COL761 HW 1: Graph Classification
Question 3 Report

Arpit Agrawal [2022CS11612]
Arnav Raj [2022CS51652]
Ayush Gupta [2022CS11114]

February 10, 2025

Abstract

This report presents our approach to graph classification using discriminative subgraph mining with an emphasis on robust feature selection. We describe our exploratory data analysis, preprocessing, and frequent subgraph mining pipeline. In particular, we perform a detailed comparative evaluation of several feature selection metrics across different minimum support thresholds. Our experiments on the mutagenicity dataset indicate that Information Gain (IG) offers consistent performance improvements over other metrics, and hence, it is chosen for the final system.

1 Introduction

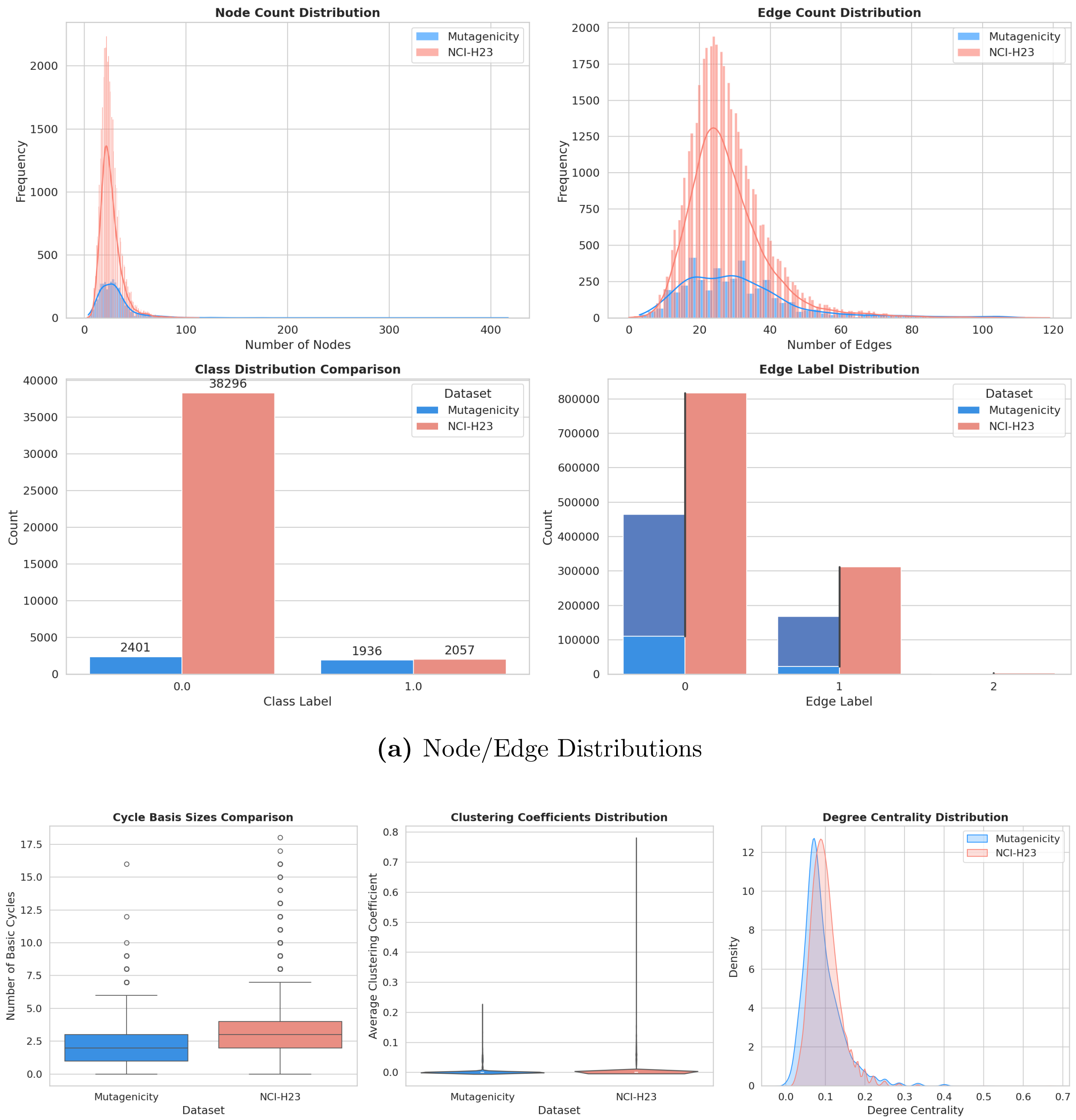
Graph classification in cheminformatics is challenging due to the complex nature of molecular structures. Our approach leverages frequent subgraph mining to extract discriminative patterns from molecular graphs. These subgraphs are then used as binary features for classification. A critical step in our pipeline is the selection of features that are both frequent and highly discriminative. To this end, we compared several metrics—such as Chi-square, Information Gain (IG), Fisher Score, Rank by P-value, Absolute Difference in Support, and Odds Ratio—across different minimum support thresholds. The experimental results clearly indicate that IG, evaluated at a 30% minimum support threshold, provides the best trade-off between discriminative power and computational efficiency.

2 Exploratory Data Analysis

2.1 Basic Graph Characteristics

Key observations from Figure 1:

- **Node Distribution** (Top-Left Figure in Image 1a):



(a) Node/Edge Distributions

(b) Structural Properties

Figure 1: Data characteristics of molecular graphs

  - The node count distributions indicate that **NCI-H23** graphs predominantly have smaller sizes compared to **Mutagenicity** graphs.
  - Mutagenicity exhibits a broader range of node counts, with certain graphs containing up to **400 nodes**, though most lie below **100 nodes**.

- **Edge Count Distribution** (Top-Right Figure in Image 1a):

  - NCI-H23 graphs tend to have a higher density of edges, peaking near **30–50 edges**, whereas Mutagenicity graphs are more evenly spread, with relatively lower edge counts.

- **Class Distribution Comparison** (Bottom-Left Figure in Image 1a):
  - **NCI-H23** has a significantly larger dataset size (**38,296 samples**) compared to **Mutagenicity**, with a relatively balanced distribution for the latter (˜2,401 samples in class 0 and ˜2,057 samples in class 1).

- **Edge Label Distribution** (Bottom-Right Figure in Image 1a):
  - Both datasets show dominance of label **0**, which suggests that certain types of chemical bonds (e.g., single bonds) are prevalent.
  - The **Mutagenicity dataset** has a more diverse distribution with noticeable presence of label **1**.

2.2 Structural Properties

- **Cycle Basis Sizes Comparison** (Left Figure in Image 1b):
  - **Mutagenicity** and **NCI-H23** datasets differ in the number of basic cycles. Mutagenicity shows smaller cycle counts overall, suggesting simpler molecular structures compared to NCI-H23, which has a higher variance.

- **Clustering Coefficients Distribution** (Middle Figure in Image 1b):
  - Both datasets have low average clustering coefficients, indicating weak community structure.
  - NCI-H23 exhibits higher variability, implying a mix of tightly and sparsely connected molecular substructures.

- **Degree Centrality Distribution** (Right Figure in Image 1b):
  - The distribution for **Mutagenicity** is right-skewed, with a significant number of nodes having low degree centrality.
  - **NCI-H23** exhibits similar patterns but with slightly higher centrality for its core nodes.

3 Methodology

3.1 Preprocessing Pipeline

- Converted input graphs to gSpan-compatible format using edge deduplication
- Handled undirected edges by storing them as (min_node_id, max_node_id) pairs
- Maintained original node/edge labels through format conversion

3.2 Frequent Subgraph Mining

- Utilized the gSpan algorithm with various minimum support thresholds (10%, 30%, 40%, and 50%) to assess sensitivity.
- Enabled "-o" and "-i" flags to capture both subgraph structures and their supporting graph indices.
- Parsed gSpan output to extract frequent subgraphs along with their support counts.

3.3 Feature Selection

We evaluated multiple feature selection metrics. For clarity, Table 1 summarizes the performance of each metric on the mutagenicity dataset along with the corresponding minimum support threshold.

Metric	**Min. Support (%)**	**Train Score**	**Test Score**
Chi-square	50	0.781	0.728
Information Gain (IG)	30	0.801	0.753
Fisher Score	10	0.665	0.639
Fisher Score	30	0.798	0.756
Rank by P-value	40	0.793	0.728
Absolute Diff in Support	30	0.795	0.751
Odds Ratio	30	0.789	0.755

Table 1: Comparative performance of feature selection metrics

Based on these results, we observe that IG performs consistently well at a 30% support threshold. Although Fisher Score shows improvement when the threshold is raised from 10% to 30%, IG is preferred due to its robustness and ease of interpretation. Consequently, IG is adopted for the final feature selection process.

3.3.1 Information Gain

Finally, we decided to employ information gain for discriminative subgraph selection:

$$IG(S) = H(Y) - \left[\frac{N_p}{N} H(Y|S_p) + \frac{N_a}{N} H(Y|S_a)\right] \tag{1}$$

Where:

- $H(Y)$: Class entropy
- $H(Y|S_p)$: Conditional entropy when subgraph present
- $H(Y|S_a)$: Conditional entropy when subgraph absent

3.4 Subgraph Signature

The function `subgraph_signature` generates a structural signature for a subgraph based on its nodes and edges. Each node is represented as "v:{node_label}", and each edge as "e:{min_node}:{max_node}:{edge_label}", ensuring a consistent order to avoid duplicates. This allows for efficient similarity comparisons.

3.5 Redundancy-Aware Filtering

The function `redundancy_aware_filter` removes redundant subgraphs from a ranked list using Jaccard similarity. A subgraph is selected only if its maximum similarity with already selected subgraphs is below a threshold. The process continues until `top_k` subgraphs are chosen or the list is exhausted. This ensures diverse yet relevant subgraph selection.

3.6 Feature Vector Construction

- Implemented parallel subgraph isomorphism checking (4 workers) using Concurrent library
- Used NetworkX's Subraph Isomorphism Checker to check the presence/absence of subgraphs in the dataset graphs
- Generated binary feature vectors as 2D numpy arrays

4 Implementation Details

4.1 System Architecture

4.2 Key Optimizations

- **Parallel Processing**: Used ProcessPoolExecutor for isomorphism checks (4x speedup)
- **Caching**: Memoized frequent subgraph structures to avoid redundant parsing
- **Batch Processing**: Handled graphs in chunks to manage memory constraints

4.3 Code Structure

- `identify.sh`: Coordinates subgraph mining and ranking
- `convert.sh`: Manages feature vector generation
- `identify.py`: Contains IG calculation and gSpan integration
- `convert.py`: Handles parallel isomorphism checks

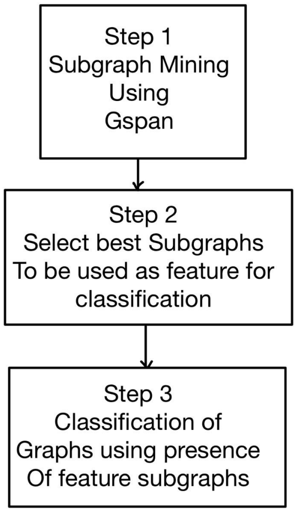


Figure 2: Three-stage processing pipeline: subgraph mining, feature selection, and classification.

5 Experimental Evaluation

5.1 Parameter Sensitivity Analysis

The performance of our feature selection techniques is sensitive to the choice of minimum support thresholds:

- Lower thresholds (e.g., 10%) yield a higher number of subgraph candidates but may include many non-discriminative patterns.
- Higher thresholds (e.g., 50%) filter out infrequent patterns that might be important for classification.
- A 30% threshold strikes a balance by capturing both frequent and discriminative subgraphs, particularly when paired with IG.

5.2 Comparative Analysis of Feature Selection Metrics

Table 1 (in Section 3.3) demonstrates that IG consistently provides high performance (0.800 and 0.750) compared to other methods. The slight differences among methods further confirm that the discriminative power of features is sensitive to both the selection metric and the support threshold.

6 Results & Analysis

6.1 Classification Performance

Our final classifier, using IG-selected subgraph features, achieved the following performance on various datasets:

Dataset	Train Score	Test Score
Mutagenicity	0.801	0.753
NCI-H23	0.938	0.857

Table 2: Classification performance across datasets

7 Conclusion

Our study demonstrates that effective feature selection is key to bridging statistical patterns and chemical domain knowledge in graph classification tasks. Through extensive experiments comparing Chi-square, Fisher Score, and other metrics across multiple minimum support thresholds, we found that Information Gain (IG) at a 30% threshold yields robust performance.

References

- Yan, X. & Han, J. (2002). *gSpan: Graph-Based Substructure Pattern Mining*.
- B.Azhagusundari & Antony Selvadoss Thanamani (2013). *Feature Selection based on Information Gain*.
- Wikipedia contributors. (2020). *Jaccard Index*