Title: Extended Connectivity Fingerprint inspired Graph Neural Network that makes use of interim iteration results

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. Two dominant paradigms have emerged in this area: Extended Connectivity Fingerprints (ECFP) and Graph Neural Networks (GNN).

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.

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

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.

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 fingerprints 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 aggregation of intermediate representations in Graph Neural networks?

Goal of the Project

The primary goal of this project is to investigate how the concept of aggregating 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 fingerprints by explicitly incorporating and aggregating intermediate node-level embeddings. These learned fingerprints will then be evaluated against traditional ECFP bit vectors on downstream tasks, such as molecular property prediction, to assess improvements in performance.

Project Design

Single-Layer GNN for Node-Level Embeddings (ECFP First Iteration Analog)

The initial phase of the project is to design and randomly initialize a single-layer GNN that produces graph embeddings similar to the output of the first iteration of ECFP. In this stage, each atom (node) is represented by a common set of features, and its embedding is updated by aggregating information from its immediate neighbors—mirroring the way ECFP updates an atom's identifier with local structural information. The goal for this part of the project will be to design a network that can produce the same number of uniquely distinguishable embeddings as ECFP in its first iteration. The quality of these embeddings will be asessed in a downstream task and their accuracy will be compared to ECFP2 bit vectors.

Multi-Layer GNN for Molecular Fingerprinting Using Interim Iteration Results

In the second phase, the approach will be extended to a multi-layer GNN that explicitly captures and aggregates the node embeddings generated at each intermediate layer. This design will allow for the dissection of contributions from each iteration, ultimately producing a global molecular fingerprint. The final molecular-level embedding will be evaluated on downstream tasks, such as molecular property prediction, and its performance will be compared against traditional ECFP bit vectors. This comparative analysis aims to assess whether incorporating interim iteration results can enhance the quality and interpretability of the learned molecular representations.

Challenges to consider

- A differentiable neighbourhood aggregation method
  In ECFP the information of neighbouring atoms is aggregated by hashing the array of an atom and its neighbours to a new identifier of the substructure containing the aggregated information of all neighbouring atoms of the substructure. To enable end to end learning with GNNs, this neighbourhood aggregation has to be differentiable.
- 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—such as softmax pooling or attention-based pooling—that can handle duplicate or highly similar intermediate representations without disrupting the learning process.

- Transforming graph embeddings into fixed size fingerprints
  When using concatenation as a neighbourhood aggregation function, it is not guaranteed that the final graph embeddings will be vectors of the same size for each molecule. However training a classifier on a downstream task requires the input vectors to be of similar dimensionality. To adress this, a differentiable operation that transforms concatenated vectors into fixed size neural fingerprints will need to be employed.

Schedule

01.03.2025 - 01.04.2025

- Reading relevant literature
- Setup of a training pipeline and the benchmark experiment
- Development of a single layer GNN that produces the same number of distinguishable node embeddings as ECFP2

01.04.2025 - 15.05.2025

- Finding a way to deal with duplicates in node embeddings
- Finding a way to transform graph embeddings into fixed size fingerprints
- Extension of the GNN to multiple layers

15.05.2025 - 15.06.2025

- This timeframe serves as buffer time if any problems arose during the implementation
- Implementation of any improvements
- Refinement and testing of different experiment structures

15.06.2025 - 30.06.2025

- Preparation of the presentation
- Writing of the report