, with data on $12,665$ drugs. This gives our proof of concept more reliability in its ability to deal with diverse amounts of data.
2. The labels given are those that of binary classification. For sake of simplicity, these tasks have more intuitive 'human' metrics for performance than regression tasks.
3. The dataset has a balanced label distribution, with roughly the same amount of positive and negatives (1's and 0's).

The sizes of the training, validation and test sets were 8866, 1266 and 2533 respectively.

Choice of models

When it comes to choosing models for benchmarking, it is tempting to search for established breakthrough models in that those of existing literature. However for purposes of proof-of-concept, simple small models with similar number of adjustment parameters offer a better standard for comparisons and control.

Our GNN structure (see code) consisted of 3 graph network layers with top k pooling followed by 3 linear layers. This offered satisfactory performance on the dataset, thus was the structure stuck with for the benchmarking.

When it came to the classical tree model, a decision was made to cut connectivity data and aggregate across node data in an ignorant way in order for decision tree algorithm to be applied. The nature of inputs of graphical data into decision trees has been discussed in the notebook, and since this method was to be **classical**, no other clever implementation was made.

Training Parameters

Not much experimentation was made in fine tuning training parameters such as learning rate or optimizer. It was thought that all runs were to be rough and for proof of concept.

The number of epochs trained in the end was decided under the criteria of not over-fitting to the training data as well as spacial and time consumption purposes.

Chosen Metric

For human intuition and explainability reasons, *Average Precision* or *AUPRC* (Area Under Precision-Recall Curve) is used rather than the trained on binary cross-entropy loss. With a total of 5819 positive labels in the dataset, this gives a baseline AUPRC of $5819/12665 \approx 0.46$.

AUPRC is a number between 0 and 1 which measures the area under the precision-recall curve. This curve is found by obtaining the precision (identification of false negatives) and recall (identification of true positives) scores of the model given varying thresholds for classification. The closer this value is to 1, the better the performance. In biomedical literature often *AUROC* (Area Under the Receiver Operating Statistic) is reported instead as it has in general a higher value (due to a higher baseline of 0.5).

Results

The following results were obtained from training.

Benchmarked Results on Test Set			
Loss Metric	Graph Neural Net	Decision Tree	Graph Transformer
(Mean) BCE	30.148	33.162	15.269
AUPRC	0.8258	0.5805	0.8367

Summary

It is not surprising that such a classical approach to the dataset (along with the mistreatment of graphical data) performed so poorly. The main interest comes from the performances of the GNN and the Graph transformer which had similar results. The graph transformer did in the end slightly, but this may be due to the fact that the graph transformer had a higher number of trainable parameters (54081 vs 21441 in my implementations).