SELECTIVE FEATURE-NODE TRANSFORMATION FOR HETEROPHILIC GRAPH LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Graph neural networks (GNNs) can struggle on heterophilic graphs, where neighborhood aggregation mixes information from nodes with different labels. Feature-node transformation addresses this challenge by connecting original nodes through shared discrete features. However, exhaustive transformation includes every active feature, regardless of its predictive relevance or structural cost. This work proposes a selective transformation framework that identifies useful auxiliary structure under a feature-node budget while retaining the original attributes and topology. A relevance score combines normalized mutual information, pairwise label agreement above chance, coverage, and incidence-edge cost. Support-based shrinkage and component standardization balance these signals when labeled observations are limited. The selected incidence graph is integrated with the original topology through signed, feature-aware gated propagation. Analysis establishes exact auxiliary graph sizes and a sufficient condition for above-chance agreement along selected two-hop paths. Evaluation on six node-classification benchmarks shows substantial reductions in auxiliary nodes and edges, accompanied by lower runtime and memory measurements. Predictive outcomes vary across datasets, while matched-budget comparisons and ablations clarify when relevance scoring helps and when simpler criteria remain competitive. These findings establish selective feature-node construction as a practical approach to balancing predictive performance and computational cost in graph learning.

1 INTRODUCTION

Graph learning supports prediction in networks whose edges encode interactions, dependencies, and shared context. A graph neural network (GNN) should extract useful relational information even when connected nodes have different labels. Recent evaluations show that this heterophily does not alone determine prediction difficulty and that preprocessing, feature quality, and baseline design strongly affect conclusions (Platonov et al., 2023b;a; Zheng et al., 2026). Feature-node transformation provides additional routes for information exchange: GRAPHITE connects original nodes through auxiliary nodes representing shared discrete features (Qiu et al., 2026). Such routes can improve learning, but constructing every feature node also creates incidences whose cost may exceed their predictive value.

Which features should become auxiliary graph nodes when the transformation has a limited structural budget? This work addresses this question through selective feature-node transformation. Its central distinction is between retaining a feature as an input attribute and constructing that feature as an additional message-passing node. Selection preserves every original attribute and edge while controlling which feature-induced paths are available.

The technical idea evaluates a feature according to its label association, the label agreement of nodes sharing it, its coverage, and the number of edges it creates. Training-label statistics estimate relevance without accessing validation or test labels. Support shrinkage reduces the influence of weakly supported estimates, and standardization makes the component weights comparable. A top-K rule then selects the auxiliary nodes, and signed gates integrate their messages with the original topology. The analysis separates exact size reduction from conditional path agreement, because a smaller or purer graph does not necessarily improve prediction.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

A controlled exhaustive comparison examines graph reduction and predictive performance; matched-budget selectors assess the combined relevance score. Path diagnostics relate the retained structure to the theoretical conditions, while ablations and budget curves examine the roles of individual score components and selection strength. The evaluation identifies when relevance scoring or simpler selection is preferable.

The main contributions are as follows.

- A budgeted feature-node transformation separates auxiliary connectivity from input attributes and preserves the original graph (Section 3.3).
- A support-adjusted score integrates label relevance, pair agreement, coverage, and incidence cost before construction (Section 3.4).
- Selection-specific analysis establishes exact size bounds and conditional two-hop agreement, complemented by predictive and computational evaluation (Section 3.5).

2 Related Work

The review covers auxiliary budgets (Section 2.1), structural relevance (Section 2.2), and selection guarantees (Section 2.3).

2.1 Auxiliary Graph Structure

Heterophilic graph learning often modifies how messages are propagated rather than which auxiliary objects are constructed (Zheng et al., 2026). Scalable graph transformers, including DIFFormer and SGFormer, extend interactions beyond local neighborhoods through diffusion or attention (Wu et al., 2023a;b).

Topology processing offers a complementary approach. Locality-aware rewiring balances information flow, locality, and sparsity (Barbero et al., 2024). Joint denoising and rewiring (JDR) aligns graph and feature spectral spaces before downstream prediction (Linkerhägner et al., 2025).

GRAPHITE creates feature-induced paths by turning discrete features into nodes (Qiu et al., 2026). Exhaustive construction leaves open which features warrant auxiliary incidences. This work addresses that choice through budgeted construction that retains original attributes and topology (Section 3.3).

2.2 Structural Feature Relevance

Feature relevance depends on how an attribute interacts with graph propagation. Graph Feature Selection (GFS) uses Topological Feature Informativeness (TFI) to distinguish features that benefit from graph convolution and routes the remaining features through a non-GNN model (Zheng et al., 2025).

A structural feature selector faces an additional question: converting a feature into a node creates one edge per occurrence. Mutual information measures label association, whereas feature-sharing pair agreement measures a particular consequence of those incidences. Neither frequency nor association captures all these effects (Platonov et al., 2023a; Zheng et al., 2026).

Limited labeled support makes the selection statistics uncertain, especially for rare features. TFI-based feature routing and spectral denoising address related feature-processing questions but do not define the same auxiliary-node selection objective (Zheng et al., 2025; Linkerhägner et al., 2025). This work integrates support-adjusted relevance and incidence cost into a single structural ranking (Section 3.4).

2.3 Selection and Guarantees

Homophily measures characterize different aspects of graph structure and should not be treated as interchangeable predictors of accuracy (Platonov et al., 2023a). Recent analysis also cautions against universal statements connecting heterophily, propagation, and learning failure without specifying their assumptions (Arnaiz-Rodríguez & Errica, 2026).

GRAPHITE analyzes homophily and graph size for its transformation (Qiu et al., 2026). After selecting only a subset of feature nodes, the relevant quantities depend on the retained feature supports and label distributions.

Similarly, selective blocking studies uncertain message propagation, which differs from counting and evaluating feature-induced paths (Choi et al., 2025). Controlled evaluation is needed to distinguish a construction effect from changes in preprocessing or predictor configuration (Platonov et al., 2023b). Exact counts and a conditional path result characterize the selected graph without asserting an unconditional accuracy improvement (Section 3.5).

3 Selective Graph Transformation

The framework comprises preliminaries (Section 3.1), problem formulation (Section 3.2), budgeted construction (Section 3.3), scoring (Section 3.4), and analysis (Section 3.5). Figure 1 illustrates the sequence from feature statistics to auxiliary connectivity.

3.1 Preliminaries

Let $G = (V, E, X)$ be a simple undirected graph with $n = |V|$ nodes, $m = |E|$ unordered edges, and binary attributes $X \in \{0,1\}^{n\times d}$. The active feature set is $\mathcal{F} = \{k : \sum_i X_{ik} > 0\}$, with $D = |\mathcal{F}|$. Each original node has a class $y_i \in \{1, \dots, C\}$. In transductive classification, all attributes and edges are available, but only labels on the training set $L \subset V$ enter the feature statistics and learning loss. Validation labels select hyperparameters; test labels evaluate prediction.

For feature k, let $V_k = \{i : X_{ik} = 1\}$, $s_k = |V_k|$, $t_k = |V_k \cap L|$, and $n_{kc} = |\{i \in V_k \cap L : y_i = c\}|$. Let $\ell = |L| \geq 2$ and $n_c = |\{i \in L : y_i = c\}|$. The chance agreement of two distinct training nodes, sampled without replacement, is

$$p_0 = \frac{\sum_{c=1}^{C} \binom{n_c}{2}}{\binom{\ell}{2}}. \tag{1}$$

Equation (1) accounts for class imbalance; all label statistics are recomputed for each training split.

3.2 Problem Formulation

The first problem is budgeted auxiliary construction. For $\mathcal{S} \subseteq \mathcal{F}$, define its incidence cost $e_{\mathcal{S}} = \sum_{k\in\mathcal{S}} s_k$ and transformed graph $G^*_{\mathcal{S}}$. Given an integer node budget $K \leq D$, the ideal selection is

$$\max_{\mathcal{S}\subseteq\mathcal{F},\, |\mathcal{S}|=K} \Phi(G^*_{\mathcal{S}}, Y_L) - \rho e_{\mathcal{S}}, \qquad \rho \geq 0, \tag{2}$$

where Y_L denotes training labels and Φ represents predictive utility. The budget fixes node cost, but incidence cost varies. Equation (2) is a design goal rather than an exactly solved objective.

The second problem is featurewise utility estimation. With score S_k, the operational selection is

$$\mathcal{S}_K \in \arg\max_{\mathcal{S}\subseteq\mathcal{F},\, |\mathcal{S}|=K} \sum_{k\in\mathcal{S}} S_k, \qquad S_k = F(r_k, g_k, c_k, q_k, \beta_k). \tag{3}$$

Here r_k, g_k, c_k, q_k, and β_k denote association, purity lift, coverage, edge cost, and support reliability, respectively. Top-K ranking solves Eq. (3); Section 3.4 defines F, which approximates rather than recovers Φ.

The third problem is to characterize the selected graph through its size and path agreement:

$$\big(|V^*_{\mathcal{S}}|, |E^*_{\mathcal{S}}|\big), \qquad \Delta_{\mathcal{S}} = H_2(\mathcal{S}) - p_0, \tag{4}$$

where H_2 is the same-label fraction of labeled two-hop paths through selected feature nodes. Section 3.5 derives conditions for $\Delta_{\mathcal{S}} \geq 0$, separately from predictive accuracy.

These problems address *unselective auxiliary construction* through budgeted transformation (Section 3.3), *incomplete structural relevance assessment* through joint scoring (Section 3.4), and *uncharacterized effects of selection* through explicit size and path analysis (Section 3.5).

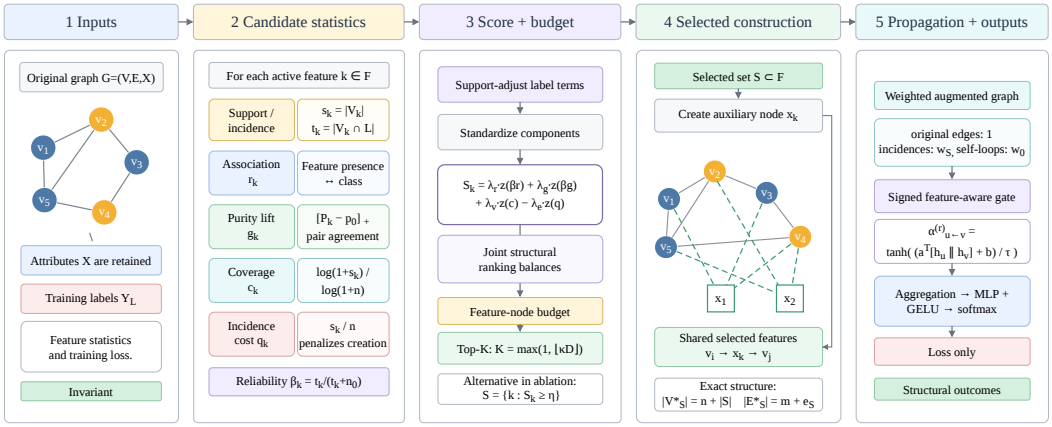


Figure 1: Selective feature-node construction. The original five-node graph and its attributes remain intact; selected feature nodes add two-hop paths between nodes sharing those features. Class colors illustrate the mechanism and are not test-label inputs to selection.

3.3 Budgeted Feature Nodes

For each selected feature $k \in \mathcal{S}$, create a node x_k and incidence edges $\{i, x_k\}$ for all $i \in V_k$. With original adjacency A and selected attribute columns $X_{\mathcal{S}}$, construction gives

$$V^*_{\mathcal{S}} = V \cup \{x_k : k \in \mathcal{S}\}, \quad E^*_{\mathcal{S}} = E \cup \{\{i, x_k\} : i \in V_k, k \in \mathcal{S}\}, \quad A^*_{\mathcal{S}} = \begin{bmatrix} A & X_{\mathcal{S}} \\ X_{\mathcal{S}}^\top & 0 \end{bmatrix}. \tag{5}$$

Equation (5) adds paths $i \rightarrow x_k \rightarrow j$ through shared features without feature-to-feature edges.

All original node attributes are retained, and an auxiliary node receives the mean attributes of its incident nodes, following feature-node transformation (Qiu et al., 2026):

$$X^*_{\mathcal{S}}[i,:] = X[i,:], \qquad X^*_{\mathcal{S}}[x_k,:] = \frac{1}{s_k} \sum_{i \in V_k} X[i,:]. \tag{6}$$

Active candidates ensure $s_k > 0$. Full and empty selections recover exhaustive construction and the original graph, respectively.

To integrate the two node types, let W assign weight 1 to original edges, $w_S > 0$ to incidences, and $w_0 > 0$ to self-loops. Define weighted degree $d_u = \sum_v W_{uv}$ and hidden state $h^{(r)}_u \in \mathbb{R}^h$ at layer r. A receiver-first signed gate and normalized message are

$$\alpha^{(r)}_{u \leftarrow v} = \tanh\left(\frac{a_r^\top [h^{(r)}_u \| h^{(r)}_v] + b_r}{\tau} \right), \qquad m^{(r)}_u = \sum_{v : W_{uv} > 0} \frac{W_{uv} \alpha^{(r)}_{u \leftarrow v}}{\sqrt{d_u d_v}} h^{(r)}_v. \tag{7}$$

Here $\|$ denotes concatenation, $a_r \in \mathbb{R}^{2h}$ and $b_r \in \mathbb{R}$ are learned parameters, and $\tau > 0$ is temperature. The gate can be signed and asymmetric.

A residual multilayer perceptron (MLP) with Gaussian error linear unit (GELU) activation processes each aggregation, and a softmax classifier produces $\hat{y}_i = f_\theta(i, G^*_{\mathcal{S}})$. Only original training nodes contribute to the objective:

$$\mathcal{L}(\theta) = -\sum_{i \in L} \sum_{c=1}^{C} \mathbf{1}[y_i = c] \log \hat{y}_{ic}. \tag{8}$$

Selection precedes Eq. (8) and is not differentiated through it. The backbone integrates the selected graph without being a separate novelty claim.

3.4 Relevance and Cost Scoring

The first score component measures association between binary feature presence b and class c. For additive smoothing $\epsilon > 0$, define the joint distribution and normalized mutual information (MI):

$$\hat{p}_k(1,c) = \frac{n_{kc} + \epsilon}{\ell + 2C\epsilon}, \qquad \hat{p}_k(0,c) = \frac{n_c - n_{kc} + \epsilon}{\ell + 2C\epsilon}, \tag{9}$$

$$r_k = \frac{1}{H(Y)} \sum_{b \in \{0,1\}, c} \hat{p}_k(b,c) \log \frac{\hat{p}_k(b,c)}{\hat{p}_k(b)\hat{p}(c)}, \qquad H(Y) = -\sum_c \hat{p}(c) \log \hat{p}(c). \tag{10}$$

Marginals are obtained by summing Eq. (9); the class marginal is independent of k. Set $r_k = 0$ if $H(Y) = 0$. Equation (10) uses presence and absence, unlike purity among feature-bearing nodes.

The second component measures that purity relative to Eq. (1):

$$P_k = \begin{cases} \dfrac{\sum_c \binom{n_{kc}}{2}}{\binom{t_k}{2}}, & t_k \geq 2, \\ p_0, & t_k < 2, \end{cases} \qquad g_k = [P_k - p_0]_+, \tag{11}$$

where $[z]_+ = \max(z, 0)$. Positive g_k rewards above-chance agreement; zero lift does not prohibit selection.

Coverage, incidence cost, and reliability are

$$c_k = \frac{\log(1 + s_k)}{\log(1 + n)}, \qquad q_k = \frac{s_k}{n}, \qquad \beta_k = \begin{cases} t_k/(t_k + n_0), & t_k + n_0 > 0, \\ 0, & t_k = n_0 = 0, \end{cases} \tag{12}$$

where $n_0 \geq 0$ is a pseudo-count. Logarithmic coverage rewards reach, linear cost penalizes incidences, and β_k shrinks weakly supported label terms.

For a statistic T_k, let $\widetilde{T}_k = (T_k - \mu_T)/(\sigma_T + \epsilon)$, where μ_T and σ_T are its population mean and standard deviation over $\mathcal{F}$. The final score is

$$S_k = \lambda_r \widetilde{\beta r}_k + \lambda_g \widetilde{\beta g}_k + \lambda_v \widetilde{c}_k - \lambda_e \widetilde{q}_k, \qquad \lambda_r, \lambda_g, \lambda_v, \lambda_e \geq 0. \tag{13}$$

Shrinkage precedes standardization, and a constant component contributes zero. Equation (13) integrates these statistics for structural selection without explicitly penalizing pairwise feature redundancy.

Top-K ranking implements Eq. (3), with ties resolved by feature index. For budget fraction $\kappa \in (0, 1]$, set $K = \max(1, \lfloor \kappa D \rfloor)$ when $D > 0$. Threshold selection, $\mathcal{S} = \{k : S_k \geq \eta\}$, provides an alternative assessed in the ablation study. Equal feature-node budgets do not imply equal incidence costs, because s_k varies across features.

3.5 Size and Path Analysis

Structural counts exclude propagation self-loops and hold independently of score quality.

Proposition 1 (Compactness and connectivity)**.** *For $\mathcal{S} \subseteq \mathcal{F}$, construction in Eq.* (5) *satisfies*

$$|V_{\mathcal{S}}^*| = n + |\mathcal{S}| \leq n + D, \qquad |E_{\mathcal{S}}^*| = m + e_{\mathcal{S}} \leq m + \mathrm{nnz}(X). \tag{14}$$

Every pair of distinct original nodes sharing a selected feature has a two-hop path through its auxiliary node.

Proof. Each retained feature creates one distinct node and s_k incidences, disjoint from the original edges and other feature incidences. Summing these counts and using $\sum_{k \in \mathcal{F}} s_k = \mathrm{nnz}(X)$ proves Eq. (14). Both incidences of a shared selected feature exist, proving the path statement. □

Let $B_k = \binom{t_k}{2}$ and $B_{\mathcal{S}} = \sum_{k \in \mathcal{S}} B_k > 0$. The same-label fraction of labeled feature-induced paths is

$$H_2(\mathcal{S}) = \frac{\sum_{k \in \mathcal{S}} \sum_c \binom{n_{kc}}{2}}{B_{\mathcal{S}}} = \sum_{k \in \mathcal{S}} \frac{B_k}{B_{\mathcal{S}}} P_k, \qquad \Delta_{\mathcal{S}} = \frac{\sum_{k \in \mathcal{S}} B_k (P_k - p_0)}{B_{\mathcal{S}}}. \tag{15}$$

Equation (15) counts a pair once per shared feature, not once per unique node pair. It is undefined without labeled pairs.

Theorem 1 (Conditional path agreement)**.** *If $P_k \geq p_0$ for every selected feature with $B_k > 0$, then $H_2(\mathcal{S}) \geq p_0$. More generally, above-chance agreement holds exactly when $\sum_{k \in \mathcal{S}} B_k(P_k - p_0) \geq 0$.*

Proof. The weights $B_k / B_{\mathcal{S}}$ in Eq. (15) are nonnegative and sum to one, so a common lower bound on P_k also bounds their weighted average. The second statement follows because the denominator defining $\Delta_{\mathcal{S}}$ is positive. □

The condition is not guaranteed by top-K selection: association or coverage can favor a feature with zero purity lift. For example, four labeled nodes with classes $(1, 1, 2, 2)$ have $p_0 = 1/3$; a sole candidate present on one node of each class must be selected at $K = 1$, yet has $H_2 = 0$. Thus neither top-K ranking nor labeled agreement guarantees improved test accuracy.

Computing supports and label statistics costs $O(\mathrm{nnz}(X) + DC)$, and sorting scores costs $O(D \log D)$. Indexed graph construction takes $O(n + m + |\mathcal{S}| + e_{\mathcal{S}})$ work. Dense auxiliary initialization can add $O(de_{\mathcal{S}})$ time and $O(d|\mathcal{S}|)$ storage. At fixed hidden width and depth, Eq. (14) bounds structural propagation work relative to exhaustive construction, while actual runtime and memory also depend on implementation and preprocessing.

4 Predictive and Structural Evaluation

The evaluation covers setup (Section 4.1), construction (Section 4.2), ranking (Section 4.3), structural properties (Section 4.4), and ablation with budget sensitivity (Section 4.5).

4.1 Evaluation Setup

The benchmarks are ACTOR, filtered Squirrel and Chameleon (Squirrel-F and Chameleon-F), Minesweeper, Cora, and CiteSeer. The citation graphs complement four heterophilic benchmarks. Table 1 gives dataset sizes and validation-selected parameters. Nonnegative attributes are binarized by $X_{ik} = \mathbf{1}[z_{ik} > 0]$; attributes containing negative values use $X_{ik} = \mathbf{1}[z_{ik} > \bar{z}_k]$, where $\bar{z}_k$ is the column mean. This preprocessing uses graph attributes, not held-out labels.

Scores are means and standard deviations across ten splits. Cora and CiteSeer use random 60/20/20 training/validation/test partitions; the remaining datasets use their benchmark split protocol. Accuracy is used except on Minesweeper, which uses area under the receiver operating characteristic curve (ROC-AUC). Scores are percentages; validation selects configurations without entering training-label statistics.

The primary comparison uses exhaustive and selective construction with the same preprocessing, split protocol, hardware, and tuning pipeline. Random, frequency, and MI selectors provide matched-node-budget comparisons for Eq. (13). Published architecture baselines provide context but are distinguished from controlled reruns, following heterophily evaluation guidance (Platonov et al., 2023b).

Table 1: Dataset sizes and validation-selected parameters. n, m, C are node, undirected-edge, and class counts; κ is the node-budget fraction. Parameters w_S, w_0, τ, n_0 follow Eqs. (7) and (12).

Dataset	n	m	C	κ	w_S	w_0	τ	n_0
ACTOR	7600	33544	5	0.75	8.00	0.30	0.01	20
Squirrel-F	2223	46998	5	0.10	0.01	0.30	0.10	20
Chameleon-F	890	8854	5	0.10	0.60	0.50	1.00	0
Minesweeper	10000	39402	2	0.75	8.00	1.00	1.00	0
Cora	2708	5429	7	0.50	0.01	8.00	0.01	5
CiteSeer	3327	4732	6	0.10	0.01	8.00	0.10	10

4.2 BUDGETED CONSTRUCTION

Table 2 links construction in Eq. (5) to prediction under Eq. (8). Selection reduces auxiliary nodes and incidences on every dataset. Auxiliary-edge reductions range from 32.1% on ACTOR to 92.5% on Squirrel-F.

Table 2: Controlled exhaustive (All) and selective construction. D/K are auxiliary-node counts and $e_{\mathcal{F}}/e_{\mathcal{S}}$ are incidence-edge counts. Scores are mean $\pm$ standard deviation.

Dataset	All score	Selective score	D/K	$e_{\mathcal{F}}/e_{\mathcal{S}}$	Edge reduction
ACTOR	37.67 ± 0.77	37.15 ± 1.51	932/699	40977/27832	32.1%
Squirrel-F	45.35 ± 1.71	43.71 ± 2.32	1593/159	32481/2451	92.5%
Chameleon-F	48.93 ± 2.58	44.72 ± 5.27	1980/198	9903/1099	88.9%
Minesweeper	87.33 ± 1.30	86.22 ± 1.50	7/5	10000/4935	50.6%
Cora	85.03 ± 1.74	85.45 ± 1.59	1432/716	49216/27622	43.9%
CiteSeer	72.79 ± 3.98	74.19 ± 2.33	3703/370	105165/8626	91.8%

Selection improves the two citation means but reduces the four heterophilic means relative to exhaustive construction. Chameleon-F shows the clearest cost of aggressive selection: removing 88.9% of incidences accompanies a 4.21-point accuracy decrease.

Table 3 places the predictor beside an MLP, graph convolutional network (GCN), graph attention network (GAT), GraphSAGE, frequency-adaptive GCN (FAGCN), generalized PageRank GNN (GPRGNN), and graph transformers (Kipf & Welling, 2017; Veličković et al., 2018; Hamilton et al., 2017; Bo et al., 2021; Chien et al., 2021; Wu et al., 2023b;a). Its Minesweeper score of 94.78 also exceeds the controlled exhaustive value of 87.33, so this cross-pipeline gap cannot be attributed entirely to selection.

Table 3: Architecture context using published baseline results. All scores are percentages; Minesweeper uses ROC-AUC. These baseline values are not matched-pipeline reruns.

Method	ACTOR	Squirrel-F	Chameleon-F	Minesweeper	Cora	CiteSeer
MLP	35.04 ± 1.53	33.91 ± 1.55	38.44 ± 5.14	50.99 ± 1.47	75.45 ± 1.88	71.53 ± 0.70
GCN	30.21 ± 0.86	35.57 ± 1.86	40.06 ± 4.38	72.32 ± 0.93	87.50 ± 1.68	75.77 ± 0.96
GAT	28.86 ± 0.99	32.74 ± 3.02	40.11 ± 2.80	87.59 ± 1.35	87.11 ± 1.48	76.43 ± 1.31
GraphSAGE	34.95 ± 1.06	34.43 ± 2.68	39.33 ± 4.53	90.54 ± 0.66	87.90 ± 1.73	76.43 ± 1.19
FAGCN	36.18 ± 1.52	36.52 ± 1.72	39.83 ± 3.93	84.69 ± 2.05	88.66 ± 2.11	76.82 ± 1.48
GPRGNN	35.42 ± 1.33	34.97 ± 2.83	40.50 ± 4.55	83.94 ± 0.98	88.86 ± 1.42	76.49 ± 1.00
SGFormer	25.89 ± 0.80	34.54 ± 2.96	42.79 ± 4.06	52.06 ± 0.50	86.24 ± 1.58	70.74 ± 1.25
DIFFormer	26.31 ± 1.19	33.17 ± 2.84	39.16 ± 4.10	69.25 ± 0.93	86.61 ± 3.04	76.65 ± 1.52
GRAPHITE	37.69 ± 1.57	43.06 ± 2.89	45.08 ± 4.04	94.78 ± 0.41	88.23 ± 1.65	76.41 ± 1.57
Selective	37.15 ± 1.51	43.71 ± 2.32	44.72 ± 5.27	86.22 ± 1.50	85.45 ± 1.59	74.19 ± 2.33

4.3 RELEVANCE-BASED SELECTION

Table 4 isolates the ranking choice at the same feature-node budget. Frequency selection emphasizes support in Eq. (12), while MI selection uses association in Eq. (10). The combined score has the highest mean on ACTOR, but random and MI selection lead on Squirrel-F, and frequency leads on Chameleon-F and Minesweeper. The evidence therefore supports graph-dependent benefits of joint scoring.

Equation (13) balances association, purity, and cost, but matching node budgets does not establish superiority at equal edge cost.

The selective row here and Table 2 represent separate evaluation series and are not pooled. Their mean differences do not support paired significance claims without split-level results and matched configurations.

Table 4: Selection at matched feature-node budgets. Scores are mean ± standard deviation; Minesweeper uses ROC-AUC.

Selection method	ACTOR	Squirrel-F	Chameleon-F	Minesweeper
Random	36.14 ± 1.47	43.83 ± 2.04	43.51 ± 4.48	84.04 ± 4.30
Frequency	36.57 ± 1.42	43.33 ± 1.66	46.08 ± 2.91	86.39 ± 1.73
MI	34.23 ± 1.10	43.83 ± 1.81	44.56 ± 4.09	84.26 ± 1.34
Selective	37.36 ± 0.89	43.59 ± 2.07	44.89 ± 5.19	86.09 ± 1.56

4.4 Compactness and Path Agreement

The counts in Table 2 agree with the subset bounds in Proposition 1. Table 5 connects those reductions to time, memory, and the path statistic in Eq. (15). Time and memory decrease on every dataset. On CiteSeer, time changes from 127.73 to 22.78 seconds and memory from 15830.8 to 2572.0 MB, equivalent to a $5.61\times$ time ratio and an 83.8% memory reduction.

Table 5: Computational measurements and feature-path diagnostics. Each All/Selective pair compares exhaustive and selected construction. Time is in seconds and memory in MB; measurement scope and diagnostic label masks require verification, as discussed in Section 4.5.

Dataset	Time: All/Selective	Memory: All/Selective	p_0	$H_2(\mathcal{F})$	$H_2(\mathcal{S})$
ACTOR	139.59/86.67	10363.8/8602.9	0.2119	0.2269	0.2789
Squirrel-F	91.84/61.38	11275.9/7102.7	0.2302	0.2023	0.3208
Chameleon-F	28.18/15.03	3061.8/1648.4	0.2137	0.2588	0.5968
Minesweeper	138.58/88.11	8100.1/7429.0	0.6800	0.6807	0.6869
Cora	75.02/43.70	7982.6/5034.5	0.1793	0.2324	0.2980
CiteSeer	127.73/22.78	15830.8/2572.0	0.1776	0.2402	0.5040

Selected path agreement exceeds both chance and exhaustive agreement on all six datasets. This is consistent with positive aggregate lift in Eq. (15), but does not prove that each selected feature satisfies the sufficient condition of Theorem 1. Chameleon-F provides a useful contrast: its selected agreement rises to 0.5968 while controlled accuracy decreases.

The computational measurements support smaller graphs without implying hardware-independent speed guarantees.

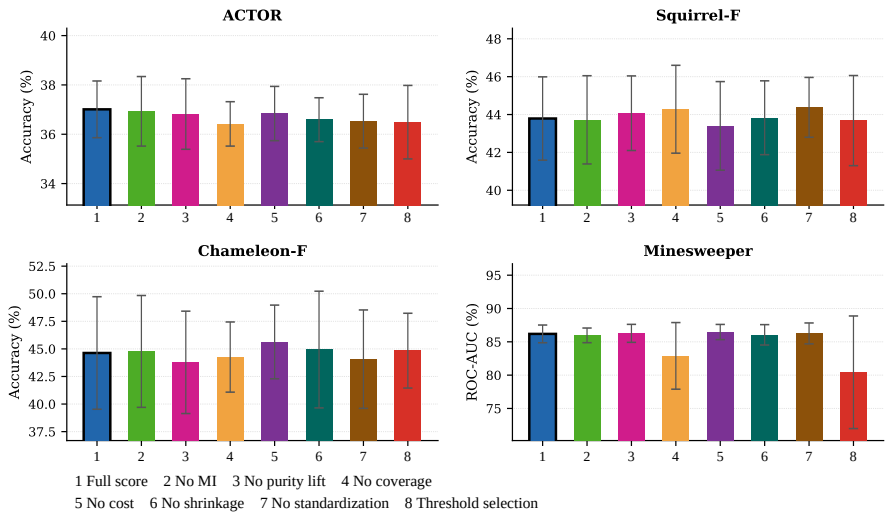


Figure 2: Score ablations on four heterophilic datasets. The numbered variants are defined below the panels. Bars retain the existing measurements; the error-bar convention is unspecified.

4.5 Ablation and Budget Sensitivity

Figure 2 removes score components and processing steps from Eqs. (10), (12), and (13). The full score is not the strongest variant on every graph. Removing coverage or standardization can improve the Squirrel-F mean, and removing cost can improve Chameleon-F. Minesweeper instead shows a marked decline without coverage or with threshold selection. These differences favor validation-based configuration rather than a universal ranking.

Figure 3 varies the feature-node budget and plots the resulting prediction and incidence count. Incidence cost increases with the retained fraction, whereas prediction can be non-monotonic. Small budgets can discard complementary features; larger budgets approach exhaustive construction.

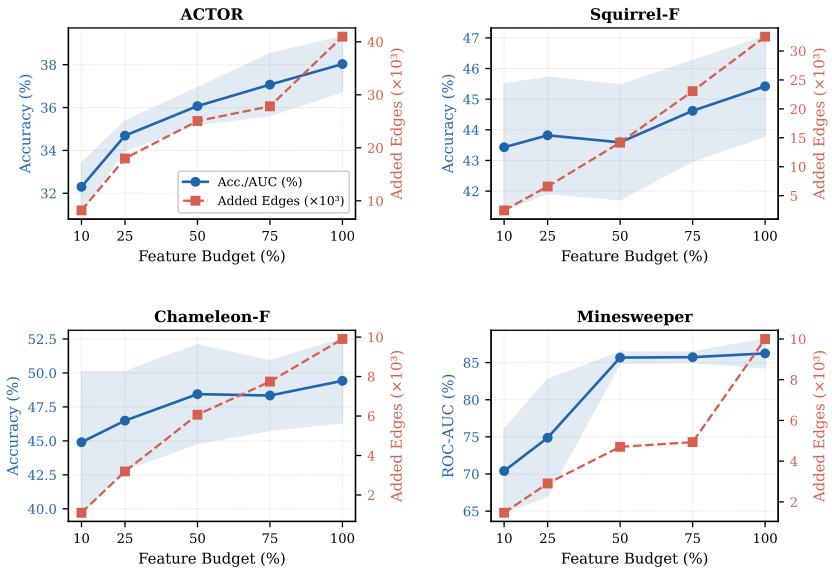


Figure 3: Sensitivity to the feature-node budget. Left axes show accuracy or ROC-AUC; right axes show added incidence edges. Shaded score bands denote one standard deviation. The node budget controls K, while edge counts measure its structural consequence.

Reproducibility. Equations (5) to (13), the split protocol, and Table 1 specify the available method and configuration details. Exact replication remains limited by unavailable code, seeds, complete tuning grids, optimizer settings, and hardware metadata. The runtime interval, memory statistic, diagnostic label mask, and ablation error-bar definition are also unspecified. ACTOR lists 931 input dimensions but 932 exhaustive feature nodes; this discrepancy requires checking against preprocessing.

Scope and future work. Marginal scores may retain redundant features or discard complementary ones. Further evaluation can examine label-budget sensitivity, selection stability, equal-edge-budget controls, and direct comparisons with GFS and JDR. Continuous-feature selection and redundancy penalties could extend the current discrete transductive setting.

5 Conclusion

This work develops selective feature-node transformation through three connected contributions: budgeted auxiliary construction that preserves original information, support-adjusted relevance and cost scoring, and selection-specific size and path analysis. The framework controls added structure and identifies a sufficient condition for above-chance labeled path agreement, separately from predictive generalization.

Evaluation on six benchmarks shows substantial graph reduction and lower computational measurements, with dataset-dependent predictive effects. Matched-budget selectors and ablations identify where the combined score helps and where simpler choices remain competitive.

AI USE STATEMENT

Generative AI assisted with manuscript organization, language editing, reference formatting, equation and proof checks. This assistance did not generate new experimental measurements or execute model-training experiments.

REFERENCES

Adrián Arnaiz-Rodríguez and Federico Errica. Oversmoothing, "oversquashing", heterophily, long-range, and more: Demystifying common beliefs in graph machine learning. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/6697bb267dc517379bc8aa326e844f8d-Abstract-Conference.html.

Federico Barbero, Ameya Velingker, Amin Saberi, Michael M. Bronstein, and Francesco Di Giovanni. Locality-aware graph-rewiring in GNNs. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.01668.

Deyu Bo, Xiao Wang, Chuan Shi, and Huawei Shen. Beyond low-frequency information in graph convolutional networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 3950–3957, 2021. URL https://ojs.aaai.org/index.php/AAAI/article/view/16514.

Eli Chien, Jianhao Peng, Pan Li, and Olgica Milenkovic. Adaptive universal generalized PageRank graph neural network. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=n6jl7fLxrP.

Yoonhyuk Choi, Taewook Ko, Jiho Choi, and Chong-Kwon Kim. Selective blocking for message-passing neural networks on heterophilic graphs. In *Proceedings of the Forty-first Conference on Uncertainty in Artificial Intelligence*, volume 286 of *Proceedings of Machine Learning Research*, pp. 737–751, 2025. URL https://proceedings.mlr.press/v286/choi25a.html.

William L. Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*, volume 30, pp. 1024–1034, 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/hash/5dd9db5e033da9c6fb5ba837a7ebea9-Abstract.html.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=SJU4ayYgl.

Jonas Linkerhägner, Cheng Shi, and Ivan Dokmanić. Joint graph rewiring and feature denoising via spectral resonance. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=zBbZ2vdLzH.

Oleg Platonov, Denis Kuznedelev, Artem Babenko, and Liudmila Prokhorenkova. Characterizing graph datasets for node classification: Homophily-heterophily dichotomy and beyond. In *Advances in Neural Information Processing Systems*, volume 36, 2023a. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/01b681025fdbda8e935a66cc5bb6e9de-Abstract-Conference.html.

Oleg Platonov, Denis Kuznedelev, Michael Diskin, Artem Babenko, and Liudmila Prokhorenkova. A critical look at the evaluation of GNNs under heterophily: Are we really making progress? In *International Conference on Learning Representations*, 2023b. URL https://openreview.net/forum?id=tJbbQfw-5wv.

Ruizhong Qiu, Ting-Wei Li, Gaotang Li, and Hanghang Tong. Graph homophily booster: Reimagining the role of discrete features in heterophilic graph learning. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2602.07256.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=rJXMpikCZ.

Qitian Wu, Chenxiao Yang, Wentao Zhao, Yixuan He, David Wipf, and Junchi Yan. DIFFormer: Scalable (graph) transformers induced by energy constrained diffusion. In *International Conference on Learning Representations*, 2023a. URL https://openreview.net/forum?id=j6zUzrapY3L.

Qitian Wu, Wentao Zhao, Chenxiao Yang, Hengrui Zhang, Fan Nie, Haitian Jiang, Yatao Bian, and Junchi Yan. SGFormer: Simplifying and empowering transformers for large-graph representations. In *Advances in Neural Information Processing Systems*, volume 36, 2023b. URL https://openreview.net/forum?id=R4xpvDTWkV.

Xin Zheng, Yi Wang, Yixin Liu, Ming Li, Miao Zhang, Di Jin, Philip S. Yu, and Shirui Pan. Graph neural networks for graphs with heterophily: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 2026. doi: 10.1109/TKDE.2026.3680353. URL https://arxiv.org/abs/2202.07082.

Yilun Zheng, Xiang Li, Sitao Luan, Xiaojiang Peng, and Lihui Chen. Let your features tell the differences: Understanding graph convolution by feature splitting. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/993760d9b1d63a385611c7c9ef8f3ecb-Abstract-Conference.html.