

Taxonomy Maintenance in the Wild over Evolving Scholarly Data: Reliability, Efficiency, and Cost-Effectiveness

DAOMIN JI*, RMIT University, Australia and The University of Queensland, Australia

HUI LUO, University of Wollongong, Australia

ZHIFENG BAO[†], The University of Queensland, Australia

JUNHAO GAN, The University of Melbourne, Australia

ZI HUANG, The University of Queensland, Australia

The rapid growth of scientific publications makes scholarly taxonomies quickly obsolete. We study *taxonomy maintenance in the wild*, a new problem that moves beyond static construction by continuously adapting taxonomies to evolving scholarly repositories, such as arXiv, for a given research topic. We propose GIST, a robust framework for maintaining evolving taxonomies. Unlike purely LLM-centric approaches, GIST grounds structure induction in expert-curated evidence by extracting partial hierarchies from the “Related Work” sections of papers. It integrates these partial taxonomies into a unified global taxonomy in a *geometric box-embedding space*, where box containment encodes the inductive bias of is-a relations. To connect semantics with geometric structure, GIST learns a bidirectional mapping between word embeddings and box embeddings. For efficient incremental updates, GIST uses novelty-aware coreset selection to update the model with representative historical signals and new evidence, avoiding costly full retraining. To handle high-velocity paper streams under user-specific token budgets, GIST further combines a hypothesized concept generator with a cost-effective evidence retrieval module. Experiments on real-world arXiv datasets show that GIST outperforms state-of-the-art baselines, improving Node F1 and Edge F1 by 11.0% and 13.1% over the strongest baseline while requiring only 9.6% of its runtime and 12.7% of its monetary cost.

CCS Concepts: • **Information systems** → **Data structures; Information integration.**

Additional Key Words and Phrases: Taxonomy Maintenance, Scholarly Data Management, Semantic Index

ACM Reference Format:

Daomin Ji, Hui Luo, Zhifeng Bao, Junhao Gan, and Zi Huang. 2026. Taxonomy Maintenance in the Wild over Evolving Scholarly Data: Reliability, Efficiency, and Cost-Effectiveness. *Proc. ACM Manag. Data* 4, 4 (SIGMOD), Article 289 (September 2026), 28 pages. <https://doi.org/10.1145/3837127>

1 Introduction

Taxonomies provide structured abstractions of domain knowledge by organizing concepts through hierarchical “is-a” relationships. They have long served as foundational infrastructure in data management, supporting tasks such as knowledge base construction [5, 52], semantic indexing and query reformulation [13, 16], Text-to-SQL translation [47, 48], and product catalog management [28].

*This work was done when Daomin Ji was a visiting student at The University of Queensland.

[†]Zhifeng Bao is the corresponding author.

Authors’ Contact Information: Daomin Ji, daomin.ji@student.rmit.edu.au, RMIT University, Melbourne, Victoria, Australia and The University of Queensland, Brisbane, Queensland, Australia; Hui Luo, huil@uow.edu.au, University of Wollongong, Wollongong, New South Wales, Australia; Zhifeng Bao, zhifeng.bao@uq.edu.au, The University of Queensland, Brisbane, Queensland, Australia; Junhao Gan, junhao.gan@unimelb.edu.au, The University of Melbourne, Melbourne, Victoria, Australia; Zi Huang, huang@itee.uq.edu.au, The University of Queensland, Brisbane, Queensland, Australia.



This work is licensed under a Creative Commons Attribution 4.0 International License.

© 2026 Copyright held by the owner/author(s).

ACM 2836-6573/2026/9-ART289

<https://doi.org/10.1145/3837127>

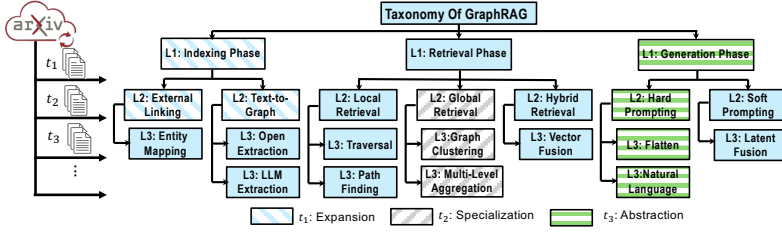


Fig. 1. The rapid development of research topic *GraphRAG*

In this work, we focus on topic-centric taxonomies over scholarly data—fine-grained hierarchies tailored to a specific research topic—that organize knowledge from high-level problems and methodological families down to concrete design choices. A critical characteristic of scholarly taxonomies is their *volatility*: Unlike taxonomies in general-purpose knowledge bases, whose conceptual hierarchies evolve relatively gradually, the scientific landscape for certain research topics evolves rapidly, as illustrated in Example 1. Consequently, existing taxonomies can quickly become obsolete, potentially impairing the key scholarly data management workflows: (1) *Reviewer Assignment*: Outdated classification schemes fail to capture emerging sub-fields, leading to misclassified papers and mismatched reviewer assignments [3]; (2) *Information Retrieval*: Digital libraries relying on static indices, such as ACM Digital Library [2] and IEEE Xplore [24], cannot effectively support semantic navigation for newly coined terminology, creating gaps between user queries and relevant literature; (3) *Conference Organization* [8, 54]: conference programs fail to structurally reflect the shifting distribution of research interests, resulting in coarse-grained sessions.

EXAMPLE 1. Consider Graph Retrieval-Augmented Generation (*GraphRAG*) in Fig. 1, a rapidly evolving domain where researchers customize LLMs by organizing external knowledge as graphs, retrieving graph-structured evidence, and integrating it into generation. Since Microsoft introduced the initial *GraphRAG* method in 2024 [15], the field has expanded rapidly. As of January 2026, over 730 papers on arXiv leverage or discuss this paradigm. As the literature evolves, the taxonomy may undergo the following structural refinements at different timestamps: (1) **expansion** (at t_1), which introduces a new node under an existing non-leaf category (e.g., adding External Linking as a sub-category under Indexing Phase, parallel to Text-to-Graph); (2) **specialization** (at t_2), which splits a coarse concept into finer-grained variants (e.g., dividing Global Retrieval into Graph Clustering and Multi-Level Aggregation); and (3) **abstraction** (at t_3), which adds an intermediate high-level parent to unify multiple existing paradigms (e.g., adding Hard Prompting under the original parent Generation Phase to unify Flatten and Natural Language).

This motivates the problem of maintaining *topic-centric* taxonomies over a large, evolving scholarly repository (e.g., arXiv). Given a target research topic and an evolving scholarly repository that reflects the shifting research landscape, the goal is to automatically construct and incrementally adapt a taxonomy, ensuring it evolves alongside the underlying literature.

Most existing scholarly taxonomy construction methods [29, 32] are designed for small, static corpora and cannot scale to large, dynamic settings due to three key limitations: (1) They heavily rely on LLMs not only to extract salient signals but also to reason about hierarchical structure. Consequently, they are vulnerable to hallucination: without reliable textual cues, LLM-inferred hierarchies often contain plausible yet unsupported concepts or relations. (2) They lack native support for continual updates. With new arrivals, one must either re-infer the taxonomy from scratch over all accumulated documents, or build a delta taxonomy from the new batch and merge it via entity resolution [55], which may fix naming inconsistencies but often fails to resolve relational conflicts, causing drift (e.g., cycles or duplicated branches). As evidenced in Sec. 7.1, reconstruction

from scratch increases monetary and time costs by up to 3.9 $\times$, while naive incremental update degrades performance by 14%–18%. In Example 1, a single research topic typically can involve hundreds or thousands of relevant papers, making *full-corpus processing* over all relevant papers infeasible.

To guide our method design, we identify three key desiderata. **D1: Reliable structure induction.** It should infer taxonomic concepts and relations from reliable signals (rather than unconstrained LLM reasoning) to reduce hallucination. **D2: Efficient incremental updates.** Given the continuous arrival of new papers, it should support efficient incremental updates that incorporate emerging knowledge without expensive recomputation. **D3: Cost-effective evidence acquisition.** Under a fixed budget (e.g., token budget), it should selectively process the most informative documents to maximize evidence utility per unit cost. To meet the above desiderata, we develop GIST, a Geometric-Inferred and Self-refining Taxonomy Maintenance framework (Fig. 2). Our contributions include:

- To satisfy *D1*, GIST derives taxonomic concepts and relations from expert-curated textual cues by using LLMs primarily as *structure extractors*: it distills topic-relevant partial taxonomies from the *Related Work* (or equivalent) sections of retrieved papers where authors explicitly organize and categorize prior work, and integrates these high-precision yet partial structures into a unified global taxonomy, mitigating hallucinated nodes and edges. This *taxonomy integration* is formulated in a geometric box-embedding space, where box containment encodes inductive bias for “is-a” relations. To bridge semantic and geometric spaces, GIST learns a *self-supervised bidirectional word-box mapping model* from the extracted partial taxonomies and applies reliability-aware filtering to retain high-confidence signals. (Sec. 3.1–3.2)
- With this mapping model, GIST avoids redundant reprocessing of historical content by transforming only novel concepts from newly extracted partial taxonomies into geometric boxes. To support *D2* and continually adapt the mapping model, we introduce a novelty-aware coreset selection strategy. This strategy updates the bidirectional mapping model using only a representative subset of historical signals alongside new evidence, enabling efficient assimilation of emerging concepts without full retraining. (Sec. 3.3)
- To support *D3*, GIST introduces two tightly coupled modules: (1) Instead of using only the input topic keywords or existing taxonomy concepts as search signals—which can be overly broad and often retrieve redundant evidence—we propose a *hypothesized concept generation* module that predicts likely emerging concepts for the next iteration (Sec. 4). (2) Given a user-specified token budget, we develop a *cost-effective evidence retrieval* module that aims to select an optimal query set of hypothesized concepts and retrieve (and process) only the most informative newly arrived papers to extract the partial taxonomies.
- We build an *incremental semantic index* atop GIST to enhance downstream paper search accuracy. This index is also maintained incrementally—mapping only newly arrived papers to the evolving taxonomy—thereby avoiding costly full repository rescanning to rebuild the index. (Sec. 6)
- Experiments on the real-world *arXiv* scholarly corpus across diverse topics show that GIST consistently constructs taxonomies with higher coverage and structural quality than the baselines. In particular, GIST achieves average relative improvements of 11.0% and 13.1% in *Node Soft F1* and *Edge Soft F1* over the strongest baseline, while requiring only 9.6% of its runtime and 12.7% of its monetary cost; the incremental update further delivers a 3.1 $\times$ speedup over full retraining.

2 Problem and Solution Overview

2.1 Problem Formulation

Concept and Taxonomy. A scholarly taxonomy is a rooted directed acyclic graph (DAG) $\mathcal{T} = (\mathcal{V}, \mathcal{E})$, where each node $v \in \mathcal{V}$ is a *concept*—a named semantic *category* used to classify documents

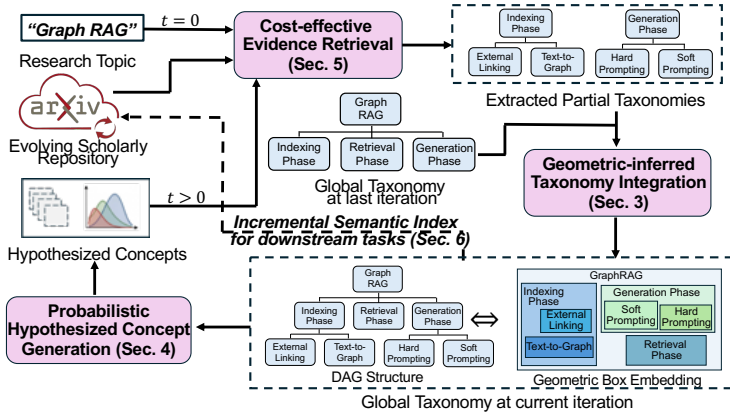


Fig. 2. The workflow of the proposed GIST

within the target topic, represented by a textual label. Each directed edge $(p, q) \in \mathcal{E}$ encodes a hypernym–hyponym (parent–child) relation, meaning that q is a sub-category of p . The taxonomy is rooted at a single concept r representing the overarching research topic.

Scholarly Data Repository. The scholarly data repository is an evolving set of documents, e.g., research papers, denoted at time t as $\mathcal{P}_t = \{d_1, d_2, \dots, d_{|\mathcal{P}_t|}\}$. Each document d_i comprises textual content (e.g., title, abstract, and full text). Crucially, we adopt an *open-world* setting (e.g., arXiv), where uncured and continuously incoming documents yield a repository that mixes topic-relevant documents with off-topic or otherwise irrelevant documents.

The continuous evolution of the repository necessitates *dynamic structural refinement* of the taxonomy:

Structural Refinement. We classify structural refinement into three types based on how a new concept relates to the current hierarchy:

- **Expansion Refinement** occurs when a novel paradigm emerges at the same hierarchical level as existing categories under a common parent. It is realized by adding a new *sibling node* to form a parallel (sub)-branch.
- **Specialization Refinement** occurs when an existing leaf node becomes too coarse as the literature grows. It is realized by introducing finer-grained *child nodes* to increase granularity.
- **Abstraction Refinement** occurs when a set of sibling nodes naturally suggests a higher-level shared category. It is realized by inserting an *intermediate node* to group these siblings.

Based on the preliminaries, we define our problem as follows:

DEFINITION 1 (TAXONOMY MAINTENANCE IN THE WILD). Given an input research topic r and an evolving repository $\mathcal{P}_t$, the problem of taxonomy maintenance in the wild is twofold:

- **Initial Construction** ($t = 0$): Construct an initial taxonomy $\mathcal{T}_0$ rooted at r using the initial corpus snapshot $\mathcal{P}_0$, establishing the baseline semantic structure of the domain.
- **Continuous Evolution** ($t > 0$): For each subsequent timestamp, refine $\mathcal{T}_t$ in light of newly arriving scientific documents $\Delta\mathcal{P}_{t+1} = \mathcal{P}_{t+1} \setminus \mathcal{P}_t$ by applying structural refinements (e.g., expansion, specialization, and abstraction), yielding an updated taxonomy $\mathcal{T}_{t+1}$ that remains consistent with the evolving $\mathcal{P}_{t+1}$.

2.2 Solution Overview

To address this challenge, we propose GIST, a Geometric-Inferrred and Self-refining Taxonomy Maintenance framework (Fig. 2) that operates in a continuous three-module cycle:

- **Cost-effective Evidence Retrieval.** Guided by the input research topic for initialization ($t = 0$) and, for subsequent iterations ($t > 0$), by the hypothesized concepts generated by the *Probabilistic Hypothesized Concept Generation* module, it employs a novel *utility-driven, budget-constrained planner* to maximize evidence utility under a user-specified token budget (and corresponding monetary cost): it selects (i) which hypotheses to instantiate as retrieval queries and (ii) which retrieved papers to parse, and then applies an LLM-guided extractor to distill partial taxonomies from their *Related Work* (or equivalent) sections. (Sec. 5).
- **Geometric-inferred Taxonomy Integration.** This module integrates newly extracted partial taxonomies with the global taxonomy from iteration $t - 1$, yielding the updated taxonomy at iteration t . We propose a bidirectional word–box mapping model that embeds concepts in a *geometric box space* and leverages containment relationships to guide robust taxonomy integration. The model is further designed for reliable and efficient self-supervised training via a dual-objective data selection strategy. (Sec. 3)
- **Probabilistic Hypothesized Concept Generation.** This module forecasts likely *structural refinement directions* (e.g., expansion, specialization, and abstraction) in the current taxonomy $\mathcal{T}_t$ and generates corresponding *hypothesized concepts* as query candidates for the next iteration. We instantiate this step with a geometric concept emergence model that leverages the inductive bias of box embeddings: unoccupied regions (geometric “gaps”) within a parent concept’s embedding box indicate latent space where emerging concepts are likely to reside. (Sec. 4)

Remark: While we instantiate GIST for evolving scholarly data, the framework is broadly applicable to other evolving hierarchical knowledge structures whose schema itself drifts over time, including product catalogs, biomedical concept hierarchies, and domain-specific knowledge bases. Adapting to a new domain requires changes only at the evidence-extraction stage: the regex section detector and Related-Work prompt would be replaced by a domain-specific extractor (e.g., a category-attribute parser for e-commerce catalogs, or an ontology-aligned span extractor for biomedical texts). The geometric integration, hypothesized concept generation, and budget-aware retrieval modules are domain-agnostic and require no modification.

3 Geometric-inferred Taxonomy Integration

Merging partial taxonomies with the global taxonomy presents challenges beyond simple name resolution. Entity Resolution methods [17, 23] alone cannot resolve structural conflicts—such as granularity mismatches or cycles—often resulting in structural incoherence. Consequently, we formulate this integration problem within a geometric box-embedding space, utilizing geometric containment as a strict inductive bias for hierarchy construction (Sec. 3.1). To bridge the semantic and geometric spaces, we propose a bidirectional word–box mapping model (Sec. 3.2). Crucially, to address the challenges of continual learning and noisy training signals, we employ a self-supervised dual-objective training scheme that selectively filters high-quality signals for robustness while retaining a representative historical set for efficient incremental updates (Sec. 3.3).

3.1 Box Representations for Taxonomy Structures

In the geometric box-embedding space [38, 62], each concept $v \in \mathcal{V}$ is encoded as an axis-aligned hyper-rectangle (hereafter referred to as a *box*) in a d -dimensional Euclidean space, as illustrated in the right panel of Fig. 3. Formally, we adopt a center-offset parameterization:

$$B_v = B(\mathbf{c}_v, \mathbf{b}_v) = \left\{ \mathbf{x} \in \mathbb{R}^d \mid |\mathbf{x}^{(k)} - \mathbf{c}_v^{(k)}| \leq \mathbf{b}_v^{(k)}, \forall k \in \{1, \dots, d\} \right\}.$$

Here, $\mathbf{c}_v \in \mathbb{R}^d$ denotes the *center* of the box, and $\mathbf{b}_v \in \mathbb{R}_{\geq 0}^d$ is the non-negative *offset* vector determining its spatial extent along each dimension. d is the dimensionality of the box-embedding space, and k indexes each coordinate axis from 1 to d .

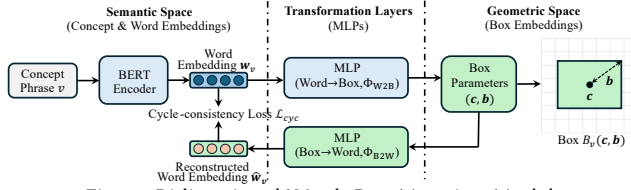


Fig. 3. Bidirectional Word-Box Mapping Model.

In the geometric space, hypernym-hyponym relations are encoded via containment constraints. Specifically, for any directed edge $(p, q) \in \mathcal{E}$, the parent concept p must spatially *contain* its child concept q (i.e., $B_p \supseteq B_q$). For example, in Fig. 2, the box corresponding to *Generation Phase* must enclose those of its child concepts, including *Hard Prompting* and *Soft Prompting*.

3.2 Bidirectional Word-Box Mapping Model

3.2.1 Model Architecture. In order to bridge the semantic space and geometric (box embedding) space, we propose a bidirectional word-box mapping model that jointly learns two complementary transformations, as illustrated in Fig. 3:

- **Word $\rightarrow$ Box (Φ_{W2B}):** Given a concept v , a frozen pretrained encoder (e.g., BERT [12]) first converts it into a word embedding $\mathbf{w}$. To capture the non-linear transformation from the semantic space to the geometric space, we employ a multilayer perceptron (MLP) that maps $\mathbf{w}$ to a joint latent representation of dimension $2d$. We split this output to obtain the center $\mathbf{c}$ and an offset vector $\mathbf{b}$: $[\mathbf{c}; \mathbf{b}] = \text{MLP}_{\text{enc}}(\mathbf{w})$ where $[\cdot; \cdot]$ denotes vector concatenation.
- **Box $\rightarrow$ Word (Φ_{B2W}):** Similarly, a second MLP implements the reverse transformation by projecting a box embedding $B(\mathbf{c}, \mathbf{b})$ back into a continuous query embedding in the semantic space. We concatenate the center and offset vectors to form a unified geometric feature, which is then mapped to a semantic vector $\mathbf{w}$ via a decoder MLP: $\mathbf{w} = \text{MLP}_{\text{dec}}([\mathbf{c}; \mathbf{b}])$.

3.2.2 Training Objectives. The model is trained using self-supervised signals derived from the extracted parent-child relations. We optimize a joint objective consisting of three loss functions:

- **Geometric Containment Loss $\mathcal{L}_{\text{cont}}$.** To enforce hierarchy, we require that the parent box B_p spatially encapsulates its child B_q . We quantify this by measuring the volume of the child covered by the parent: $\text{cov}(B_p, B_q) = \frac{\text{Vol}(B_p \cap B_q)}{\text{Vol}(B_q)} \in [0, 1]$, where $\text{Vol}(B) = \prod_{k=1}^d 2b^{(k)}$. The loss is minimized when containment is complete: $\mathcal{L}_{\text{cont}}(p, q) = 1 - \text{cov}(B_p, B_q)$.
- **Cycle-Consistency Loss $\mathcal{L}_{\text{cycle}}$.** Since we lack ground-truth geometric coordinates to supervise the inverse Φ_{B2W} mapping directly, we adopt a self-supervised cycle-consistency approach [65]. We penalize the reconstruction error between the original word embedding $\mathbf{w}$ and its round-trip projection through the box space: $\mathcal{L}_{\text{cycle}}(v) = \frac{1}{2} \left(1 - \cos(\Phi_{B2W}(\Phi_{W2B}(\mathbf{w})), \mathbf{w}) \right)$. This ensures that the learned geometric representations retain high semantic fidelity to the input space.
- **Volume Regularization Loss $\mathcal{L}_{\text{vol}}$.** To avoid degenerate solutions where box volumes collapse to zero (leading to trivial containment), we enforce a minimum-volume constraint by penalizing boxes whose volume falls below a threshold $V_{\min}$: $\mathcal{L}_{\text{vol}} = \frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} \max(0, V_{\min} - \text{Vol}(B_v))$.

Overall Objective. The total loss is a weighted sum: $\mathcal{L} = \mathcal{L}_{\text{cont}} + \mathcal{L}_{\text{cycle}} + \lambda_{\text{vol}} \mathcal{L}_{\text{vol}}$, where λ_{vol} balances the regularization strength against the primary structural and semantic objectives.

3.3 Robust and Efficient Incremental Training

To optimize these objectives without manual annotation, we derive self-supervised signals from parent-child relations in the extracted partial taxonomies. However, the continuous influx of papers introduces a *learning dilemma*: Updating the model is necessary to capture structural changes, but

doing so without historical data can cause *catastrophic forgetting* [30]; in contrast, retraining on the entire dataset is computationally infeasible. This process is further complicated by inherent noise in the self-supervision signals, stemming from three sources: (1) *Subjectivity*, arising from divergent author perspectives and terminology; (2) *Temporal Drift*, where the ground-truth structure itself evolves (e.g., the emergence of new concepts); (3) *Extraction Errors*, caused by inevitable inaccuracies in LLM processing. Consequently, relying solely on direct supervision from raw extracted pairs is prone to inconsistency and error accumulation.

To address these issues, GIST introduces a *dual-objective data selection strategy* designed for efficient and robust incremental training. This strategy selects two types of data: 1) **New data** ($\mathcal{D}_{\text{new}}$). A *Reliability-Aware Graph Selection* retains only the most reliable relation pairs, thereby reducing noise in the supervision signal. 2) **Historical data** ($\mathcal{D}_{\text{old}}$). A *novelty-aware coreset selection* mechanism selects a compact yet representative subset of historical training instances, preserving accumulated knowledge without incurring the cost of retraining from scratch.

3.3.1 Reliability-Aware Graph Selection ($\mathcal{D}_{\text{new}}$). To distill high-quality signals from $\mathcal{D}_{\text{new}}$, we construct a weighted candidate graph from extracted partial taxonomies and then select a subset of high-support relations that ensures broad concept participation.

Graph construction from partial taxonomies. To mitigate terminological heterogeneity, we employ Entity Resolution (ER) [9] to normalize diverse concept names into canonical identifiers. This process consolidates synonyms, such as *Cardinality Estimation* and *Selectivity Estimation*, via a blocking-and-matching pipeline, instantiated using either deep learning models [37, 55] or lightweight LLMs [14]. We then aggregate the normalized parent-child relations into a weighted *candidate concept graph* $\mathcal{G}_{\text{cand}} = (\mathcal{V}_{\text{cand}}, \mathcal{E}_{\text{cand}})$, where $\mathcal{V}_{\text{cand}}$ denotes the set of unique candidate concepts. For each directed edge $(u, v) \in \mathcal{E}_{\text{cand}}$, we assign a support weight w_{uv} equal to the number of unique source papers that contain this relation. Larger w_{uv} indicates stronger corroboration across independent evidence sources and is used as a proxy for relation reliability.

Coverage-Reliability Selection via Submodular Maximization. Given the set of candidate relations $\mathcal{E}_{\text{cand}}$, we aim to select a subset $\mathcal{D}_{\text{rel}} \subseteq \mathcal{E}_{\text{cand}}$ to serve as reliable supervision signals. This selection is constrained by a cardinality budget and must balance two objectives: (i) *Coverage*, ensuring broad semantic representation by maximizing the number of unique participating concepts; and (ii) *Reliability*, prioritizing relations with strong support weights. We formulate this trade-off as a constrained maximization problem. Let $\text{Cov}(\mathcal{D}_{\text{rel}})$ denote the set of unique concepts covered by the selected relations. We define the normalized coverage score as $f_{\text{cov}}(\mathcal{D}_{\text{rel}}) = |\text{Cov}(\mathcal{D}_{\text{rel}})|/|\mathcal{V}_{\text{cand}}|$. Then, we quantify reliability using log-normalized edge weights to mitigate the effect of heavy-tailed distributions. Defining the normalized weight $\tilde{w}_{uv} = \log(1 + w_{uv})/\log(1 + w_{\text{max}})$, the reliability score is given by $f_{\text{rel}}(\mathcal{D}_{\text{rel}}) = \frac{1}{K_{\text{rel}}} \sum_{(u,v) \in \mathcal{D}_{\text{rel}}} \tilde{w}_{uv}$. The overall optimization problem is:

$$\begin{aligned} \max_{\mathcal{D}_{\text{rel}} \subseteq \mathcal{E}_{\text{cand}}} \quad & F(\mathcal{D}_{\text{rel}}) = f_{\text{cov}}(\mathcal{D}_{\text{rel}}) + \lambda_{\text{rel}} f_{\text{rel}}(\mathcal{D}_{\text{rel}}) \\ \text{s.t.} \quad & |\mathcal{D}_{\text{rel}}| \leq K_{\text{rel}}. \end{aligned} \quad (1)$$

where $K_{\text{rel}} = \lfloor \eta_{\text{rel}} |\mathcal{D}_{\text{new}}| \rfloor$ is the cardinality budget determined by the retention fraction η_{rel} , and λ_{rel} is a hyperparameter balancing the two terms.

Greedy Optimization and Guarantee. The objective function F is non-negative, monotone, and submodular, as it combines a coverage term exhibiting diminishing marginal returns with a strictly modular (linear) reliability term. (We provide a formal proof in the technical report [26].) Leveraging these properties, we employ a standard greedy algorithm that iteratively selects the edge e^* maximizing the marginal gain $\Delta F(e \mid \mathcal{D}_{\text{rel}})$, which is computationally efficient and theoretically grounded, providing a $(1 - 1/e)$ -approximation guarantee to the optimal solution [43].

Remark. Reliability-aware filtering grounds GIST’s robustness in *cross-paper consensus* rather than any single extraction signal, operating along two complementary axes. *Synchronically*, key concepts and relations within a domain are typically attested by multiple papers, so missing detections (e.g., papers without an explicit Related Work section) and noisy extractions are absorbed by corroborating evidence: uncorroborated relations are downweighted through the support weights w_{uv} , while well-supported structures are preserved. *Diachronically*, the same principle yields a *self-correcting mechanism* for niche subfields, where low-support branches may reflect either idiosyncratic framing by a single author or the genuine emergence of a novel domain. The continuous update mechanism resolves this ambiguity *a posteriori*: branches the community adopts accumulate cross-paper support and their w_{uv} grows over time, ensuring survival under selection, while unreinforced branches remain low-weight and are gradually displaced. The filter therefore neither commits prematurely to any single author’s framing nor over-penalizes emerging structure, adapting instead to the empirical distribution of cross-paper agreement as the repository grows.

3.3.2 Incremental Model Update via Novelty-Aware Coreset Selection ($\mathcal{D}_{old}$). To enable continual refinement of the dual mapping model without full retraining, we draw inspiration from prior research on coreset selection [7, 11, 58]. At each iteration, the full historical dataset $\mathcal{D}_{old}$ is approximated by a compact, weighted *coreset* $\mathcal{D}_{core}$, defined as a subset of size $|\mathcal{D}_{core}| = \eta_{core} |\mathcal{D}_{old}|$, where $\eta_{core} \in (0, 1]$ is a hyperparameter controlling the compression ratio. This coreset is then utilized for subsequent training.

Primary Goal: Loss Preservation. The primary goal of the coreset adheres to the principles of *Empirical Risk (Loss) Minimization (ERM)* [42], which aims to select $\mathcal{D}_{core}$ with weights $\{w'_e\}_{e \in \mathcal{D}_{core}}$ such that its weighted empirical risk $\tilde{\mathcal{L}}_{old}(\theta) = \sum_{e \in \mathcal{D}_{core}} w'_e \ell(e; \theta)$ closely approximates the full historical risk $\mathcal{L}_{old}(\theta)$:

$$\min_{\mathcal{D}_{core} \subset \mathcal{D}_{old}, \{w'_e\}} \left| \mathcal{L}_{old}(\theta) - \sum_{e \in \mathcal{D}_{core}} w'_e \ell(e; \theta) \right| \quad (2)$$

where $w'_e \geq 0$ (with $\sum w'_e = 1$) denotes the importance weight assigned to each selected instance to mirror the full-data objective. The key issue is thus to choose an optimal sampling distribution over $\mathcal{D}_{old}$ from which to construct $\mathcal{D}_{core}$ and assign the weights $\{w'_e\}$.

Complementary Goal: Novelty for Dynamic Data. Traditional coreset selection assumes a static dataset, ignoring potential redundancy between the historical coreset and incoming data $\mathcal{D}_{new}$. We address this by enforcing a *novelty* constraint: *retained historical instances must be complementary to the new batch*. To quantify this, we define a novelty score $n(e)$ for each historical edge $e = (p, q) \in \mathcal{D}_{old}$. First, we represent each edge as the concatenation of its parent and child word embeddings $(\mathbf{w}_p, \mathbf{w}_q)$: $\mathbf{r}(e) = [\mathbf{w}_p; \mathbf{w}_q]$. We then measure the maximum similarity of e to the incoming batch $\mathcal{D}_{new}$: $s(e) = \max_{e' \in \mathcal{D}_{new}} \cos(\mathbf{r}(e), \mathbf{r}(e'))$. The final novelty score is $n(e) = 1 - s(e)$, which prioritizes historical instances that are semantically distinct from the new data.

Novelty-Aware Coreset Selection. Based on the above idea, we formulate coreset selection as an optimization problem to find a sampling distribution that balances fidelity to the historical risk (Π_{ERM}) with novelty relative to the new data.

$$\max_{\Pi_{core} \in \Delta(\mathcal{D}_{old})} -\text{KL}(\Pi_{core} \parallel \Pi_{ERM}) + \lambda_{nov} \mathbb{E}_{e \sim \Pi_{core}} [n(e)], \quad (3)$$

where $\Delta(\mathcal{D}_{old}) = \{\Pi_{core} : \mathcal{D}_{old} \rightarrow [0, 1] \mid \sum_e \Pi_{core}(e) = 1\}$ denotes the set of all valid probability distributions over the $\mathcal{D}_{old}$. Here, Π_{ERM} represents the loss distribution at the previous timestamp, derived directly from the preceding training stage. $\text{KL}(\cdot \parallel \cdot)$ denotes the Kullback–Leibler divergence, and λ_{nov} is a hyperparameter controlling the trade-off between loss preservation and novelty.

To derive the optimal sampling distribution Π_{core} and the corresponding importance weights w'_e , we establish Propositions 1 and 2. Detailed proofs are provided in the technical report [26].

Algorithm 1 Taxonomy Integration

Require: Previous global taxonomy $\mathcal{T}_{t-1}$; partial taxonomies $\{\mathcal{T}_{par,1}, \dots, \mathcal{T}_{par,n}\}$; Φ_{W2B} ; containment threshold τ .

Ensure: Updated global taxonomy $\mathcal{T}_t$.

- 1: Construct the unified concept set $\mathcal{V}_t \leftarrow \mathcal{V}(\mathcal{T}_{t-1}) \cup \bigcup_{i=1}^n \mathcal{V}(\mathcal{T}_{par,i})$.
 - 2: For each $v \in \mathcal{V}_t$, compute its box embedding $B_v \leftarrow \Phi_{W2B}(\mathbf{w}_v)$.
 - 3: Initialize the candidate edge set $\mathcal{E}_{cand} \leftarrow \emptyset$.
 - 4: **for** each ordered pair $(u, v) \in \mathcal{V}_t \times \mathcal{V}_t, u \neq v$ **do**
 - 5: Compute the geometric containment score $s_{u \rightarrow v} \leftarrow \text{cov}(B_u, B_v)$.
 - 6: **if** $s_{u \rightarrow v} \geq \tau$ **and** $\text{Vol}(B_u) > \text{Vol}(B_v)$ **then**
 - 7: Add (u, v) to $\mathcal{E}_{cand}$.
 - 8: **end if**
 - 9: **end for**
 - 10: Apply transitive reduction to obtain a minimal edge set $\mathcal{E}_{final} \leftarrow \text{TransitiveReduction}(\mathcal{V}_t, \mathcal{E}_{cand})$.
 - 11: **return** $\mathcal{T}_t \leftarrow (\mathcal{V}_t, \mathcal{E}_{final})$.
-

PROPOSITION 1 (OPTIMAL SAMPLING DISTRIBUTION). *The unique sampling distribution Π_{core}^* that maximizes the novelty-aware objective in Eq. 3 is given by $\Pi_{core}^*(e) = \frac{\Pi_{ERM}(e) \exp\{\lambda_{nov} n(e)\}}{\sum_{e'} \Pi_{ERM}(e') \exp\{\lambda_{nov} n(e')\}}$.*

PROPOSITION 2 (OPTIMAL CORESET WEIGHTING). *To ensure that the coreset risk $\tilde{\mathcal{L}}_{old}(\theta)$ is a consistent estimator of the full historical risk $\mathcal{L}_{old}(\theta)$, each instance $e \in \mathcal{D}_{core}$ is assigned a normalized inverse-probability weight $w'_e = \frac{1/\Pi_{core}^*(e)}{\sum_{e_j \in \mathcal{D}_{core}} 1/\Pi_{core}^*(e_j)}$.*

3.4 Taxonomy Integration

As presented in Alg. 1, we construct the unified global taxonomy $\mathcal{T}_t$ by merging concepts from the previous state $\mathcal{T}_{t-1}$ with novel concepts from the partial taxonomies at timestamp t . Specifically, we form a unified concept set $\mathcal{V}_t = \mathcal{V}(\mathcal{T}_{t-1}) \cup \bigcup_{i=1}^n \mathcal{V}(\mathcal{T}_{par,i})$ and project each concept $v \in \mathcal{V}_t$ into the box-embedding space via $B_v = \Phi_{W2B}(\mathbf{w}_v)$ (Lines 1–2). We then infer candidate hypernym–hyponym relations by evaluating geometric containment for each ordered pair of concepts (Lines 4–9). Concretely, we compute the containment score $s_{u \rightarrow v} = \text{cov}(B_u, B_v)$ and introduce a candidate edge $u \rightarrow v$ only if $s_{u \rightarrow v} \geq \tau$ (Lines 5–7), where τ is a predefined threshold (set to 0.9 by default). To ensure that the candidate relation set is acyclic under approximate containment scores, we retain only edges directed from larger to smaller boxes, i.e., $\text{Vol}(B_u) > \text{Vol}(B_v)$ (Line 6). Finally, since containment is transitive (e.g., $B_c \subset B_b \subset B_a$), we apply transitive reduction to remove redundant indirect relations and obtain a minimal, consistent hierarchy $\mathcal{T}_t$ (Line 10).

4 Probabilistic Hypothesized Concept Generation

Based on the global taxonomy $\mathcal{T}_t$, we propose a Probabilistic Hypothesized Concept Generation module that produces a set of hypothesized concepts for iteration $t + 1$. These hypotheses serve as query candidates for the subsequent evidence retrieval stage: by targeting likely emerging topics, they provide more informative search signals than reusing existing taxonomy concepts, enabling *cost-effective retrieval*. To this end, we introduce a *Geometric Concept Emergence Model* grounded in the geometric box embedding space, where containment constraints act as an intrinsic inductive bias for concept emergence under different structural refinements (Sec. 4.1). Finally, we enable efficient online updates via a self-supervised, lightweight Method-of-Moments estimator that infers parameters directly from the taxonomy structure (Sec. 4.2).

4.1 Geometric Concept Emergence Model

Given the taxonomy at iteration t , our goal is to estimate $P(v \mid \mathcal{T}_t)$ —the probability that a new concept $v \in \Delta \mathcal{V}_{t+1} = \mathcal{V}_{t+1} \setminus \mathcal{V}_t$ emerges at $t+1$. We model emergence as two coupled decisions:

- **Structural Refinement Context Selection.** We first select a structural refinement context $C \in \mathbb{C}(\mathcal{T}_t)$, which specifies (i) an anchor (parent) node p together with (when applicable) the subset of its children involved, and (ii) the refinement operator to apply. For example, at t_3 in Example 1, the structural refinement context corresponds to applying the refinement operator *abstraction* to the anchor parent node *Generation Phase* and its existing children *Flatten* and *Natural Language*.
- **Context-aware Generation.** Conditioned on the selected context C , we then generate the specific emerging concept v that is compatible with the geometric constraints implied by C .

Thus, we introduce a latent random variable C ranging over the refinement-context set $\mathbb{C}(\mathcal{T}_t)$, and factorize the concept *emergence* probability by marginalizing over all possible contexts: $P(v \mid \mathcal{T}_t) = \sum_{C \in \mathbb{C}(\mathcal{T}_t)} P(v \mid C = C) P(C = C \mid \mathcal{T}_t)$. Moreover, we adopt the modeling assumption that, for any candidate concept v , there is at most one *compatible* context C such that $P(v \mid C = C) > 0$. For instance, *Local Retrieval* can arise under the *Retrieval Phase* but is incompatible with the *Indexing Phase*. While a concept could in principle fit multiple contexts, this assumption holds for the vast majority of cases in practice and keeps the factorization tractable. Hence, the summation collapses to the single valid term:

$$P(v \mid \mathcal{T}_t) = P(v \mid C = C) P(C = C \mid \mathcal{T}_t), \quad (4)$$

where C denotes the (unique, if it exists) context compatible with v .

We instantiate this factorization in a geometric box-embedding space by representing each concept v with a box embedding B and encoding structural relations via containment constraints. This yields two modeling objectives: (1) *context-aware generation* $P(B \mid C = C)$, which characterizes how an emerging concept can be geometrically realized under the constraints induced by C ; and (2) *context selection prior* $P(C = C \mid \mathcal{T}_t)$, which prioritizes the most likely refinement contexts under the current taxonomy.

4.1.1 Context-aware Generation. The key advantage of modeling emergence in the geometric box space is that geometric relations impose strong inductive biases, enabling us to infer plausible emerging nodes under different structural refinement contexts. As illustrated in Fig. 4: (1) *Specialization* inserts a new child under an existing node, requiring the child box to be contained within the parent box; (2) *Abstraction* inserts an intermediate node to group a set of siblings, requiring the new box to enclose the grouped children while remaining contained within the original parent; and (3) *Expansion* inserts a new child under a parent with existing children, requiring the new box to be contained within the parent while remaining geometrically separated from existing sibling boxes.

Leveraging these geometric inductive biases, we cast context-aware generation as sampling a valid box B from the feasible region induced by the refinement context C , which reduces $P(B \mid C = C)$ to learning distributions over unit intervals $[0, 1]$.

Parent-normalized Reparameterization. We apply a scale-invariant, parent-normalized reparameterization to make the distribution tractable. We re-parameterize the problem into a parent-normalized coordinate system anchored at the context anchor (parent) node p . For any box $B_q = (c_q, b_q)$ such that $B_q \subseteq B_p$, we define:

$$\rho_q = b_q \oslash b_p \in (0, 1]^d, \quad \delta_q = (c_q - c_p) \oslash b_p \in [-1, 1]^d, \quad (5)$$

where $\oslash$ denotes element-wise division. Here, ρ_q and δ_q denote the *parent-normalized offset* (relative box size) and the *parent-normalized center* (relative placement) of B_q , respectively.

Geometric Coupling and Sequential Factorization. Directly modeling the joint distribution $P(\rho_v, \delta_v \mid C = C)$ is still problematic, since treating size and center independently ignores their strict geometric coupling, often yielding invalid boxes that violate containment boundaries. To enforce these constraints, we decompose the generative objective via the chain rule:

$$P(B \mid C = C) = P(\delta_v \mid \rho_v, C = C) \cdot P(\rho_v \mid C = C). \quad (6)$$

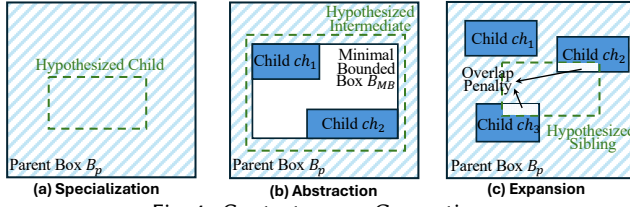


Fig. 4. Context-aware Generation.

This factorization corresponds to a sequential process: we first sample the relative *offset* (ρ_v), and then sample the relative *center* (δ_v) conditioned on that size to ensure geometric validity.

Feasible-Interval Modeling. Given a context C , the feasible set $\mathcal{F}(C)$ admits a per-dimension interval characterization: for each dimension k , $(\rho_v, \delta_v) \in \mathcal{F}(C)$ implies $\rho_v^k \in [\underline{\rho}^k, \bar{\rho}^k]$, $\delta_v^k \in [\underline{\delta}^k, \bar{\delta}^k]$, where the bounds are induced by C . Notably, the feasible interval of δ_v^k depends on the chosen ρ_v^k : *larger boxes admit a smaller range of valid centers under containment/enclosure constraints.*

This interval motivates an auxiliary-variable reparameterization. For each dimension k , we introduce unit random variables $u_\rho^k, u_\delta^k \in [0, 1]$ and map them affinely to the context-induced feasible intervals. Specifically, we parameterize ρ_v^k as $\rho_v^k = \underline{\rho}^k + u_\rho^k(\bar{\rho}^k - \underline{\rho}^k)$, $u_\rho^k \in [0, 1]$, and, conditioned on the sampled ρ_v^k , we parameterize the placement coordinate as $\delta_v^k = \underline{\delta}^k + u_\delta^k(\bar{\delta}^k - \underline{\delta}^k)$, $u_\delta^k \in [0, 1]$. We model these unit variables with the Beta distribution $\text{Beta}(\alpha, \beta)$ for two reasons: its native support on $[0, 1]$ guarantees geometric validity without clipping, and its shape flexibility captures diverse density profiles, from uniform uncertainty to peaked confidence, within a single probabilistic framework. We apply this formulation to all context types $\text{CTX} \in \{\text{SPEC}, \text{ABS}, \text{EXP}\}$, with distinct Beta parameters for $u_{\rho, \text{CTX}}^k$ and $u_{\delta, \text{CTX}}^k$. The problem thus reduces to specifying the boundary constraints $[\underline{\rho}, \bar{\rho}]$ and $[\underline{\delta}, \bar{\delta}]$ per context:

- **Specialization.** The new box B must be contained by the parent box B_p , imposing $B \subseteq B_p$. Accordingly, the relative size is bounded by the parent's unit extent, $[\rho^k, \bar{\rho}^k] = [0, 1]$, and the center is constrained to remain within the residual room, yielding $[\delta^k, \bar{\delta}^k] = [-(1 - \rho_v^k), 1 - \rho_v^k]$.
- **Abstraction.** Let $B_{\text{MB}} = (\delta_{\text{MB}}, \rho_{\text{MB}})$ denote the minimal bounding (MB) box of the grouped children. The new box must satisfy $B_{\text{MB}} \subseteq B \subseteq B_p$, which yields: (1) *Size Bound*: $[\rho^k, \bar{\rho}^k] = [\rho_{\text{MB}}^k, 1]$; and (2) *Center Bound*: conditioned on ρ_v^k , $[\delta^k, \bar{\delta}^k] = \left[\max\{-1 + \rho_v^k, \delta_{\text{MB}}^k + \rho_{\text{MB}}^k - \rho_v^k\}, \min\{1 - \rho_v^k, \delta_{\text{MB}}^k - \rho_{\text{MB}}^k + \rho_v^k\} \right]$.
- **Expansion.** This context imposes two constraints: (1) *Parent-child containment.* The new node must lie inside the parent box. In this case, the feasible intervals coincide with the specialization case. Accordingly, we define a *base proposal distribution* $P_{\text{base}}(B \mid C = C)$ that adopts the same parametric form as specialization but uses distinct Beta parameters; (2) *Sibling overlap penalty.* Unlike specialization, v should avoid regions already explained by existing siblings, since taxonomic principles encourage siblings to be semantically distinct rather than redundant [45]. Let $\mathcal{B}_{\text{Ch}(p)} = \{B_s\}_{s \in \text{Ch}(p)}$ denote the existing child boxes under p . We quantify overlap using the IoU between B and the union of siblings, denoted as $\mathcal{O}(B, \mathcal{B}_{\text{Ch}(p)})$, and define $P(B \mid C = C) \propto P_{\text{base}}(B \mid C = C) \exp(-\mathcal{O}(B, \mathcal{B}_{\text{Ch}(p)})/\tau_{\text{sib}})$, where $\tau_{\text{sib}} > 0$ controls the penalty strength.

4.1.2 Context Selection Prior. The Context Selection Prior $P(C = C \mid \mathcal{T}_t)$ quantifies the likelihood that a candidate context C will be refined at iteration $t+1$. We premise this on the geometric intuition that insertion likelihood is proportional to the *void volume* available within that context. We define $\text{Vol}_{\text{void}}(C)$ for each structural pattern: 1) *Specialization void.* For a leaf node p becoming a parent:

Algorithm 2 Unit-Sample Recovery from Pseudo-Observations

Require: Pseudo-observation z_i ; parent-normalized geometry (ρ_i, δ_i) ;

Ensure: Unit samples $(\mathbf{u}_{\rho,i}, \mathbf{u}_{\delta,i}) \in [0, 1]^d$ and weight ω_i .

```

1: if  $\text{CTX}_i = \text{ABS}$  then
2:   Compute the minimal bounding box:  $(\delta_{\text{MB}}, \rho_{\text{MB}}) \leftarrow \text{MB}(\cdot)$ .
3: end if
4: for each dimension  $k = 1, \dots, d$  do
5:   Bounds: derive the per-dimension feasible intervals  $[\underline{\rho}^k, \bar{\rho}^k]$  and  $[\underline{\delta}^k, \bar{\delta}^k]$  implied by  $\text{CTX}_i$  and the local context geometry.
6:   Invert to unit samples:  $u_{\rho,i}^k \leftarrow \frac{\rho_i^k - \underline{\rho}^k}{\bar{\rho}^k - \underline{\rho}^k}, u_{\delta,i}^k \leftarrow \frac{\delta_i^k - \underline{\delta}^k}{\bar{\delta}^k - \underline{\delta}^k}$ .
7: end for
8: Set  $\omega_i \leftarrow 1$ .
9: if  $\text{CTX}_i = \text{EXP}$  then
10:   $\omega_i \leftarrow \exp(O(B_i)/\tau_{\text{sib}})$ .
11: end if
12: return  $(\mathbf{u}_{\rho,i}, \mathbf{u}_{\delta,i}, \omega_i)$ .
```

$\text{Vol}_{\text{void}}(C) = \text{Vol}(B_p)$. 2) *Abstraction void*. For grouping children under p : $\text{Vol}_{\text{void}}(C) = \text{Vol}(B_p) - \text{Vol}(B_{\text{MB}})$. 3) *Expansion void*. For adding a sibling under p : $\text{Vol}_{\text{void}}(C) = \text{Vol}(B_p) - \text{Vol}\left(\bigcup_{s \in \text{Ch}(p)} B_s\right)$.

Finally, we normalize over all candidate contexts $C_i \in \mathbb{C}(\mathcal{T}_t)$ to obtain a valid distribution:

$$P(C = C_i \mid \mathcal{T}_t) = \frac{\text{Vol}_{\text{void}}(C_i)}{\sum_{C_j \in \mathbb{C}(\mathcal{T}_t)} \text{Vol}_{\text{void}}(C_j)}.$$

4.2 Self-supervised Parameter Estimation

To instantiate the Beta parameters in $P(v \mid \mathcal{T}_t)$, we propose a self-supervised estimation strategy on the current taxonomy $\mathcal{T}_t$. Specifically, we generate *pseudo-observations* by masking existing nodes to emulate structural refinement events (Sec. 4.2.1), and fit the corresponding Beta distributions via a weighted Method-of-Moments (MoM) estimator [60], enabling efficient and incremental updates.

4.2.1 Pseudo-Observation Generation. Since novel nodes in $\mathcal{T}_{t+1}$ are unobserved at time t , we estimate these parameters via self-supervision on the current taxonomy $\mathcal{T}_t$. We construct pseudo-observations $\{z_i\}$ by selecting a *context* from $\mathcal{T}_t$ and temporarily *removing* a node that naturally fits the corresponding refinement pattern; the remaining structure induces a feasible region for that node. We consider three masking strategies: 1) **Specialization**. For each parent p whose only child is v (so that removing v leaves p a leaf, mirroring the specialization pattern), we take the child context at p and remove v . 2) **Abstraction**. For each chain $v_1 \rightarrow v_2 \rightarrow \text{Ch}(v_2)$, we take the intermediate context between v_1 and $\text{Ch}(v_2)$ and remove the intermediate node v_2 . 3) **Expansion**. For each parent p with at least two children, we take the sibling context under p and perform leave-one-out by removing one $v \in \text{Ch}(p)$; the siblings that remain in the context distinguish this case from specialization. This process yields a collection of pseudo-observations $\{z_i = (C_i, v_i)\}$, where C_i denotes the induced context and v_i denotes the masked target node.

4.2.2 Parameter Estimation via Weighted Method of Moments. We estimate parameters of the unit variables via a *weighted* Method-of-Moments (MoM), which matches the weighted empirical mean and variance of unit samples to the corresponding theoretical moments. The estimation proceeds in two stages (Alg. 2 and Alg. 3).

Stage I (Alg. 2): unit-sample recovery and debiasing. For each pseudo-observation z_i , this stage first computes the context-induced size and center bounds per dimension including the auxiliary information required for the ABS context (Alg. 2, Line 1–5), and then inverts the affine mapping to obtain unit samples in $[0, 1]$ (Line 6). For the EXP context, since the overlap penalty

Algorithm 3 Online Weighted MoM for Beta(α, β)**Require:** Stream of weighted unit samples $\{(u_j, \omega_j)\}$ with $u_j \in [0, 1]$.**Ensure:** MoM estimates $(\hat{\alpha}, \hat{\beta})$.

```

1: Maintain running statistics:  $\Omega = \sum_j \omega_j$ ,  $\Omega_{sq} = \sum_j \omega_j^2$ ,  $\mu_\omega$ , and  $M_{2,\omega}$ .
2: Initialize  $\Omega, \Omega_{sq}, \mu_\omega, M_{2,\omega} \leftarrow 0$ .
3: for each  $(u_j, \omega_j)$  do
4:    $\Omega^{new} \leftarrow \Omega + \omega_j$ ;  $\Delta \leftarrow u_j - \mu_\omega$ .
5:    $\mu_\omega^{new} \leftarrow \mu_\omega + \frac{\omega_j}{\Omega^{new}} \Delta$ .
6:    $M_{2,\omega} \leftarrow M_{2,\omega} + \omega_j \Delta (u_j - \mu_\omega^{new})$ .
7:    $\Omega_{sq} \leftarrow \Omega_{sq} + \omega_j^2$ ;  $\Omega \leftarrow \Omega^{new}$ ;  $\mu_\omega \leftarrow \mu_\omega^{new}$ .
8: end for
9:  $\hat{\nu}_\omega \leftarrow \frac{\Omega M_{2,\omega}}{\Omega^2 - \Omega_{sq}}$ ;  $\kappa \leftarrow \frac{\mu_\omega(1-\mu_\omega)}{\hat{\nu}_\omega} - 1$ .
10:  $\hat{\alpha} \leftarrow \mu_\omega \kappa$ ;  $\hat{\beta} \leftarrow (1 - \mu_\omega) \kappa$ .
11: return  $(\hat{\alpha}, \hat{\beta})$ .

```

biases observations, the inverse-probability weighting $\omega_i = \exp(O(B_i)/\tau_{sib})$ is applied to recover moments of the underlying base proposal (Lines 9–12).

Stage II (Alg. 3): online weighted moment updates. We stream the weighted unit samples and maintain running statistics— Ω (weight sum), Ω_{sq} (sum of squared weights), μ_ω (weighted mean), and $M_{2,\omega}$ (weighted second central moment)—via a weighted Welford update [60] (Alg. 3, Lines 1–7), and then compute $(\hat{\alpha}, \hat{\beta})$ in closed form (Lines 9–10). This yields incremental, up-to-date priors for each context and dimension without revisiting past data.

5 Cost-effective Evidence Retrieval

When a new iteration t begins and new research papers arrive, we leverage the hypothesized concepts derived in the previous iteration as *query candidates* for evidence retrieval. The key challenge is to derive an *optimal query plan*: decide which candidate structural refinement contexts C_h to explore and how much budget to allocate to each under a strict token budget. A naive Monte Carlo strategy will repeatedly sample candidate concepts from each hypothesis distribution $P(B \mid \mathcal{T}_{t-1})$, which leads to two key limitations. First, since hypotheses are inferred solely from $\mathcal{T}_{t-1}$, some may not be supported by the repository, wasting budget on irrelevant retrievals. Moreover, the strategy is both costly and high-variance: it requires many samples for stable behavior, and retrieval quality can be dominated by sampling noise. To address these issues, we propose a novel budget-aware utility-driven planner that anchors each candidate context to the repository and estimates its expected relevance utility and expected parsing cost, yielding an optimal budget-feasible query plan (Sec. 5.1). Then, we materialize this plan to retrieve a final set of papers and extract partial taxonomies from them. (Sec. 5.2).

5.1 Budget-aware Utility-driven Planner

5.1.1 Query Planning as an Optimization Problem. At iteration t , we consider the candidate structural refinement contexts extracted from the taxonomy at the previous timestamp $\mathbb{C}(\mathcal{T}_{t-1}) = \{C_h\}_{h=1}^{H_{t-1}}$, together with the newly arrived papers $\Delta\mathcal{P}_t$. The planner aims to derive a query-allocation policy Π_t over these contexts that jointly accounts for the following objectives and constraints:

Primary Objective: Utility Anchoring. The planner *anchors* each candidate context C_h by estimating its empirical evidence utility U_h , which captures how strongly the available papers can support the corresponding hypothesis. Under an allocation policy Π_t , the expected relevance gain is $\sum_{h=1}^{H_{t-1}} \Pi_{t,h} U_h$, which favors allocating more budget to contexts whose hypotheses are better supported by the available papers. We detail the utility function U later in Sec. 5.1.2.

Secondary Objective: Context Selection Prior. Beyond utility maximization, we impose the *context realization prior* that biases exploration toward contexts that are more likely to admit valid emerging concepts under the current taxonomy $\mathcal{T}_{t-1}$. Let $A_h \in \{0, 1\}$ indicate whether context C_h is selected under $\mathcal{T}_{t-1}$. We instantiate this preference via the prior allocation $\Pi_{t-1}^{\text{prior}} = P(A_h=1 \mid \mathcal{T}_{t-1})$, and regularize the planning policy Π_t toward Π_{t-1}^{prior} through $\text{KL}(\Pi_t \parallel \Pi_{t-1}^{\text{prior}})$.

Budget Constraint: Token-Feasible Planning. The per-iteration token budget K_t limits the *total* parsing cost incurred when reading related-work sections in iteration t . Since different contexts may retrieve papers of different lengths, we summarize the context-induced parsing burden by an *expected per-paper cost* $\bar{\ell}_h$. Under allocation Π_t , the expected cost per parsed paper is the mixture $\sum_{h=1}^{H_{t-1}} \Pi_{t,h} \bar{\ell}_h$. We first estimate a token-feasible parsing batch size by converting the budget into a maximum number of parsable new papers: $\bar{\ell}(\Delta\mathcal{P}_t) = \frac{1}{|\Delta\mathcal{P}_t|} \sum_{d \in \Delta\mathcal{P}_t} \ell(d)$, $M_t = \min\left\{|\Delta\mathcal{P}_t|, \left\lfloor \frac{K_t}{\bar{\ell}(\Delta\mathcal{P}_t)} \right\rfloor\right\}$. If we plan to parse M_t papers in this iteration, then the expected total token consumption becomes $M_t \sum_{h=1}^{H_{t-1}} \Pi_{t,h} \bar{\ell}_h$, and budget feasibility requires $M_t \sum_{h=1}^{H_{t-1}} \Pi_{t,h} \bar{\ell}_h \leq K_t$.

Optimization Formulation. Combining these components, we have the following constrained maximization problem:

$$\max_{\Pi_t} \sum_{h=1}^{H_{t-1}} \Pi_{t,h} U_h - \lambda_{\text{plan}} \text{KL}(\Pi_t \parallel \Pi_{t-1}^{\text{prior}}) \quad \text{s.t.} \quad M_t \sum_{h=1}^{H_{t-1}} \Pi_{t,h} \bar{\ell}_h \leq K_t, \quad (7)$$

where $\lambda_{\text{plan}} > 0$ controls the trade-off between maximizing empirical evidence gain and adhering to the context-realization prior.

5.1.2 Gaussian Proxy Utility and Cost Estimation. The key challenge in Eq. 7 is to instantiate, for each candidate context C_h , (i) an evidence utility U_h and (ii) an expected per-paper parsing cost $\bar{\ell}_h$, in a way that is both principled and computationally tractable.

Gaussian-Proxy Utility. Recall that each context C_h induces a hypothesis distribution over candidate boxes in the learned geometric space, $\pi_h(B) \triangleq P(B \mid C_h, \mathcal{T}_{t-1})$. Drawing $B \sim \pi_h$ and mapping it through the learned box-to-word function $\mathbf{x} = \Phi_{\text{B2W}}(B)$ induces a (generally intractable) distribution over query embeddings. We score a context by its *expected* relevance to the newly arrived papers:

$$U_h \triangleq \mathbb{E}_{d \sim \hat{p}_t} [r_h(d)], \quad r_h(d) \triangleq \mathbb{E}_{B \sim \pi_h} [k(\Phi_{\text{B2W}}(B), \mathbf{s}(d))], \quad (8)$$

where $\hat{p}_t$ is uniform over $\Delta\mathcal{P}_t$, $\mathbf{s}(d)$ denotes the embedding of paper d , and $k(\cdot, \cdot)$ is a similarity kernel in the word embedding space.

We adopt three design choices to make Eq. 8 tractable and scalable. (i) *Gaussian proxy.* We approximate the induced query distribution by a Gaussian, $\mathbf{x}_h \approx \mathcal{N}(\mu_h, \Sigma_h)$, with (μ_h, Σ_h) estimated via a first-order Delta approximation (i.e., moment propagation through $\Phi_{\text{B2W}}(\cdot)$) [46]. This yields a compact, low-variance representation of the context-induced uncertainty and enables closed-form scoring with common kernels. (ii) *RBF kernel.* We use an RBF kernel $k(\cdot, \cdot)$ as a smooth, bounded, distance-based relevance measure; under a Gaussian proxy, its expectation admits a closed form [6], avoiding expensive per-context Monte Carlo over B . (iii) *Sampling.* Computing U_h exactly requires scanning all $|\Delta\mathcal{P}_t|$ papers. We instead estimate the outer expectation using m i.i.d. samples, achieving sublinear runtime with a tunable accuracy–efficiency trade-off.

Query-specific Cost Estimation. To incorporate token costs, we treat $r_h(d)$ as a soft relevance weight and define a query-induced distribution over papers $p(d \mid h) \propto \hat{p}_t(d) r_h(d)$. Let $\ell(d)$ denote the token length of the related-work section of paper d . We define the expected per-paper parsing

cost induced by context C_h as $\bar{\ell}_h = \mathbb{E}_{d \sim p(d|h)}[\ell(d)] = \mathbb{E}_{d \sim \hat{p}_t}[r_h(d) \ell(d)] / \mathbb{E}_{d \sim \hat{p}_t}[r_h(d)]$. Using the same sampled papers $\{d_j\}_{j=1}^m$, we estimate $\bar{\ell}_h$ efficiently by $\hat{\ell}_h = \sum_{j=1}^m r_h(d_j) \ell(d_j) / \sum_{j=1}^m r_h(d_j)$.

Based on the above, Eq. 7 admits a unique closed-form solution, summarized below, with a full proof provided in the technical report [26].

PROPOSITION 3 (OPTIMAL BUDGET-AWARE KL-REGULARIZED POLICY). *For Eq. 7, there exists $\lambda^* \geq 0$ such that the unique maximizer satisfies $\Pi_{t,h}^* = \frac{\Pi_{t-1,h}^{\text{prior}} \exp((U_h - \lambda^* \bar{\ell}_h) / \lambda_{\text{plan}})}{Z(\lambda^*)}$, where $Z(\lambda) = \sum_{h'=1}^{H_{t-1}} \Pi_{t-1,h'}^{\text{prior}} \exp((U_{h'} - \lambda \bar{\ell}_{h'}) / \lambda_{\text{plan}})$. The optimal λ^* can be found by one-dimensional bisection to satisfy the budget constraint (with equality when active): $M_t \sum_{h=1}^{H_{t-1}} \Pi_{t,h}^* \bar{\ell}_h \leq K_t$.*

5.2 Plan Materialization and Paper Selection

With the optimal exploration plan Π_t established, we proceed to materialize and execute it. This involves determining the number of hypothetical boxes to sample per direction, transforming them into query embeddings to initialize the search, and selecting the papers that most effectively refine the taxonomy.

Exploration Strategy Materialization. We convert the fractional plan Π_t^* into integer per-context counts by allocating $\tilde{n}_h = \Pi_{t,h}^* M_t$ and choosing integers $n_h \in \mathbb{Z}_{\geq 0}$ with $\sum_{h=1}^{H_{t-1}} n_h = M_t$, obtained via unbiased rounding with a largest-remainder tie-break.

Query Sampling and Retrieval. For each context C_h , draw n_h box samples B and map each to a query embedding $\mathbf{x} = \Phi_{B2W}(B)$, forming the query set $\mathcal{X}_t$. For each $\mathbf{x} \in \mathcal{X}_t$, retrieve its top- L nearest papers from the *unparsed index* $\mathcal{R}_t$ —the embedding index over papers that have arrived by iteration t but have not yet been parsed—and let $\mathcal{P}_t^{\text{cand}} \subseteq \mathcal{R}_t$ be the deduplicated union of these candidates. Let $k(\cdot, \cdot) \in [0, 1]$ be the similarity and define $a(d, \mathbf{x}) = k(s(d), \mathbf{x})$ for $d \in \mathcal{P}_t^{\text{cand}}$, $\mathbf{x} \in \mathcal{X}_t$. **Submodular objective.** We select a subset $S \subseteq \mathcal{P}_t^{\text{cand}}$ by solving the knapsack-constrained facility-location problem:

$$\max_{S \subseteq \mathcal{P}_t^{\text{cand}}} F(S) = \sum_{\mathbf{x} \in \mathcal{X}_t} \max_{d \in S} a(d, \mathbf{x}) \quad \text{s.t.} \quad \sum_{d \in S} \ell(d) \leq K_t, \quad (9)$$

Eq. 9 is a monotone submodular maximization under a single knapsack constraint, for which the proof is provided in the technical report [26]. Similarly, we solve it using a standard cost-aware lazy-greedy procedure with partial enumeration. Note that Eq. 9 does not introduce an additional budget: it enforces the same per-iteration cap K_t on the *realized* selection S , whereas the constraint in Eq. 7 is its planning-stage counterpart that bounds the *expected* parsing cost when shaping the allocation Π_t .

LLM-guided Partial Taxonomy Extraction. The extraction process first employs regular-expression-based matching to isolate the *Related Work* sections. Subsequently, GIST prompts an LLM to extract partial taxonomies. Detailed regex matching process and prompt specifications are provided in our technical report [26].

6 Incremental Semantic Indexing

At iteration t , we maintain a taxonomy-induced inverted index $\mathcal{I}_t$ to support efficient downstream retrieval. Newly arrived papers $\Delta \mathcal{P}_t$ are indexed immediately by assigning them to relevant concept nodes via embedding similarity. For previously indexed papers, we condition updates on the refinement type to avoid full-corpus re-indexing: (1) *Expansion*. Introducing a new node v requires no historical re-indexing: $\mathcal{I}_t[v]$ is initialized as empty and populated only by matches from $\Delta \mathcal{P}_t$ and future streams. (2) *Specialization*. Specializing a node u into children $\mathcal{S}_u$ requires redistributing only the affected posting list $\mathcal{I}_{t-1}[u]$ across $\mathcal{S}_u$, avoiding any full-repository scan. (3) *Abstraction*.

Creating a new parent node a over an existing child set S_a requires only posting-list aggregation: $\mathcal{I}_t[a] \leftarrow \bigcup_{c \in S_a} \mathcal{I}_{t-1}[c]$. This is a local operation requiring no historical re-scoring.

Supporting Paper Retrieval. Given a user query q , we route it to the taxonomy by selecting the top- k semantically similar concepts $\mathcal{U}_q$. We then form a compact candidate set by unioning the posting lists of these concepts and their descendants: $\mathcal{P}_t(q) = \bigcup_{u \in \mathcal{U}_q} (\mathcal{I}_t[u] \cup \bigcup_{v \in \text{Desc}(u)} \mathcal{I}_t[v])$. Finally, papers in $\mathcal{P}_t(q)$ are ranked using standard retrieval models (e.g., BM25 or dense retrieval), enabling efficient candidate generation without repetitive full-corpus scanning.

7 Experiments

In this section, we conduct experiments to evaluate: 1) the effectiveness and efficiency of our proposed method on the taxonomy maintenance in the wild task (Sec. 7.1); 2) the effectiveness and efficiency of the semantic indexing based on the generated taxonomy (Sec. 7.2); 3) the sensitivity of key hyper-parameters (Sec. 7.3); 4) the ablation study (Sec. 7.4); 5) diagnostic studies on extraction robustness, numerical stability, and predictive precision (Sec. 7.5); and 6) a qualitative case study (Sec. 7.6).

7.1 Evaluation of Taxonomy Maintenance

Benchmarks. We simulate an “in-the-wild” setting using the Computer Science (CS) subset of the arXiv corpus [19], which comprises over 270K LaTeXML-converted HTML documents. To establish ground truth, we curate a benchmark of 12 survey papers spanning diverse CS domains. We select survey papers that are either published in top-tier venues or highly cited, manually extracting and verifying their author-curated taxonomies. To capture the evolutionary dynamics of each topic, we assign distinct start times to each domain, ensuring that the topic has reached a sufficient level of development prior to the simulation. We then partition the remaining timeline into seven equal intervals $t = 1, \dots, 7$, where sequential paper arrivals trigger iterative taxonomy updates; the first interval supplies the initial corpus snapshot ($\mathcal{P}_0$ in Definition 1), so the initial construction (*Init*) is evaluated at $t=1$. To avoid data leakage, we remove the benchmark survey papers from the arXiv corpus. Benchmark statistics are reported in Table 1, and further details on the survey papers are provided in the technical report [26].

Remark. We adopt a single highly authoritative survey per domain, selected from top-tier venues or with high citation counts, rather than aggregating across multiple surveys. Aggregation would require either redefining the evaluation protocol against an ensemble of ground truths or manually reconciling disparate hierarchies into a unified gold standard, both of which would introduce substantial curator-induced bias. We acknowledge that survey-derived taxonomies favor cleaner, consensus-driven hierarchies over fully organic ones; however, they capture the structured knowledge that downstream users and systems actually rely on, and purely organic benchmarks at scale are notoriously difficult to obtain. We leave the construction of such benchmarks as a promising direction for future work.

Baselines. To comprehensively evaluate GIST in the “in-the-wild” setting, we design baselines through a modular pipeline that factorizes: (i) a *taxonomy construction backbone*, (ii) an *updating mechanism* for sequential arrivals, and (iii) a *paper search strategy* for selecting evidence papers from the newly arrived papers. We denote each baseline instance as $\text{BACKBONE}_{\text{UPDATE}, \text{SEARCH}}$.

Backbones. We instantiate four representative baselines spanning the major taxonomy-construction paradigms: 1) TaxoGen [64], a point-embedding clustering method that iteratively partitions a topic term graph; 2) HyperExpan [39], a hyperbolic-embedding expansion method that places new concepts in negatively-curved space and attaches them via geometric distances; 3) TaxoAlign [32] builds taxonomies via a three-stage pipeline: locating topic-related knowledge slices, aggregating these

Table 1. Statistics of the benchmark.

Domain	Research Topic (Input)	Levels	Nodes
DB	Schema Design & Tuning (SDT)	4	24
	LLM-based Text-to-SQL (LLM2SQL)	5	37
	HTAP Databases (HTAP)	2	28
	Graph ANN Vector Search (ANN-VS)	3	21
	Distributed Graph Algorithms (DGA)	3	31
SE	Auto Program Repair (APR)	3	14
	AI for Software Testing (AI4ST)	3	48
ML	Machine Unlearning (MU)	3	16
	Fairness in ML (FairML)	3	19
IR	Multimodal Recommenders (MRS)	3	29
HCI	Industrial Extended Reality (I-XR)	3	58
Robotics	Multi-Robot Task Allocation (MRTA)	5	23

Table 2. Taxonomy quality and efficiency comparison across all compared methods. Each row U+A under a backbone denotes the variant $\text{BACKBONE}_{U,A}$, where $U \in \{S, I\}$ is the updating mechanism and $A \in \{K, C, H\}$ the evidence acquisition strategy. F1 metrics are reported as $\text{mean}_{\pm\text{std}}$ over 5 random seeds. $\dagger$ denotes a statistically significant improvement of our method $\text{GIST}_{I,H}$ over *all baselines* under paired t -tests at $p < 0.01$.

	Node Soft F1			Edge Soft F1			Runtime	Tokens	Cost
	Init	Final	Δ/ep	Init	Final	Δ/ep	(min)	(k)	(\$)
<i>TaxoGen</i>									
S+K	24.8 $\pm$ 0.3	31.2 $\pm$ 0.4	+1.07 $\pm$ 0.05	19.7 $\pm$ 0.2	24.7 $\pm$ 0.3	+0.83 $\pm$ 0.04	76.4	–	–
S+C	24.8 $\pm$ 0.3	33.8 $\pm$ 0.5	+1.50 $\pm$ 0.06	19.7 $\pm$ 0.2	27.1 $\pm$ 0.4	+1.23 $\pm$ 0.05	77.8	–	–
I+K	24.8 $\pm$ 0.3	29.4 $\pm$ 0.3	+0.77 $\pm$ 0.04	19.7 $\pm$ 0.2	23.6 $\pm$ 0.3	+0.65 $\pm$ 0.04	22.7	–	–
I+C	24.8 $\pm$ 0.3	31.7 $\pm$ 0.4	+1.15 $\pm$ 0.05	19.7 $\pm$ 0.2	25.4 $\pm$ 0.3	+0.95 $\pm$ 0.04	23.6	–	–
<i>HyperExpan</i>									
S+K	32.8 $\pm$ 0.4	40.7 $\pm$ 0.5	+1.32 $\pm$ 0.06	26.7 $\pm$ 0.3	33.8 $\pm$ 0.4	+1.18 $\pm$ 0.05	91.2	–	–
S+C	32.8 $\pm$ 0.4	43.6 $\pm$ 0.6	+1.80 $\pm$ 0.07	26.7 $\pm$ 0.3	36.4 $\pm$ 0.5	+1.62 $\pm$ 0.06	92.7	–	–
I+K	32.8 $\pm$ 0.4	38.7 $\pm$ 0.4	+0.98 $\pm$ 0.05	26.7 $\pm$ 0.3	32.4 $\pm$ 0.3	+0.95 $\pm$ 0.04	27.1	–	–
I+C	32.8 $\pm$ 0.4	41.2 $\pm$ 0.5	+1.40 $\pm$ 0.06	26.7 $\pm$ 0.3	34.8 $\pm$ 0.4	+1.35 $\pm$ 0.05	28.4	–	–
<i>TaxoAlign</i>									
S+K	38.6 $\pm$ 0.7	56.2 $\pm$ 0.8	+2.93 $\pm$ 0.13	32.3 $\pm$ 0.6	47.8 $\pm$ 0.7	+2.58 $\pm$ 0.11	451.8	2438	7.31
S+C	38.6 $\pm$ 0.7	60.4 $\pm$ 0.9	+3.63 $\pm$ 0.14	32.3 $\pm$ 0.6	51.3 $\pm$ 0.8	+3.17 $\pm$ 0.12	458.4	2552	7.64
I+K	38.6 $\pm$ 0.7	46.4 $\pm$ 0.6	+1.30 $\pm$ 0.10	32.3 $\pm$ 0.6	39.7 $\pm$ 0.5	+1.23 $\pm$ 0.09	113.4	871	2.62
I+C	38.6 $\pm$ 0.7	49.8 $\pm$ 0.7	+1.87 $\pm$ 0.11	32.3 $\pm$ 0.6	42.6 $\pm$ 0.6	+1.72 $\pm$ 0.10	117.2	928	2.78
<i>TaxoAdapt</i>									
S+K	42.8 $\pm$ 0.8	61.7 $\pm$ 0.9	+3.15 $\pm$ 0.13	35.8 $\pm$ 0.7	53.6 $\pm$ 0.8	+2.97 $\pm$ 0.12	384.6	3756	11.27
S+C	42.8 $\pm$ 0.8	66.2 $\pm$ 0.9	+3.90 $\pm$ 0.14	35.8 $\pm$ 0.7	57.8 $\pm$ 0.8	+3.67 $\pm$ 0.12	391.8	3895	11.69
I+K	42.8 $\pm$ 0.8	52.4 $\pm$ 0.7	+1.60 $\pm$ 0.10	35.8 $\pm$ 0.7	44.1 $\pm$ 0.6	+1.38 $\pm$ 0.09	114.3	947	2.84
I+C	42.8 $\pm$ 0.8	56.7 $\pm$ 0.8	+2.32 $\pm$ 0.12	35.8 $\pm$ 0.7	47.6 $\pm$ 0.7	+1.97 $\pm$ 0.10	117.3	1003	3.01
<i>GIST</i>									
S+K	47.5 $\pm$ 0.5	58.6 $\pm$ 0.5	+1.85 $\pm$ 0.07	40.8 $\pm$ 0.4	50.4 $\pm$ 0.4	+1.60 $\pm$ 0.06	113.4	641	1.91
S+C	47.5 $\pm$ 0.5	62.3 $\pm$ 0.6	+2.47 $\pm$ 0.08	40.8 $\pm$ 0.4	53.7 $\pm$ 0.5	+2.15 $\pm$ 0.07	114.6	682	2.04
S+H	47.5 $\pm$ 0.5	76.0 $\pm$ 0.6	+ 4.75 $\pm$ 0.10	40.8 $\pm$ 0.4	67.6 $\pm$ 0.5	+ 4.47 $\pm$ 0.09	117.3	713	2.14
I+K	47.5 $\pm$ 0.5	55.3 $\pm$ 0.4	+1.30 $\pm$ 0.06	40.8 $\pm$ 0.4	48.2 $\pm$ 0.4	+1.23 $\pm$ 0.06	35.8	421	1.26
I+C	47.5 $\pm$ 0.5	60.5 $\pm$ 0.5	+2.17 $\pm$ 0.07	40.8 $\pm$ 0.4	53.1 $\pm$ 0.4	+2.05 $\pm$ 0.06	36.7	463	1.38
I+H	47.5 $\dagger$ $\pm$ 0.5	73.5 $\dagger$ $\pm$ 0.5	+4.33 $\dagger$ $\pm$ 0.10	40.8 $\dagger$ $\pm$ 0.4	65.4 $\dagger$ $\pm$ 0.4	+4.10 $\dagger$ $\pm$ 0.09	37.5	498	1.49

slices to form an initial taxonomy, and performing LLM-based refinement; and 4) TaxoAdapt [29] starts with an LLM-generated taxonomy and iteratively *grounds* it to the collected documents. It employs top-down classification and selective expansion, driven by corpus density and unmapped signals, to generate a final corpus-aligned taxonomy. To provide initial concept seeds for TaxoGen and HyperExpan, we use BERTopic [21] to extract them from the input corpus.

Updating mechanisms. We instantiate each backbone under two standard updating mechanisms: (1) *Scratch (S)*: Ignores the previous state and leverages the full cumulative corpus up to interval t .

For the baselines (TaxoAlign and TaxoAdapt), this involves reconstructing the taxonomy entirely from the cumulative set of papers. For GIST, this entails retraining the mapping-model parameters from scratch using all accumulated data. (2) *Incremental (I)*: Updates the taxonomy $\mathcal{T}_{t-1} \rightarrow \mathcal{T}_t$ using only the newly arrived papers in interval t . For the baselines, we implement a generate-and-merge approach: (i) running the backbone on the new batch to generate a “delta” taxonomy $\Delta\mathcal{T}_t$; and (ii) merging $\Delta\mathcal{T}_t$ with $\mathcal{T}_{t-1}$ via concept alignment and edge union. For GIST, we employ the proposed incremental training strategy (Sec. 3.3).

Evidence Acquisition strategies. We consider three strategies: (1) *Topic-keyword search (K)*: queries using the initial topic keywords as the query at each update; (2) *Concept search (C)*: queries using concepts in the current taxonomy; and (3) *Hypothesized-concept search (H)*: queries using the hypothesized concepts (specific to our method). To ensure a fair comparison across methods that may rely on different parts of each paper, we cap the amount of raw text processed from newly retrieved papers at every update. Specifically, the total evidence budget per iteration is limited to 400,000 tokens in the first iteration and 100,000 tokens thereafter, resulting in a 1,000,000-token budget per topic. We use GPT-5-mini for all methods. Note that this budget caps only the *evidence text supplied as initial LLM input* (i.e., the raw paper content parsed per update); the total token counts reported in Table 2 additionally include each method’s own prompting scaffolds, intermediate reasoning calls, and generated outputs, and can therefore exceed the evidence budget.

Evaluation Metrics. We evaluate three dimensions: effectiveness, efficiency, and monetary cost. For *effectiveness*, we report the standard *Node Soft F1* and *Edge Soft F1* metrics [32]. *Node Soft F1* quantifies concept-level agreement via semantic maximum-weight bipartite matching, while *Edge Soft F1* measures agreement on parent–child relations, weighted by the similarity of their endpoints. At each iteration we evaluate against the final ground truth ($t = 7$). *Efficiency* is measured by total runtime across all iterations, and *monetary cost* by total API expense.

Implementation and Hyper-parameters. We implement GIST in PyTorch and run all experiments on a Linux server equipped with an NVIDIA RTX 4090 GPU. For geometric representation, we set the box-embedding dimension to $d = 12$. In the geometric-guided taxonomy integration module, we set the reliability weight to $\lambda_{\text{rel}} = 0.3$ and the novelty weight to $\lambda_{\text{nov}} = 0.2$. For the *Cost-effective Evidence Retrieval* module, we set the planning weight to $\lambda_{\text{plan}} = 0.2$. For Gaussian proxy utility estimation, we employ a sample size of $m = 1000$, which keeps the sampling error of the estimated utilities negligible in practice. The source code is publicly available in [20].

Results. The comparison results are presented in Table 2. We make the following observations:

- **Overall.** The GIST series achieves the best taxonomy quality across every quality metric in the grid. The only baselines with lower runtime or monetary cost are the two embedding-only methods (TaxoGen and HyperExpan), but their taxonomy quality is far below competitive levels (Final Node F1 below 44%). Among methods with competitive quality, the proposed efficient variant GIST_{I,H} attains the best cost–quality trade-off: an 11.0% / 13.1% improvement in Final Node / Edge F1 over the strongest LLM baseline TaxoAdapt_{S,C}, statistically significant at $p < 0.01$, while requiring only 9.6% of its runtime and 12.7% of its monetary cost. Note that HyperExpan and TaxoGen do not involve LLM processing and therefore incur no API cost and run efficiently. However, their performance lags substantially behind LLM-based methods, reflecting the inherent trade-off between cost and taxonomy quality.
- **Backbone.** Because the *Init* score is measured at $t=1$ before any updating mechanism or evidence acquisition strategy takes effect, it isolates the contribution of the backbone itself. The four baseline backbones cluster into three quality tiers reflecting their representational power. The point-embedding backbone (TaxoGen) cannot model the asymmetric containment relations required for is-a hierarchies and remains below 35% *Final Node Soft F1* under any update or search

configuration. The hyperbolic backbone (HyperExpan) modestly improves this via negatively-curved geometry but still tops out around 44%. The LLM-driven backbones (TaxoAlign and TaxoAdapt) reach 60–66% by leveraging LLM reasoning, yet exhibit the high token cost and structural hallucination characteristic of free-form generation. GIST’s box-embedding backbone, in contrast, attains the highest *Init* (47.5 Node, 40.8 Edge) – evidence that combining expert-curated partial taxonomies with a box-embedding space, which natively encodes containment, yields a stronger structural prior than any of the alternative backbones.

- **Updating Mechanism.** An effective updating mechanism is essential for the task of taxonomy maintenance in the wild. Without one (i.e., retraining from scratch at each interval), runtime grows substantially for every method, and monetary cost grows dramatically for the LLM-intensive backbones TaxoAdapt and TaxoAlign. With our dedicated updating mechanism and novelty-aware coreset selection, GIST’s incremental variant GIST_{I,H} retains 96.7% of the fully retrained GIST_{S,H}’s quality (Final Node F1: 73.5 vs. 76.0; Final Edge F1: 65.4 vs. 67.6, a 3.3% relative drop) while cutting runtime from 117.3 to 37.5 minutes (3.1× speedup) and monetary cost from \$2.14 to \$1.49 (30.4% reduction). This demonstrates that scratch retraining is largely unnecessary when guided by the hybrid evidence strategy. By contrast, naive incremental updating in TaxoAdapt and TaxoAlign reduces runtime by 70.1% / 74.4% and monetary cost by 74.2% / 63.6%, but at the expense of a 14.4% / 17.5% relative quality drop. The gap between GIST’s 3.3% drop and the 14–18% drop suffered by these baselines shows that simple merging is insufficient for preserving structural integrity, whereas GIST’s coreset-based mechanism keeps nearly all of the full-retraining quality at a fraction of the cost.
- **Evidence Acquisition Strategy.** The evidence acquisition strategy via *hypothesized concepts* drives the fastest taxonomy growth per epoch. It enables more targeted information acquisition and avoids retrieving redundant papers that contribute only to already-known concepts. Compared with leveraging existing *concepts* and *input search topics*, GIST’s per-epoch gain (Δ/ep) is 2.0× and 3.3× higher on Node Soft F1 (GIST_{I,H}: +4.33, GIST_{I,C}: +2.17, GIST_{I,K}: +1.30), with the same 2.0× / 3.3× gap on Edge (+4.10 vs. +2.05 / +1.23). Querying with *concepts* from the current taxonomy is also more likely to surface papers capable of expanding the taxonomy than querying with the original research topics alone. Conversely, relying on input keywords yields the smallest per-epoch progress among LLM-based incremental variants.

7.2 Effectiveness in Downstream Literature Search

We evaluate the effectiveness and efficiency of the proposed semantic indexing utilizing the maintained taxonomies. We reuse the arXiv corpus and the benchmark, with the following modifications.

Query and Ground-Truth Construction. For each topic, we treat each concept node as an information need and generate two query types: (i) *keyword queries* using the concept name, and (ii) *natural-language queries* by prompting an LLM to produce a concise, user-style search query from the concept name and description; we generate 5 queries of each type per topic. We derive ground truth from the corresponding survey paper: for each concept query, the relevant set comprises papers explicitly assigned to that concept/category by the survey. We ensure these papers are present in the repository by adding any missing ones to the arXiv corpus.

Evaluation Metrics. We report standard ranked-retrieval metrics averaged over topics, including $n\text{DCG}@10$ and MRR , under three retrieval paradigms: *BM25*, *Dense*, and *Hybrid*. To quantify indexing efficiency, we additionally report: (i) *index size* of the taxonomy-induced inverted index at $t=7$ (posting lists + metadata), (ii) *cumulative index build/update time* across all iterations, counting only the time to materialize and maintain the index given the constructed taxonomy (i.e., assigning papers to concept nodes and updating posting lists), and (iii) *query latency*.

Table 3. Effectiveness comparison on paper search tasks. GIST and GIST* maintain the same taxonomy and produce identical indexes, hence identical retrieval effectiveness; The symbols $\dagger$ and $\ddagger$ indicate that GIST significantly outperforms all non-GIST baselines under paired t -tests at $p < 0.01$ and $p < 0.05$, respectively.

Method	BM25		Dense		Hybrid	
	nDCG@10	MRR	nDCG@10	MRR	nDCG@10	MRR
Flat-Ret	0.374	0.333	0.402	0.356	0.438	0.378
TaxoAlign	0.382	0.339	0.409	0.363	0.448	0.387
TaxoAdapt	0.397	0.350	0.423	0.373	0.463	0.401
GIST / GIST*	0.409$\dagger$	0.360$\dagger$	0.439$\dagger$	0.386$\dagger$	0.479$\ddagger$	0.420$\ddagger$

Table 4. Evaluation of indexing efficiency. Index (MB), Build (min), and Per-paper (ms) count only the *additional* taxonomy-induced index (posting lists + metadata) built on top of the base BM25/dense retrieval indexes shared by all methods.

Method	Index (MB)	Build (min)	Per-paper (ms)	Query (ms)
Flat-Ret	0.00	0.0	0.0	58.0
TaxoAlign	2.76	6.5	82.3	45.7
TaxoAdapt	3.15	7.4	91.8	41.9
GIST*	2.64	6.8	86.5	46.7
GIST	2.64	3.4	41.2	45.4

Baselines. We compare GIST against taxonomies produced by TaxoAlign and TaxoAdapt (using their best-performing variants), and a flat retrieval baseline (*Flat-Ret*) that performs retrieval directly over the corpus without any taxonomy-induced semantic index. To isolate the benefit of incremental indexing, we further include GIST*, which uses the same taxonomy as GIST but *rebuilds* the semantic index at each iteration via full-corpus scanning.

Results. Table 3 reports downstream search effectiveness. We observe that all taxonomy-indexed methods consistently outperform the *Flat-Ret* baseline across BM25, Dense, and Hybrid retrieval, validating that a structured taxonomy improves semantic access to scholarly corpora. Moreover, GIST achieves the best effectiveness overall. Compared with the strongest baseline TaxoAdapt, GIST yields consistent gains; averaging nDCG@10 and MRR, it improves performance by 2.9%, 3.6%, and 4.1% under BM25, Dense, and Hybrid, respectively. Table 4 reports indexing efficiency under two complementary views: cumulative *build time* across the 7 timestamps, and the *per-paper marginal cost* that characterizes streaming maintenance. GIST halves the cumulative build time of GIST* (full-scan indexing, 3.4 vs. 6.8 minutes) while preserving the same index size (2.64 MB), comparable query latency (45.4 vs. 46.7 ms), and identical retrieval effectiveness. Decomposed at the per-paper level, the marginal indexing cost of GIST is only 41.2 ms, roughly 2 $\times$ faster than GIST* (86.5 ms) and the rebuild-from-scratch baselines (TaxoAlign and TaxoAdapt at 82.3–91.8 ms). Together, these results demonstrate that the induced taxonomy can serve as an accurate index structure at light maintenance overhead.

7.3 Hyper-parameter Study

We analyze the sensitivity of GIST to the key coefficients λ_{rel} (Eq. 1), λ_{nov} (Eq. 3), and λ_{plan} (Eq. 7), as well as the box dimension d , reporting the final Node and Edge Soft F1 of GIST averaged over the 12 benchmark topics (Fig. 5). We make the following observations: 1) The reliability weight λ_{rel} balances concept coverage against evidence quality: performance peaks at $\lambda_{\text{rel}}=0.3$ and stays close to the peak within $[0.2, 0.5]$, while values toward either extreme over-emphasize one side of the trade-off—coverage-only selection ($\lambda_{\text{rel}}=0$) admits noisy relations, whereas overly reliability-heavy selection over-prunes coverage—and degrade taxonomy quality. 2) The novelty weight λ_{nov} controls the trade-off between loss preservation and diversity in coreset selection: a moderate

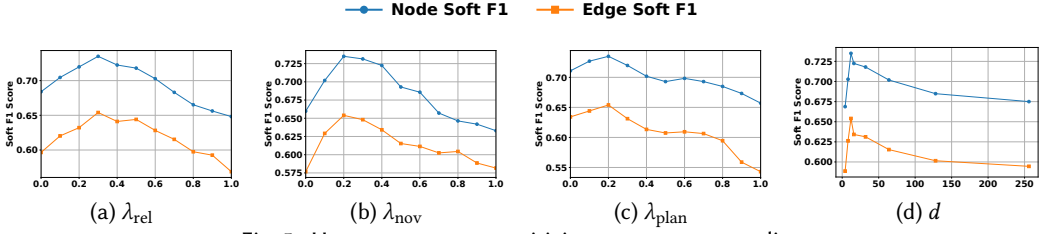


Fig. 5. Hyper-parameter sensitivity on taxonomy quality.

setting ($\lambda_{nov}=0.2$) works best. Disabling novelty entirely causes the sharpest drop in Fig. 5(b), as redundant coresets waste the update budget, whereas larger values sacrifice loss preservation for novelty and reduce coherence. 3) The planning weight λ_{plan} balances the context-realization prior against utility-driven exploitation: a moderate value ($\lambda_{plan}=0.2$) is optimal. Smaller values cost only mildly, concentrating the budget myopically on currently high-utility contexts, while larger values over-regularize toward the prior and miss emerging gaps, degrading more sharply as the prior dominates. 4) The box dimension d determines geometric expressiveness: small d underfits the hierarchical structure, performance peaks around $d=12$, and larger dimensions gradually *degrade* quality—consistent with overfitting the limited self-supervised signals—while raising computational cost, justifying our default of $d=12$.

7.4 Ablation Study

Beyond the configuration-level ablation analysis in Sec. 7.1, we conduct two finer-grained ablations that isolate individual design choices: the graph-selection objective (Table 5) and the budget-aware planner objective (Table 6). (1) *For the graph-selection objective*, coverage-only selection ($\lambda_{rel}=0$) admits noisy partial taxonomies and reduces Final Node F1 by 8.2% relative to the full model, whereas reliability-only selection (removing f_{cov}) over-prunes the candidate graph and reduces it by 5.6%; since f_{cov} optimizes for recall while f_{rel} optimizes for precision, the two terms are anti-correlated by design, and only their joint optimization recovers the reported 73.5/65.4 Final Node/Edge F1. (2) *For the planner objective*, utility-only planning increases token consumption by 186% (to 1426k) while delivering a Node F1 that is 9.4% lower than the full model; adding the prior Π^{prior} alone yields a 2.7% relative improvement in Node F1 over utility-only planning but leaves token usage unconstrained, whereas adding the budget constraint alone limits tokens to 542k and yields a 4.8% relative improvement. Only the complete three-term formulation simultaneously attains high quality (73.5% Node F1) and tight cost (498k tokens), confirming that the five components— f_{cov} , f_{rel} , U_h , Π^{prior} , and the budget constraint—are jointly indispensable.

7.5 Diagnostic Studies

Robustness of Related Work Detection and Extraction. We verify two properties of the partial taxonomy extraction pipeline: (i) the coverage of the regex-based detector, and (ii) the accuracy of the LLM-based extractor in recovering the author-curated partial taxonomy from the detected sections. We randomly sample 40 papers from the arXiv benchmark and manually annotate each with its *ground-truth Related Work span* and *ground-truth partial taxonomy*, against which we report detector *coverage* and *false-positive rate*, as well as extractor Node Soft F1 and Edge Soft F1. The regex detector achieves 95% coverage at a 0% false-positive rate; the missed 5% corresponds to 2 papers that contain no explicit Related Work section and are skipped entirely. This confirms that authors overwhelmingly organize related work in a dedicated section with an identifiable heading—a regularity the lightweight detector reliably exploits. The LLM extractor attains 94.7% Node Soft F1 and 93.5% Edge Soft F1 on the detected sections, indicating that the recovered partial

Table 5. Ablation on the reliability–coverage decomposition.

Variant	Node F1		Edge F1	
	Init	Final	Init	Final
f_{cov} only ($\lambda_{\text{rel}}=0$)	41.7	67.5	35.7	58.9
f_{rel} only (no coverage)	43.4	69.4	37.3	60.8
Full (GIST)	47.5	73.5	40.8	65.4

Table 6. Ablation on the budget-aware planner objective.

Variant	Final Node F1	Tokens (k)	Cost (\$)
U_h only	66.6	1426	4.27
$U_h + \Pi^{\text{prior}}$	68.4	1403	4.20
$U_h + \text{budget}$	69.8	542	1.62
Full (GIST)	73.5	498	1.49

taxonomies closely align with the manually annotated ground-truth structures and introduce only limited extraction noise into the subsequent taxonomy-integration stage.

Numerical Stability under Deep Hierarchies. We verify whether box volumes shrink sharply or remain numerically well-behaved as taxonomy depth grows. To this end, we measure the smallest learned box volume and the smallest per-dimension side length across all nodes on two topics: the LLM-based Text-to-SQL taxonomy (5 levels), one of the deepest topics in our benchmark, and an exceptionally deep 11-level taxonomy derived from a broad topic query of *Machine learning*. To isolate the effect of the volume regularization, we compare two variants of the bidirectional mapping model under otherwise identical hyperparameters: *w/o Reg.*, which removes the regularization term, and *w/ Reg.*, our full model that penalizes any box whose volume falls below the threshold $V_{\min}$. As a result, without regularization both quantities collapse toward the FP32 underflow limit, and training diverges with NaN gradients on the deeper topic. With regularization, the smallest volumes remain at 1.2×10^{-4} and 3.5×10^{-5} —roughly 33–34 orders of magnitude above the FP32 underflow limit ($\approx 10^{-38}$)—while the smallest side lengths stay at 0.37 and 0.26, indicating that inner boxes retain a meaningful spatial extent and the model converges stably in both settings.

Predictive Concept Precision. To verify whether the void volumes identified by our concept emergence model correspond to plausible emergence regions, we measure *Predictive Concept Precision*. For each timestamp transition $t \rightarrow t+1$, we collect *all* hypothesized concepts used for retrieval queries under each refinement context in $\mathcal{T}_t$, identify *all* newly emerging concepts at $t+1$, and compute the fraction of hypothesized concepts that are matched by at least one newly emerging concept under the soft-matching criterion. We report precision rather than recall because ground-truth concept sets at intermediate timestamps are not available in our benchmark, which provides ground-truth taxonomies only at the final timestamp. Averaged over six transitions and 12 benchmark topics, GIST achieves 54.2% precision. The precision is also relatively consistent across the three refinement contexts, with 48.5%, 55.4%, and 58.7% for Specialization, Abstraction, and Expansion, respectively. These results suggest that the geometric gaps provide a useful soft prior for identifying plausible emergence regions. Unrecovered concepts are handled by the budget-aware retrieval module, as hypothesized concepts are used only as a soft prior rather than a hard filter.

7.6 Case Study: LLM-based Text-to-SQL

We present a qualitative case study on *LLM-based Text-to-SQL*, which has one of the deepest and most fine-grained ground-truth taxonomies in our benchmark.

Taxonomy Evolution. Table 7 tracks the taxonomy’s evolution over time. Initially ($t=1$), the structure captures a coarse pipeline with broad subtopics like consistency post-processing and few-shot prompting. By the middle stage ($t=4$), it becomes significantly more granular, separating *Schema Linking* from general pre-processing and refining inference workflows into *Chain-of-Thought* and *Decomposition*, reflecting the 2023–2024 literature. Finally ($t=7$), it captures emerging trends difficult to surface via keyword search, such as *Autonomous Agents*.

Table 7. Representative taxonomy evolution stages.

Stage	Representative New Concepts	Example Evidence Papers
$t=1$	Post-processing → Consistency Method; Few-shot → Replacing the task description; Model Evaluation	BINDER: <i>Binding language models in symbolic languages</i> (2022)
$t=4$	Pre-processing → Schema Linking (Traditional vs. LLM-based); Inference → Workflow (CoT / Decomposition)	ACT-SQL (2023); DIN-SQL (2023)
$t=7$	Workflow → Autonomous Agents; richer consistency post-processing	Spider 2.0 (2024); R3 (2024); MCS-SQL (2024)

Table 8. Representative paper search results.

Concept & Query	Top-5 Results
Chain-of-Thought (CoT) “chain-of-thought prompting for text-to-SQL with correction”	Flat-Ret: <i>Divide and prompt</i> (2023); <i>Exploring chain of thought style prompting...</i> (2023); ACT-SQL (2023); <i>Text-to-SQL empowered...</i> (2023) GIST: <i>Divide and prompt</i> (2023); ACT-SQL (2023); CHESS (2024); CoE-SQL (2024); OpenSearch-SQL (2025)
Traditional Schema Linking “schema linking with skeleton retrieval / de-semanticization”	Flat-Ret: <i>De-semanticization...</i> (2023); DBCopilot (2023); C3 (2023); <i>OpenSQL Framework</i> (2024); RSL-SQL (2024) GIST: <i>De-semanticization...</i> (2023); CRUSH4SQL (2023); PURPLE (2024); SGU-SQL (2024); OpenSearch-SQL (2025)

Taxonomy-Indexed Search. We demonstrate the taxonomy’s value as a semantic index in Table 8. User queries are first aligned to the most relevant concept node, restricting retrieval to that concept’s subtree. This minimizes topic drift for overloaded terms (e.g., “consistency”). As shown, GIST concentrates ranking on ground-truth relevant papers (bolded) more effectively than flat retrieval (*Flat-Ret*), which often retrieves tangentially related work.

8 Related Work

Knowledge Graph Maintenance. Maintaining structured knowledge under continuous evolution is a long-standing data management problem [59]. Prior efforts can be broadly organized into three categories. (i) *Incremental knowledge-base construction*. Systems such as DeepDive [51] and Fonduer [61] extract entities and relations from newly arriving documents to grow a knowledge base over time. (ii) *Entity resolution and schema alignment*. A second line consolidates equivalent records across heterogeneous sources through learned matchers and unified resolution frameworks [17, 23, 37, 55]. (iii) *Continual learning under distribution drift*. Methods such as Elastic Weight Consolidation (EWC) [30] adapt model parameters to shifting input distributions, typically guarding against catastrophic forgetting via regularization or rehearsal. These categories share a common problem setting: the underlying schema is treated as *fixed*, and maintenance proceeds by growing coverage, consolidating duplicates, or adapting parameters within it. GIST instead targets a setting in which the schema itself—the concept hierarchy—evolves through expansion, abstraction, and specialization. This difference introduces two new challenges: (i) the supervision signal drifts with the hierarchy, so past training signals cannot be naively replayed as in rehearsal-based continual learning; and (ii) maintenance requires coupled growth and structural revision over concepts, rather than insertion under a fixed schema or instance-level record matching.

Taxonomy Construction. Automated taxonomy construction generally falls into two paradigms. *Entity-driven* methods expand or complete existing structures by predicting concept placement [27, 40, 45, 49, 53], recently incorporating LLMs for low-resource tasks [41]. Conversely, *document-driven* methods induce hierarchies directly from corpora via clustering [64], seed-guided growth [50], or

emerging LLM-based prompting [29, 32, 34, 63, 66]. These methods treat taxonomy construction as a generation task over fixed, small corpora, leaving LLM-based variants prone to structural hallucination. We instead target taxonomy maintenance in the wild in a dynamic, in-the-wild setting and reframe it as a *structural integration* problem: LLMs are restricted to extracting author-curated hierarchical signals from *Related Work* sections, while reliability-aware integration, hypothesized concept prediction from geometric voids, and budget-aware retrieval jointly address the efficiency and cost challenges of streaming paper arrivals.

Geometric Embeddings for Hierarchical Relations. To capture the asymmetry of hierarchical relations, researchers have explored non-Euclidean geometries, including 1) hyperbolic models [18, 39, 44], which embed concepts as points in a negatively curved space whose exponentially expanding volume compactly accommodates tree-like structure; 2) order embeddings [4, 33, 56], which model is-a as a partial order, originally via coordinate-wise dominance and subsequently extended to probabilistic and density-based formulations; and 3) *box embeddings* [1, 10, 36, 57], which represent each concept as an axis-aligned hyperrectangle so that is-a relations correspond to spatial inclusion between boxes. In this work, we choose box embeddings as they provide an explicit geometric prior in which the void volumes between boxes furnish a signal of where new concepts can plausibly emerge. Neither property is available from hyperbolic or order-based representations. Furthermore, extending the existing box-embedding framework, we introduce two components tailored to in-the-wild taxonomy maintenance. First, we propose a novel bidirectional mapping model to bridge the semantic word-embedding space and the geometric box-embedding space. Second, we introduce a geometric concept emergence model that predicts emerging concepts to guide evidence retrieval.

Data Integration. Data integration is a fundamental problem that arises across diverse data modalities, including tabular data [9, 25, 31, 35, 37, 55], graph data [5, 23, 52, 59], and text corpora [51, 61]. Our work is most closely related to the integration of graph-structured data, as it dynamically and incrementally consolidates multiple partial taxonomies into a unified, evolving taxonomy. For *graph data*, integration centers on aligning and fusing heterogeneous graph-structured sources. Representative efforts include large-scale knowledge graph construction that merges Wikipedia, WordNet, and other sources into unified ontologies [5, 52, 59], as well as entity alignment across knowledge graphs, where embedding-based and active learning methods jointly align entities and schemata [23]. A related direction leverages integrated graph structures for downstream retrieval and reasoning, such as retrieval-augmented generation over textual graphs [22] and hierarchical, query-focused summarization over entity-relation graphs [15]. Unlike conventional graph integration methods that align and fuse a fixed collection of pre-existing graphs, GIST continuously incorporates partial taxonomies extracted from newly arriving documents and revises the global concept hierarchy as the underlying scholarly repository evolves.

9 Conclusion

We address taxonomy maintenance in the wild over evolving scholarly repositories. We propose GIST, which leverages human-curated partial taxonomies as reliable signals and integrates them via geometric containment relationships, together with a probabilistic hypothesized concept generation module and a cost-effective evidence retrieval module. Experiments on the real-world arXiv corpus (ar5iv) across 12 topics show that GIST achieves strong taxonomy quality with substantially lower runtime and monetary cost than LLM-driven baselines.

Acknowledgments

Zhifeng Bao is supported in part by ARC DP240101211 and FT240100832. Junhao Gan is supported in part by ARC DP230102908.

References

- [1] Ralph Abboud, Ismail Ilkan Ceylan, Thomas Lukasiewicz, and Tommaso Salvatori. 2020. BoxE: A Box Embedding Model for Knowledge Base Completion. In *Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS)*. 1–13.
- [2] Association for Computing Machinery. [n.d.]. ACM Digital Library. <https://dl.acm.org>. Accessed: 2025-11-25.
- [3] Association for Computing Machinery. 2012. The 2012 ACM Computing Classification System. <https://www.acm.org/publications/class-2012>. Accessed: 2025-11-12.
- [4] Ben Athiwaratkun and Andrew Gordon Wilson. 2018. Hierarchical Density Order Embeddings. In *International Conference on Learning Representations (ICLR)*. <https://openreview.net/forum?id=HJCXZQbAZ>
- [5] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. DBpedia: A Nucleus for a Web of Open Data. In *The Semantic Web – ISWC 2007 + ASWC 2007 (Lecture Notes in Computer Science)*, Vol. 4825. Springer, 722–735. https://doi.org/10.1007/978-3-540-76298-0_52
- [6] Christopher M. Bishop. 2006. *Pattern Recognition and Machine Learning*. Springer.
- [7] Chengliang Chai, Jiabin Liu, Nan Tang, Ju Fan, Dongjing Miao, Jiayi Wang, Yuyu Luo, and Guoliang Li. 2023. Goodcore: Data-effective and data-efficient machine learning through coresets selection over incomplete data. *Proceedings of the ACM on Management of Data* 1, 2 (2023), 1–27.
- [8] Lydia B Chilton, Juho Kim, Paul André, Felicia Cordeiro, James A Landay, Daniel S Weld, Steven P Dow, Robert C Miller, and Haoqi Zhang. 2014. Frenzy: collaborative data organization for creating conference sessions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1255–1264.
- [9] Peter Christen. 2012. *Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection*. Springer. <https://doi.org/10.1007/978-3-642-31164-2>
- [10] Shib Sankar Dasgupta, Michael Boratko, Dongxu Zhang, Luke Vilnis, Xiang Lorraine Li, and Andrew McCallum. 2020. Improving Local Identifiability in Probabilistic Box Embeddings. In *Advances in Neural Information Processing Systems (NeurIPS)*, Vol. 33. 182–192.
- [11] Yuhao Deng, Chengliang Chai, Kaisen Jin, Linan Zheng, Lei Cao, Ye Yuan, and Guoren Wang. 2025. Two birds with one stone: Efficient deep learning over mislabeled data through subset selection. *Proceedings of the ACM on Management of Data* 3, 3 (2025), 1–28.
- [12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. 4171–4186.
- [13] Bolin Ding, Haixun Wang, Ruoming Jin, Jiawei Han, and Zhongyuan Wang. 2012. Optimizing Index for Taxonomy Keyword Search. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*. ACM, 493–504. <https://doi.org/10.1145/2213836.2213892>
- [14] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The Llama 3 Herd of Models. *arXiv preprint arXiv:2407.21783* (2024). <https://arxiv.org/abs/2407.21783>
- [15] Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitan-sky, Robert Osazuwa Ness, and Jonathan Larson. 2024. From local to global: A graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130* (2024).
- [16] Marcus Fontoura, Vanja Josifovski, Ravi Kumar, Christopher Olston, Andrew Tomkins, and Sergei Vassilvitskii. 2008. Relaxation in Text Search using Taxonomies. *Proceedings of the VLDB Endowment* 1, 1 (2008), 672–683. <https://doi.org/10.14778/1453856.1453930>
- [17] Sainyam Galhotra, Donatella Firmani, Barna Saha, and Divesh Srivastava. 2018. Robust entity resolution using random graphs. In *Proceedings of the 2018 International Conference on Management of Data*. 3–18.
- [18] Octavian-Eugen Ganea, Gary Bécigneul, and Thomas Hofmann. 2018. Hyperbolic Entailment Cones for Learning Hierarchical Embeddings. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*. PMLR, 1646–1655. <https://proceedings.mlr.press/v80/ganea18a.html>
- [19] Deyan Ginev, Bruce R. Miller, and Michael Kohlhase. 2025. ar5iv: HTML5-Converted arXiv Articles (Dataset). <https://sigmathling.kwarc.info/resources/ar5iv-dataset-2024/>. Sources up to October 2025; HTML5 conversion of arXiv articles using LaTeXML. Not a live preview service.
- [20] GIST 2026. GIST: Source Code. <https://github.com/DataAutonomyLab/gist>. Accessed: 2026-07-29.
- [21] Maarten Grootendorst. 2022. BERTopic: Neural topic modeling with a class-based TF-IDF procedure. *arXiv preprint arXiv:2203.05794* (2022).
- [22] Xiaoxin He, Yijun Tian, Yifei Sun, Nitesh V. Chawla, Thomas Laurent, Yann LeCun, Xavier Bresson, and Bryan Hooi. 2024. G-Retriever: Retrieval-Augmented Generation for Textual Graph Understanding and Question Answering. *arXiv preprint arXiv:2402.07630* (2024). <https://arxiv.org/abs/2402.07630>

- [23] Jiacheng Huang, Zequn Sun, Qijin Chen, Xiaozhou Xu, Weijun Ren, and Wei Hu. 2023. Deep active alignment of knowledge graph entities and schemata. *Proceedings of the ACM on Management of Data* 1, 2 (2023), 1–26.
- [24] Institute of Electrical and Electronics Engineers. [n.d.]. IEEE Xplore Digital Library. <https://ieeexplore.ieee.org>. Accessed: 2025-11-25.
- [25] Daomin Ji, Hui Luo, Zhifeng Bao, and J. Shane Culpepper. 2025. Table Integration in Data Lakes Unleashed: Pairwise Integrability Judgment, Integrable Set Discovery, and Multi-Tuple Conflict Resolution. *The VLDB Journal* 34, 3 (2025), 36:1–36:24. <https://doi.org/10.1007/s00778-025-00917-9>
- [26] Daomin Ji, Hui Luo, Zhifeng Bao, Junhao Gan, and Zi Huang. 2026. Taxonomy Maintenance in the Wild over Evolving Scholarly Data: Reliability, Efficiency, and Cost-Effectiveness. *arXiv preprint arXiv:2607.09149* (2026).
- [27] Minhao Jiang, Xiangchen Song, Jieyu Zhang, and Jiawei Han. 2022. TaxoEnrich: Self-Supervised Taxonomy Completion via Structure-Semantic Representations. In *Proceedings of The Web Conference (WWW)*. Lyon, France. https://hanj.cs.illinois.edu/pdf/www22_mjiang.pdf
- [28] Bhargav Kanagal, Amr Ahmed, Sandeep Pandey, Vanja Josifovski, Jeffrey Yuan, and Lluís García-Pueyo. 2012. Supercharging Recommender Systems using Taxonomies for Learning User Purchase Behavior. *Proceedings of the VLDB Endowment* 5, 10 (2012), 956–967. https://vldb.org/pvldb/vol5/p956_bhargavkanagal_vldb2012.pdf
- [29] Priyanka Kargupta, Nan Zhang, Yunyi Zhang, Rui Zhang, Prasenjit Mitra, and Jiawei Han. 2025. TaxoAdapt: Aligning LLM-based multidimensional taxonomy construction to evolving research corpora. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vienna, Austria, 29834–29850.
- [30] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. 2017. Overcoming Catastrophic Forgetting in Neural Networks. *Proceedings of the National Academy of Sciences* 114, 13 (2017), 3521–3526. <https://doi.org/10.1073/pnas.1611835114>
- [31] Pradap Konda, Sanjib Das, Paul Suganthan G. C., AnHai Doan, Adel Ardalan, Jeffrey R. Ballard, Han Li, Fatemah Panahi, Haojun Zhang, Jeffrey F. Naughton, Shishir Prasad, Ganesh Krishnan, Rohit Deep, and Vijay Raghavendra. 2016. Magellan: Toward Building Entity Matching Management Systems. *Proceedings of the VLDB Endowment* 9, 12 (2016), 1197–1208. <https://www.vldb.org/pvldb/vol9/p1197-pkonda.pdf>
- [32] Avishek Lahiri, Yufang Hou, and Debarshi Kumar Sanyal. 2025. TaxoAlign: Scholarly Taxonomy Generation Using Language Models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. 30191–30211.
- [33] Alice Lai and Julia Hockenmaier. 2017. Learning to Predict Denotational Probabilities for Modeling Entailment. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*. 721–730. <https://aclanthology.org/E17-1068/>
- [34] Hai Lan, Tingting Wang, Zhifeng Bao, Guoliang Li, Daomin Ji, Ge Lee, Feng Luo, Zi Huang, Hailang Qiu, and Gang Hua. 2026. AgenticScholar: Agentic Data Management with Pipeline Orchestration for Scholarly Corpora. *Proceedings of the ACM on Management of Data* 4, 3 (SIGMOD) (2026), 1–28.
- [35] Ge Lee, Shixun Huang, Zhifeng Bao, Felix Naumann, Shazia Sadiq, and Yanchang Zhao. 2026. Shape-Agnostic Table Overlap Discovery: A Maximum Common Subhypergraph Approach. *Proceedings of the ACM on Management of Data* 4, 3 (SIGMOD) (2026), 1–26. <https://doi.org/10.1145/3802108>
- [36] Xiang Li, Luke Vilnis, Dongxu Zhang, Michael Boratko, and Andrew McCallum. 2019. Smoothing the Geometry of Probabilistic Box Embeddings. In *International Conference on Learning Representations (ICLR)*. <https://openreview.net/forum?id=H1xSNiRcF7>
- [37] Yuliang Li, Jinfeng Li, Yoshihiko Suhara, AnHai Doan, and Wang-Chiew Tan. 2020. Deep entity matching with pre-trained language models. *Proceedings of the VLDB Endowment* 14, 1 (2020), 50–60. <https://doi.org/10.14778/3421424.3421431>
- [38] Yuyin Lu, Hegang Chen, Pengbo Mao, Yanghui Rao, Haoran Xie, Fu Lee Wang, and Qing Li. 2024. Self-supervised Topic Taxonomy Discovery in the Box Embedding Space. *Transactions of the Association for Computational Linguistics (TACL)* (2024). https://doi.org/10.1162/tacl_a_00712
- [39] Mingyu Derek Ma, Muhao Chen, Te-Lin Wu, and Nanyun Peng. 2021. HyperExpan: Taxonomy Expansion with Hyperbolic Representation Learning. In *Findings of the Association for Computational Linguistics: EMNLP 2021*. Association for Computational Linguistics, 4182–4194. <https://doi.org/10.18653/v1/2021.findings-emnlp.353>
- [40] Yuning Mao, Xiang Ren, Jiaming Shen, Xin Gu, and Jiawei Han. 2018. End-to-End Reinforcement Learning for Automatic Taxonomy Induction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia, 2462–2472. <https://doi.org/10.18653/v1/P18-1229>
- [41] Sahil Mishra, Ujjwal Sudev, and Tanmoy Chakraborty. 2024. FLAME: Self-Supervised Low-Resource Taxonomy Expansion using Large Language Models. *arXiv preprint arXiv:2402.13623* (2024). <https://arxiv.org/abs/2402.13623>
- [42] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. 2018. *Foundations of Machine Learning* (2 ed.). The MIT Press, Cambridge, MA, USA.

- [43] George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. 1978. An analysis of approximations for maximizing submodular set functions—I. *Mathematical Programming* 14, 1 (1978), 265–294. <https://doi.org/10.1007/BF01588971>
- [44] Maximilian Nickel and Douwe Kiela. 2017. Poincaré Embeddings for Learning Hierarchical Representations. In *Advances in Neural Information Processing Systems (NeurIPS)*. 6338–6347.
- [45] Robert C Nickerson, Upkar Varshney, and Jan Muntermann. 2013. A method for taxonomy development and its application in information systems. *European Journal of Information Systems* 22, 3 (2013), 336–359.
- [46] Gary W Oehlert. 1992. A note on the delta method. *The American Statistician* 46, 1 (1992), 27–29.
- [47] Diptikalyan Saha, Avriila Floratou, Karthik Sankaranarayanan, Umar Farooq Minhas, Ashish R Mittal, and Fatma Özcan. 2016. ATHENA: an ontology-driven system for natural language querying over relational data stores. *Proceedings of the VLDB Endowment* 9, 12 (2016), 1209–1220.
- [48] Jaydeep Sen, Chuan Lei, Abdul Quamar, Fatma Özcan, Vasilis Efthymiou, Ayushi Dalmia, Greg Stager, Ashish Mittal, Diptikalyan Saha, and Karthik Sankaranarayanan. 2020. ATHENA++: Natural language querying for complex nested SQL queries. *Proceedings of the VLDB Endowment* 13, 12 (2020), 2747–2759.
- [49] Jiaming Shen, Meng Jiang, Xian Li, Jian Li, and Jiawei Han. 2020. TaxoExpan: Self-supervised Taxonomy Expansion with Position-Enhanced Graph Neural Network. In *Proceedings of The Web Conference (WWW)*. https://hanj.cs.illinois.edu/pdf/www20_jshen.pdf
- [50] Jiaming Shen, Zeqiu Wu, Dongming Lei, Chao Zhang, Xiang Ren, Michelle T. Vanni, Brian M. Sadler, and Jiawei Han. 2019. HiExpan: Task-Guided Taxonomy Construction by Hierarchical Tree Expansion. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP-IJCNLP)*. <https://arxiv.org/abs/1910.08194>
- [51] Jaeho Shin, Sen Wu, Feiran Wang, Christopher De Sa, Ce Zhang, and Christopher Ré. 2015. Incremental Knowledge Base Construction Using DeepDive. *Proc. VLDB Endow.* 8, 11 (2015), 1310–1321. <https://doi.org/10.14778/2809974.2809991>
- [52] Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2008. YAGO: A Large Ontology from Wikipedia and WordNet. *Web Semantics: Science, Services and Agents on the World Wide Web* 6, 3 (2008), 203–217. <https://doi.org/10.1016/j.websem.2008.06.001>
- [53] Yushi Sun, Hao Xin, Kai Sun, Yifan Ethan Xu, Xiao Yang, Xin Luna Dong, Nan Tang, and Lei Chen. 2024. Are Large Language Models a Good Replacement of Taxonomies? *Proceedings of the VLDB Endowment* 17, 11 (2024), 2919–2932. <https://doi.org/10.14778/3681954.3681973>
- [54] Masahiro Tanaka, Yasuyuki Mori, and Andrzej Bargiela. 2002. Granulation of keywords into sessions for timetabling conferences. *Proceedings of soft computing and intelligent systems (SCIS 2002)* (2002), 1–5.
- [55] Jianhong Tu, Ju Fan, Nan Tang, Peng Wang, Guoliang Li, Xiaoyong Du, Xiaofeng Jia, and Song Gao. 2023. Unicorn: A unified multi-tasking model for supporting matching tasks in data integration. *Proceedings of the ACM on Management of Data* 1, 1 (2023), 1–26.
- [56] Ivan Vendrov, Ryan Kiros, Sanja Fidler, and Raquel Urtasun. 2016. Order-Embeddings of Images and Language. In *International Conference on Learning Representations (ICLR)*. <https://arxiv.org/abs/1511.06361>
- [57] Luke Vilnis, Xiang Li, Shikhar Murty, and Andrew McCallum. 2018. Probabilistic Embedding of Knowledge Graphs with Box Lattice Measures. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*. 263–272. <https://doi.org/10.18653/v1/P18-1025>
- [58] Jiayi Wang, Chengliang Chai, Nan Tang, Jiabin Liu, and Guoliang Li. 2022. Coresets over multiple tables for feature-rich and data-efficient machine learning. *Proceedings of the VLDB Endowment* 16, 1 (2022), 64–76.
- [59] Gerhard Weikum. 2021. Knowledge Graphs 2021: A Data Odyssey. *Proc. VLDB Endow.* 14, 12 (2021), 3233–3238. <https://doi.org/10.14778/3476311.3476393>
- [60] D. H. D. West. 1979. Updating Mean and Variance Estimates: An Improved Method. *Commun. ACM* 22, 9 (1979), 532–535.
- [61] Sen Wu, Luke Hsiao, Xiao Cheng, Braden Hancock, Theodoros Rekatsinas, Philip Levis, and Christopher Ré. 2018. Fonduer: Knowledge Base Construction from Richly Formatted Data. In *Proceedings of the 2018 International Conference on Management of Data (SIGMOD '18)*. ACM, 1301–1316. <https://doi.org/10.1145/3183713.3183729>
- [62] Wei Xue, Yongliang Shen, Wenqi Ren, Jietian Guo, Shiliang Pu, and Weiming Lu. 2024. Insert or Attach: Taxonomy Completion via Box Embedding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*. <https://aclanthology.org/2024.acl-long.212.pdf>
- [63] Qingkai Zeng, Yuyang Bai, Zhaoxuan Tan, Shangbin Feng, Zhenwen Liang, Zhihan Zhang, and Meng Jiang. 2024. Chain-of-Layer: Iteratively Prompting Large Language Models for Taxonomy Induction from Limited Examples. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM)*. <https://arxiv.org/abs/2402.07386>
- [64] Chao Zhang, Fangbo Tao, Xiushi Chen, Jiaming Shen, Meng Jiang, Brian Sadler, Michelle Vanni, and Jiawei Han. 2018. TaxoGen: Unsupervised Topic Taxonomy Construction by Adaptive Term Embedding and Clustering. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2701–2709. <https://doi.org/10.1145/3219819.3220064>

- [65] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*. 2223–2232.
- [66] Kun Zhu, Lizi Liao, Yuxuan Gu, Lei Huang, Xiaocheng Feng, and Bing Qin. 2025. Context-aware hierarchical taxonomy generation for scientific papers via LLM-guided multi-aspect clustering. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. 15627–15645.

Received 17 January 2026; revised 19 May 2026; accepted 12 June 2026