BENCHMARKING GRAPH NEURAL NETWORKS FOR EXTRAPOLATIVE BIOACTIVITY PREDICTION ACROSS PROTEIN TARGETS

Martin Brokeš
Department of Informatics and Chemistry
University of Chemistry and Technology
Prague
brokesm@vscht.cz

ABSTRACT

Predicting small-molecule bioactivity remains a fundamental task in cheminformatics and drug discovery. While classical machine learning models often report strong performance under random dataset splits, such evaluations frequently overestimate real-world generalization due to chemical similarity leakage between training and test sets. In this work, we benchmark two graph neural network (GNN) architectures – Chemprop and a Gated Graph Neural Network (GGNN) – against a strong tree-based baseline, XGBoost, implemented via scikit-learn. The benchmark spans ten large-scale binary classification datasets, each corresponding to a distinct human protein target curated from the Papyrus database. To explicitly assess extrapolation ability, we employ three dataset splitting strategies of increasing difficulty: random, cluster-based, and aggregate cluster-based splits. For each dataset and split, extensive hyperparameter optimization is conducted using successive Optuna runs. Model performance is evaluated using multiple classification metrics and statistical significance is assessed using Tukey–Hones multiple comparison tests. Our results demonstrate that while XGBoost is competitive under random splits, graph-based neural models significantly outperform it under cluster-based splits, highlighting their superior extrapolative capabilities.

1 Introduction

Machine learning (ML) methods have become integral to modern drug discovery pipelines, particularly for predicting compound–target bioactivity [8]. Traditional approaches rely on fixed molecular descriptors combined with models such as random forests or gradient-boosted trees. Although these methods achieve strong performance under random train–test splits, such evaluations often suffer from overly optimistic estimates due to scaffold overlap between splits [11].

Graph neural networks (GNNs) address this limitation by learning representations directly from molecular graphs. Message passing neural networks (MPNNs) [7] and gated graph neural networks (GGNNs) [9] have demonstrated improved generalization, particularly in extrapolative regimes where test compounds differ structurally from training data.

Despite the growing popularity of GNNs, systematic benchmarks that explicitly test extrapolation performance across diverse biological targets remain limited. Furthermore, comparisons are often confounded by inconsistent dataset curation, evaluation protocols, or insufficient hyperparameter optimization.

In this work, we present a controlled benchmark comparing Chemprop [11], GGNNs [9], and XGBoost [6] across ten human protein targets. We explicitly study extrapolation by evaluating models under random, cluster, and aggregate cluster splits.

Our contributions are as follows:

- We construct a curated benchmark of ten large-scale bioactivity datasets derived from Papyrus, following a consistent activity thresholding protocol.
- We systematically compare Chemprop, GGNN, and XGBoost across random, cluster, and aggregate cluster splits.
- We perform extensive hyperparameter optimization using Optuna for each model–dataset–split combination.
- We assess statistical significance of performance differences using Tukey–Hones tests.

2 Dataset Construction

2.1 Bioactivity Labeling

Compound–target interaction data were obtained from the Papyrus database version 05.6 [10]. Each compound i associated with a target protein t is assigned a median inhibition constant $\tilde{K}_{i,t}$ expressed in logarithmic units.

Following the protocol of Lenselink et al. [8], binary activity labels are defined as:

$$y_{i,t} = \begin{cases} 1 & \text{if } \tilde{K}_{i,t} < 6.5 \\ 0 & \text{if } \tilde{K}_{i,t} \geq 6.5 \end{cases} \tag{1}$$

This thresholding approach balances biological relevance with robustness to experimental noise and has been shown to yield stable classification performance in large-scale bioactivity datasets.

2.2 Target Selection

Targets were selected according to the following criteria:

- Target organism is *Homo sapiens*
- At least 5% of compounds labeled as active
- Unique protein family assignment based on UniProt annotations

One cancer-related target without family annotation was included after confirming its evolutionary independence.

From all eligible targets, the ten largest were selected to ensure sufficient statistical power. Table 1 summarizes the selected targets, including UniProt identifiers, protein names, and family annotations.

Table 1: Names and protein families of selected protein targets

ID	Name	Protein Family
Q16637	Gemin-1	SMN family
P42336	PI3-kinase subunit alpha	PI3/PI4-kinase family
P08684	Cytochrome P450 3A4	Cytochrome P450 family
Q12809	hERG1	Potassium channel family
P04637	Phosphoprotein p53	p53 family
Q9Y468	Lethal(3)malignant brain tumor-like protein 1	–
P00918	Carbonic anhydrase 2	Alpha-carbonic anhydrase family
P14416	Dopamine D2 receptor	G-protein coupled receptor 1 family
P03372	Estrogen receptor	Nuclear hormone receptor family
P22303	Acetylcholinesterase	Type-B carboxylesterase/lipase family

3 Methods

3.1 Problem Formulation

For each target, we consider a binary classification problem:

$$f_\theta : \mathcal{G} \to [0, 1] \tag{2}$$

where $\mathcal{G}$ denotes the space of molecular graphs and f_θ predicts the probability of activity.

3.2 Dataset Splitting

Each dataset $\mathcal{D}$ is partitioned into training, validation, and test sets using three strategies:

1. **Random split**: $\mathcal{D}$ is randomly partitioned.
2. **Cluster split**: Compounds are clustered based on molecular similarity; entire clusters are assigned to splits [11].
3. **Aggregate cluster split**: A stricter clustering strategy that minimizes inter-split similarity.

These strategies progressively increase the difficulty of the learning task and better approximate real-world lead optimization and virtual screening scenarios.

3.3 Models

3.3.1 Chemprop

Chemprop is a message-passing neural network that operates directly on molecular graphs. At each message-passing step, atom-level hidden states are updated by aggregating information from neighboring atoms and bonds. For a molecular graph $G = (V, E)$, atom hidden states $\mathbf{h}_v^{(t)}$ are updated as:

$$\mathbf{m}_v^{(t)} = \sum_{u \in \mathcal{N}(v)} M(\mathbf{h}_v^{(t-1)}, \mathbf{h}_u^{(t-1)}, \mathbf{e}_{uv}) \tag{3}$$

$$\mathbf{h}_v^{(t)} = U(\mathbf{h}_v^{(t-1)}, \mathbf{m}_v^{(t)}) \tag{4}$$

A readout function aggregates node embeddings:

$$\mathbf{h}_G = \sum_{v \in V} \mathbf{h}_v^{(T)} \tag{5}$$

3.3.2 Gated Graph Neural Network

The GGNN model used in this work is based on gated recurrent updates applied to graph-structured data. Node states are iteratively updated using gated mechanisms analogous to GRUs, allowing the network to regulate information flow across message-passing steps:

$$\mathbf{a}_v^{(t)} = \sum_{u \in \mathcal{N}(v)} \mathbf{W}_{e_{uv}} \mathbf{h}_u^{(t-1)} \tag{6}$$

$$\mathbf{h}_v^{(t)} = \text{GRU}(\mathbf{h}_v^{(t-1)}, \mathbf{a}_v^{(t)}) \tag{7}$$

Gating allows adaptive control of information flow across message-passing iterations [9].

3.3.3 XGBoost

As a non-neural baseline, we employ XGBoost via the scikit-learn interface. Molecular inputs are represented using fixed-length descriptor vectors – Morgan fingerprints. XGBoost serves as a strong baseline due to its robustness, interpretability, and strong performance on tabular data. XGBoost optimizes an additive tree ensemble:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(\mathbf{x}_i) \tag{8}$$

by minimizing:

$$\mathcal{L} = \sum_i \ell(y_i, \hat{y}_i) + \sum_k \Omega(f_k) \tag{9}$$

where ℓ is the logistic loss and Ω is a regularization term [6].

3.4 Hyperparameter Optimization

For each model–dataset–split combination, hyperparameters are optimized using three successive Optuna runs [4]. Initially, 80 trials were performed over broad parameter ranges. The search space was then refined based on the top 10 trials, where the new bounds were set according to the minimum and maximum values present in the top 10 trials. In case of categorical parameters, only those present in the top 10 trials were retained. With this setting, another 80 trials were conducted. Subsequently, the search space was further refined using the same approach. In the final stage, 160 trials were performed, resulting in the identification of the optimal hyperparameter combination. Best-performing hyperparameter configurations were selected based on validation performance.

3.5 Evaluation Metrics

We report:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{10}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{11}$$

$$\text{F1} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{12}$$

$$\text{MCC} = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{13}$$

ROC-AUC is computed as the area under the receiver operating characteristic curve.

Statistical significance is assessed using Tukey–Hones tests [5].

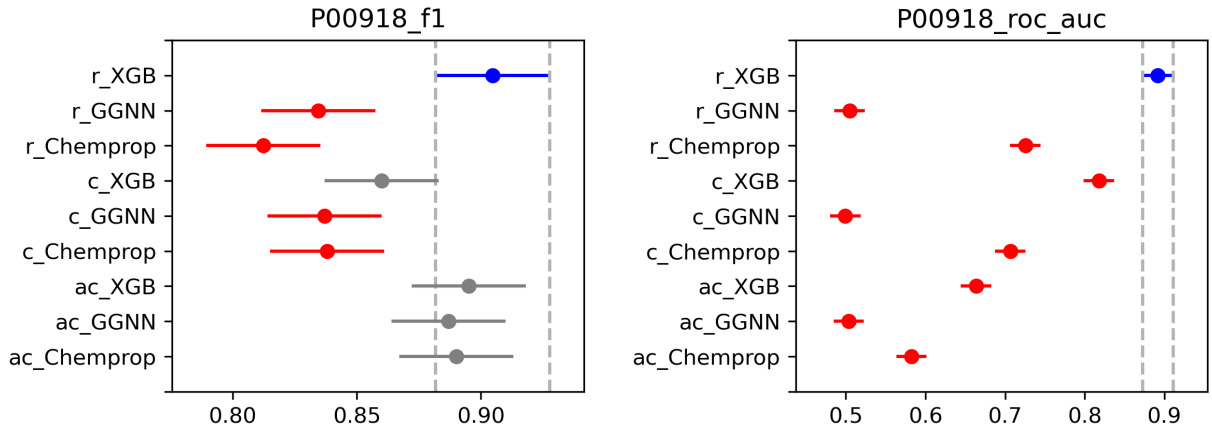
4 Results and Discussion

The quantitative results obtained from the benchmark experiments reveal substantial limitations that preclude meaningful comparison between the evaluated models. In particular, both graph neural network architectures – Chemprop and the Gated Graph Neural Network (GGNN) – were effectively trained for a maximum of two epochs due to typographical errors present in the Optuna search space definitions during the study. As a consequence, neither neural model reached a regime of convergence, resulting in severe underfitting across all datasets and splitting strategies.

This undertraining manifests consistently across evaluation metrics, including F1-score, Matthews correlation coefficient (MCC), and ROC-AUC. The resulting performance values for the neural models are close to random guessing in many cases and exhibit high variance across Optuna trials. Under these conditions, differences between Chemprop and GGNN are not statistically meaningful and do not reflect the intrinsic modeling capacity of either architecture.

In contrast, XGBoost exhibits stable and interpretable performance trends across datasets and splitting strategies. While absolute performance varies by target, a clear monotonic degradation is observed when moving from random splits to cluster-based and aggregate cluster-based splits. This behavior which can be seen in Figure 1b is consistent across most metrics and aligns with expectations regarding increasing extrapolation difficulty. On the other hand, metrics with not that apparent decreasing trend for XGBoost also exist (Figure 1a). However, these can be dataset specific and ought to be interpreted in combination with ROC-AUC score.

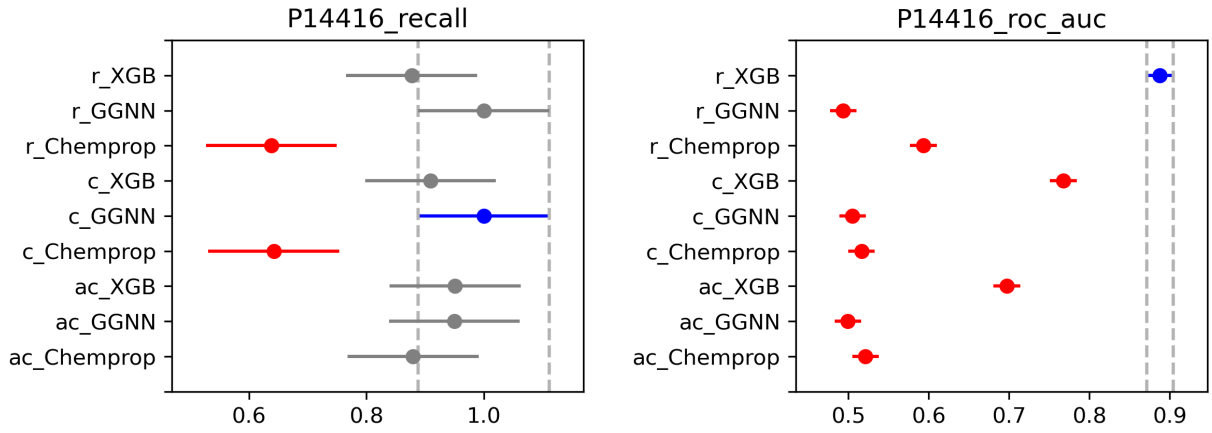
In isolated dataset–metric combinations, GNN-based models appear to outperform XGBoost, as exemplified by the recall score in Figure 2. However, interpreting such metrics in isolation is misleading. Due to severe underfitting, the GNN-based models fail to learn meaningful decision boundaries and, when combined with highly imbalanced datasets, may yield artificially inflated recall values (e.g., close to 1.0) despite producing near-constant predictions. Under these conditions, high recall does not indicate effective discrimination between active and inactive compounds. Consequently, all classification metrics must be interpreted jointly with the ROC-AUC, which remains close to 0.5 for GNN-based models, indicating performance near random guessing.



(a) F1 score for P00918 dataset.

(b) ROC-AUC score for P00918 dataset.

Figure 1: Tukey–Hones plots depicting the distributions of F1 and ROC-AUC scores for P00918 dataset.



(a) Recall for P14416 dataset.

(b) ROC-AUC score for P14416 dataset.

Figure 2: Tukey–Hones plots depicting the distributions of recall and ROC-AUC for P14416 dataset.

Overall, the numerical results obtained for the neural models are not suitable for drawing conclusions regarding their relative effectiveness. The only robust and interpretable signal present in the benchmark is the degradation trend observed for XGBoost as extrapolation requirements increase.

Only representative examples are displayed in above figures. The remaining plots for all metrics across all datasets can be found here[3].

5 Conclusion

Our results highlight the importance of realistic splitting strategies. As documented by Tukey–Hones plots, classical fingerprint-based models gradually degrade in extrapolative bioactivity prediction.

Future work should repeat the benchmark under computational conditions that permit full convergence of neural models. Only then can reliable conclusions be drawn regarding the relative extrapolative performance of graph neural networks and tree-based methods in bioactivity prediction.

The benchmarking process has not been completed yet due to computational extensiveness and the typographical errors present in Optuna search space during the first benchmarking procedure. The completed benchmarking workflow can be found here[2].

The completed and successfully tested QSPRpred API integration of the GGNN model can be found here[1].

References

[1] Ggnn integration into qsprpred. `https://github.com/brokesm/QSPRpred/tree/add-graph-dl`, 2025. Accessed: 2025-12-02.

[2] Metódy výpočetnej inteligencie – semestrálna práca. `https://github.com/brokesm/mvi-sp`, 2025. Accessed: 2025-11-01.

[3] Metódy výpočetnej inteligencie – semestrálna práca – grafy. `https://github.com/brokesm/mvi-sp/tree/main/plots`, 2025. Accessed: 2025-12-28.

[4] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework, 2019.

[5] Practical Cheminformatics. Even more thoughts on ml method comparison. `https://practicalcheminformatics.blogspot.com/2025/03/even-more-thoughts-on-ml-method.html`, 2025. Accessed: 2025-11-05.

[6] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 785–794. ACM, August 2016.

[7] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry, 2017.

[8] Eelke B. Lenselink, Niels ten Dijke, Brandon Bongers, George Papadatos, Herman W.T. van Vlijmen, Wojtek Kowalczyk, Adriaan P. IJzerman, and Gerard J.P. van Westen. Beyond the hype: Deep neural networks outperform established methods using a chembl bioactivity benchmark set. *bioRxiv*, 2017.

[9] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. Gated graph sequence neural networks, 2017.

[10] Béquignon OJM, Bongers BJ, Jespers W, IJzerman AP, van der Water B, and van Westen GJP. Papyrus: a large-scale curated dataset aimed at bioactivity predictions., 2023.

[11] 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. Analyzing learned molecular representations for property prediction, 2019.