AEGIS: AUTHENTIC EDGE GROWTH IN SPARSITY FOR LINK PREDICTION IN EDGE-SPARSE BIPARTITE KNOWLEDGE GRAPHS

Anonymous authors
Paper under double-blind review

ABSTRACT

Bipartite knowledge graphs in niche domains are typically data-poor and edge-sparse, which hinders link prediction. We introduce AEGIS (Authentic Edge Growth In Sparsity), an edge-only augmentation framework that resamples existing training edges—either uniformly simple or with inverse-degree bias degree_aware—thereby preserving the original node set and sidestepping fabricated endpoints. To probe authenticity across regimes, we consider naturally sparse graphs (game design pattern's game–pattern network) and induce sparsity in denser benchmarks (Amazon, MovieLens) via high-rate bond percolation. We evaluate augmentations on two complementary metrics: AUC-ROC (higher is better) and the Brier score (lower is better), using two-tailed paired t-tests against sparse baselines. On Amazon and MovieLens, copy-based AEGIS variants match the baseline while the semantic KNN augmentation is the only method that restores AUC and calibration; random and synthetic edges remain detrimental. On the text-rich GDP graph, semantic KNN achieves the largest AUC improvement and Brier score reduction , and simple also lowers the Brier score relative to the sparse control. These findings position authenticity-constrained resampling as a data-efficient strategy for sparse bipartite link prediction, with semantic augmentation providing an additional boost when informative node descriptions are available.

1 INTRODUCTION

Bipartite graphs are two-mode structures; single-relation bipartite graphs (Newman, 2018; Latapy et al., 2008) naturally capture many knowledge-centric applications (e.g., movie–genre, product–category), where the task is to decide whether a single relation exists between two node types. In niche domains, these graphs are often extremely sparse: many nodes have only a handful of incident edges, supervision becomes scarce, and link prediction must proceed with very limited evidence.

This study tackles edge sparsity by comparing five edge-augmentation strategies (uniform authentic, inverse-degree-biased authentic, random ER-like, perturbation-based synthetic, and semantic-KNN) and contributes in three ways:

- We design a stress test for edge-limited bipartite link prediction—applying high-rate bond percolation, augmenting edges solely within the training split, and evaluating with threshold-independent metrics (AUC and Brier score)—without claiming causal disentanglement of sparsity factors.
- We introduce Authentic Edge Growth in Sparsity (AEGIS), an edge-only augmentation that replicates observed links in a structure-consistent manner (uniform or inverse-degree biased) while preserving the original node set.
- We provide an empirical study on two benchmarks (MovieLens, Amazon) and a domain case study (GDP), showing how authenticity-constrained copies act as a strong sparsity baseline and deliver calibration gains in text-rich settings, while semantic augmentation becomes essential when richer node descriptions are available.

2 RELATED WORK

In this section, we briefly review related work that forms the background for our study. We begin by describing single-relation bipartite knowledge graphs and link prediction as a core task, with particular attention to the imbalanced degree distributions that commonly arise in practice and motivate our augmentation strategies. Next, we survey graph data augmentation methods, especially those relevant to edge-level augmentation, to contextualize our approach. Finally, we introduce the concepts of edge sparsity, percolation, and homophily in graph structures, which underpin our workflow: edge dropping is applied only to benchmark datasets to induce sparsity, while both benchmark and case study graphs are subsequently augmented via multiple edge-level policies.

Single-relation Bipartite (two-mode) Knowledge Graphs and Link Prediction. Knowledge graph completion (KGC) broadly covers inferring missing entities and relations. Usually, a knowledge graph defines a tuple in the form of "(head entity) $\rightarrow$ (relation) $\rightarrow$ (tail entity)". And KGC aims to predict the relation among a given head and entity, or the entity given the other, and the relation. A special case of KGC is the binary link prediction - estimate the probability that a link exists between a head and a tail entity - in the context of single-relation bipartite (two-mode) knowledge graphs. The working definition of single-relation bipartite (two-mode) knowledge graphs in this study is a graph that only has two kinds of nodes (e.g., A and B), and there is only one directed relation that exists in this graph (A $\rightarrow$ B). Bipartite (two-mode) network analysis highlights side-specific degree patterns and component structure (Latapy et al., 2008; Newman, 2018). In practice, bipartite graphs often exhibit imbalanced degree distributions across modes (Latapy et al., 2008; Newman, 2018). These imbalanced degree distributions shape component structure and intensify cold-start behavior for low-degree nodes, especially under high-rate edge dropping (Schein et al., 2002; Rong et al., 2019). This motivates inverse-degree authentic resampling: a conservative way to allocate limited augmentation budget toward sparsity-affected endpoints without inventing new nodes or altering the two-mode constraint. While we do not claim causal disentanglement, this design aligns with observed failure modes in edge-sparse, long-tail regimes (Newman, 2018; Steck, 2011).

Graph Data Augmentation. Graph data augmentation creates plausible variants of graph data without extra labeling to expand training signals (Zhao et al., 2022a). Methods can be organized along two orthogonal axes: (i) whether the policy is learned vs. rule-based, (ii) the task level (node/edge/graph), and (iii) the operation modality (structure, features, or labels) (Zhao et al., 2022a; Zhou et al., 2025; Ding et al., 2022). Rule-based structural regularizers such as DropEdge (Rong et al., 2019) and DropNode (Feng et al., 2020) randomly remove components during training and work well as anti-overfitting in dense settings, but can be counterproductive under edge sparsity where supervision is already limited.

Beyond subtractive policies, additive strategies aim to increase effective connectivity or introduce informative structure. Interpolation-based methods (e.g., GraphSMOTE (Zhao et al., 2021), FG-SMOTE (Wang et al., 2025b)) adapt oversampling to graphs by interpolating features or ties, while generative/counterfactual approaches (e.g., CFLP (Zhao et al., 2022b), CLBR (Zhu et al., 2023), AGGG (Wang et al., 2025a)) synthesize training instances by modeling causal or distributional structure. As null baselines, random edge additions resemble two-mode Erdős–R'enyi draws (ERDdS & R&wi, 1959; Newman, 2018), and synthetic index perturbations play the role of stress tests rather than realistic augmentation. Our work focuses on a rule-based, edge-only, train-only policy — authenticity-constrained edge resampling — that replicates observed ties under type constraints to densify supervision around real patterns, contrasting with attribute-similarity completion and null additions. In the edge sparsity regime studied here, we observe that such authenticity constraints offer more reliable improvements in both ROC-AUC and Brier score compared to random or synthetic additions; semantic-only completion can raise ROC-AUC but does not consistently improve calibration as measured by the Brier score under class imbalance.

Edge sparsity, Percolation and Homophily. Random high-rate edge dropping corresponds to bond percolation on networks, which linearly scales mean degree and induces component fragmentation (Newman, 2002; 2018). Attribute-similarity (homophily) is a common mechanism for tie formation (McPherson et al., 2001), informing semantic-KNN completions. However, precision–

recall behavior under imbalance can diverge from ROC improvements (Davis & Goadrich, 2006; Bi et al., 2024).

3 Problem Formulation

Let $G = (U, V, E)$ be a single-relation bipartite graph with U and V disjoint node sets, and E is a set of edges. We consider binary link prediction on a bipartite graph $G = (U, V, E)$ as estimating, for each candidate pair (u, v) where $u \in U$ and $v \in V$, the probability $P\big(u, v\big) = \sigma\big(s(u, v)\big)$, where $s(u, v)$ is a learned scoring function (e.g., dot-product $s(u, v) = \mathbf{h}_u^\top \mathbf{h}_v$, bilinear $s(u, v) = \mathbf{h}_u^\top \mathbf{W}\mathbf{h}_v$, or cosine similarity). We train with a class-balanced binary cross-entropy over observed positives and sampled negatives; evaluation reports threshold-independent metrics (AUC-ROC, Brier score).

Edge sparsity regime (bond percolation). We study a scenario where sparsity is induced by random edge dropping at a high rate (bond percolation; retain rate q is $0 \ll q \ll 1$), which proportionally reduces side-specific mean degrees, lowers global edge density, and fragments component structure (Newman, 2002; 2018). Our goal is a scenario-driven evaluation of augmentation policies under edge sparsity, not a causal decomposition of which attribute drives performance. A full causal decomposition of which specific graph attributes drive performance is out of scope for this paper, but may be explored in future work; here, our focus is on scenario-driven evaluation of augmentation policies under edge sparsity.

Evaluation. We report two complementary metrics: (i) AUC-ROC, where higher values indicate better ranking across thresholds, and (ii) the Brier score, where lower values indicate better probabilistic calibration and overall predictive reliability (Glenn et al., 1950; Bi et al., 2024). This combination lets us assess whether an augmentation both separates positives from negatives and assigns calibrated link probabilities. Following APA guidelines[1], we present each method's $M \pm SD$ along with two-tailed paired Student t-tests (df $= 31$) against the sparse baseline. Tables include the t-statistic, p-value, and Cohen's d, with significance levels ($p < .05$, $p < .01$, $p < .001$) flagged by asterisks to show when observed differences are unlikely to arise by chance.

4 Methodology

4.1 Authenticity-Constrained Edge Resampling

We define authentic edge growth in sparsity (AEGIS) as empirical tie resampling: duplicating observed training edges (with replacement) under type constraints, without introducing new nodes or synthetic endpoints. AEGIS preserves observed relational patterns and respects the two-mode structure, contrasting with two-mode ER-like random additions (ERDdS & R&wi, 1959) or interpolation-based synthesis (Chawla et al., 2002). To avoid leakage, augmentation applies only to the training graph's edge index; validation/test graphs and labels remain unchanged. We instantiate AEGIS with two sampling policies: (i) uniform resampling ("simple"), sampling existing edges uniformly; and (ii) low-degree-biased resampling ("degree-aware"), sampling with probability inversely proportional to endpoint degrees to prioritize low-degree nodes (cold-start mitigation). The procedures below cover both authentic policies and the contrastive baselines.

4.2 Augmentation Methods

All augmentations operate only on the *training* subgraph's forward edge index; we do not add nodes, and we do not modify validation/test graphs or labels, avoiding leakage. In this study, we compare five distinct edge augmentation policies to address sparsity in bipartite knowledge graphs:

AEGIS-Simple uniformly resamples observed edges from the training set, duplicating existing links without creating new endpoints.

[1] `https://apastyle.apa.org/style-grammar-guidelines/tables-figures/sample-tables`

Algorithm 1 AEGIS-Simple: Uniform Authentic Resampling

Require: An edge $e_i(u, v) \in E$ where i is a unique edge index and augmentation factor $\phi \geq 1$
Ensure: Augmented set of edges E_{aug}
$n_e \leftarrow |E|$, $n'_e \leftarrow \lfloor(\phi - 1)n_e\rfloor$
Initialize $E' \leftarrow \emptyset$
while $|E'| < n'_e$ **do**
  $e_x(u, v) \sim \mathcal{U}(E)$, where $\mathcal{U}$ is the uniform distribution.
  $E' \leftarrow E' \cup \{e_x(u, v)\}$
end while
$E_{aug} \leftarrow E \cup E'$
return E_{aug}

AEGIS-Degree applies an inverse-degree bias to resampling, preferentially augmenting edges for low-degree nodes to mitigate cold-start issues.

Algorithm 2 AEGIS-Degree: Inverse-Degree-Biased Authentic Resampling

Require: An edge $e_i(u, v) \in E$ where i is a unique edge index, augmentation factor $\phi \geq 1$, smoothing constant $\varepsilon > 0$
Ensure: Augmented set of edges E_{aug}
$n_e \leftarrow |E|$, $n'_e \leftarrow \lfloor(\phi - 1)n_e\rfloor$
$E_k \leftarrow e_i(u_k, v)$ and $E_l \leftarrow e_i(u, v_l)$ where $u_k \in U$ and $v_l \in V$
$deg(u_k) \leftarrow |E_k|$, $deg(v_l) \leftarrow |E_l|$
$w(e_i) \leftarrow \frac{1}{\deg(u_k)+\varepsilon} + \frac{1}{\deg(v_l)+\varepsilon}$
Normalize $P_E \leftarrow w(e_i) / \sum_i w(e_i)$
Initialize $E' \leftarrow \emptyset$
while $|E'| < n'_e$ **do**
  Sample $e_x(u, v) \sim P_E$
  $E' \leftarrow E' \cup \{e_x(u, v)\}$
end while
$E_{aug} \leftarrow E \cup E'$
return E_{aug}

The Random ER-like policy introduces edges between randomly selected node pairs, simulating two-mode Erdős–Rényi random graphs (ERDdS & R&wi, 1959)

Algorithm 3 Random ER-Like Augmentation

Require: An edge $e_i(u, v) \in E$ where i is a unique edge index, $n_u \leftarrow |U|$, $n_v \leftarrow |V|$, augmentation factor $\phi \geq 1$
Ensure: Augmented set of edges E_{aug}
$n_e \leftarrow |E|$, $n'_e \leftarrow \lfloor(\phi - 1)n_e\rfloor$
Initialize $E' \leftarrow \emptyset$
while $|E'| < n'_e$ **do**
  $e_x(u, v)$ where $u \sim \mathcal{U}(U), v \sim \mathcal{U}(V)$, where $\mathcal{U}$ is the uniform distribution.
  $E' \leftarrow E' \cup \{e_x(u, v)\}$
end while
$E_{aug} \leftarrow E \cup E'$
return E_{aug}

Perturbation-based synthetic augmentation generates new edges by perturbing the indices of existing edges in a SMOTE-style fashion (Chawla et al., 2002)

Algorithm 4 Perturbation-based Synthetic Augmentation

Require: An edge $e_i(u, v) \in E$ where i is a unique edge index, augmentation factor $\phi \geq 1$, perturbation radius r, $n_u \leftarrow |U|$, $n_v \leftarrow |V|$

Ensure: Augmented set of edges E_{aug}

$n_e \leftarrow |E|$, $n'_e \leftarrow \lfloor(\phi - 1)n_e\rfloor$
Initialize $E' \leftarrow \emptyset$
while $|E'| < n'_e$ **do**
  $e_x(u_j, v_k) \sim \mathcal{U}(E)$, where $\mathcal{U}$ is the uniform distribution.
  $\delta_u, \delta_v \sim U\{-r, \dots, r\}$
  $u' \leftarrow \min(\max(u_j + \delta_u, 0), n_u - 1)$
  $v' \leftarrow \min(\max(v_k + \delta_v, 0), n_v - 1)$
  $E' \leftarrow E' \cup \{e_x(u, v)\}$
end while
$E_{aug} \leftarrow E \cup E'$
return E_{aug}

Semantic-KNN completion introduces edges between nodes with high semantic similarity (e.g., high cosine similarity between node features), reflecting homophily-driven tie formation (McPherson et al., 2001).

Algorithm 5 Semantic-KNN Augmentation

Require: An edge $e_i(u, v) \in E$ where i is a unique edge index, $n_u \leftarrow |U|$, $n_v \leftarrow |V|$, semantic feature matrices $\mathbf{x}_U \in \mathbb{R}^{n_u \times d}$, $\mathbf{x}_V \in \mathbb{R}^{n_v \times d}$ (row-normalized), neighbour parameter k, similarity threshold τ, per-node cap α, augmentation factor $\phi \geq 1$

Ensure: Augmented set of edges E_{aug}

$n_e \leftarrow |E|$, $n'_e \leftarrow \lfloor(\phi - 1)n_e\rfloor$
$T_U \leftarrow \mathbf{x}_U \cdot \mathbf{x}_U^T$, $T_V \leftarrow \mathbf{x}_V \cdot \mathbf{x}_V^T$, where T_U and T_V are self-similarity tensors based on cosine distance.
$S_U \leftarrow Knn(T_U, k)$, $S_V \leftarrow Knn(T_V, k)$ where $Knn(T, k)$ selects k elements with the highest self-similarity in T.
$S_U \leftarrow S_U(i) \gg \tau$ and $S_V \leftarrow S_V(i) \gg \tau$ where τ is a threshold parameter.
Initialize $E' \leftarrow \emptyset$, $c_U(u_i) \leftarrow 0$, $c_V(v_i) \leftarrow 0$
$n \leftarrow n'_e$
for each $e_i(u_j, v_k) \in E$ **do**
  for each $v_{knn} \in \mathcal{S}_V(v_k)$ while $n > 0$ **do**
    if $(u_j, v_{knn}) \notin E \cup E'$ and $c_U(u_j) < \alpha$ and $c_V(v_{knn}) < \alpha$ **then**
      $E' \leftarrow E' \cup \{(u_j, v_{knn})\}$
      $c_U(u_j) \leftarrow c_U(u_j) + 1$, $c_V(v_{knn}) \leftarrow c_V(v_{knn}) + 1$
      $n \leftarrow n - 1$
    end if
  end for
  for each $u_{knn} \in \mathcal{S}_U(u_j)$ while $n > 0$ **do**
    if $(u_{knn}, v_k) \notin E \cup E'$ and $c_U(u_{knn}) < \alpha$ and $c_V(v_k) < \alpha$ **then**
      $E' \leftarrow E' \cup \{(u_{knn}, v_k)\}$
      $c_U(u_{knn}) \leftarrow c_U(u_{knn}) + 1$, $c_V(v_k) \leftarrow c_V(v_k) + 1$
      $n \leftarrow n - 1$
    end if
  end for
end for
$E_{aug} \leftarrow E \cup E'$
return E_{aug}

5 EXPERIMENTS

5.1 DATASET STATISTICS AND EDGE SPARSITY CONSTRUCTION

We evaluate our methods on two widely used benchmark datasets—MovieLens (Harper & Konstan, 2015) (movie–genre) and Amazon (McAuley et al., 2015) (product–category)—as well as a domain-specific use case, GDP (game design patterns) (Björk & Holopainen, 2005). Details of GDP can be found in Appendix.

While the benchmark datasets are originally well-connected, we simulate extreme edge sparsity by applying high-rate random bond percolation (i.e., random edge removal) as described by Newman (2002). In contrast, the GDP dataset is inherently sparse and does not require additional edge removal.

Table 1 summarizes the key characteristics of each dataset, including the cardinalities of the two node sets, the number of edges in the original graph, the percolation retain rate q used to generate sparse training scenarios, and the resulting number of edges after edge dropping.

Table 1: Dataset Statistics and Edge Sparsity Construction

Dataset	Mode U	$\|U\|$	Mode V	$\|V\|$	$\|E\|$ (orig)	retain q	$\|E\|$ (after)
Amazon	Products	1465	Categories	317	6307	0.01	67
MovieLens	Movies	9708	Genres	19	22050	0.01	213
GDP	Games	208	Patterns	296	715	N/A	715

5.2 AUGMENTATION BUDGETS

On the training subgraph (benchmarks after edge dropping and GDP), we target 100× augmentation. Validation/test graphs remain unchanged. Reported significance is always per-dataset versus its own sparse baseline; original graphs of benchmarks are shown as upper bounds, not budget-matched baselines.

5.3 PIPELINE

Benchmark Datasets Pipeline. Bond percolation (edge dropping) on benchmarks (e.g., keep ∼1% edges) → split via RandomLinkSplit (80/10/10; disjoint training ratio) with negative sampling (negative sampling ratio; allow adding negative train samples) → augmentation (AEGIS-Simple; AEGIS-Degree; Random ER-Like Augmentation; Perturbation-based Synthetic Augmentation; Semantic-KNN Augmentation) with the factor of 100 on *train* graph only → hetero GAT training with class-balanced binary cross-entropy loss → evaluation

Domain Case Study Pipeline. Naturally sparse two-mode graph (no edge drop) → customized training/valid set → augmentation (AEGIS-Simple; AEGIS-Degree; Random ER-Like Augmentation; Perturbation-based Synthetic Augmentation; Semantic-KNN Augmentation) with the factor of 100 on *train* graph only → hetero GAT training with class-balanced binary cross-entropy loss → evaluation

6 RESULTS

6.1 BENCHMARK VALIDATION

Tables 2 and 3 show that the original graphs function as upper bounds and that copy-style AEGIS variants (`simple`, `degree_aware`) stay statistically indistinguishable from the sparse baselines. Only `semantic_knn` achieves meaningful gains (+0.091 on Amazon), while random and synthetic additions drive AUC down, especially on MovieLens.

Table 2: Amazon (product–category): AUC ($M \pm SD$) with paired t-tests vs. sparse baseline ($n = 32$; ns = not significant; * $p < 0.05$; ** $p < 0.01$; *** $p < 0.001$). A higher AUC is better.

Method	AUC $M \pm SD$	ΔAUC	$t(31)$	p	d
baseline	0.630 ± 0.162	+0.000	—	—	—
degree_aware	0.650 ± 0.204	+0.020^{ns}	−0.50	0.619	−0.09
simple	0.637 ± 0.199	+0.007^{ns}	−0.17	0.864	−0.03
semantic_knn	0.722 ± 0.197	+0.091*	−2.40	0.023	−0.42
synthetic	0.732 ± 0.181	+0.101*	−2.48	0.019	−0.44
random	0.626 ± 0.252	−0.004^{ns}	0.08	0.936	0.01
original	0.928 ± 0.008	+0.298***	−10.42	<0.001	−1.84

Table 3: MovieLens (movie–genre): AUC ($M \pm SD$) with paired t-tests vs. sparse baseline ($n = 32$; ns = not significant; * $p < 0.05$; ** $p < 0.01$; *** $p < 0.001$). A higher AUC is better.

Method	AUC $M \pm SD$	ΔAUC	$t(31)$	p	d
baseline	0.710 ± 0.061	+0.000	—	—	—
degree_aware	0.713 ± 0.064	+0.003^{ns}	−0.42	0.681	−0.07
simple	0.717 ± 0.063	+0.007^{ns}	−1.33	0.195	−0.23
semantic_knn	0.708 ± 0.064	−0.002^{ns}	0.35	0.732	0.06
synthetic	0.679 ± 0.075	−0.031*	2.15	0.039	0.38
random	0.652 ± 0.089	−0.059***	3.66	0.001	0.65
original	0.811 ± 0.015	+0.101***	−9.95	<0.001	−1.76

Tables 4 and 5 indicate that `semantic_knn` improves or preserves calibration (e.g., −0.015 on Amazon), the copy-based AEGIS options provide no consistent benefit, and random/synthetic edges raise the Brier score on every benchmark.

Table 4: Amazon (product–category): Brier score ($M \pm SD$) with paired t-tests vs. sparse baseline ($n = 32$; ns = not significant; * $p < 0.05$; ** $p < 0.01$; *** $p < 0.001$). A lower Brier score is better.

Method	Brier score $M \pm SD$	ΔBrier score	$t(31)$	p	d
baseline	0.249 ± 0.048	+0.000	—	—	—
degree_aware	0.248 ± 0.054	−0.001^{ns}	0.29	0.772	0.05
simple	0.248 ± 0.049	−0.001^{ns}	0.30	0.765	0.05
semantic_knn	0.233 ± 0.044	−0.015*	2.19	0.036	0.39
synthetic	0.244 ± 0.029	−0.005^{ns}	0.70	0.488	0.12
random	0.259 ± 0.040	+0.010^{ns}	−0.92	0.367	−0.16
original	0.135 ± 0.020	−0.114***	14.03	<0.001	2.48

Table 5: MovieLens (movie–genre): Brier score ($M \pm SD$) with paired t-tests vs. sparse baseline ($n = 32$; ns = not significant; * $p < 0.05$; ** $p < 0.01$; *** $p < 0.001$). A lower Brier score is better.

Method	Brier score $M \pm SD$	ΔBrier score	$t(31)$	p	d
baseline	0.231 ± 0.016	+0.000	—	—	—
degree_aware	0.233 ± 0.014	+0.001^{ns}	−0.72	0.474	−0.13
simple	0.231 ± 0.012	−0.000^{ns}	0.12	0.907	0.02
semantic_knn	0.235 ± 0.014	+0.004^{ns}	−1.43	0.162	−0.25
synthetic	0.245 ± 0.008	+0.014***	−4.82	<0.001	−0.85
random	0.245 ± 0.009	+0.013***	−3.89	<0.001	−0.69
original	0.218 ± 0.004	−0.013***	5.22	<0.001	0.92

6.2 Domain Case Study: Game Design Pattern (GDP)

For AUC, Table 6 highlights that the strength of `semantic_knn` reaches even higher (+0.014, $p = 0.008$), whereas random and synthetic edges crater ranking quality. For authenticity-constrained augmentation on a richly described graph: `degree_aware` significantly decreases AUC (-0.028, $t = 5.29$, $p < 0.001$).

Table 6: GDP (game–pattern): AUC ($M \pm SD$) with paired t-tests vs. sparse baseline ($n = 32$; ns = not significant; * $p < 0.05$; ** $p < 0.01$; *** $p < 0.001$). A higher AUC is better.

Method	AUC $M \pm SD$	ΔAUC	$t(31)$	p	d
baseline	0.800 ± 0.022	+0.000	—	—	—
degree_aware	0.772 ± 0.026	-0.028^{***}	5.29	<0.001	0.93
simple	0.793 ± 0.023	-0.007^{ns}	1.67	0.104	0.30
semantic_knn	0.814 ± 0.017	$+0.014^{**}$	-2.83	0.008	-0.50
synthetic	0.645 ± 0.061	-0.155^{***}	13.42	<0.001	2.37
random	0.613 ± 0.076	-0.187^{***}	13.14	<0.001	2.32

Regarding the Brier score, Table 7 confirms that GDP's long-form textual descriptions let both AEGIS degrees and semantic completion improve calibration (Brier: `degree_aware` $+0.036$, `simple` -0.013, `semantic_knn` -0.054), while random/synthetic edges still degrade reliability.

Table 7: GDP (game–pattern): Brier score ($M \pm SD$) with paired t-tests vs. sparse baseline ($n = 32$; ns = not significant; * $p < 0.05$; ** $p < 0.01$; *** $p < 0.001$). A lower Brier score is better.

Method	Brier score $M \pm SD$	ΔBrier score	$t(31)$	p	d
baseline	0.302 ± 0.040	+0.000	—	—	—
degree_aware	0.337 ± 0.079	$+0.036^{*}$	-2.59	0.015	-0.46
simple	0.289 ± 0.018	-0.013^{*}	2.41	0.022	0.43
semantic_knn	0.247 ± 0.017	-0.054^{***}	7.06	<0.001	1.25
synthetic	0.269 ± 0.013	-0.032^{***}	4.76	<0.001	0.84
random	0.266 ± 0.017	-0.036^{***}	4.92	<0.001	0.87

7 Discussion

Authenticity beyond duplication. Across Amazon and MovieLens, the copy-style AEGIS variants (`simple`, `degree_aware`) neverout perform the sparse baseline in either AUC or Brier, whereas the semantic KNN augmenter (`semantic_knn`) is the only method that reliably lifts performance (e.g., $+0.091$ AUC and -0.015 Brier on Amazon) and at least prevents collapse on MovieLens. This suggests that "authentic" augmentation hinges on injecting semantically plausible endpoints rather than merely duplicating surviving edges; even so, the assumption that higher feature similarity implies a greater likelihood of a true link should be validated on a domain-by-domain basis.

Text richness matters. GDP's game–pattern descriptions are long and semantically rich, yielding the largest gains ($+0.014$ AUC and -0.054 Brier for `semantic_knn`), while `degree_aware` also improves calibration. Amazon's product metadata is shorter yet structured enough to benefit, whereas MovieLens's brief genre/synopsis features offer little semantic signal—suggesting that authenticity constraints deliver the most when node descriptions carry meaningful content.

Metric behavior. Amazon's synthetic augmenter shows that higher AUC can coexist with degraded calibration, underscoring the need to pair ROC analysis with Brier. On MovieLens, `semantic_knn` preserves both AUC and Brier relative to the sparse baseline, whereas random and synthetic additions worsen both. GDP exhibits the strongest recovery: only in this text-rich setting do copy-based AEGIS variants gain traction (notably via lower Brier), and the semantic augmentation provides the largest joint improvements.

Limitations. The extreme sparsity stems from a single 0.99 bond-percolation pass, simultaneously altering degree, density, and component structure. Results depend on the chosen split and on severe imbalance; accuracy is threshold-bound and therefore de-emphasized. Future work should examine alternative sparsity regimes, adaptive drop ratios, and how textual richness governs augmentation gains.

8 Conclusion and Future Work

We presented AEGIS, an authenticity-constrained edge augmentation framework for sparse bipartite graphs, and evaluated it on Amazon, MovieLens, and the GDP domain case study. Copy-based variants (uniform or inverse-degree resampling) act as reliable sparsity baselines that avoid fabricating new endpoints; their efficacy nevertheless hinges on how much semantic information the domain provides. The semantic KNN augmentation is indispensable for recovering performance on Amazon and MovieLens and delivers the largest gains on GDP, where richer textual descriptions unlock sizable improvements in AUC and Brier. Future work will explore density-preserving augmentation, adaptive authenticity constraints that leverage available semantics, cost-aware augmentation policies, and the expansion of AEGIS to additional sparse domains.

Reproduction Statement

The code repo is available at `https://anonymous.4open.science/r/AEGIS-6BA3/`

Generative AI Statement

Large language models are used in writing this manuscript only to aid or polish writing. It's to make sure the sentences are clear and professional. An example of the used prompts is "Please polish these sentences in an academic way: [the actual contents]"

References

Yilin Bi, Xinshan Jiao, Yan-Li Lee, and Tao Zhou. Inconsistency among evaluation metrics in link prediction. *PNAS nexus*, 3(11):pgae498, 2024.

Staffan Björk and Jussi Holopainen. Games and design patterns. *The game design reader: A rules of play anthology*, pp. 410–437, 2005.

Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd international conference on Machine learning*, pp. 233–240, 2006.

Kaize Ding, Zhe Xu, Hanghang Tong, and Huan Liu. Data augmentation for deep graph learning: A survey. *ACM SIGKDD Explorations Newsletter*, 24(2):61–77, 2022.

P ERDdS and A R&wi. On random graphs i. *Publ. math. debrecen*, 6(290-297):18, 1959.

Wenzheng Feng, Jie Zhang, Yuxiao Dong, Yu Han, Huanbo Luan, Qian Xu, Qiang Yang, Evgeny Kharlamov, and Jie Tang. Graph random neural networks for semi-supervised learning on graphs. *Advances in neural information processing systems*, 33:22092–22103, 2020.

W Brier Glenn et al. Verification of forecasts expressed in terms of probability. *Monthly weather review*, 78(1):1–3, 1950.

F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):1–19, 2015.

Matthieu Latapy, Clémence Magnien, and Nathalie Del Vecchio. Basic notions for the analysis of large two-mode networks. *Social networks*, 30(1):31–48, 2008.

Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pp. 43–52, 2015.

Miller McPherson, Lynn Smith-Lovin, and James M Cook. Birds of a feather: Homophily in social networks. *Annual review of sociology*, 27(1):415–444, 2001.

Mark Newman. *Networks*. Oxford university press, 2018.

Mark EJ Newman. Spread of epidemic disease on networks. *Physical review E*, 66(1):016128, 2002.

Natalya F Noy, Deborah L McGuinness, et al. Ontology development 101: A guide to creating your first ontology, 2001.

Yu Rong, Wenbing Huang, Tingyang Xu, and Junzhou Huang. Dropedge: Towards deep graph convolutional networks on node classification. *arXiv preprint arXiv:1907.10903*, 2019.

Andrew I Schein, Alexandrin Popescul, Lyle H Ungar, and David M Pennock. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 253–260, 2002.

Harald Steck. Item popularity and recommendation accuracy. In *Proceedings of the fifth ACM conference on Recommender systems*, pp. 125–132, 2011.

Zichong Wang, Zhipeng Yin, Fang Liu, Zhen Liu, Christine Lisetti, Rui Yu, Shaowei Wang, Jun Liu, Sukumar Ganapati, Shuigeng Zhou, et al. Graph fairness via authentic counterfactuals: Tackling structural and causal challenges. *ACM SIGKDD Explorations Newsletter*, 26(2):89–98, 2025a.

Zichong Wang, Zhipeng Yin, Yuying Zhang, Liping Yang, Tingting Zhang, Niki Pissinou, Yu Cai, Shu Hu, Yun Li, Liang Zhao, et al. Fg-smote: Towards fair node classification with graph neural network. *ACM SIGKDD Explorations Newsletter*, 26(2):99–108, 2025b.

Tianxiang Zhao, Xiang Zhang, and Suhang Wang. Graphsmote: Imbalanced node classification on graphs with graph neural networks. In *Proceedings of the 14th ACM international conference on web search and data mining*, pp. 833–841, 2021.

Tong Zhao, Wei Jin, Yozen Liu, Yingheng Wang, Gang Liu, Stephan Günnemann, Neil Shah, and Meng Jiang. Graph data augmentation for graph machine learning: A survey. *arXiv preprint arXiv:2202.08871*, 2022a.

Tong Zhao, Gang Liu, Daheng Wang, Wenhao Yu, and Meng Jiang. Learning from counterfactual links for link prediction. In *International Conference on Machine Learning*, pp. 26911–26926. PMLR, 2022b.

Jiajun Zhou, Chenxuan Xie, Shengbo Gong, Zhenyu Wen, Xiangyu Zhao, Qi Xuan, and Xiaoniu Yang. Data augmentation on graphs: A technical survey. *ACM Computing Surveys*, 57(11):1–34, 2025.

Shixuan Zhu, Qi Shen, Chuan Cui, Yu Ji, Yiming Zhang, Zhenwei Dong, and Zhihua Wei. Data-augmented counterfactual learning for bundle recommendation. In *International Conference on Database Systems for Advanced Applications*, pp. 314–330. Springer, 2023.

A Appendix

The Game Design Patterns (GDP) dataset serves as a (semi-)ontology for the game design domain, offering formalized descriptions, properties, and constraints for each concept (Noy et al., 2001). Central to GDP is the notion of a "pattern"—a recurring interaction that can be instantiated in diverse games, independent of genre or theme. For example, the "Alignment" pattern refers to "the goal of forming a linear arrangement of game elements." [2] This pattern is exemplified in games such as Tic-Tac-Toe, Candy Crush Saga, and Tetris.

Patterns in GDP not only describe game mechanics but also exhibit structural relationships with other patterns: they can enable ("instantiate"), modify, or potentially conflict with the deployment

[2] `http://virt10.itu.chalmers.se/index.php/Alignment`

of other patterns. As such, GDP provides a shared vocabulary for game designers to communicate, analyze, and create games.

However, the identification and verification of game design patterns is a highly specialized and expert-driven process. Despite the vast number of games, only a limited number of patterns have been formally proposed, and even fewer game–pattern relationships have been verified by experts. This results in an inherently sparse bipartite graph, making GDP an ideal testbed for evaluating augmentation strategies in edge-sparse domains.