Random Forest GNN in Chemical Domain

Sofia Blyufer 321128019

May 8, 2025

Abstract

In this report, we investigate the application of Random Forest Graph Neural Networks (RF-GNN) to chemical data, focusing on the Directed Message Passing Neural Network (DMPNN) and Graph Attention Network (GAT) models. The motivation for this study comes from the need to improve molecular property predictions, which are crucial for drug discovery, toxicity prediction, and material science. Using three widely recognized chemical datasets—Tox21, BACE, and ClinTox—we compare the performance of these GNNs in predicting molecular properties. Our results demonstrate that DMPNN gains higher roc-auc scores than GAT in all of our benchmark tasks. This is likely due to its better handling of bond information and 3D molecular configurations. Additionally, we find that ensemble methods, particularly averaging predictions across multiple subsets, enhance model performance across all datasets. These findings underscore the potential of RF-GNN and suggest further exploration with more sophisticated models and larger datasets.

Introduction

Graph Neural Networks (GNNs) are becoming very popular in chemistry and materials science because they can effectively model and predict properties of complex molecules. GNNs use the natural graph-like structure of chemical compounds, where atoms are represented as nodes and bonds as edges, to understand the relationships and interactions within molecules. This ability has led to significant improvements in predicting molecular properties and understanding chemical processes[1].
In this study, we work with three wildly used chemical datasets:

- Tox21: contains information on the toxicity of various compounds[2]
- BACE: includes inhibitors of the human β-secretase 1 (BACE-1), which is relevant for Alzheimer's disease research[3]

- ClinTox: provides toxicity measurements for drugs that have been approved by the FDA and for those that failed clinical trials due to toxicity[4]

All these datasets contain SMILES representation[5] of the relevant molecules and labels (some of the above are multi-label). However, using the rdkit[6] python library, more features can be added depending on the atoms and known chemical data (such as molecular mass).
This work was inspired by the Random Forest Boosted Graph Neural Network (RF-GNN) model, which has shown promising results in detecting social bots[7]. RF-GNN is a combination of a random forest algorithm with GNNs instead of decision trees as base classifiers.
RF, i.e. random forest, is a classic machine learning algorithm that improves decision trees by using an ensemble approach. The general algorithm is[8]:

Algorithm 1 Random Forest(S,k,d'):

---

1. Sub-sample k sample sets $S_1, ..., S_k$ of size $m =\mid S \mid$ with replacement from S

2. For each S_i, learn an unbounded-depth decision tree h_i, with the following change:
   at each decision node, use only a randomly sampled subset of $d' < d$ features (without replacement)

3. Return $H(x) = \frac{1}{k} \sum_{i1}^{k} h_i(x)$

Our research question is:
can an ensemble approach with GNN base classifiers improve results of GNNs on molecular data?
In our experiments, we evaluates two specific types of GNNs: Directed Message Passing Neural Network (DMPNN)[9] and Graph Attention Network (GAT)[10]. The DMPNN focuses on directed edges to capture directional dependencies between atoms, which improves the model's ability to predict molecular properties (such as toxicity). The GAT model uses attention mechanisms to assign different importance to nodes and edges, improving the model's ability to learn more complicated relationships in the graph. GAT and MPNN architectures were covered in our course, therefore we would like to explain DMPNN further.

DMPNN[9]

Figure 1: Illustration of DMPNN Architecture

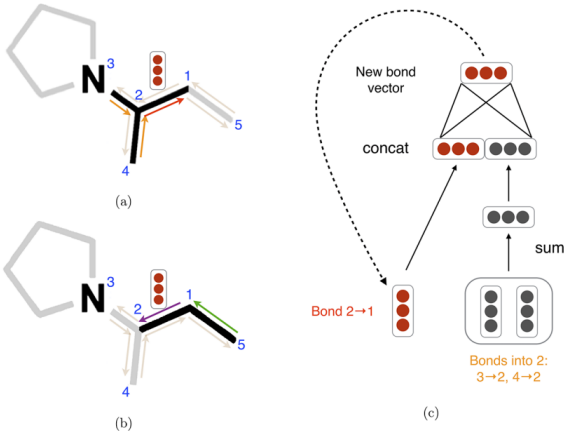


Figure 1. Illustration of bond-level message passing in our proposed D-MPNN. (a) Messages from the orange directed bonds are used to inform the update to the hidden state of the red directed bond. By contrast, in a traditional MPNN, messages are passed from atoms to atoms (for example, atoms 1, 3, and 4 to atom 2) rather than from bonds to bonds. (b) Similarly, a message from the green bond informs the update to the hidden state of the purple directed bond. (c) Illustration of the update function to the hidden representation of the red directed bond from diagram (a).

DMPNNs are an improvement of MPNNs (Message passing netural networks) that we have covered in class. This is a case of "Generalized Message Passing" from the lecture.
Generally, MPNNs operate on undirected graphs where nodes represent atoms and edges represent bonds. The network operates in two main phases:

- Message Passing Phase: During this phase, information is transmitted across the molecule graph through a series of steps. Each node (atom) updates its hidden state by aggregating messages from its neighboring nodes. The update for a node v at step $t+1$ t+1 is generally given by:

$$h_v^{t+1} = U\left(h_v^t, \sum_{w \in N(v)} M\left(h_v^t, h_w^t, e_{vw}\right)\right)$$

  where h_v^t is the hidden state of node v at step t, M is the message function, U is the update function, $N(v)$ is the set of neighbors of v , and e_{vw} is the feature of the edge between v and w.

- Readout Phase: After the message-passing steps T, the final hidden states of the nodes are combined to form a graph-level representation, which is then used for predicting molecular properties. This is achieved by a readout function R :

$$pred = \hat{y} = R\left(\left\{h_v^T \mid v \in G\right\}\right)$$

The DMPNN archticture modifies the above framework by focusing on directed edges (bonds) instead of nodes:

- Edge-Focused Messages: In DMPNN, messages are associated with directed edges rather than nodes. This design helps to prevent the problem of "totters," where messages might loop back and forth between nodes, creating noise in the representation. For instance, in DMPNN, a message from node 1 to node 2 will be propagated to nodes 3 and 4 in the next iteration, but not back to node 1, avoiding unnecessary loops.

- Directed Edge Hidden States: Instead of updating node-based hidden states, DMPNN updates hidden states associated with directed edges. The message passing and update functions are adapted accordingly:

$$h_{vw}^{t+1} = U\left(h_{vw}^{t}, \sum_{k \in N(v) \setminus \{w\}} M\left(h_{vk}^{t}, x_v, x_w\right)\right)$$

Method

The library "deepchem" [11] in python was used. This library has most of the popular datasets in chemistry and materials science, including Tox21, BACE and ClinTox, which were presented above. To fully understand the efficiency of the proposed approach, we have created baselines. The base classifiers we tested in the experiments were DMPNN and GAT. The experiments were conducted as follows:

1. Baseline GNN Models: Regular GNN models (DMPNN and GAT) were used on the Tox21, BACE, and ClinTox datasets to establish baseline performance metrics.

2. Training Multiple Models: For each dataset, N models were trained on different subsets of the data. N is a hyperparameter that was optimized during the experiments. This step helps differentiating between the effect of averaging and the effect of using subgraphs.

3. Similarliy to RFGNN[7], N models were trained on a subset of the data, while also dropping random parts of the molecules in each subset.

Step 3 follows the algorithm:

Algorithm 2 Modified RF-GNN for Molecular Data

For N in num of base classifiers:

1. Sample randomly fraction f of the trainset - $S' < S$
2. For each sample $s \in S'$: randomly drop a fraction of the vertices, and any dangling bonds (i.e. subgraph)
3. train GNN of new trainset

To predict return: $H(x) = \frac{1}{k} \sum_{i1}^{k} gnn_i(x)$

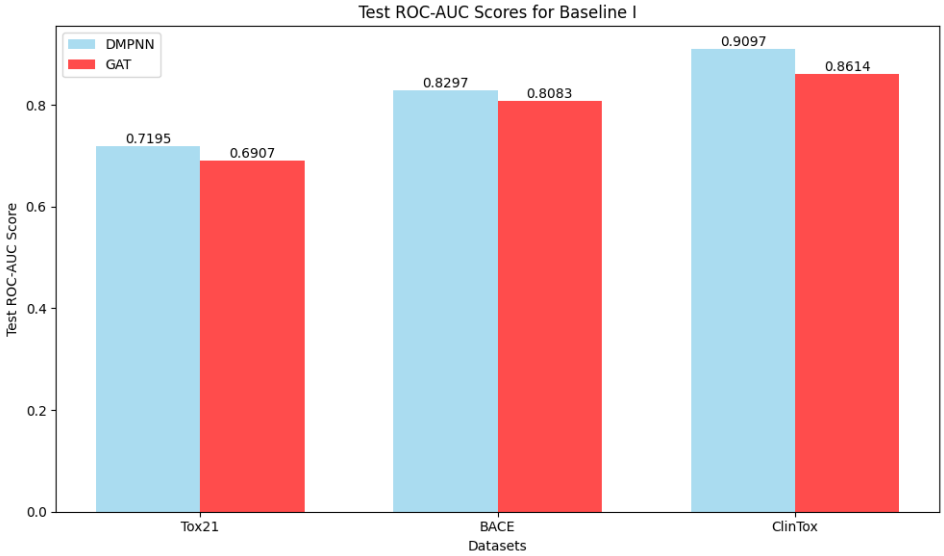
A few notes regarding the metrics calculation:

- Tox21 and ClinTox are multi-label. To adhere to the way deepchem calculates roc-auc score, we calulate the score for each label separately and then average on the number of labels
- To calculate the probabilities for each prediction when calculating a few models, the probabilities are averaged one all the models.
- This applies to the calculations for the ML random forest model as well.

Results

Baseline I - Regular Models:

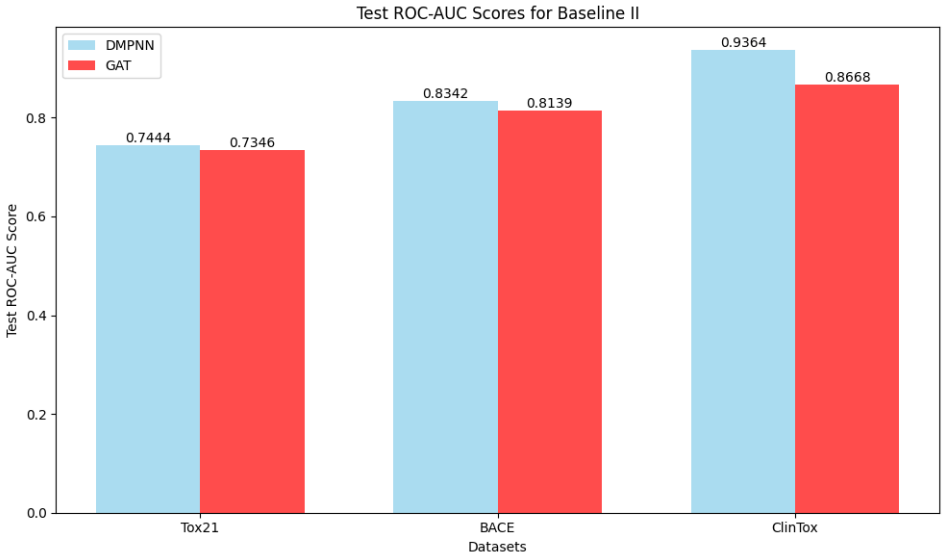
Figure 2: Roc-Auc scores for Tox21, BACE and ClinTox datasets for DMPNN and GAT models



In the above results we can see that in the current framework, DMPNN delivers higher roc-auc scores than GAT for all datasets.

Baseline II - Averaging Numerous GNN Base Classifiers:

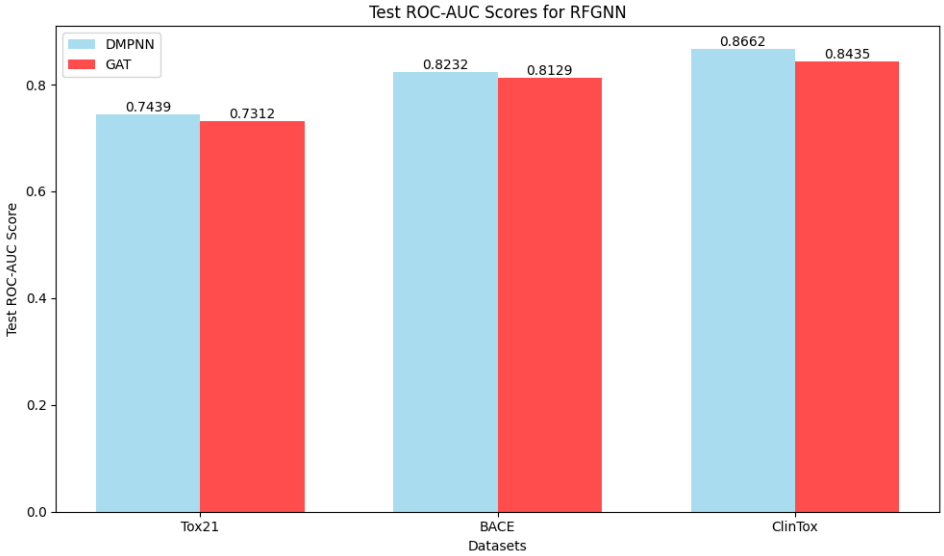
Figure 3: Roc-Auc scores for Tox21, BACE and ClinTox datasets for averaging base classifiers of DMPNN and GAT



Firstly, we can see that DMPNN still outperforms GAT in roc-auc score. Secondly, across all models and datasets, the results are higher than the previous experiment. In other words, an average of a few GNNs improved results for the current datasets.

RF-GNN Results:

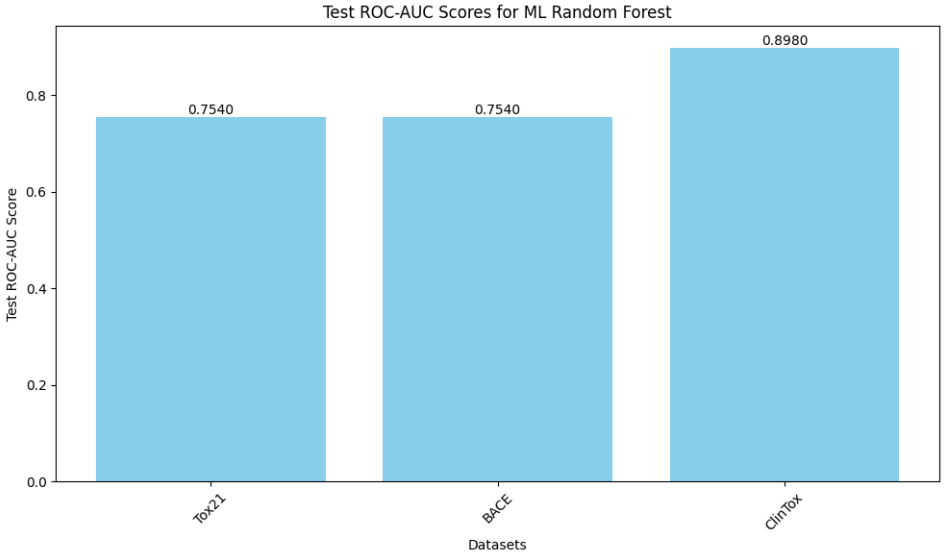
Figure 4: Roc-Auc scores for Tox21, BACE and ClinTox datasets for RF-GNN



The above results are for our proposed algorithm. We can see that DMPNN roc-auc score is still higher in all cases. Additionally, the results are higher than baseline 1. However, the results are lower than for the baseline 2 case, of a simple averaging.

For a complete comparison, we have also run a "regular" random forest classifier on the datasets:

Figure 5: Roc-Auc scores for Tox21, BACE and ClinTox datasets for RF



Interestingly, for Tox21, RF provides better results than the first baseline. For Clintox the result is better than GAT but not DMPNN. For Clintox and Tox21 the resuls are better than RF-GNN.

Conclusions and Future Directions

First, we observed that DMPNN performs better than GAT. This difference can be attributed to the nature of the datasets. Molecules in SMILES format are typically represented in 2D, but their toxicity and interactions with proteins depend heavily on their 3D structure. DMPNN, which incorporates bond information, appears to have better 3D awareness compared to GAT, which mainly focuses on the nodes. Additionally, DMPNN distinguishes between single and double bonds more effectively, which can explain its higher results.

Our results also show that averaging predictions across multiple subsets improves performance for all datasets and models. This demonstrates the strength of bagging algorithms. However, we believe our hyperparameter tuning could have been better, as RF-GNN did not perform as well as a simple average. For future work, we recommend running experiments with more estimators if more computational resources are available. We also suggest exploring other GNNs designed for specific tasks in chemistry, such as chirality GNNs. Although we tested three datasets of interest, there are many other datasets that should be tested with RF-GNN in the future, especially those too large for us to handle with our resources as students.

Refrences

1. Reiser, P., Neubert, M., Eberhard, A. et al. Graph neural networks for materials science and chemistry. Commun Mater 3, 93 (2022). https://doi.org/10.1038/s43246-022-00315-6

2. Tox21 Machine Learning Data Set: http://bioinf.jku.at/research/DeepTox/tox21.html

3. Drug Design Data Resource: https://drugdesigndata.org/about/grand-challenge-4/bace

4. ClinTox Dataset: https://huggingface.co/datasets/zpn/clintox

5. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules David Weininger Journal of Chemical Information and Computer Sciences 1988 28 (1), 31-36 DOI: 10.1021/ci00057a005

6. https://www.rdkit.org/

7. Shi, Shuhao, et al. "Rf-gnn: Random forest boosted graph neural network for social bot detection." arXiv preprint arXiv:2304.08239 (2023).

8. “Introduction to Machine Learning” lecture notes by Prof. Nir Rosenfeld

9. Analyzing Learned Molecular Representations for Property Prediction Kevin Yang, Kyle Swanson, Wengong Jin, Connor Coley, Philipp Eiden, Hua Gao, Angel Guzman-Perez, Timothy Hopper, Brian Kelley, Miriam Mathea, Andrew Palmer, Volker Settels, Tommi Jaakkola, Klavs Jensen, and Regina Barzilay Journal of Chemical Information and Modeling 2019 59 (8), 3370-3388 DOI: 10.1021/acs.jcim.9b00237

10. Velickovic, Petar, et al. "Graph attention networks." stat 1050.20 (2017): 10-48550.

11. https://deepchem.io/