Exposé
Topic: ECFP inspired Graph Neural Network that makes use of interim iteration results

Albert Dinstl

March 2025

Background

In molecular property prediction, it is crucial to develop effective molecular representations—commonly known as molecular fingerprints—to serve as inputs for machine learning models. The traditional paradigm in this area is the discrete and rule based ECFP algorithm. Another approach is using Graph Neural Networks (GNN) which apply a data driven deep learning strategy to learning powerful molecular representations. A third strategy is Neural Graph Fingerprints (NGF), which blend ECFP with the adaptive learning capabilities of GNNs.

Extended Connectivity Fingerprint (ECFP)

ECFP generates fixed-size binary vectors by iteratively aggregating local neighborhood information of atoms. In each iteration, the algorithm combines an atom's features with those of its neighbors, hashing the result to create discrete structural identifiers that progressively capture larger substructures. These identifiers set corresponding bits in the fingerprint, yielding a representation that encodes the presence or absence of specific substructures.[**ECFP**]

Graph neural network (GNN)

GNNs learn molecular representations in an end-to-end differentiable manner. They propagate and transform node and edge features through a series of message-passing layers and then aggregate the resulting information with a global pooling operation at the readout layer.

Neural Graph Fingerprints (NGF)

Neural Graph Fingerprints build on the concept of ECFP by replacing its discrete, non-differentiable operations with differentiable neural network components. This key innovation allows NGF to generate molecular fingerprints that

retain the interpretability and structural insights of ECFP while benefiting from gradient-based optimization. In contrast to standard GNN approaches that are primarily geared toward learning representations for end-to-end tasks, NGF is specifically designed to mimic the functionality of traditional fingerprints. [**NGF**]

The key difference between these approaches is that ECFP relies on fixed, rule-based, and non-differentiable operations like hashing and bit setting, whereas GNNs leverage continuous, differentiable functions that allow the model to optimize the feature extraction process during training. Neural Graph Fingerprints (NGF) bridge these paradigms by substituting the discrete steps of ECFP with differentiable neural operations.

Motivation

Research interest and questions

This thesis explores how to integrate the discrete and iterative approach of ECFP into the design of Graph Neural Networks aiming at producing high quality embeddings for downstream tasks like molecular property prediction. This research seeks to answer the following questions:

- How can discrete iterative methods like ECFP be integrated into the continuous and differentiable nature of Graph Neural Networks?
- What benefits arise from leveraging the concatenation of intermediate representations in Graph Neural networks?

Goal of the project

The primary goal of this project is to investigate how the concept of combining interim iteration results—as employed in ECFP—can be integrated into the design of Graph Neural Networks. Specifically, the project aims to develop a GNN architecture that generates molecular embeddings by explicitly incorporating and aggregating intermediate node-level embeddings and combining them with concatenation.

Approach

The initial phase of the project involves using a frozen, randomly initialized Graph Isomorphism Network (GIN) [**GIN**] as a fixed feature extractor for molecular graphs. This GIN converts each molecular graph into a corresponding embedding, which is then used as input to train an MLP classifier. The performance of this classifier—measured by ROC-AUC—will be directly compared to that obtained using traditional ECFP bit vectors as molecular representations. Additionally, a frozen Neural Graph Fingerprint (NGF) [**NGF**] model will serve

as another baseline, with its embeddings used to train a separate MLP classifier. In the second phase, the focus shifts to designing an enhanced GIN that integrates mechanisms such as Jumping Knowledge concatenation [**JK**] to aggregate interim iteration outputs into the final graph embedding. The aim is to assess how closely the embeddings produced by this enhanced, frozen GIN match the discriminative power of ECFP when used for downstream tasks via an MLP classifier.
Additionally, the expressiveness of the embeddings and fingerprints generated by each method will be evaluated. This assessment will involve determining the number of unique embeddings or fingerprints produced from a given molecular dataset, thereby providing insight into the discriminative capacity of each representation approach.

Challenges to consider

A differentiable neighbourhood aggregation method

In ECFP, the local neighborhood of each atom is aggregated by hashing the atom's feature array along with its neighbors' features into a new substructure identifier. This process, while effective for generating fixed binary fingerprints, is inherently non-differentiable. For a GNN to support end-to-end learning via gradient descent, the neighborhood aggregation must be implemented using differentiable operations. [**NGF**]

Duplicate removal

In traditional ECFP fingerprint generation, if the same identifier is produced for different atoms or substructures, the fingerprint simply marks the corresponding bit as active, effectively collapsing duplicates into a single indicator. This non-differentiable process is acceptable for fixed, binary fingerprints. However, in a GNN, every operation must be differentiable to allow for gradient-based learning. Therefore, the GNN must employ differentiable aggregation methods that can handle duplicate or highly similar intermediate representations without disrupting the learning process.

Transforming graph embeddings into fixed size fingerprints

When using concatenation for neighborhood aggregation, the resulting graph-level embedding may vary in length from one molecule to another, as it depends on the number of nodes (atoms). However, downstream tasks such as classification require input vectors of a consistent dimensionality. To address this, a differentiable operation must be applied to transform the variable-length, concatenated embeddings into fixed-size neural fingerprints.

Milestones

Milestone 1 - 01.03.2025-01.04.2025

- Read relevant literature
- Setup a training pipeline and the benchmark experiment

Milestone 2 - 01.04.2025-15.05.2025

- Compare frozen downstream task performance between ECFP4, ECFP6, NGF2, NGF3, GIN2 and GIN3

Milestone 3 - 15.05.2025-15.06.2025

- Develop enhanced GIN and assess frozen downstream task performance
- Train enhanced GIN and assess downstream task performance

Milestone 4 - 15.06.2025-30.06.2025

- Presentation of the findings - Date: 25.06.2025
- Submission of the written report