Graph Mining Course Project Progress Report

Submission Date: January 7, 2026
Course: Graph Mining [4041]
Instructor: Dr. Zeinab Maleki
Project Title: Improving Node Classification in Heterophilic Graphs Using Similarity-Weighted H2GCN

Student Information

- **Student Name(s):** Shokufa Shalchi, Zahra Aboutalebi
- **Student ID(s):** 40124913, 40116943
- **Email(s):** s.shalchi@ec.iut.ac.ir, z.abootalebi@ec.iut.ac.ir

Executive Summary

Since the submission of the project proposal, we have successfully established our experimental framework and baseline comparisons. Initially, we validated the **H2GCN** model by running the authors' original code on the standard **Cora** dataset.

Due to the unavailability of the originally proposed **GitHub** dataset, we transitioned to the **Reddit** dataset. To handle its massive size under memory constraints, we generated a subgraph using **Snowball Sampling**. We evaluated both H2GCN and a standard GCN on the same sampled Reddit subgraph. Our results show that H2GCN achieves significantly higher accuracy than the standard GCN (approximately 72% vs. 31%). This indicates that although Reddit is not a strongly heterophilic graph, the separation mechanism of H2GCN remains effective on this sampled subgraph. These findings motivate further improvements through similarity-aware aggregation.

Progress on Objectives

Objective 1: Reproduce and evaluate the H2GCN model

Status: Completed. We utilized the original source code provided by the H2GCN authors. * **Validation:** First, we successfully executed the code on the **Cora** dataset to ensure the environment and dependencies were correctly configured. * **Adaptation:** We then adapted the data loading pipeline to accept our custom Reddit subgraph, resolving input type mismatches (SparseTensor vs. Dense arrays). * **Outcome:** The model runs stably for 200 epochs on the Reddit subgraph.

Objective 2: Establish a Baseline Comparison (GCN)

Status: Completed. We implemented a standard Graph Convolutional Network (GCN) using PyTorch Geometric to serve as a homophilic baseline. *

Outcome: The GCN was trained on the same Reddit subgraph. * **Finding:** GCN achieved lower accuracy compared to H2GCN on the sampled Reddit subgraph, highlighting the effectiveness of H2GCN's separation strategy even outside strongly heterophilic benchmarks.

Objective 3: Integrate feature similarity (Next Step)

Status: Pending. With the baselines established and the data pipeline fixed, we are now ready to implement the proposed similarity-weighting mechanism into the aggregation layer.

Work Accomplished

1. Dataset Selection and Preprocessing

- **Change of Plan (GitHub to Reddit):**
  In the proposal, we planned to first run experiments on the GitHub Developer Graph dataset. However, after investigation, we found that this dataset is no longer accessible and has been removed from standard repositories. Therefore, we transitioned to the **Reddit dataset** as an alternative real-world benchmark.

- **Reason for Choosing Reddit:**
  Reddit was selected because it is:
  - a **large-scale real social graph** suitable for node classification,
  - **not artificially optimized for heterophily models**, unlike citation datasets,
  - a **more realistic challenge** for evaluating heterophilic graph learning methods.

- **Sampling Strategy:**
  Due to memory limitations, we extracted a **50k-node connected subgraph using Snowball Sampling** to preserve local graph structure while avoiding out-of-memory errors.

- **Contradictory Results vs. Original Paper:**
  In our experiments, H2GCN achieved higher accuracy than GCN (approximately 72% vs. 31%) on the sampled Reddit subgraph. This differs from common assumptions that Reddit primarily favors homophilic models. While Cora is a small citation network with strong homophily, Reddit exhibits more diverse and noisy neighborhood structures, where the separation-based aggregation of H2GCN can still provide benefits. This suggests that the effectiveness of H2GCN is not limited to purely heterophilic graphs but also depends on local structural patterns introduced by sampling.

- **Final Insight:**
  Our results indicate that on the sampled Reddit subgraph, H2GCN consis-

tently outperforms a standard GCN, even when the GCN is trained for a large number of epochs. This suggests that simple smoothing-based aggregation is insufficient for this graph structure, and that separation-based designs such as H2GCN provide a more robust representation. These observations further motivate our proposed extension using similarity-weighted aggregation.

2. H2GCN Evaluation (Original Code Adaptation)

We ran the H2GCN model on the Reddit subgraph for 200 epochs. We faced challenges regarding data types (the code expected specific input formats not natively provided by standard loaders), which we resolved.

H2GCN Results (Selected Epochs): The model reached a plateau around **72%** accuracy.

```
Epoch 005 | Train Loss: 2.6636 | Test Acc: 0.4481
Epoch 020 | Train Loss: 1.7846 | Test Acc: 0.6180
Epoch 050 | Train Loss: 1.3472 | Test Acc: 0.6937
Epoch 080 | Train Loss: 1.2190 | Test Acc: 0.7107
Epoch 110 | Train Loss: 1.1667 | Test Acc: 0.7208
Epoch 140 | Train Loss: 1.1459 | Test Acc: 0.7189
Epoch 180 | Train Loss: 1.1272 | Test Acc: 0.7198
Epoch 200 | Train Loss: 1.1236 | Test Acc: 0.7202
```

3. GCN Implementation and Comparison

We implemented a standard GCN model to test the performance of a homophilic-based architecture on this data.

GCN Results (Selected Epochs): The GCN model demonstrated superior convergence, reaching approximately **31%** accuracy.

```
Epoch 010 | Loss 3.1730 | Acc 0.2211
Epoch 040 | Loss 3.0073 | Acc 0.2421
Epoch 070 | Loss 2.8462 | Acc 0.2544
Epoch 100 | Loss 2.6979 | Acc 0.2731
Epoch 130 | Loss 2.5758 | Acc 0.2906
Epoch 160 | Loss 2.5020 | Acc 0.2915
Epoch 190 | Loss 2.4076 | Acc 0.3142
Epoch 200 | Loss 2.3827 | Acc 0.3184
```

Interpretation: The relatively low accuracy of the standard GCN suggests that naïve neighborhood aggregation is not sufficient for this sampled Reddit subgraph. Despite Reddit not being explicitly heterophilic, the presence of noisy or label-inconsistent neighborhoods limits the effectiveness of homophily-based smoothing. In contrast, H2GCN benefits from separating ego and neighbor information, leading to substantially better performance.

Challenges Encountered and Resolutions

- **Challenge 1: Dataset Unavailability:** The GitHub dataset was inaccessible.
  Resolution: Switched to Reddit, a standard benchmark in the field.
- **Challenge 2: Hardware Constraints (RAM):** The Reddit graph caused OOM errors.
  Resolution: Implemented Snowball Sampling to create a representative 50k-node subgraph.
- **Challenge 3: Code Compatibility:** The original H2GCN code failed on the new dataset due to input formatting (Sparse/Dense mismatches).
  Resolution: We debugged the data loader and wrote a custom adapter to ensure the Reddit vectors matched the model's expected input tensors.

Next Steps

1. **Similarity Weighting:** We will now modify the H2GCN aggregation function to include a coefficient based on feature similarity (e.g., Cosine Similarity).
2. **Hypothesis Testing:** We aim to see if adding similarity weights can help H2GCN bridge the performance gap with GCN on this specific subgraph.

References

1. J. Zhu, et al., "Beyond Homophily in Graph Neural Networks: Current Limitations and Effective Designs," *NeurIPS*, 2020.
2. T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *ICLR*, 2017.

Student Signature(s): Shokufa Shalchi, Zahra Aboutalebi January 7, 2026